



A Doctrinal Reconstruction of Engineering Informatics: Data Pipelines, Digital Twins, and Decision Authority in Cyber-Physical Systems

Felix Eling^{1*}¹ ORCID: <https://orcid.org/0009-0008-1071-0133>* Email ID: feling@osiriuniversity.org

ARTICLE INFO

Article history:

Received: 01-08-2026

Received in revised form:
02-08-2026

Accepted: 12-09-2026

Available online: 13-09-2026

Keywords:

Cyber-Physical Systems

Data Provenance

Digital Twins

Engineering Data Pipelines

Informatics Methods

Intelligent Operations

ABSTRACT

Engineering informatics is equipped with a variety of technologies including data pipelines, digital twins, cyber-physical loops, and operational analytics; however, there is an absence of a clear theoretical understanding in regard to how all these technologies collectively create reliable engineering knowledge. This paper provides a framework for the above area of study through identifying four classes of problems in the field: data pipelines, digital twins and cyber-physical systems, intelligent operations, and informatics methods. A doctrinal methodology was developed based on primary-instrument analysis, which viewed foundational literature, reference architectures, and international standards as mutually influencing resources rather than simply a collection of technologies. The central artifact of the field is not any singular model or algorithm but the traceable path from sensor evidence to decision authority. Four doctrinal tests were formulated: provenance continuity, semantic interoperability, twin fidelity with declared uncertainty, and contestable operational authority. These tests created a Traceable Engineering Knowledge Cycle that connects acquisition, representation, synchrony, decision, actuation and audit. The paper also addresses why technologically advanced systems may still fail as engineering knowledge systems and creates a research agenda for pipeline, twin and intelligent operation technologies that remain verifiable by industrial, safety and organisational constraints.

1. INTRODUCTION

Engineering organisations now instrument plants, fleets, grids, and products at a density that would have been implausible two decades ago, yet the resulting data rarely travel as knowledge. Streams are captured, stored, and visualised; models are trained; dashboards are populated; and still the decisive questions remain unsettled: what may be believed, who may act, and on what evidence. The strategic initiative that popularised Industrie 4.0 treated this coupling of physical production and information systems as a national industrial project rather than as a local software upgrade [1]. Subsequent surveys of the same programme show that connectivity, analytics, and cyber-physical integration have become a research field in their own right [2]. What has not kept pace is a doctrine capable of saying when those integrations succeed as engineering knowledge systems rather than as collections of tools.

That territory is usefully organised around four problem-classes: engineering data pipelines, digital twins and cyber-physical systems, intelligent operations, and informatics methods, including data-driven decision systems. Those four classes are usually discussed in separate literatures. Pipeline research speaks the language of acquisition, quality, and warehouses. Digital-twin research speaks of virtual counterparts and synchrony. Cyber-physical research speaks of feedback loops and layered architectures [3]. Operations

research speaks of prediction, maintenance, and closed-loop control [4]. The doctrinal problem is that an organisation can be competent in each literature and still fail at the join: a high-fidelity twin fed by an untraceable pipeline, a predictive model that cannot be contested, or an actuation path that no longer remembers the evidence that authorised it.

This article is a theoretical reconstruction, not a laboratory report and not a systematic map of every digital-twin paper. The article presents primary standards, reference architecture and academic canon as tools that restrict the possible claims to be made by Engineering Informatics. The NASA/United States Air Force definition of the Digital Twin as a multi-physics, multi-scale, ultra-accurate simulation of a vehicle defined what was possible for engineering informatics within the domain of aerospace [5]. The subsequent manufacturing formulation of the twin as a "virtual copy" of the product to be maintained synchronized with its real-world counterpart defined the possible in the context of industry [6]. Both formulations represent commitments of evidence, updates and residual uncertainty. Since then, the state-of-the-art in industry has extended these commitments throughout the entire lifecycle including design, manufacture and service [7]. In addition, studies on classification have demonstrated that most so-called digital twins are simply digital models or digital representations of physical products without closure [8]. Studies examining characterizations have similarly indicated that despite almost a decade of significant scientific publication, the concept still does not possess stability [9].

The contribution of this paper is a unifying doctrine for that instability. It argues that the object of engineering informatics is the *traceable passage from sensor evidence to decision authority*. Four tests follow from that object: provenance continuity, semantic interoperability, twin fidelity with declared uncertainty, and contestable operational authority. The tests are assembled into a Traceable Engineering Knowledge Cycle (TEKC) that links acquisition, representation, synchrony, decision, actuation, and audit. The reconstruction is written from those four problem-classes because they are the materials from which a field-level doctrine can be built. The paper therefore analyses the intellectual territory of engineering informatics rather than any one tool or sector.

Four question areas can be used to structure the arguments. What kind of methodology should be used when the evidence comes from standards and architectures instead of experiments? How does a data pipeline add value when it is considered an instrument for gathering knowledge (as opposed to just "plumbing")? In terms of the duties of fidelity and synchrony that Digital Twins and Cyber Physical Systems imply, how much do those duties require to be met? Under what circumstances will Intelligent Operations continue to be Engineering Decisions versus Opaque Automations?

2. DOCTRINAL METHOD AND ANALYTICAL STANCE

2.1 Why a Doctrinal Paper in Engineering Informatics

Doctrinal research is also well-known within legal studies; the researcher constructs a coherent narrative based on laws (statutes), court decisions (judgments) and rules (principles). The engineering-informatics community has an equivalent collection. Standards from organizations such as ISO, IEC, IEEE, NIST and national Industrie 4.0 standards for example, define all that a "digital twin" can be, what a "data quality" process can be, what an "enterprise control interface" can be, and so forth. There are also canonical papers that specify what must be contained in a "twin," and how a "cyber physical" architecture should be structured. A doctrine paper does not substitute for experimentation. Rather, it investigates if the fundamental instruments when read together will indicate a structural framework which subsequent experimental work would need to observe.

2.2 Primary Instruments and Secondary Commentary

The analysis distinguishes three classes of source. Primary sources are established norms and authoritative standards: Digital Twin Standard (ISO 23247) for Manufacturing; Cyber Physical Systems Framework (NIST); RAMI 4.0; Enterprise-Control Integration (IEC 62264); Data Quality Standards (ISO 8000); Industrial Security Standards (IEC 62443); Internet of Things (IEEE 2413); and Specifications for the Asset Administration Shell. Secondary academic sources include articles that have developed into canonical works since they have been used as defining works instead of illustrative works [5]. Tertiary commentaries consist of case studies, classification schemes, and surveys on how industrial practices align with such definitions [8].

