LOOMLAB
Neuronetiq Ltd

CORE ARCHITECTURE

Electrical Digital Twin Architecture for Low-Voltage Systems

A Graph-Structured, Validation-Driven Systems Model

Published by LoomLab (Neuronetiq Ltd)
Classification: Public / Defensive Publication
Version: 1.0
February 2026

EXECUTIVE SUMMARY

Low-voltage electrical systems in marine, motorsport, off-grid, and industrial environments are increasingly complex, safety-critical, and lifecycle-sensitive.

SCOPE OF APPLICABILITY

This publication describes an architectural framework for modeling and validating low-voltage electrical systems, with optional accommodation for mixed AC/DC boundary conditions. It does not constitute engineering advice, regulatory guidance, or certification documentation. Application of this architecture to real-world systems must be performed by qualified professionals in accordance with applicable codes and standards.

COMPLIANCE AND SAFETY NOTICE

This publication is provided for informational purposes only and does not provide engineering advice, installation instructions, or certification evidence. Electrical failures in safety-critical systems can create hazards. Any implementation must be independently reviewed, tested, and validated using appropriate engineering practices.

This publication is provided "as is" without warranties of any kind. LoomLab (Neuronetiq Ltd) disclaims liability for any damages arising from use or reliance on this publication.

No representation is made that use of this architecture ensures safety, regulatory compliance, or fitness for a particular purpose. Readers shall not rely on this document as a substitute for independent engineering analysis, regulatory consultation, or professional design review. Use of the architectural concepts described herein remains at the sole risk of the implementer.

PROBLEM STATEMENT

Low-voltage electrical systems in marine, motorsport, off-grid, and industrial environments are increasingly complex, safety-critical, and lifecycle-sensitive.

Traditional design tools treat schematic capture, harness routing, validation, manufacturing documentation, and operational telemetry as separate workflows using separate data. This fragmentation introduces compounding risk:

- Design drift between schematic and physical harness
- Validation inconsistency across revisions
- Manufacturing errors from stale documentation
- Lifecycle traceability gaps between as-designed and as-built
- Undetected voltage and thermal risk from incremental system changes
- Version misalignment across engineering, production, and field teams

This paper introduces a structured Electrical Digital Twin Architecture built on a versioned, graph-based systems model. The architecture unifies:

- Electrical topology representation
- Engineering constraint validation
- Manufacturing metadata binding
- Operational telemetry integration
- Lifecycle revision history

The objective is not to replace CAD or schematic tools, but to establish a single source of electrical system truth that persists across the entire system lifecycle.

1. THE PROBLEM: FRAGMENTED ELECTRICAL SYSTEM MODELING

Most low-voltage systems today are modeled across multiple disconnected layers. Each layer uses its own tooling, its own data format, and its own versioning:

Domain	Typical Tooling	Fundamental Limitation
Schematic	ECAD tools	Net-centric; limited lifecycle context or physical constraints
Harness	CAD routing software	Geometry without validation intelligence
Manufacturing	Spreadsheets, ERP	Disconnected from topology logic; manual synchronization required
Validation	Manual calculation, isolated rules engines	Context-local; cannot evaluate system-wide interactions
Operation	Telemetry platforms, monitoring dashboards	No link to original design topology or validation state

These systems lack a persistent architectural model tying them together.

The result: engineering teams operate on representations of the system, not the system itself.

When a change is made in one layer, it does not automatically propagate to the others. The schematic says one thing. The harness drawing says another. The BOM says a third. The installed system may match none of them.

This is not an edge case. It is the normal state of electrical system engineering in most organizations.

1.1 The Cost of Fragmentation

Fragmentation produces measurable consequences:

- Warranty failures from undocumented modifications
- Field retrofits that violate original design constraints
- Compliance gaps when standards change but documentation does not
- Engineering hours spent reconciling divergent documents
- Safety incidents from accumulated, untracked system drift

These costs are not hypothetical. They are structural outcomes of disconnected tooling.

