



LARGE LOAD & DATA CENTER SERIES • EC-WP-1004

Operating a Registered Computational Load

Controls, Governance, Evidence, and Audit Defense for the New Registration

The compliance operating model for a registered computational load: the program architecture, internal controls, governance, the evidence infrastructure that withstands audit, and the disciplines that convert obligations into demonstrable compliance.

Book Five of the Energy Compliance Large Load & Data Center Series

The Operating Layer

Disclaimer & Legal Notice

This publication is the work product of Energy Compliance, Inc. and is provided for general information and educational purposes only. It does not constitute legal advice, regulatory guidance, or formal interpretation of any NERC Reliability Standard, FERC order, Department of Energy directive, or state commission rule.

This publication may not be used or cited as authority in any audit, enforcement, regulatory, or legal proceeding. The integration of large loads and data centers into the bulk power system is a domain of active rulemaking, accelerated standard development, and ongoing state and federal proceedings. Every factual assertion in this whitepaper was drawn from primary regulatory sources or authoritative public filings current as of the date of writing; readers should confirm the current status of every authority cited here before acting on it. Compliance with NERC Reliability Standards and applicable state and federal rules depends on facts and circumstances unique to each registered entity, and the reader should consult qualified counsel and subject-matter experts when making compliance decisions.

This edition updates the June 2026 original. Statements about NERC procedure, the computational load registration framework, and the Project 2026-02 standards were checked against primary sources available on October 17, 2026.

Where a matter is proposed but not yet final at the time of writing, such as a registry criterion awaiting FERC approval or a standard awaiting Board adoption, it is described as such. Energy Compliance, Inc. does not represent that any proposed rule or pending order will be adopted in the form described, or at all.

NERC, FERC, the United States Department of Energy, the Public Utility Commission of Texas, ERCOT, PJM, and the names of Regional Entities are referenced here for descriptive purposes only. Energy Compliance, Inc. is not affiliated with, sponsored by, or endorsed by the North American Electric Reliability Corporation, the Federal Energy Regulatory Commission, the Department of Energy, the Public Utility Commission of Texas, ERCOT, PJM Interconnection, L.L.C., or any NERC Regional Entity.

Energy Compliance, Inc. assumes no responsibility or liability for errors, omissions, or outcomes resulting from the use of this publication. Citations to regulatory text are summarized and paraphrased; readers requiring the operative language should consult the source document directly.

Foreword

This professional reference is one of a series Energy Compliance, Inc. publishes for registered entities and the people who run their compliance programs.

I have spent more than thirty years on every side of the bulk electric system. I have operated control centers as a Reliability Coordinator, Transmission Operator, and Power System Operator. I have audited grid facilities and signed off on findings as a senior compliance auditor. I have worked enforcement matters from inside the regulator's process. For the last several years I have advised registered entities directly through the firm I founded.

The entities that handle reliability well share a common habit. They take the standards seriously without confusing them with reliability itself. They know that a NERC Reliability Standard is a floor, not a ceiling. They know that compliance is what an auditor evaluates, but reliability is what the system either delivers or does not. They prepare for audits by building programs that survive real questions, not binders that look thick.

Registration is the start of the work. Once a site is registered as a Computational Load Owner or Computational Load Operator, the question stops being whether obligations exist. They do. The question becomes whether the program can demonstrate compliance to a regulator under pressure. The generation side of the house already knows the disciplines that produce that demonstration: internal controls, governance, evidence architecture, and operating habits that make every requirement an output of

the normal program instead of a one-off response to an audit notice. This fifth book in the series translates that body of practice into the shape it has to take for a registered computational load.

These references are written for the compliance manager who wants to understand the system, not just memorize requirements. For the legal counsel who has to brief a board honestly about what the next eighteen months of large-load regulation will require. And for the data center owner or engineering partner who has been told their facility may now be a registered entity, and wants to know what that actually means.