This categorization has implications. A study may determine that definitions differ but it cannot alone define what an object in a twin will be responsible to perform. A standard may establish what responsibilities the object must fulfill, but it cannot by itself describe why companies continue to develop

and deploy dashboard applications that violate these responsibilities. Doctrine is the process of developing a way of seeing both types of sources so that the responsibilities and failures associated with each type can be viewed through the same lens.

2.3 Analytical Procedure

The process can be broken down into four stages. First, the field is to be considered as a collection of maps for four classes of problems as opposed to being considered as a set of tools. Second, the characteristics of each of these problem-class-maps are to be identified from both their respective original instrumentation, and in addition the formal definitions of the problem class made by subsequent scholarship. Third, the contradictions inherent to multiple pillars will be articulated as a series of theoretical problems (e.g., a pipeline without provenance cannot feed a twin; a twin without uncertainty cannot authorise actuation; an operation without contestability cannot remain an engineering decision). Fourth, the contradictions are to be resolved as much as possible given available data in terms of testing and a cycle rather than in terms of some proprietary architecture.

The method is openly interpretive. Selection bias is a real risk in any narrative reconstruction, and the paper does not claim PRISMA completeness. Mitigation takes two forms: preference for sources that later reviews themselves treat as obligatory, and refusal to invent bibliographic details. Claims that cannot be tied to a named instrument are marked as proposals of this reconstruction, not as findings of a hidden experiment.

3. THE INTELLECTUAL TERRITORY OF ENGINEERING INFORMATICS

3.1 Four Pillars, One Field

Engineering informatics sits at the intersection of engineering science, information systems, and computational analytics. The four problem-classes used here are more concrete than that intersection metaphor. Engineering data pipelines concern the acquisition, conditioning, and governance of high-volume data across instrumented assets. Digital twins and cyber-physical systems concern model-based representations, sensor fusion, and the feedback loops that close the gap between physical assets and decision software. Intelligent operations concern data-driven decision systems, predictive maintenance, and the insertion of analytics into engineered workflows. Informatics methods concern data architectures, schemas, and engineering-grade machine-learning practice.

Read separately, the four pillars look like neighbouring specialisms. Read together, they describe a single passage. Data must be acquired and governed before they can represent an asset. A representation must remain coupled to the asset before it can support a decision. A decision must remain attributable before it can be allowed to actuate. Informatics methods are the grammar of that passage, not an optional fifth topic.

3.2 What the Field Is Not

The reconstruction is sharper if exclusions are stated. Clinical predictive analytics belong primarily to health-informatics research unless the contribution is an engineering data system that happens to carry physiological signals. Purely legal analyses of algorithmic justice belong to digital-justice research unless they constrain operational authority inside an engineered loop. Learning-analytics dashboards belong to the knowledge sciences unless they are used as industrial human-machine interfaces. The point is not territorial policing. It is that the distinctive object of engineering informatics is the engineered coupling of data, model, and actuation under physical constraint.

Industry 4.0 popularised that coupling as a production narrative [1]. The 5C architecture later gave the narrative a layered form: connection, conversion, cyber, cognition, and configuration [3]. Cyber-physical production research then insisted that manufacturing CPS are not simply computers attached to machines but systems whose computational and physical processes are inseparable [4]. Digital-twin research supplied the missing representational centre: a virtual counterpart that is supposed to remain current with the physical entity across its life cycle [6]. The field of engineering informatics is what remains when those four statements are treated as one problem.

3.3 The Instability of the Central Terms

The digital twin is the most unstable of those terms. Early aerospace use bound the twin to an as-built vehicle, available data, and a fleet-management purpose [5]. Manufacturing use loosened the bound until

almost any simulation, shadow, or dashboard could be so named [8]. Systematic characterisation has identified recurring properties (physical entity, virtual entity, connection, data, and purpose) while also documenting that few implementations instantiate all of them [9]. Survey work has repeated the same warning in application after application [10]. The warning is doctrinal, not merely bibliographic. If the word no longer carries duties, it cannot authorise action.

Data are similarly unstable. The FAIR principles require findability, accessibility, interoperability, and reuse, and they were written for scholarly stewardship rather than for a shop-floor historian [11]. Industrial data-quality standards impose process duties that FAIR does not [12]. Provenance models such as W3C PROV supply a vocabulary for derivation, attribution, and activity that neither FAIR nor ISO 8000 fully replaces [13]. A pipeline that satisfies storage service-level agreements can still fail every one of those duties. The field therefore needs a way to say what a pipeline must preserve if it is to count as an engineering knowledge instrument.

Table 1 maps the four problem-classes to their primary instruments and to the doctrinal objects this paper extracts from them.

Table 1. Problem-classes of engineering informatics, primary instruments, and doctrinal objects.

Problem-class	Primary instruments	Doctrinal object extracted here
Engineering data pipelines	ISO 8000; ISO/IEC 25012; W3C PROV; DAMA-DMBOK	Provenance continuity from acquisition to reuse
Digital twins and cyber-physical systems	ISO 23247; NIST SP 1500-201; RAMI 4.0; 5C architecture	Fidelity, synchrony, and declared uncertainty
Intelligent operations	IEC 62264; ISO 55000; IEC 62443; industrial analytics literature	Contestable decision authority over actuation
Informatics methods	IEEE 2413; Asset Administration Shell; ISO/IEC 22989; OPC UA	Semantic interoperability across life-cycle views

The table is a claim, not a catalogue. Each cell will be argued in the sections that follow. The remainder of the paper asks what happens when the four objects are required to hold at once.

4. DATA PIPELINES AS INSTRUMENTS OF ENGINEERING KNOWLEDGE

4.1 From Acquisition to Governed Evidence

Engineering pipelines do more than move data - they also move claims that assets have been measured at certain times with sensors of specified calibration and precision. The engineering pipeline is not just about moving bytes; it's about carrying the claim that the temperature was measured on this particular asset by this particular instrument under this particular calibration with this particular uncertainty at this particular time.

ISO 8000 views data quality as a managed process and not simply something that will occur when the data is stored. ISO/IEC 25012 similarly identifies both the inherent quality of data (i.e., the degree to which it is accurate, complete, consistent, etc.) and the system dependent qualities (e.g., how it has been processed) and if these are not maintained during transmission then subsequent analytic processes will inherit this silent flaw [14].

