

The Cyber Digital Twin, and how a document becomes part of it

AUTHORS

Jim McKenney

Principal Investigator, Eigenia B.V.

PUBLISHED

Sep 2026

ABSTRACT

The Cyber Digital Twin is a computable model of an industrial environment. It holds a facility's designed and actual equipment, what is inside that equipment, the threats aimed at it, the people who run it, the events that change its risk and what a failure would cost, as one graph organised in eight layers. Most of what the twin knows about a facility arrives first as a document. This primer describes how a document becomes part of the model. The original is captured unchanged and fingerprinted as the record of note, converted to Markdown, held at a human review gate, classified and mined for entities and relationships, embedded for retrieval, and written into the graph against an ontology. The primer then covers how live operational evidence keeps the model current as a facility drifts from its drawings, how competing claims are scored for confidence and reconciled by a named reviewer, and how a semantic reasoning layer answers multi-hop questions by explicit traversal. Every answer the twin gives can be traced to the documents and records behind it.

cyber-digital-twin

document-ingestion

provenance

record-of-note

human-review

ontology

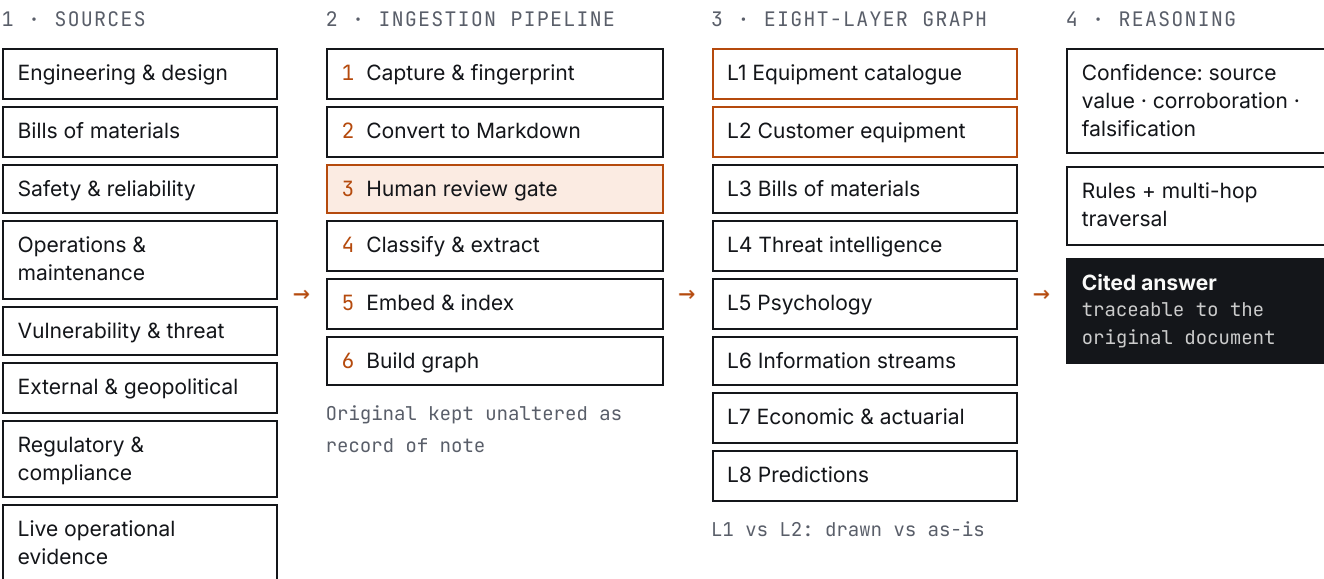
property-graph

vector-search

reconciliation

multi-hop-reasoning

GRAPHICAL ABSTRACT



Documents and feeds enter one pipeline, pass a human review gate, and become cited nodes and edges in an eight-layer graph. Reconciled facts and explicit rules produce answers that can be traced to their sources.

1 What the Cyber Digital Twin is

The Cyber Digital Twin is a computable model of an industrial environment. It holds the equipment a facility was designed to have, the equipment it actually has, the software and hardware inside that equipment, the threats aimed at it, the people who run it, the events in the wider world that change its risk, and what a failure would cost. These are held as one graph of nodes and relationships, so a question that crosses several of them can be answered by calculation rather than by assembling a meeting.