They come from three decades of doing this work and watching it succeed and fail. I've written them in the same voice I use in a control room or in front of a Regional Entity audit team. Direct. Tied to evidence. Honest about what the standards do and do not require.

Energy Compliance, Inc. exists because most of the consulting offered to registered entities today is structured for billable hours rather than for outcomes. We do not bring five people to a meeting that needs one. We automate what should be automated and apply senior judgment to the rest. If that is what you are looking for in a compliance partner, the back of this reference tells you how to reach us.

If not, the reference still belongs to you. Take what is useful. Apply it well. And remember the only test that ultimately matters: when the system needs to perform, does it?

Rob Smith

Founder and Principal, Energy Compliance, Inc., Bloomington, Minnesota

How to Use This Reference

This whitepaper is the fifth book of the Energy Compliance Large Load and Data Center Series. The four earlier books described the grid problem, the standards trajectory, the registration mechanics, and the owner's playbook. This book is for the entity that has been registered, or is preparing to be, and now has to run the program that demonstrates ongoing compliance. The October 2026 update, EC-WP-1005, documents the registration framework and standards as they now stand, and it governs wherever an earlier book describes the superseded proposal.

The grid layer (EC-WP-1000)

What the demand shock is, why the existing planning and interconnection frameworks cannot absorb it, and how the federal and state interconnection regimes are being rebuilt in response.

The standards layer (EC-WP-1001)

NERC's Large Loads Action Plan, the Level 2 and Level 3 Alerts, the technical white papers, the reliability guideline, and the mandatory standards then under development.

The entity layer (EC-WP-1002)

The April 2026 Computational Load Entity proposal, the registration test it used, and what becoming a NERC-registered functional entity actually requires. NERC has since replaced that proposal; read EC-WP-1005 for the current framework.

The owner layer (EC-WP-1003)

The data center owner's playbook under Texas Senate Bill 6: the interconnection trigger, the study-fee and site-control regime, the curtailment bargain, and the standalone-versus-co-located choice.

The operating layer (EC-WP-1004)

The compliance operating model for a registered computational load: internal controls, governance, evidence infrastructure, and the day-to-day disciplines that convert obligations into demonstrable compliance.

The 2026 update (EC-WP-1005)

The Computational Load Owner and Computational Load Operator functions, the 50 MW and 100 kV site threshold, what CLO-001-1, CLO-002-1 and CLO-003-1 require, and the compliance calendar.

Every factual assertion in this reference is grounded in a primary source identified in the Sources and Authorities appendix. Where a regulatory matter was proposed or pending at the time of writing, it is described as such. Energy regulation in this area moves on monthly timescales, and the status of a docket or a NERC project when you read this may differ from its status when it was written.

Executive Summary

Registration is the starting line, not the finish. The earlier books in this series explained why large computational loads are being brought into the reliability framework, how NERC is building the obligations, who must register, and what a Texas owner must weigh. This book takes up what happens after the registration determination: the work of operating, day in and day out, as a NERC-registered entity. That work is a program. It is a standing system of controls, evidence, and disciplined practice, not a one-time filing, and the organizations that treat it as a program are the ones that pass audits and come through enforcement intact.

The framework this book serves has firmed up since June 2026. On October 8, 2026, the NERC Board approved Rules of Procedure revisions creating two functional entities, the Computational Load Owner and the Computational Load Operator, and their registry criteria. The three Project 2026-02 standards, CLO-001-1, CLO-002-1 and CLO-003-1, passed their final ballot in October and are headed to the Board and then FERC. NERC's implementation plans expect registration to begin in 2027. None of it is in effect yet. EC-WP-1005 covers the details. This book covers how to run the program once those obligations land.

For a data center operator that has never carried reliability obligations, this is unfamiliar territory. The instinct of a technology business is to move fast, document lightly, and fix problems quietly. Inside the Compliance Monitoring and Enforcement Program that instinct is wrong. An undocumented control cannot be demonstrated to an auditor, and an attempt to conceal a violation is an aggravating factor under NERC's Sanction Guidelines. This book translates the established discipline of NERC compliance into the context of a registered computational load, and it does so the way EC works: analyst first, anchored in evidence, built to stand up to a regulator.