In addition to managing quality, the FAIR principles introduce a stewardship language that many industrial pipeline systems do not currently follow [11]. As long as the assets being measured have no permanent identifier, their histories cannot be searched. When there are no documented access procedures, accessing measurements becomes a craft. As long as different systems use different terms to describe the same things, merging them into a single stream requires writing a new program each time. And if reuse is to be possible (and thus reusable), license, provenance and context must also be provided so that what today's historians find useful can be used by tomorrow's twin historians.

4.2 Provenance as the Pipeline's Central Duty

Provenance is the record of what produced a datum, from which other entities, by which activities, and under whose responsibility. These relationships can be described using the W3C PROV Data Model; an ontology specifically designed to describe these types of relationships [13]. In the context of engineering, these same relationships do not exist as academic metadata. The existence of these relationships would represent the difference between a value that could potentially enter into a Safety Case and a value that is likely only used to provide additional information to a dashboard. A pipeline that omits details such as instrument ID, units of measure, sampling frequency or transform histories, does not simply lose

documentation; it breaks the causal chain required for later sections to demonstrate twin fidelity and operational authority.

Existing data management bodies of knowledge have identified metadata, quality and lineage as primary knowledge domains rather than as secondary considerations [15]. However, the technical challenges associated with establishing these domains within industrial environments lies in their implementation as catalogs located outside of pipelines. A doctrine-based interpretation reverses this paradigm. For pipeline output to be utilized by twin outputs, provenance cannot be treated as an optional appendage. It represents the conditions under which a stream becomes evidence.

4.3 Conditioning, Semantics, and the Temptation of the Lake

High-volume engineering data invite the lake metaphor: land everything, decide later. The metaphor is operationally attractive and doctrinally dangerous.

When using the "lake" model of Data Management the idea behind this model is to collect all data into one large pool. This approach is appealing from a functional standpoint, but problematic from a semantic perspective. Any conditioning applied to the data (normalisation, synchronisation, outliers, missing values, etc.) cannot be delayed until the arrival of a data scientist. These are conditions placed upon the datum which create a claim regarding the datum. If these transformations occur too late and are not documented then according to PROV they constitute another process or activity. In addition, if they are not properly documented then the downstream twin is no longer seeing the actual plant [13].

The second aspect of data management that is also often overlooked is semantic binding. While IEEE 2413 establishes an architecture for IoT systems that advocates for cross domain descriptions as opposed to device level innovation, the Asset Administration Shell takes it a step further [16]. Each asset receives a standardized digital representation that can hold attributes, actions and documents regardless of vendor [17]. Therefore, a pipeline that holds anonymous time series data while the meaning of the asset resides in a privately maintained spreadsheet has already made a decision against interoperability. As such, later sections will view this as a failure of method within Informatics, as opposed to a convenient localized solution.

4.4 Security and Integrity of the Evidence Path

A pipeline is also an attack surface. Intrusion into industrial and cloud-connected data paths can corrupt the very evidence on which twins and operations depend [18]. Deep learning detectors and related security analytics are therefore not a neighbouring cybersecurity topic only. They are part of pipeline doctrine once the pipeline is understood as a knowledge instrument. The same point holds for classification and clustering of Internet of Things services: identity, grouping, and platform integration are preconditions of trustworthy acquisition, not optional analytics layered on afterwards [19]. A pipeline that cannot say which device spoke, whether the speech was altered, and how the speech was grouped cannot later claim to represent a plant.

The doctrinal test extracted here is *provenance continuity*. A pipeline passes the test when every value that may later authorise a model update or an actuation can be traced to an acquisition event, a transformation history, a semantic binding, and an integrity condition. It fails the test when any of those links is optional, reconstructable only by tribal knowledge, or silently overwritten.

5. DIGITAL TWINS AND CYBER-PHYSICAL SYSTEMS

5.1 Duties Implied by the Canonical Definitions

The aerospace statement of the digital twin bound the concept to an as-built product, ultra-realistic models, and available fleet data [5]. The later systems-engineering statement treated the twin as a means of mitigating unpredictable emergent behaviour across a life cycle [6]. Manufacturing research then placed the twin at the centre of Industry 4.0 by pairing a physical entity, a virtual entity, and the data connections between them [7]. Those three statements do not agree in every particular, but they agree on a duty: the virtual counterpart is answerable to a physical original. A model that is never updated is a model. A dashboard that never writes back is a shadow. A twin, if the word is to keep its force, must remain in a declared relation of fidelity to something that exists outside the software [8].

ISO 23247 gives that duty a manufacturing form. The standard describes a digital-twin framework with observable manufacturing elements, a device communication entity, a digital-twin entity, and a user entity, and it treats synchronisation as a structural requirement rather than as an implementation hint [20]. NIST's cyber-physical systems framework, independently, analyses CPS through domains, facets, and aspects, and it refuses to reduce the system to either the computer or the plant [21]. RAMI 4.0 adds a three-dimensional reference architecture that relates hierarchy levels, life-cycle value streams, and layers from asset to business [22]. None of these instruments is a product specification. Each is a claim about what must be present if the resulting system is to count as a twin or a CPS rather than as a simulation study.

5.2 Fidelity, Synchrony, and the Residual of Uncertainty

The main issue with "fidelity" (the quality of the digital twin) is that it is commonly confused with "photorealism." Research on characterization has made clear that the correct question to ask about the degree of representation of a virtual object or system is simply whether the object or system is sufficient for a specific use case within a specified update routine [9]. A structural twin used for life prediction of a physical asset, and a kinematic twin used for simulation of robotic motion planning, could each be considered sufficiently accurate (or "faithful") to their respective purposes using vastly different levels of detail. Both would require stating how much detail was used when developing them. Surveys repeatedly find that this declaration is missing [23]. The lack of such a statement is not merely an oversight related to documenting the development of the digital shadow; it is a conceptual gap, since an accuracy level cannot be compared to the risk of the decision being supported by the accuracy level of the digital shadow.

"Synchrony" is the second aspect of the above obligation. As described previously, there are classifications of digital shadows based on characteristics of how data is transmitted from/to the physical world. The two primary categories include one-way communication versus closed-loop systems [8]. Closed-loop status does not automatically imply superiority. For example, a relatively static physical structure in a civil engineering application might be better represented by a batch updated digital shadow, whereas a rapidly moving industrial process might pose safety risks if its digital shadow were subject to delays prior to writing commands to actuators. Regardless, the responsibility is to identify which rate of updates, latency, and valid time window should be applied to the type of decision the digital shadow is authorized to make regarding the physical world [24]. Even though a digital shadow may be highly geometrically accurate, failing to establish matching rates and timing of those updates and latency represents a failure of this responsibility.