It is not a monitoring tool and it does not replace the systems of record a facility already runs. It reads from them, and from the engineering and compliance documents that describe the facility, and it keeps a traceable line from every element in the model back to the source that put it there.

How it works, in brief

Information enters the twin from many kinds of source: engineering drawings and models, bills of materials, electrical and network models, safety and reliability studies, operating and maintenance records, vulnerability and threat catalogues, geopolitical events, and the

compliance frameworks that apply in each geography and sector. The core engineering sources are expressed against open standards and joined on a common asset identity, so the same pump, controller or feeder is recognised as one object whichever source describes it.

Once joined, the graph can be traversed. Vulnerability, exploitation and threat data attach to the components they affect; dependency edges carry that exposure towards the functions that rely on it; and simulation over the graph produces a distribution of consequences rather than a list of findings. The result can always be explained, because every node and edge that contributed to it can be traced to the document or record it came from.

That traceability depends on how documents are taken in. Most of what the twin knows about a facility arrives first as a document, and most of this primer is about that process.

2 Eight layers, each answering one question

The graph is organised in eight layers. Each answers a distinct question about the facility, and edges between layers carry weights that are updated as new information arrives. The layers move at very different speeds: a newly published vulnerability changes the bill-of-materials layer within the hour, while the human layer shifts over months.

L1 Equipment catalogue	What should exist, according to the design and the reference model?
L2 Customer equipment	What actually exists, where, and in what condition?
L3 Bills of materials	What software, hardware and cryptography is inside, and how deep do its dependencies go?
L4 Threat intelligence	Who is actively trying to attack a configuration of this kind?
L5 Psychology	How will the people who run the facility behave, and misbehave?
L6 Information streams	What is happening in the world now that changes the risk picture?
L7 Economic & actuarial	What would a failure cost, and what is it rational to spend against it?
L8 Predictions	What happens next, with what probability, and what should be done about it?

Why the first two layers matter most

The first layer is the facility as drawn: the reference design, the vendor's specification, the world that was sold. The second is the facility as it is, including the undocumented change, the substituted part and the asset nobody recorded. Many serious incidents in hardened industrial environments have exploited the difference between the two. The twin treats that difference as an object it can compute with rather than an assumption that the drawing is accurate.

Both layers are built largely from documents. The rest of this primer describes how those documents are taken in, kept, checked and turned into graph structure.

3 Many sources, one model

A facility is described by far more than its drawings. How it was designed, how it is meant to fail safely, how it is operated and maintained, what threatens it, what is happening around it and which rules it must meet are all written down somewhere, usually by different teams in

different formats. The twin takes in all of them, so that the questions it answers can draw on every kind of evidence at once.

CATEGORY	TYPICAL SOURCES	LAYERS
Engineering & design	P&ID diagrams (DEXPI / IEC 61131 XML), CAD and BIM models, design specifications, interface control documents, single line diagrams, electrical and network models	L1 · L2
Bills of materials & logistics	Software, hardware, cryptographic and manufacturing bills of materials; bills of lading; firmware and software versions	L3
Safety & reliability	HAZOP studies, FMEA / FMECA analyses, RAMS reports, safety instrumented system documentation and safety requirement specifications, hazard logs, minimum operating requirements	L1 · L2 · L7
Operations & maintenance	Asset inventories, network topology maps, device configurations, operating manuals, maintenance plans, work orders, commissioning records, test suites, incident reports, training documents, emergency plans	L2 · L5
Vulnerability & threat	CVE, CWE, CPE and CAPEC catalogues; MITRE ATT&CK and EMB3D; threat intelligence reports; security alerts and bulletins	L3 · L4
External & geopolitical	Geopolitical events and external pressures, political climate and civil unrest, sanctions and supply disruption, sector and regional news	L6
Regulatory & compliance	Compliance frameworks by geography and vertical — for example the Cyber Resilience Act, NIS2 and IEC 62443 — together with sector rules and customer contractual requirements	L1–L3 · L7

Some of these arrive as files uploaded by a customer; others, such as vulnerability catalogues and threat bulletins, arrive as structured feeds that are refreshed continuously. The same principle applies to both: every item enters with its source and date recorded, and every element it contributes to the model points back to where it came from. The remainder of this primer follows a customer document through that process.