Five findings frame the analysis

Compliance is a control system, not a document. The durable foundation is a framework of internal controls (defined ownership, repeatable processes, and self-monitoring) that makes compliance the normal output of how the organization operates, rather than a scramble before each audit. The CMEP uses internal controls to tailor audit scope, and the Sanction Guidelines treat the presence and quality of an internal compliance program as a mitigating factor. In my experience a mature control environment is the best single predictor of a clean audit.

Evidence is the currency, and it is built continuously. A requirement is satisfied in an audit only if the entity can produce evidence that demonstrates it: the document, the log, the record, the dated artifact. Evidence cannot be manufactured the week before an audit. It has to be

generated and retained as the work happens, which makes the program's evidence infrastructure as important as the technical work it documents.

The technical obligations are now written down. CLO-001-1, CLO-002-1 and CLO-003-1 spell out what a Computational Load Owner and Operator will have to do: provide and verify modeling data, satisfy operators' data specifications, follow Operating Instructions, maintain Interpersonal Communication and trained personnel, record disturbances to stated specifications, and coordinate protection settings. They passed final ballot in October 2026 but are not yet adopted or approved. An entity can start building these practices, and the evidence that proves them, now.

The CMEP lifecycle is continuous and varied. Compliance is monitored through audits, self-certifications, spot checks, investigations, self-reports, self-logging, periodic data submittals, and complaints, and noncompliance is resolved through mitigation. A registered entity has to be ready for any of these at any time, and that is only possible if the program runs continuously rather than episodically.

Audit defense is built into the work, not bolted on at the end. The RSAW, the worksheet through which compliance with a standard is demonstrated, is strongest when each requirement gets a direct narrative, specific evidence, and the explicit logic connecting the two. A program that produces audit-ready narratives as a byproduct of ordinary operation, and that pressure-tests itself through mock audits, defends itself.

The practical message: a registered computational load should stand up its compliance program with the same seriousness it applies to its core technology. The program in this book is designed to be implemented once and operated across every site an organization owns or runs, and it is the operational answer to the obligations the rest of the series traces. This is where the abstract becomes routine, and where a new class of registered entity learns to demonstrate, every day and on demand, that it is reliable.

Chapter 1. The Compliance Operating Model

Becoming a registered entity changes what kind of organization you are. A computational load that registers with NERC does not simply pick up a checklist of new tasks. It takes on a standing obligation to operate reliably and to show a regulator, on demand, that it does. Meeting that obligation takes an operating model, not a project: a way of running the organization in which compliance is a normal output instead of something assembled under deadline pressure.

1.1 From project to program

The most common and most costly mistake a newly registered entity makes is to treat compliance as a series of episodic projects. Prepare for the audit, file the self-certification, respond to the data request, each handled when its deadline arrives. That posture fails because the obligations are continuous and they overlap. An audit examines the whole period since the prior audit, or since the entity first became subject to the standards. NERC encourages a Self-Report at the time an entity becomes aware it has, or may have, violated a standard, so the clock on that decision starts the day the issue is found. And evidence has to exist already for work performed long before any deadline. A program runs continuously. It assigns ownership, executes repeatable processes on a defined cadence, generates and retains evidence as the work happens, and monitors itself for problems. The difference between a project and a program is the difference between passing one audit and being reliably compliant.

The shift from project to program is what most separates a first-time registered entity from a long-registered one. A project has a beginning, a middle, and an end. A program has cycles but no end. A facility that treats registration as a project will mobilize teams, produce deliverables, hit milestones, and demobilize. Then the next audit cycle begins and the program needed to support the registration is gone. The entities that succeed make the program durable from the start: ongoing ownership, recurring rhythms, evidence pipelines that produce artifacts as a byproduct of normal operations, and a posture that treats compliance as part of how the facility runs.