"Uncertainty," is the last component of the above obligation, and perhaps the least documented. Accounts of the digital twin focused primarily on applications involving simulations have generally defined the virtual side of the twin as a living model that includes all assumptions inherent in creating/developing it [25]. More recent implementation frameworks define scope and requirements for establishing a digital twin, including the requirement to acknowledge what the digital twin is *not* [26]. If a digital twin is developed with no consideration for uncertainty associated with its development, subsequent actions taken based upon that twin will likely result in some form of actuation based solely upon an assumed single value or point estimate. This does not represent intelligent action; rather it represents a mischaracterization.

5.3 Cyber-Physical Layering and the Place of the Twin

The 5C architecture remains the most cited layering of manufacturing CPS: smart connection, data-to-information conversion, cyber, cognition, and configuration [3]. Cyber-physical production research has since insisted that these layers are not a maturity ladder that every factory must climb in order, but a way of locating where computation and physics become inseparable [4]. The twin sits most naturally at the cyber and cognition layers: it is the representation that makes configuration (actuation, reconfiguration, supervisory control) answerable to conversion (conditioned evidence). Remove the twin and the architecture still functions as a control stack. Remove the duties of fidelity and the stack still functions, but it no longer knows what it is controlling in the sense that later audit can reconstruct.

Reference discussions of autonomy and twins for future manufacturing made a related point a decade ago: autonomy without an inspectable counterpart relocates skill without relocating responsibility [27]. Shop-floor realisations of cyber-physical production systems then showed how quickly the twin becomes a practical necessity once products, processes, and resources must be represented together [28]. Those papers do not settle implementation. They settle a doctrinal order. Representation precedes authorised reconfiguration.

5.4 Applications Do Not Dissolve the Duties

Application-based surveys of industry have illustrated examples of "twin" approaches used for design, manufacturing, prognostics, and service [29]. Pairing large data sets using industrial analytics and "twins," has highlighted how volume and variability increase, rather than decrease the necessity for the representative center [30]. Reference models for smart manufacturing have simply restated this pairing as a research agenda [31]. The additional wealth of applications does not reduce the obligations defined above. A predictive maintenance twin that does not provide information regarding what sensor(s), transformation(s) and model versions were responsible for producing alerts will be no more industrial because it is helpful; it will simply be more difficult to justify.

The doctrine-based test identified above is *twin fidelity with declared uncertainty*. A twin or CPS representation meets the test of twin fidelity with declared uncertainty when there is documentation of all four of the following: its purpose, the nature of updates to the representation (regime), its adequacy limits (or other limitations), and the residual uncertainties. When these items are implied or when a digital model or shadow is referred to as a "twin," or when authorization to perform an action occurs based on a representation whose time frame for validation has passed; the representation fails to meet the test of twin fidelity with declared uncertainty.

6. INTELLIGENT OPERATIONS AND DECISION AUTHORITY

6.1 From Prediction to an Authorised Act

Intelligent operations represent the point when engineering informatics goes from simply describing the world to actually changing it. Predictive maintenance, quality intervention, energy dispatch, rerouting, and supervisory control represent decisions that commit plant, people and capital. The enterprise-control system integration standard IEC 62264 (descended from ISA-95) already distinguishes business logistics, manufacturing operations, and control and treats interfaces between these levels as design objects [32]. ISO 55000 also treats asset management as a coordinated process that realises value from assets, not as a maintenance department's private analytics [33]. An operational model that cannot be sited on those interfaces is still not an engineering decision. It is a recommendation seeking a home.

Industrial analytics literature frequently leaves out this locational work. A model is trained, accuracy reported and "deployment" left as a sentence. Deployment is where doctrine is tested. Who may accept the recommendation? Under what residual uncertainty? With what fallback if pipeline or twin is stale? With what record if the act later fails. Those questions aren't ethics bolted on afterwards. They are the conditions under which an operation remains intelligent in the engineering sense.

6.2 Contestability as an Engineering Requirement

Contestability requires that the reasons behind a decision made by someone in a position of engineering, supervisory authority, or investigating authority can be explained through references to evidence, models, and policies, as opposed to simply referencing a vendor log. Contestability is therefore a stronger requirement than designing interfaces so they are understandable (explainable) but it includes this requirement. The use of soft computing and machine learning techniques for automation in industry is common now for sensors, controllers, and actuators [34]. Real time learning and nature inspired learning mechanisms have also been proposed as a way to close the loop sooner than classical tuning [35]. An organisation's ability to demonstrate how its speed increases will still require it to provide an explanation for the loop. If the loop cannot be reconstructed, the organisation has not automated judgement. It has hidden it.

Again, security should be considered as part of the operating doctrine rather than alongside it. Security within industrial communications has been defined in terms of zones, conduits and security levels as design considerations [36]. As such a predictive model that can easily be poisoned, a twin that can be desynchronised or an actuator command stream that accepts unsigned commands would fail as a process regardless of whether the predictive model's performance off-line is good. The previous discussion around intrusion detection systems on data streams takes on greater emphasis here: a compromised evidence path results in a compromised authority [18].

6.3 Human, Supervisory, and Autonomous Modes

Not all intelligent operations should operate autonomously. The existing Business Layer, Functional Layers and 5C configuration level of RAMI 4.0 indicate that some organisational decisions still exist when

the data for those decisions are technically available [22]. The 5C configuration level can represent a human supervisor, rules engine, or closed controller [3]. Doctrine does not dictate a single operating mode. Rather it dictates that the operating mode needs to be declared; that the transition from one operational mode to another is defined; and that the record of who (or what) acted on behalf of the system be just as long-lived as the records of the sensors used in the system. A system that oscillates back and forth between advisory, supervisory, and closed-loop modes, but does not define which operational mode is currently active has not become more intelligent. It has become less auditable.

Asset Management Language provides insight into this issue. Realising value includes accepting risk levels in addition to maintaining uptime [33]. Therefore, an operation that increases a local Key Performance Indicator (KPI), through unrecognised risk (i.e. running closer to a limit, ignoring alarms of a specific type/class, delaying inspection etc.) may be considered "smart" locally, while being indefensible globally. In such cases, the TEKC will consider these possibilities as Cycle Failures at both the Decision Node and Audit Nodes rather than successes of optimization.