4 Document ingestion

The document as record of note

A conclusion drawn by the twin is only as defensible as the evidence behind it. When an auditor, a regulator or an engineer asks why the model believes a valve sits upstream of a controller, the answer has to be a specific page of a specific drawing, in the exact form it was received. The ingestion pipeline is built around that requirement. The original document is the record of note; everything derived from it — text, extracted entities, relationships, vectors and graph nodes — is treated as a working copy that must be able to point back to it.

The pipeline runs in a fixed order. Each step records when it completed, so the state of any document can be read at any time and the history of how it entered the model can be reconstructed afterwards.



FIG. 1 The ingestion pipeline. Each step records when it completed. Nothing is embedded or added to the graph until the review gate is passed.

Step 1 · Capture and fingerprint

When a document is uploaded, the original bytes are written unchanged to durable object storage, and the storage location is recorded. Nothing downstream ever edits this copy. At the same moment a provenance record is created in a relational ledger. The record carries a cryptographic fingerprint of the original file: any later copy can be checked against it, and any alteration, however small, produces a different fingerprint.

PROVENANCE RECORD HOLDS	WHY IT IS KEPT
Document code and name	A stable identifier that every derived item carries.
Original filename, type and size	What was received, exactly as received.
Fingerprint of the original	Proof the stored copy is the copy that was submitted.
Facility, organisation, client	Which part of which estate the document describes, and who owns it.
Stage timestamps	When it was uploaded, converted, embedded and added to the graph.

Step 2 · Conversion to Markdown

Documents arrive as PDFs, scans, spreadsheets, word-processor files and drawings. A document conversion engine turns each one into Markdown: plain text with its structure preserved — headings, lists, tables, captions and reading order. Markdown is compact, and it keeps the structure that tells a model which heading a paragraph sits under or which column a value belongs to. Working from it is faster and cheaper than working from the original format, and much less error-prone than working from raw extracted text.

Engineering diagrams need more than text extraction. A process and instrumentation diagram carries most of its meaning in symbols, connecting lines and tag labels rather than in sentences. Multimodal models that read images as well as text are used to describe these drawings in Markdown: the equipment shown, its tags, the lines that connect it, flow direction, and the instruments and controls attached. The output is a structured written account of the drawing that the later steps can process like any other text.

The converted file is stored separately from the original, with its own fingerprint and storage location, and the time of conversion is recorded. The ledger therefore holds two linked copies: the original, which is evidence, and the Markdown, which is the working text.

Step 3 · The human review gate

After conversion a routing rule is checked, based on the facility and the type of document. Where that combination has been approved for automatic processing, the document continues on its own. Everywhere else it stops and waits for review. Nothing is embedded or added to the graph until a person has approved it.

Review is where reconciliation happens. The reviewer compares the converted text against the original, particularly where a model has described a drawing, and checks the document against what the twin already believes about the same equipment. A drawing that disagrees with the recorded inventory is not an error to be discarded; it may be the first evidence of the gap between the first and second layers. The reviewer decides whether the difference is a conversion fault to correct, a superseded revision, or a genuine discrepancy in the facility to be carried into the model and flagged.

WHY THE GATE IS THE DEFAULT

Automation is opened only for document types and facilities where review has shown the output to be reliable. A new drawing standard, a new site or a poor-quality scan starts behind the gate, and moves past it on evidence rather than by assumption.

Step 4 · Classification, entities and relationships

An approved document passes through five processing stages, always in the same order.

STAGE	WHAT IT DOES
1 Classify	Assigns the document a type — drawing, datasheet, procedure, bill of materials, certificate — which decides how the later stages treat it.
2 Recognise entities	Finds the named things in the text: equipment tags, models, manufacturers, firmware versions, protocols, locations and people.
3 Extract relationships	Records how those things relate: feeds, controls, is powered by, contains, is installed at, depends on.
4 Embed	Turns the text into vectors so it can be searched by meaning, not only by keyword.
5 Build graph	Writes the entities and relationships into the twin as nodes and edges, classed against the ontology.