1.2 The three layers of the operating model

A sound compliance operating model has three layers that reinforce one another. The first is governance and internal controls, the framework of ownership, policies, and self-monitoring that makes compliance repeatable (Chapter 2). The second is the evidence infrastructure, the systems and discipline through which the entity captures and retains the records that demonstrate compliance (Chapter 3). The third is technical execution, the substantive reliability practices specific to a computational load (Chapter 4). The Compliance Monitoring and Enforcement Program, covered in Chapters 5 and 6, then tests all three. An entity that builds

only the technical layer will do the right things and be unable to prove it. In an audit, that looks the same as not having done them.

The model also has to reach every level of the organization, and each level runs on its own clock. Executive ownership thinks in quarters and years. The compliance function that runs the program thinks in months and quarters. The floor, where events are detected, handled, and documented, thinks in shifts and days. A program built only on the floor can handle today's deadlines but cannot defend a structural pattern in front of an auditor. A program built only in the executive suite can describe its intent but cannot show that intent ever reached the floor. Design the levels together. The seams between them are where compliance programs fail most often.

1.3 Accountability and culture

Underneath the structure sits culture and accountability. A program works only if someone owns it: a defined compliance function with the authority and resources to run it, and each obligation assigned to a responsible role. It also works only if the operating culture supports it. Engineers document as they work. Problems get surfaced and self-reported, not hidden. Leadership treats compliance as part of operating the facility, not as overhead. For a computational-load operator coming from a technology-sector culture, this is the hardest and most important adjustment. The enforcement framework credits self-reporting and treats concealment as an aggravating factor, and an organization that internalizes that early avoids the most damaging category of enforcement exposure.

Compliance culture is built through repetition. The behaviors that make a program work (documenting decisions, retaining evidence, escalating concerns, treating obligations as constraints on action rather than suggestions) are reinforced by the organization's daily rhythms. Programs that make those behaviors part of routine work build a compliance culture without ever calling it that. Programs that treat compliance as a separate workstream run by a separate team build a culture of compliance avoidance, even when they intend the opposite. The cultural work cannot be separated from the operational work, and the senior leaders who own one have to own the other.

This is the layer at which the program either becomes durable or stays fragile. A program that depends on the attention of one or two key people is fragile by construction. It survives only as long as they stay in their roles, and it starts to decay the moment they leave. A program built on documented procedures, named owners, scheduled cadences, and an evidence pipeline that runs on its own is durable by construction. It survives personnel changes because the work has been institutionalized rather than personalized. Making that transition is the work of the first registration cycle, and entities that finish it early end up with a program that ages well.

Chapter 2. Internal Controls and Governance

Strong internal controls are the foundation everything else rests on. Before evidence, before technical practice, before any audit, comes the control framework: the structure that makes compliance the predictable result of how the organization operates. Regulators look hard at internal controls. The CMEP names them as an input when a Compliance Enforcement Authority tailors audit scope, and the Sanction Guidelines weigh the presence and quality of an internal compliance program when setting a penalty. An entity that can demonstrate mature controls is treated very differently from one that cannot.

2.1 What an internal control is

An internal control, in the compliance sense, is a defined, repeatable mechanism that makes sure a requirement is met and surfaces a problem if it is not. It has the same anatomy regardless of the obligation it supports: a responsible owner, a defined process run on a stated cadence, a record produced as evidence, and a way of detecting and escalating failure. A control is not a policy statement that something will be done. It is the mechanism that does it and leaves proof. Measure a program's strength by how much of its compliance comes from controls operating routinely, and how little depends on individuals remembering to act.