6.4 The Operational Test

The doctrinal test extracted here is *contestable operational authority*. A given operation will satisfy the requirements for intelligent authorization if the operation in question is within either the enterprise control or asset management interfaces, can be recreated through policy and evidence, can be defined as operating in one of the declared modes and can be justified based upon the applicable security zoning. However, it may fail the requirements for intelligent authorization if, while the model is accurate, it does not have a home; if the operations which were authorized cannot be recreated; or if speed has been gained at the cost of eliminating the record of authorization.

7. INFORMATICS METHODS: INTEROPERABILITY, PROVENANCE, AND SEMANTIC CONTINUITY

7.1 Methods as Grammar, Not Toolkit

The fourth problem-class, informatics methods, is easy to misread as a list of techniques: schemas, architectures, and engineering-grade machine learning. The doctrinal reading is stricter. Informatics methods are the grammar that makes the first three pillars mutually intelligible. Without that grammar, a pipeline, a twin, and an operation can each be locally excellent and jointly mute.

ISO/IEC 22989 supplies a conceptual vocabulary for artificial intelligence systems, including terms for data, models, and life cycle, that industrial papers often reinvent [37]. IEEE 2413 supplies a cross-domain Internet of Things architecture rather than a device catalogue [16]. The Asset Administration Shell supplies a vendor-spanning digital representation of assets [17]. OPC Unified Architecture, standardised as the IEC 62541 series, supplies an information-modelling and secure-communication substrate that many industrial stacks already treat as infrastructure [38]. These instruments do not dictate an algorithm. They dictate whether two algorithms can share a world.

7.2 Semantic Continuity Across Life-Cycle Views

RAMI 4.0's life-cycle and value-stream axis is the most compact statement of the continuity problem [22]. A type is designed; an instance is manufactured; the instance is used, maintained, and retired. Data created in one view are routinely unusable in the next because identifiers, units, and property definitions were treated as local. ISO 23247's observable manufacturing elements are an attempt to name what must remain observable across those views [20]. The Asset Administration Shell is an attempt to carry the name with the asset [17]. Semantic continuity is what those attempts are for.

Provenance, already discussed as a pipeline duty, reappears here as a method duty. PROV's entities, activities, and agents are not a metadata fashion [13]. They are a way of writing the life-cycle axis so that a later twin update or operational act can still point to a design revision, a calibration, or a human override. A method stack that can train a model but cannot write that sentence has not reached engineering grade.

7.3 Engineering-Grade Learning

Engineering-grade machine learning is defined by its respect of the first three tests. Training data must inherit pipeline provenance [12]. Features and labels must inherit semantic bindings [16]. Model outputs that update a twin must inherit fidelity and uncertainty declarations [20]. Outputs that authorise acts must inherit contestability [32]. Wide applicability of soft computing techniques in industrial automation has

shown us just how large the range of possible approaches now are, and this is not our problem [34]. Our problem is using an engineering plant as if it were a database.

The doctrinal test extracted here is *semantic interoperability*. A given method satisfies the doctrinal test when identifiers, units, property names and provenance relationships among datum remain sufficiently consistent so that a datum collected in one view can be used in a second view provided the necessary transformations are declared. That same method fails the doctrinal test when some ad-hoc script, private spreadsheet or unnamed model is used for "integration".

8. A UNIFYING DOCTRINE: THE TRACEABLE ENGINEERING KNOWLEDGE CYCLE

8.1 Statement of the Doctrine

The four tests can now be stated as one doctrine. Engineering informatics is the discipline that makes the passage from sensor evidence to decision authority traceable, semantically continuous, fidelity-bounded, and contestable. A system that cannot pass those tests may still be useful software. It is not yet a complete engineering knowledge system in the sense that these four problem-classes jointly require.

The Traceable Engineering Knowledge Cycle (TEKC) is the operational form of that doctrine. It has six nodes:

1. *Acquisition*: instrumented capture of a physical or operational event, including identity, time, units, and integrity conditions.
2. *Representation*: binding of the event to a shared information model (schema, shell, or ontology) so that it can travel.
3. *Synchrony*: update of the virtual counterpart within a declared validity window and fidelity limit.
4. *Decision*: selection of an act, including the residual uncertainty and the mode (advisory, supervisory, or closed).
5. *Actuation*: commitment of the act to the physical or organisational system, with a durable record of authority.
6. *Audit*: reconstruction of the passage, including negative results, overrides, and expired windows.

Figure 1 sets out the cycle and the four tests that police its joins.

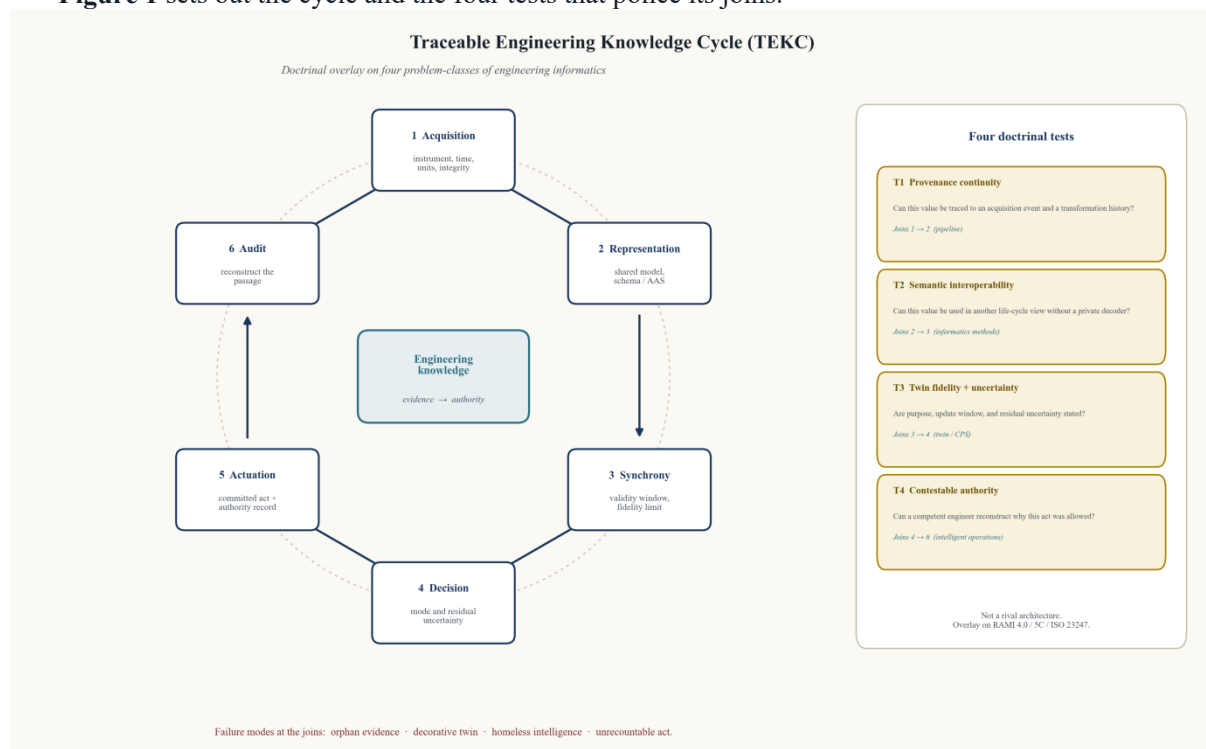


Figure 1. The Traceable Engineering Knowledge Cycle, showing six nodes from acquisition to audit and the four doctrinal tests that police the joins between the problem-classes of engineering informatics.