2. ARCHITECTURAL PRINCIPLE: THE ELECTRICAL SYSTEM AS A VERSIONED GRAPH

At the core of this architecture is a formalized graph model that represents the electrical system as a first-class engineering artifact.

2.1 Graph Primitives

- Nodes represent electrical entities: components, connectors, terminals, power sources, loads, distribution points, and grounding points.
- Edges represent conductive relationships: wires, bus segments, bundle members, and ground return paths.
- Bundles represent grouped conductors that share physical routing and are subject to collective thermal and fill constraints.
- Attributes encode electrical properties: conductor gauge, insulation class, base ampacity parameters (subject to derating factors), material composition, length, and environmental exposure classification.

2.2 Model Characteristics

This graph model is:

Deterministic Given the same inputs within declared model inputs and constraint definitions, it produces the same outputs. No hidden state. No probabilistic behavior.

Versioned Every revision is a discrete, identifiable state. Changes produce new versions, not mutations.

Constraint-aware Engineering rules are evaluated against the graph structure, not against isolated values.

Domain-agnostic The core representation is not specific to marine, motorsport, or any single application domain. Domain specificity is applied through constraint profiles, not structural modification.

2.3 Beyond the Netlist

Unlike traditional netlists, which record only connectivity, a graph-based electrical model:

- Preserves topology semantics: the structure encodes meaning, not just connection.
- Supports dependency-aware re-evaluation: a change to one element can trigger re-evaluation of dependent constraints across the graph.
- Maintains explicit lifecycle versioning: the model records what it was, not just what it is.

This distinction is critical. A netlist answers "what is connected to what." A graph-based systems model answers "what is the engineering state of this system, and is it valid?"

3. SEPARATION OF CONCERNS: REPRESENTATION VS. VALIDATION

The architecture intentionally separates three functional layers:

3.1 Layer A: Structural Layer

The canonical topology of the system. This layer defines:

- What exists (components, conductors, connectors)
- How it is connected (edges, routing, grouping)
- Physical attributes (gauge, length, material)

This is the persistent system of record.

3.2 Layer B: Constraint Layer

Engineering rules and constraints that may be used to evaluate system validity, including:

- Voltage drop limits per circuit classification
- Ampacity thresholds per conductor and bundle
- Continuous load derating rules
- Temperature derating factors
- Bundle fill and thermal constraints
- Fuse and protection device coordination
- Ground path impedance limits

Constraints are expressed as evaluable rules, not embedded assumptions.

3.3 Layer C: Context Profiles

Domain-specific rule overlays that adapt constraint evaluation to the operating environment:

- Marine: ABYC E-11 aligned rules, salt atmosphere derating, circuit classification (critical vs. non-critical)

- Motorsport: high-vibration derating, signal integrity constraints, weight-optimized conductor selection
- Off-grid: solar charge integration, battery management constraints, load prioritization rules
- Industrial: NEC aligned rules, conduit fill calculations, three-phase balancing

3.4 Why Separation Matters

This three-layer separation enables:

Deterministic rule evaluation Constraints are explicit and testable. No implicit assumptions buried in drawing conventions.

Profile-based validation The same physical system can be validated against different regulatory or domain contexts without modifying the structural model.

Reproducible results Given the same topology and the same constraint profile, validation produces identical results across time, users, and environments.

The graph is not merely a connectivity diagram. It is the substrate for constraint resolution.

4. DETERMINISTIC VALIDATION PROPAGATION

Electrical failures rarely occur in isolation.

A single change in conductor length or gauge can propagate:

- Voltage sag at a distant load
- Fuse oversizing that masks a fault condition
- Thermal overload in a bundled harness segment
- Bundle fill violations that exceed physical routing capacity

4.1 System-Level Evaluation

The architecture supports system-level constraint evaluation where:

- A change to any graph element triggers re-evaluation of all dependent constraints.
- Constraints are evaluated in a deterministic and reproducible manner based on declared relationships within the model.
- Validation results are bound to the specific graph revision that produced them.