A well-designed control has four properties. It identifies a specific risk. It specifies an action that mitigates the risk. It produces an evidence artifact when it operates. And it has a named owner accountable for its operation. Miss any one and there is a gap somewhere: the risk is unaddressed, the action is undefined, the evidence is missing, or no one is accountable. Designing controls that meet all four conditions is what separates a compliance program from a compliance posture, and every other capability in the program is built on that discipline.

Borrowing control frameworks from other compliance regimes is tempting and often counterproductive. CIP has the most mature control-design literature in the NERC family, but CIP controls were designed for cybersecurity risks: unauthorized access, malicious code, privilege escalation. A Computational Load Owner or Operator program addresses electrical and operational risks: disturbance behavior, model fidelity, protection coordination, and response to Operating Instructions. A CIP-style control imported into that program may produce paperwork without reducing risk, and the audit team will notice. Each regime deserves controls designed for its own risks.

2.2 Designing the control framework

A registered computational load should design its controls around the obligations it actually carries. For each applicable requirement, the entity should be able to answer four questions: who owns it, what process satisfies it, what evidence the process produces, and how a failure is

detected and corrected. Mapping every requirement to those four answers produces the control framework. The mapping is itself a valuable artifact, because it shows a regulator that the entity has systematically connected each obligation to a mechanism that meets it. Where a requirement has no mapped control, the mapping has surfaced a gap. That is what it is for.

In practice, designing the framework is a sequence of decisions about what to measure and how often. A good control identifies a specific risk, defines the evidence that shows the control is operating, and puts ownership at a level where the owner can actually see whether it works. Generic controls borrowed from a different program, particularly a CIP program built for a different threat model, are worse than none. They create the appearance of coverage while the real risk goes unaddressed. A defensible framework comes from spending more time on fewer controls, each tied to a specific operating reality.

2.3 Governance and ownership

Controls need an organizational home. A registered computational load should establish a defined compliance function, a person or team with the authority, independence, and resources to administer the program, and should assign each obligation to a named role so accountability is explicit. Governance also means a regular cadence of internal review: periodic self-assessment against the applicable requirements, management review of the program's health, and a documented process for raising and resolving issues. This layer lets an entity find its own problems before a regulator does. That is the difference between a Self-Report on the entity's terms and a finding on the regulator's.

Governance and ownership is where programs fail most often in the first year. In a computational-load facility staffed for operational performance rather than regulatory engagement, the temptation is to hand compliance to an existing role as an additional duty: facility manager, IT lead, engineering supervisor. That can work in a small organization where the duty is real, the time is allocated, and senior leadership backs it. It fails when the duty is nominal, the time never materializes, and the only support is the individual's own initiative. The programs that survive the first audit cycle have made compliance a real role, backed by senior leadership and shown on the organization chart.

2.4 Why regulators reward strong controls

The internal-controls posture pays off directly under the enforcement framework. Under the CMEP, a Compliance Enforcement Authority may approve an entity to self-log minimal-risk noncompliance only after a formal review of its internal controls. Under the Sanction Guidelines, an effective internal compliance program is a mitigating factor, and the Guidelines say plainly that failing to prevent or detect one violation does not by itself mean the program is ineffective. So when a violation does occur, an entity with strong, documented controls is positioned to show that the lapse was an exception to a sound program rather than evidence of a broken one. Building strong controls is the most effective investment an entity can make in its own enforcement posture, and it is the foundation the rest of this book builds on.

Strong controls also change the audit itself. The CMEP has the audit team tailor scope using risk-based inputs that include internal controls. In my experience, a team that sees a credible control framework treats it as a leading indicator of program maturity and examines accordingly.

The audit runs shorter, intrudes less, and is more productive for both sides. The investment in control design pays back in compliance outcomes and in the audit relationship the program operates inside.

The reverse holds too. A program without credible controls, or with controls that exist on paper but do not operate, forces the audit team to verify each requirement independently, dig through unstructured evidence, and probe for the underlying disciplines. The audit takes longer, the findings pile up, and the relationship that comes out of the cycle is more adversarial. Entities that go through two or three audits like that find their compliance costs rising over time even as the obligations stay the same.