The cycle is deliberately not another reference architecture competing with RAMI 4.0, 5C, or ISO 23247. Those architectures already exist and should be used [3]. TEKC is a doctrinal overlay: a way of

asking, at each join, whether knowledge is still present. A plant can implement RAMI 4.0 and still fail TEKC if the audit node is missing. A laboratory can implement a high-fidelity twin and still fail TEKC if acquisition has no provenance.

8.2 Characteristic Failure Modes

Four failure modes follow almost mechanically.

Orphan evidence. Acquisition occurs, but representation is local. The pipeline is full and the twin is blind. This is a failure of semantic interoperability, usually dressed up as a data-lake success.

Decorative twin. Representation and even visualisation are rich, but synchrony is undeclared or expired. Operators treat the image as the plant. This is a failure of fidelity with declared uncertainty [8].

Homeless intelligence. A model scores well and is placed on a dashboard that no enterprise-control interface owns. When it is right, no one knows who should act; when it is wrong, no one knows who did. This is a failure of contestable authority [32].

Unrecountable act. Actuation occurs, perhaps safely, but audit cannot reconstruct the evidence, the model version, or the mode. The organisation has acted and cannot later explain itself. This is a simultaneous failure of provenance and authority [13].

These modes are not exotic. They are the ordinary ways in which technically sophisticated systems fail as knowledge systems. Application surveys that celebrate use cases without asking after these modes systematically overstate maturity [29].

8.3 Using the Tests without a New Stack

Table 2 converts the tests into questions that can be asked of an existing architecture. The point is diagnostic, not prescriptive of vendors.

Table 2. Doctrinal tests as diagnostic questions for existing engineering data systems.

Test	Diagnostic question	Typical evidence of a pass	Typical evidence of a fail
Provenance continuity	Can this value be traced to an acquisition event and a transformation history?	PROV-like records; calibration and unit lineage	Anonymous historian tags; undocumented scripts
Semantic interoperability	Can this value be used in another life-cycle view without a private decoder?	AAS / OPC UA / shared identifiers	Spreadsheet crosswalks; one-off extract-transform-load
Twin fidelity with declared uncertainty	Is purpose, update window, and residual uncertainty stated?	ISO 23247-aligned scope; validity stamps	“Real-time” labels; unnamed digital shadows
Contestable operational authority	Can a competent engineer reconstruct why this act was allowed?	IEC 62264 interface; mode and policy record	Accuracy-only reports; unsigned actuation

The table can be used in design reviews and in industrial assurance. It does not replace safety cases or security zoning [36]. It tells those practices where the informatics contribution must be visible.

9. DISCUSSION

9.1 Implications for Design

Designers of engineering data systems should start treating the pipeline as a knowledge instrument (and not just as plumbing) beginning with sensor specifications. It will be cheaper than it sounds. Acquiring identifiers, clocks, units, and transformation logs is less expensive than rebuilding them for a regulator, insurer, or failed-act investigation [15]. Choose your semantic models based upon continuity across life-cycle views rather than first-application convenience [17]. Scope twins by purpose and uncertainty; which may involve building smaller twins than the budget allows [26]. Give each operation a home on an enterprise-control/asset-management interface before giving them a neural network [32].

Research in industrial automation combining learning methods to sensors, controllers, and actuators has value because it is inherently constrained by these constraints [35]. So too do research areas such as IoT service classification and security analytics along the evidence path [19]. Techniques are welcome. Techniques that cannot pass the four tests must be documented as components (not intelligent operations).

9.2 Implications for Standards and Reporting

The standards corpus is already rich. ISO 23247, NIST SP 1500-201, RAMI 4.0, IEC 62264, ISO 8000, IEC 62443, IEEE 2413, and the Asset Administration Shell do not need a successor so much as they need to be read together [20]. The gap this paper names is not the absence of a fifteenth framework. It is the absence of a doctrine that says what those frameworks jointly oblige. The four problem-classes already correspond to the four tests, so the reading can be hosted without a new architecture.

Two reporting consequences follow. First, studies that report a new model without stating acquisition provenance, fidelity limits, or operational interface should be treated as incomplete as engineering-informatics contributions, even if the algorithm is novel. Second, theoretical and methodological papers should not be regarded as weaker cousins of experimental ones. The present reconstruction is offered as one form of that theoretical work: a disciplined account of what the primary instruments already jointly require.

9.3 Implications for Research

Several research priorities follow.

First, *provenance at industrial tempo*. PROV is expressive, but many plants will not write full graphs at sampling frequency [13]. Work is needed on compact, standards-aligned provenance that can travel with high-rate streams without becoming theatre.

Second, *fidelity contracts*. Characterisation research has listed properties of twins [9]. What is still thin is a contract language (purpose, validity window, residual uncertainty, prohibited uses) that can be checked as routinely as a schema.

Third, *homes for models*. Enterprise-control integration is mature as a standard and immature as a practice for machine-learned recommenders [32]. Studies that locate a model on IEC 62264 levels, including the human mode, would do more for intelligent operations than another accuracy table.

Fourth, *audit as a first-class node*. Security zoning and safety cases already know how to demand records [36]. Informatics research rarely treats the audit node as a design object with latency, retention, and reconstructability requirements of its own.

Fifth, *honest negative results*. Classification work that distinguishes models, shadows, and twins should be treated as a research success when it reports that a celebrated implementation is not a twin [8]. Over-claiming is not only a rhetorical problem. It is how decorative twins enter actuation paths.