Step 5 · Embedding and indexing

The Markdown is divided into passages of roughly a thousand characters, and each passage keeps the heading it sits under, so a fragment of text always knows which section of which document it belongs to. Each passage is passed to an embedding model that represents its

meaning as a vector of 4,096 dimensions. The passages and their vectors are written to a vector store, each one carrying the document code, its position in the document and a link to the provenance record.

Writing is done as a single transaction. Any passages previously stored for the same document are removed and the new set is written in their place, so reprocessing a document never leaves stale fragments behind from an earlier, longer version. The ledger then records when embedding happened, how many passages were produced, and which model and vector size were used — so it is always known how a given set of vectors was made.

Search runs in two stages. A compact approximate index over compressed vectors finds a short list of likely matches quickly; those candidates are then scored exactly against the full-precision vectors. The first stage keeps search fast as the collection grows; the second keeps the ranking accurate.

Step 6 · Ontology and graph construction

Extracted entities are not written into the graph as free text. Each is classed against an ontology — a controlled vocabulary of equipment classes, component types and relationship kinds, aligned to the open standards the twin already uses for process schematics, bills of materials and electrical networks. The same pump described in a drawing, a datasheet and a maintenance procedure is therefore resolved to one node rather than three. Relationships are typed in the same way, so the graph can distinguish a pump that *feeds* a vessel from one that is merely *located near* it.

The graph is held in a property-graph store, where nodes and edges each carry their own attributes. Every node and edge created from a document carries a reference to the document and passage it came from. Calculations over the graph — dependency traversal, reachability, exposure propagation and consequence simulation — therefore run over structure that can always be cited.

5 Provenance, citation and change over time

Four properties of the pipeline together make the twin a source of truth rather than a summary of one.

PROPERTY	WHAT IT GUARANTEES
Provenance	Every derived item links to a ledger record, and every ledger record to an unaltered original, verified by fingerprint.
Citation	Any answer, node or calculated result can be traced to the passage and the original document behind it.
Reconciliation	Disagreement between a document and the model is resolved by a named reviewer, and recorded, before it reaches the graph.
Tracking over time	Stage timestamps and fingerprints show when each document entered the model and in what form. A revised drawing arrives with a new fingerprint, so a change to the facility is visible as a change to the evidence.

One pipeline, one set of rules

Documents enter through a single pipeline, whichever screen they are uploaded from. When a model improves or a batch needs reprocessing, affected documents are returned to a pending state and pass through the same steps again, including the same review gate. There is no second route into the graph, so there is no second standard of evidence.

6 Drift and live evidence

Why the facts change

A facility does not stay as it was drawn. Configurations are changed during maintenance, parts are substituted, systems are added and connections appear that no document records. This drift is continuous, and it is why the twin cannot be built from design documents alone. It also takes in live operational evidence that shows the facility as it is running now.

OPERATIONAL SOURCE	WHAT IT SHOWS
Directory attack-path mapping	Privilege relationships and the paths an attacker could take through identity infrastructure.
Network telemetry and packet captures	Which devices actually communicate, with whom, and over which protocols.
System, application and host logs	Events, access and behaviour on each system over time.
Active configurations	How devices are set up today, which may differ from how they were commissioned.
Vulnerability scans	Exposures observed on the running estate rather than inferred from documents.
Asset registers and as-built documentation	What was installed, where, and to which revision.
Maintenance records, work orders and change orders	What was altered, when, and under whose authority.
Onboarding of new systems and applications	What has been added since the last drawing was issued.

Working with existing sources of truth

Many customers already keep a system of record for their estate, such as a network source-of-truth platform holding devices, racks, cabling and IP addressing. Where they do, it stays the system of record and the twin integrates with it, as it does with other integration platforms. Whatever the route, every item passes through the same document processing pipeline: a provenance record, conversion where needed, vectors with reranked retrieval, the property graph and the reasoning layer described in §8. There is no side entrance to the model.

7 Reconciling the source of truth

When sources disagree

With this many sources, disagreement is normal. A drawing shows a valve that the maintenance log says was removed; a configuration file lists a firewall rule the network telemetry shows is not being enforced; two datasheets give different firmware versions

for the same controller. The twin has to decide what it believes, and it has to be able to show why.