2.5 Vendors, contractors, and change management

Vendor and contractor management is one of the most concrete operational disciplines the operating layer requires, and computational-load facilities lean on vendors more heavily than the generation entities whose practices most closely parallel the new registration. UPS and power-electronic equipment is vendor-specific. Cooling-system protection is vendor-specific. Network architecture, and the protection logic that ties the facility into the operational environment, is often a system integrator's work. Each of those relationships produces decisions that affect compliance, from protection settings to commissioning practices to documentation. The registered entity remains responsible for compliance no matter who did the underlying work. CLO-001-1 R11 makes the point sharply: the owner must verify model parameters for equipment, including the IT load, that it may not fully control. Contracts that include compliance-relevant scopes of work, change-management procedures, documentation deliverables, model and settings data rights, and the right to audit the vendor's work are foundational to a defensible program. Vendor management built around procurement defaults instead of compliance requirements is one of the most reliable sources of audit exposure in the operating layer.

Change management is where vendor relationships most often produce compliance exposure. A vendor that updates firmware, changes protection settings, or reconfigures the facility without the registered entity's knowledge has moved the compliance baseline without giving the entity a chance to assess the change. Under CLO-001-1 R11, a change that alters dynamic response starts a 180-day clock for new Model Verification documentation, and a qualified change can trigger a new study. The program needs controls that catch vendor-driven changes, evaluate them against the compliance baseline, and either approve or reverse them. Entities that build those controls with their vendors find them running quietly in the background. Entities that do not discover the gap during an audit, when the team asks how a particular configuration came to be.

Chapter 3. The Evidence Infrastructure

In compliance, if it is not documented, it did not happen. That principle surprises organizations new to the reliability framework more than any other. The entity may be doing everything right, maintaining its models, supporting studies, running its facility carefully, but in an audit what it can demonstrate is the only thing that counts. The CMEP is explicit: the entity's data and information must show compliance for the entire audit period. Evidence is the currency of compliance, and the evidence infrastructure matters as much as the work it records.

3.1 Evidence as a first-class output

The defining discipline of a mature program is that it treats evidence as a primary output of every compliance activity. When a model is verified, the verification produces a dated record of what was verified, by whom, against what, and with what result. When a study is supported, the data provided and the date it went out are captured. When a control operates, it leaves an artifact. Generating evidence as the work happens is the only reliable way to be ready for an audit that may, years later, examine the period when the work was done. Evidence reconstructed after the fact is weaker, sometimes unavailable, and always more expensive.

This is the discipline that makes a program audit-defensible. Evidence is what the auditor reviews to decide whether the program complied with a requirement. Programs that produce it as a byproduct of normal operations (automated logs from systems already in use, ticket records of changes already tracked, minutes from meetings already held) get reliable evidence at low marginal cost. Programs that produce evidence only when an audit is announced get unreliable evidence at high cost. The design choice that separates them is treating evidence as a deliverable the program produces continuously, not a presentation it assembles when asked.

3.2 What to retain, and for how long

Each NERC standard states its own evidence retention periods in its compliance section, and the three CLO standards illustrate how much they vary. CLO-001-1 calls for three years for some requirements and "since the last audit" for others. CLO-002-1 ranges from 90 calendar days for operational data delivery, to 90 days of voice recordings plus 12 months of documentation for Operating Instructions and communications, to two calendar years of documentation for training. CLO-003-1 calls for three calendar years for its recording requirements and "since the last audit" for protection settings. Each standard also says that where its retention period is shorter than the time since the last audit, the Compliance Enforcement Authority may ask for other evidence of compliance for the full period. The CMEP says the same thing: an entity will not be found in noncompliance solely because it rightfully discarded records under a standard's retention period, but it will still have to demonstrate compliance through other means.