Complementary reviews should be read as stress tests of the same doctrine rather than as a second literature. The early account of digital-twin roles inside cyber-physical production systems already treated representation as a production object rather than as a visualisation [39]. Enabling-technology surveys later listed the stack (Internet of Things, cloud, modelling, and analytics) while still warning that open problems of trust and integration remain [40]. Generalisation work has tried to state what must be true of any twin if the term is to travel across domains [41]. Life-cycle accounts of twin-driven design, manufacturing, and service show why a pipeline that works in one view can fail in the next [42]. Smart-factory architectures that close multi-agent feedback on big-data coordination make the same point from the operations side [43]. These sources do not add new tests. They confirm that the four tests already have independent witnesses.

9.4 Limits of the Reconstruction

This article did not measure one specific plant; this article did not compare different vendors; and this article does not provide an organized system for every 2020-2026 digital twin review. These are limitations on its work. These are also limitations of the type of research (genre) it was trying to accomplish. A doctrinal reconstruction may fail in two general ways: by overstating obligations imposed by the instruments, and by understating obligations already practiced. The four tests and the TEKC are therefore to be viewed as ideas or proposals to be tested and modified, not as a fixed rule. Subsequent empirical research is appropriate for further development.

This study is also manufacturing-heavy because that is where the most advanced tools presently exist. The energy, transportation, built environment, and process sectors will challenge the four tests in similar ways but with a greater variety of application. There is nothing within the doctrine to prohibit this variation. What is required is that new areas clearly identify what they consider adequate levels of acquisition, fidelity, and authority to use the term "digital twin" instead of using that term as if it were a completed concept.

10. CONCLUSION

The engineering informatics domain is well-equipped with a large number of tools but lacks a body of theory regarding how those tools collectively ensure the creation of trustworthiness through their production of reliable engineering information. As a whole area of study, including data pipelines, digital twins, cyber-physical systems, smart operation, and engineering-informatics methodologies, the space from instrumented events to actions that an organization can continue to justify represents a single path. This article constructed this path using the foundational mechanisms and basic tenets that currently exist within the space. Four tests (provenance continuity, semantic interoperability, twin fidelity with declared uncertainty, and contestable operational authority) were developed and then synthesized into a traceable engineering knowledge cycle. The cycle was not intended to be a new architecture, but instead to serve as a means by which to query at each connection point, whether the knowledge has been maintained. The normal failure modes that occur when these connection points are ignored include orphan evidence, decorative digital-twins, homeless intelligence, and undocumented actions.

From the perspective of designers, this implies specifying identifiers, clocks, units, fidelity limits, and operational homes prior to the selection of a learning method. From the perspective of standards bodies, this implies reading existing frameworks together rather than creating yet another disconnected reference model. Finally, for the field as a whole, this implies that theoretical works such as this have no role as pre-experimental studies. Instead, it illustrates one form in which the discipline of engineering informatics can become actionable. Future empirical and methodological investigations may elect to use these tests or modify/replace them; however, future studies should not consider a data-pipeline, a digital-twin or an operational-model to be complete until the path from evidence to authority is also traceable.

REFERENCES

- [1] Kagermann, H., Wahlster, W., & Helbig, J. (2013). *Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Final report of the Industrie 4.0 Working Group*. acatech / Forschungsunion.
- [2] Xu, L. D., Xu, E. L., & Li, L. (2018). Industry 4.0: State of the art and future trends. *International Journal of Production Research*, 56(8), 2941–2962. <https://doi.org/10.1080/00207543.2018.1444806>
- [3] Lee, J., Bagheri, B., & Kao, H.-A. (2015). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3, 18–23. <https://doi.org/10.1016/j.mfglet.2014.12.001>
- [4] Monostori, L., Kádár, B., Bauernhansl, T., Kondoh, S., Kumara, S., Reinhart, G., Sauer, O., Schuh, G., Sihn, W., & Ueda, K. (2016). Cyber-physical systems in manufacturing. *CIRP Annals*, 65(2), 621–641. <https://doi.org/10.1016/j.cirp.2016.06.005>
- [5] Glaessgen, E. H., & Stargel, D. S. (2012). The digital twin paradigm for future NASA and U.S. Air Force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*. American Institute of Aeronautics and Astronautics. <https://doi.org/10.2514/6.2012-1818>
- [6] Grieves, M., & Vickers, J. (2017). Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In F.-J. Kahlen, S. Flumerfelt, & A. Alves (Eds.), *Transdisciplinary perspectives on complex systems* (pp. 85–113). Springer. https://doi.org/10.1007/978-3-319-38756-7_4
- [7] Tao, F., Zhang, H., Liu, A., & Nee, A. Y. C. (2019). Digital twin in industry: State-of-the-art. *IEEE Transactions on Industrial Informatics*, 15(4), 2405–2415. <https://doi.org/10.1109/TII.2018.2873186>
- [8] Kritzinger, W., Karner, M., Traar, G., Henjes, J., & Sihn, W. (2018). Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*, 51(11), 1016–1022. <https://doi.org/10.1016/j.ifacol.2018.08.474>
- [9] Jones, D., Snider, C., Nassehi, A., Yon, J., & Hicks, B. (2020). Characterising the Digital Twin: A systematic literature review. *CIRP Journal of Manufacturing Science and Technology*, 29, 36–52. <https://doi.org/10.1016/j.cirpj.2020.02.002>
- [10] Barricelli, B. R., Casiraghi, E., & Fogli, D. (2019). A survey on digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access*, 7, 167653–167671. <https://doi.org/10.1109/ACCESS.2019.2953499>
- [11] Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., ... Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3, 160018. <https://doi.org/10.1038/sdata.2016.18>
- [12] International Organization for Standardization. (2022). *Data quality : Part 1: Overview* (ISO 8000-1:2022).