4.2 Validation Snapshots

Each graph revision produces a complete, reproducible validation state. This state records:

- Which constraints were evaluated
- Which passed, which failed, which were deferred
- The specific graph revision and constraint profile used

This enables:

Auditability Any historical validation state can be inspected.

Change tracking The validation impact of any modification is explicitly recorded.

Regression detection If a change degrades system validity, the degradation is immediately visible.

4.3 Validation Scope

Validation results are dependent on the completeness and accuracy of the input topology, selected constraint profile, and environmental assumptions. The architecture does not guarantee safety or compliance; it provides deterministic evaluation of declared constraints against declared system state. No representation is made that use of this architecture ensures conformance with any regulatory or safety standard without independent professional verification.

4.4 The Spreadsheet Failure Mode

Traditional spreadsheet-based validation is typically circuit-local and difficult to maintain as topology grows. In practice it often fails to capture cumulative effects such as:

- Cumulative voltage drop across shared conductors
- Thermal interaction between bundled circuits
- Protection device coordination across parallel paths
- Ground impedance rise from shared return conductors

System-level validation addresses these structural blind spots.

5. DIGITAL TWIN BINDING: FROM DESIGN TO OPERATION

A digital twin is not a 3D rendering. It is not a simulation model.

A digital twin is the persistent binding of:

Design topology	What was intended
+ Validation state	What was verified

+ Manufacturing intent What was built + Operational telemetry What is happening

5.1 Telemetry Binding

By mapping telemetry signals to graph entities:

- Load drift can be detected: actual current exceeds design intent.
- Undervoltage can be predicted: voltage margins are eroding.
- Fuse behavior can be validated: protection devices are correctly sized for actual operating conditions.
- Real-world conditions can be compared against design assumptions: ambient temperature, duty cycle, peak demand.

5.2 Telemetry Scope

Telemetry integration is read-only and observational. Operational data does not retroactively alter the design record. The architecture does not perform autonomous control or system actuation.

5.3 Operational Overlay

The architecture supports read-only operational overlays:

- Telemetry data is mapped to graph nodes and edges.
- Validation constraints can be re-evaluated against live conditions.
- The design model is never mutated by operational data.

This preserves deterministic design integrity while enabling operational awareness.

5.3 The Value Proposition

The operational twin answers questions that no static document can:

- "Is this system still operating within its design envelope?"
- "Has load growth exceeded the original validation margins?"
- "Which circuits are approaching their thermal limits?"
- "What is the actual voltage at this load under peak demand?"

6. LIFECYCLE AND VERSION CONTROL

Electrical systems are not static. They evolve through:

- Initial design
- Manufacturing
- Installation
- Commissioning
- Modification and retrofit
- Operational life
- Decommissioning

6.1 Immutable Revisions

Each graph revision:

- Is immutable once validated. A validated state cannot be retroactively modified.
- Carries a unique version identifier.
- Maintains change lineage: what changed, when, and why.
- Allows structural diff comparison between revisions.

6.2 Lifecycle Questions

Version-controlled graph modeling allows teams to answer:

- What changed between revision 3 and revision 7?
- Why did the validation status of circuit X change?
- Which revision was used for manufacturing documentation?
- Which revision is currently installed in the field?
- Has the installed system diverged from the validated design?

6.3 Native Traceability

Version control ensures traceability of declared revisions. It does not guarantee that physical installations match recorded topology unless verified by appropriate inspection or audit procedures.

Lifecycle traceability becomes native to the system model, not an afterthought managed in external document control systems.

This is particularly critical in:

- Safety-critical marine installations
- Regulated industrial environments
- High-performance motorsport where configuration changes between events must be tracked precisely

7. STANDARDS ALIGNMENT WITHOUT HARDCODING

The architecture does not embed regulatory text. It does not claim to be a compliance tool. It does not replace engineering judgment.