That is why I treat the stated periods as a floor. An entity that keeps evidence only to the minimum will come up short on edge cases, where an event timeline crosses a retention boundary or one artifact supports requirements with different periods. Organize retention around the program's own audit cycle instead: everything since the most recent audit plus a buffer for the next, with structural artifacts (models, verification reports, settings and their basis, one-lines) kept as long as the facility operates. In a computational-load environment, storage cost is the least binding constraint imaginable. Structure is the binding constraint. Evidence that cannot be retrieved when needed does not exist for compliance purposes. The retention obligation is unforgiving: evidence discarded or never created cannot be recovered when an audit notice arrives.

3.3 Mapping evidence to requirements

Evidence is only useful if the entity can connect it to the requirement it satisfies. The most useful organizing tool in a compliance program is therefore an evidence map: a structured link from each requirement to the specific artifacts that demonstrate compliance with it. The map does three jobs. It proves coverage. It surfaces gaps where a requirement has no supporting evidence. And it makes audit response fast and credible, because the entity can produce the right artifact for each requirement without a frantic search. Building and maintaining the map is a core, continuous activity, not a pre-audit exercise. An entity that keeps its map current is always audit-ready.

The CLO standards make the mapping concrete, because each Measure names the kind of evidence that will satisfy it. CLO-001-1 M9 points to email records or postal receipts showing the recipient and date of a modeling data submission. CLO-002-1 M8 points to dated operator logs, voice recordings or transcripts, and electronic communications. CLO-003-1 M7 points to actual data recordings and dated transmittals. Read on that level, every requirement implies an evidentiary expectation: a record that shows the requirement was met, dated, attributable to an owner, and producible on request. A program with an explicit requirement-to-evidence map answers the auditor's question with one document reference. A program without one answers it with a search through email archives, ticketing systems, and shared drives, which produces findings even when the underlying compliance is sound.

Build the map as part of program design, not at audit preparation. A program designed with the map in mind produces evidence the map can trace. A program designed without it leaves the audit-preparation team to assemble evidence retroactively, and they will find gaps doing it. Keep the map alive, updating it as controls evolve, as standards change, and as the evidence pipeline matures. Entities that do this across audit cycles find that audit preparation becomes a routine validation exercise.

3.4 Avoiding the common evidence failures

Most evidence problems fall into a few recurring patterns, and naming them helps an entity avoid them. The first is the undocumented control: the entity does the work but keeps no record, so it cannot prove it. The second is the orphaned artifact: the entity has records but cannot connect them to specific requirements, so their value is lost. The third is the retention gap: evidence existed but was not kept long enough. The fourth is the reconstruction trap: evidence assembled after an audit notice, which auditors recognize and discount. A program built on

continuous, mapped, well-retained evidence avoids all four, and in doing so turns the entity's genuine reliability work into demonstrable compliance.

There is a second set of failures inside the evidence itself. The attribution gap: evidence exists but cannot be traced to a specific person, role, or system. The timing gap: it exists but cannot be tied to a date or period within the audit window. The chain-of-custody gap: its provenance cannot be shown, so the auditor cannot satisfy itself that the evidence is what the entity says it is. The completeness gap: some evidence exists, but the auditor cannot tell whether it is the full record or a selected subset. Each of these is preventable through control design. Each, when it occurs, weakens the audit defense.

Chapter 4. The Technical Compliance Obligations

When this book was first published in June 2026, the standards for computational loads were still being drafted, and the best guide to their content was NERC's Level 3 Essential Action Alert of May 4, 2026 and its Reliability Guideline on emerging large loads. That has changed. The Project 2026-02 standards, CLO-001-1, CLO-002-1 and CLO-003-1, passed their final ballot in October 2026. NERC's October 2026 project FAQ says it will seek Board approval in December and file with FERC by December 31, 2026, the deadline FERC set in its July 16, 2026 order in Docket No. RD26-7-000. The standards take effect only after regulatory approval, with compliance phased in over periods of three to thirty-six months. EC-WP-1005 walks through each requirement and the calendar. This chapter translates the obligations into the operating practices a registered computational load has to run, and the evidence each practice must produce.