- [13] Moreau, L., & Missier, P. (Eds.). (2013). *PROV-DM: The PROV data model*. World Wide Web Consortium. <https://www.w3.org/TR/2013/REC-prov-dm-20130430/>
- [14] International Organization for Standardization. (2008). *Software engineering : Software product Quality Requirements and Evaluation (SQuaRE) : Data quality model* (ISO/IEC 25012:2008).
- [15] DAMA International. (2017). *DAMA-DMBOK: Data management body of knowledge* (2nd ed.). Technics Publications.
- [16] Institute of Electrical and Electronics Engineers. (2019). *IEEE standard for an architectural framework for the Internet of Things (IoT)* (IEEE Std 2413-2019). <https://doi.org/10.1109/IEEESTD.2019.8686444>
- [17] Plattform Industrie 4.0 / Industrial Digital Twin Association. (2022). *Details of the Asset Administration Shell: Part 1 : The exchange of information between partners in the value chain of Industrie 4.0* (Specification). Federal Ministry for Economic Affairs and Climate Action.
- [18] Pandey, J. K., Kumar, S., Lamin, M., Gupta, S., Dubey, R. K., & Sammy, F. (2022). A metaheuristic autoencoder deep learning model for intrusion detector system. *Mathematical Problems in Engineering*, 2022, Article 3859155. <https://doi.org/10.1155/2022/3859155>
- [19] Pandey, J. K., Bhuva, D., Bhuva, A., Abbas, S. H., Verma, A., & Moharekar, T. T. (2023). Classification and clustering of Internet of Things (IoT) for integrating IoT services platform. In *2023 IEEE 2nd International Conference on Industrial Electronics: Developments & Applications (ICIDEA)* (pp. 259–262). IEEE. <https://doi.org/10.1109/ICIDEA59866.2023.10295225>
- [20] International Organization for Standardization. (2021). *Automation systems and integration : Digital twin framework for manufacturing : Part 1: Overview and general principles* (ISO 23247-1:2021).
- [21] Griffor, E. R., Greer, C., Wollman, D. A., & Burns, M. J. (2017). *Framework for cyber-physical systems: Volume I, overview* (NIST Special Publication 1500-201). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.1500-201>
- [22] Deutsches Institut für Normung. (2016). *Reference Architecture Model Industrie 4.0 (RAMI4.0)* (DIN SPEC 91345:2016-04).
- [23] Semeraro, C., Lezoche, M., Panetto, H., & Dassisti, M. (2021). Digital twin paradigm: A systematic literature review. *Computers in Industry*, 130, 103469. <https://doi.org/10.1016/j.compind.2021.103469>
- [24] Liu, M., Fang, S., Dong, H., & Xu, C. (2021). Review of digital twin about concepts, technologies, and industrial applications. *Journal of Manufacturing Systems*, 58, 346–361. <https://doi.org/10.1016/j.jmsy.2020.06.017>
- [25] Boschert, S., & Rosen, R. (2016). Digital twin: The simulation aspect. In P. Hehenberger & D. Bradley (Eds.), *Mechatronic futures* (pp. 59–74). Springer. https://doi.org/10.1007/978-3-319-32156-1_5
- [26] Shao, G., & Helu, M. (2020). Framework for a digital twin in manufacturing: Scope and requirements. *Manufacturing Letters*, 24, 105–107. <https://doi.org/10.1016/j.mfglet.2020.04.004>
- [27] Rosen, R., von Wichert, G., Lo, G., & Bettenhausen, K. D. (2015). About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-PapersOnLine*, 48(3), 567–572. <https://doi.org/10.1016/j.ifacol.2015.06.141>
- [28] Uhlemann, T. H.-J., Lehmann, C., & Steinhilper, R. (2017). The digital twin: Realizing the cyber-physical production system for Industry 4.0. *Procedia CIRP*, 61, 335–340. <https://doi.org/10.1016/j.procir.2016.11.152>
- [29] Cimino, C., Negri, E., & Fumagalli, L. (2019). Review of digital twin applications in manufacturing. *Computers in Industry*, 113, 103130. <https://doi.org/10.1016/j.compind.2019.103130>
- [30] Qi, Q., & Tao, F. (2018). Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. *IEEE Access*, 6, 3585–3603. <https://doi.org/10.1109/ACCESS.2018.2793265>
- [31] Lu, Y., Liu, C., Wang, K. I.-K., Huang, H., & Xu, X. (2020). Digital Twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robotics and Computer-Integrated Manufacturing*, 61, 101837. <https://doi.org/10.1016/j.rcim.2019.101837>
- [32] International Electrotechnical Commission. (2013). *Enterprise-control system integration : Part 1: Models and terminology* (IEC 62264-1:2013).
- [33] International Organization for Standardization. (2014). *Asset management : Overview, principles and terminology* (ISO 55000:2014).
- [34] Pandey, J. K., Veeraiah, V., Kumar, M. R., Kumari, P. L., Kotti, J., Dharani, M. K., & Gupta, A. (2024). Emerging soft computing and machine learning techniques applicable in industrial and automated applications. In N. Sindhwani, R. Anand, A. George, & D. Pandey (Eds.), *Robotics and automation in Industry 4.0* (pp. 65–88). CRC Press. <https://doi.org/10.1201/9781003317456>
- [35] Pandey, J. K., Kotti, J., Parimita, Dhabliya, D., Sharma, V., Choudhary, S., & Anand, R. (2024). Integration of nature-inspired mechanisms to machine learning in real time sensors, controllers, and actuators for industrial

- automation. In N. Sindhvani, R. Anand, A. George, & D. Pandey (Eds.), *Robotics and automation in Industry 4.0* (pp. 1–25). CRC Press. <https://doi.org/10.1201/9781003317456>
- [36] International Electrotechnical Commission. (2009). *Industrial communication networks : Network and system security : Part 1-1: Terminology, concepts and models* (IEC 62443-1-1:2009).
- [37] International Organization for Standardization. (2022). *Information technology : Artificial intelligence : Artificial intelligence concepts and terminology* (ISO/IEC 22989:2022).
- [38] International Electrotechnical Commission. (2020). *OPC unified architecture : Part 1: Overview and concepts* (IEC 62541-1:2020).
- [39] Negri, E., Fumagalli, L., & Macchi, M. (2017). A review of the roles of Digital Twin in CPS-based production systems. *Procedia Manufacturing*, 11, 939–948. <https://doi.org/10.1016/j.promfg.2017.07.198>
- [40] Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital twin: Enabling technologies, challenges and open research. *IEEE Access*, 8, 108952–108971. <https://doi.org/10.1109/ACCESS.2020.2998358>
- [41] VanDerHorn, E., & Mahadevan, S. (2021). Digital Twin: Generalization, characterization and implementation. *Decision Support Systems*, 145, 113524. <https://doi.org/10.1016/j.dss.2021.113524>
- [42] Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., & Sui, F. (2018). Digital twin-driven product design, manufacturing and service with big data. *The International Journal of Advanced Manufacturing Technology*, 94(9–12), 3563–3576. <https://doi.org/10.1007/s00170-017-0233-1>
- [43] Wang, S., Wan, J., Zhang, D., Li, D., & Zhang, C. (2016). Towards smart factory for industry 4.0: A self-organized multi-agent system with big data based feedback and coordination. *Computer Networks*, 101, 158–168. <https://doi.org/10.1016/j.comnet.2015.12.017>