Nothing is discarded. Every document and record stays in the provenance ledger, including the ones that turn out to be wrong or out of date. Competing claims are held side by side, each linked to its source, so the reasoning behind any conclusion can be reopened later.

How confidence is assigned

Each claim in the model carries a confidence rating built from three considerations.

FACTOR	WHAT IT MEASURES
Source value	How authoritative the source is for this kind of claim. A live configuration is strong evidence of how a device is set now; a design drawing is strong evidence of how it was meant to be set.
Corroboration	How many independent sources support the claim. A connection seen in packet captures, recorded in a change order and shown on an as-built drawing carries more weight than one that appears in a single document.
Falsification	Whether any credible source contradicts the claim. One strong contradiction can outweigh several weak agreements, and it is recorded against the claim rather than discarded.

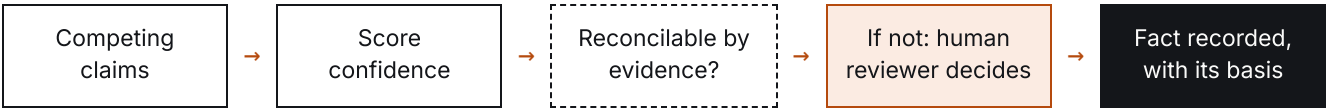


FIG. 2 Reconciliation. Where evidence settles the question, the best-supported claim becomes the working fact; otherwise a named reviewer decides.

Where the evidence settles the question, the claim with the strongest support becomes the working fact, with its confidence attached. Where variations cannot be reconciled by evidence, they are passed to a human reviewer, who makes the final decision on the facts of the matter. That decision is recorded with the reviewer, the date and the sources considered, so a later auditor sees not only what the twin believes but who decided it and on what basis.

8 Reasoning over the graph

Why similarity search is not enough

Retrieval that relies only on vector similarity finds passages that sound related to a question, but it often misreads context: two sentences can be close in wording and far apart in meaning. Simple graph extraction tries to fix this by pulling entity triples out of text, but that adds noise and carries none of the domain logic that says which connections are valid. Neither is adequate for questions an auditor or an engineer will rely on.

The twin therefore adds a semantic knowledge-graph reasoning layer on top of the property graph and the vector store. It brings structural and symbolic precision to the reasoning through four capabilities.

CAPABILITY	WHAT IT DOES
Formally programmed logic rules	Rules written in a domain-specific rule language define exactly how concepts and relationships depend on one another. Multi-hop reasoning follows those rules instead of relying on a language model to guess the connection, so the same question always produces the same answer.
Hybrid neural-symbolic reasoning	A complex question is turned into a structured logical form — in effect a graph query. The system executes it as explicit path traversal or calculation, then passes the verified facts to the language model to phrase the answer. The model is not asked to supply relationships, so it cannot invent ones that do not exist.
Concept standardisation and disambiguation	Entity linking and normalisation resolve different names for the same thing, and separate things that share a name, against the ontology. Reasoning then works from clear, unique definitions rather than loose clusters of similar text.
Multi-hop query execution	Many useful answers depend on four or five connected facts. A question such as whether a published vulnerability can affect a safety function has to pass from the vulnerability to the firmware, the controller, the equipment it drives and the function that depends on it. The layer traverses that chain explicitly and cites each source along the way.

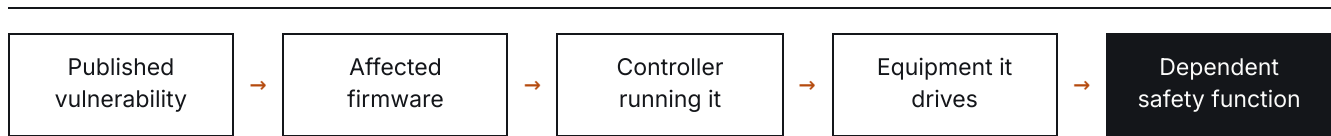


FIG. 3 A multi-hop question, traversed explicitly. Each hop is an explicit edge in the graph, and each carries a citation to the source it came from.

THE RESULT

Every answer the twin gives is built from reconciled facts, joined by explicit rules and traceable to the original documents and records. The language model writes the answer; the graph decides what is true.