7.1 What It Does

It supports:

- Constraint profiles aligned with recognized standards (ABYC E-11, NEC, SAE, ISO, IEC)
- Rule parameterization: standards define parameters, the constraint engine evaluates them
- Context-aware evaluation: rules vary by circuit classification, environment, and application domain

7.2 Standards as Constraint Profiles

Each standard is represented as a constraint profile:

- A named set of rules with specific parameters
- Applied to the graph during validation

- Versioned independently from the graph topology

When a standard is updated, a new constraint profile version is created. Historical validations retain their original constraint profile, preserving audit integrity.

7.3 Compliance Boundary

Any "alignment" refers to configurable parameterization and rule intent, not certification or authoritative interpretation. Compliance responsibility remains with the system designer and applicable regulatory authority.

References to standards are provided for context; this publication does not reproduce or interpret standards text.

7.4 Jurisdictional Flexibility

A system installed in a US-flagged vessel and a UK-flagged vessel may require different constraint profiles applied to the same topology.

The architecture supports this without structural modification.

8. INTEROPERABILITY AND EXTENSIBILITY

The graph architecture is designed to coexist with, not replace, existing engineering workflows.

8.1 Integration Points

CAD import/export Structural topology can be derived from existing schematic data or exported for schematic generation.

Manufacturing BOM Bill of materials can be generated directly from the graph model with full traceability to the validated revision.

Rule engine extension New constraint types can be added without modifying the core graph structure.

Tool integration The model can serve as a backend for specialized visualization, analysis, or reporting tools.

Simulation overlay External simulation results can be mapped to graph entities for comparison against design constraints.

8.2 Non-Disruptive Adoption

The architecture does not require organizations to abandon current tooling. It provides a canonical layer that existing tools can read from and write to.

This is critical for adoption in established engineering organizations where tool migration is costly and disruptive.

9. WHAT THIS ARCHITECTURE IS NOT

To prevent mischaracterization:

- It is not a CAD replacement. It does not draw schematics or route harnesses.
- It is not a schematic capture tool. It does not replace ECAD for circuit design.
- It is not a grid-scale power analysis platform. It is focused on low-voltage DC and mixed AC/DC systems.
- It is not a closed simulation black box. Validation is deterministic, explicit, and auditable.
- It is not an AI-driven design tool. It is an engineering systems model with explicit rules.

It is a deterministic electrical system model focused on low-voltage, safety-critical applications where system integrity, traceability, and lifecycle management matter.

10. COMPETITIVE POSITIONING

Many existing tools focus primarily on drawing, routing, or documentation workflows:

- Drawing: creating schematic representations
- Routing: determining physical harness paths
- BOM generation: listing components and materials

The architecture described here emphasizes system-level validation and lifecycle binding as a distinct layer:

- System-level dependency-aware re-evaluation across topology
- Versioned, immutable validation states
- Telemetry-bound digital twin capability
- Deterministic lifecycle engineering
- Domain-adaptive constraint profiles
- Native lifecycle traceability

It does not compete with CAD tools. It provides a complementary engineering validation and lifecycle layer not typically native to drawing-centric or routing-centric tools.

10.1 Practical Adoption Path

The architecture can be implemented incrementally:

1. Begin with topology capture of existing DC distribution (connectors, wires, segments).
2. Apply voltage drop and protection constraints to the captured topology.
3. Layer bundle and thermal modeling once harness routing information exists.
4. Add manufacturing metadata binding for BOM and build documentation traceability.
5. Bind operational telemetry (read-only) to monitor drift between design intent and operational behavior.

11. INTELLECTUAL PROPERTY CONSIDERATIONS

Graph representations are widely used to model connectivity and relationships in electrical and networked systems. This document focuses on their application to lifecycle-bound validation and operational binding in low-voltage contexts.

11.1 Defensive Publication Notice

This whitepaper constitutes a defensive publication. The combination of graph-based topology modeling, deterministic constraint evaluation, profile-based rule layering, and lifecycle-bound validation constitutes prior art as of the publication date of this document, including but not limited to implementations in marine, motorsport, off-grid, and industrial low-voltage domains.

Specifically, this publication discloses the architectural concept of:

- Representing low-voltage electrical systems as versioned graph structures.
- Applying deterministic constraint evaluation across such graphs.
- Binding validation states to immutable graph revisions.
- Applying domain-specific constraint profiles without modifying core topology.
- Binding operational telemetry to graph entities for lifecycle evaluation.

This publication discloses the architectural integration of graph-based electrical topology representation, deterministic system-level constraint evaluation, immutable revision binding, domain-profile overlays, and telemetry-to-topology lifecycle correlation in low-voltage systems. Any claims attempting to patent these combinations or their lifecycle-bound integration are disclosed herein as prior art.

11.2 Proprietary Elements

Implementation details, computational methods, optimization techniques, data schemas, and proprietary validation algorithms are not disclosed. Specifically, this whitepaper omits:

- Internal schema definitions
- Constraint evaluation algorithm specifics
- Data model implementation details
- Evaluation order logic
- Optimization techniques
- Internal rule expression language

These elements remain proprietary to Neuronetiq Ltd.

No license, express or implied, is granted by this publication under any patent, trade secret, or other intellectual property rights of Neuronetiq Ltd.

The concepts described herein are illustrative and non-exhaustive. Additional architectural variations and implementations are possible within the disclosed framework.

12. CONCLUSION

Low-voltage electrical systems are increasingly complex and safety-critical. The gap between design intent and operational reality grows with every untracked modification, every disconnected spreadsheet, every undocumented retrofit.

A unified, versioned, graph-based electrical digital twin architecture provides:

Structural clarity One canonical representation of the system.

Deterministic validation Explicit, reproducible, auditable constraint evaluation across the full topology.

Lifecycle traceability Every revision tracked. Every change recorded. Every validation state preserved.

Operational insight Design intent compared against operational reality through telemetry binding.

Without replacing existing engineering tools, this architecture establishes a canonical system model that spans design through operation.

This represents a fundamental shift:

From document-based electrical engineering To system-model-based electrical engineering.

The system of record is no longer a drawing. It is the model.

REFERENCES

[1] ABYC E-11, AC and DC Electrical Systems on Boats, American Boat & Yacht Council.

[2] NFPA 70, National Electrical Code (NEC), National Fire Protection Association.

[3] SAE J1128, Low Voltage Primary Cable, Society of Automotive Engineers.

[4] IEC 60092, Electrical Installations in Ships, International Electrotechnical Commission.

[5] ISO 13297, Small Craft - Electrical Systems, International Organization for Standardization.

References are provided for context; this publication does not reproduce or interpret standards text.

PUBLICATION METADATA

Document classification: Public / Defensive Publication

Published by LoomLab (Neuronetiq Ltd).

This document constitutes a public disclosure of architectural concepts for graph-based electrical system modeling and constraint validation in low-voltage systems.

Implementation details, algorithms, data models, and internal methods remain proprietary.

For inquiries: contact@loomlab.ai

LOOMLAB
Neuronetiq Ltd

About LoomLab

LoomLab is a structured electrical engineering platform built on a graph-based digital twin architecture. It provides system-level validation, lifecycle management, and operational awareness for low-voltage electrical systems across marine, motorsport, offshore, and industrial domains.

Website: www.loomlab.ai
Contact: contact@loomlab.ai

Published by LoomLab (Neuronetiq Ltd).

This document constitutes a public disclosure of architectural concepts. Implementation details, algorithms, data models, and internal methods remain proprietary. No license, express or implied, is granted by this publication under any patent, trade secret, or other intellectual property rights of Neuronetiq Ltd.