One point about the alert. It was addressed to Transmission Planners, Planning Coordinators, Transmission Owners, Balancing Authorities, Reliability Coordinators and Transmission Operators. It did not apply directly to data centers. Its effect on a computational load came through the entities that interconnect and operate around it, and those entities are already pressing its essential actions down the chain through data requests and interconnection requirements.

4.1 Model maintenance and validation

At the center of the obligations is the model. A computational load has to be represented accurately in the models planners use to study the system. Under CLO-001-1 R9, the Computational Load Owner must provide steady-state, dynamics, and short circuit modeling data to its Transmission Planners and Planning Coordinators according to the requirements they develop under R8. Those requirements must cover cessation, tripping, ride-through, and post-disturbance recovery parameters, the amount of IT infrastructure and its expected use, and a submission schedule of at least once every 13 calendar months. Where nothing has changed, a written confirmation is enough. Under R10, the owner has 90 calendar days to answer a planner's written technical concern with updated data or a technical justification, unless a longer period is agreed. Under R11, the owner must provide Model Verification documentation showing the model's site-specific parameters match the in-service equipment: within 365 calendar days after the in-service date, within 180 calendar days after a hardware, software, firmware, or setting change that alters the dynamic response, and then at least every ten calendar years. Any parameter that cannot be verified must be identified, with the reason. Operationally, the entity has to know its facility's electrical characteristics precisely, keep a controlled record of the model and its versions, and treat any change that touches dynamic response as a trigger. The

evidence is the model record, the dated submissions, the verification documentation, and the change history.

4.2 Data provision and study support

A computational load has to be a reliable source of the data the system needs to plan and operate around it. The Level 3 Alert asked planners to collect expected minimum and maximum consumption, dynamic characteristics including UPS settings, the split between IT and non-IT load such as cooling at various loading levels, expected ramp rates, and protective device behavior. The standards turn that into obligations on the site. Under CLO-001-1 R5, a Computational Load Owner seeking to interconnect a new site, or making a qualified change as its Planning Coordinator defines that term under R7, must coordinate and cooperate on the interconnection studies, including by providing data. Under CLO-002-1 R7, each Computational Load Owner and Operator that receives a data specification from its Transmission Operator, Balancing Authority, or Reliability Coordinator must satisfy it, across operations planning, same-day, and real-time horizons. The operating practice is a defined, owned process for answering data requests and specifications accurately and on time, plus a change-management process wired to the Planning Coordinator's qualified-change definition. The evidence is the record of what was provided and when. An entity that treats data requests as interruptions accumulates the gaps an audit exposes.

4.3 Ride-through and operating behavior

The reliability risk that drove the entire action plan is the tendency of computational loads to disconnect during disturbances. The Level 3 Alert asked Transmission Owners to coordinate with computational load customers, where possible, on customer-side improvements that maximize facility-level ride-through up to equipment capability limits. The Phase 1 CLO standards do not set a ride-through performance requirement. They do require ride-through parameters in the modeling data (CLO-001-1 R8 and R9), and a separate draft Standard Authorization Request, Disturbance Performance for Computational Loads, would establish minimum voltage and frequency ride-through. In ERCOT a ride-through obligation already applies: NOGRR282, Large Computational Load Ride-Through Requirements, was approved by the PUCT on July 9, 2026 and took effect August 1, 2026. It exempts loads that were already operating, or far enough along in interconnection, by its cutoff date, and a later qualifying modification ends the exemption. Check where each site falls. For the entity, ride-through is both an engineering task (configuring protective settings and equipment to ride through rather than trip where the equipment can safely do so) and a documentation task: recording the settings, their basis, and the engineering judgment behind them.

Operating behavior is now written into the standards. CLO-002-1 R8 requires each Computational Load Operator to comply with Operating Instructions from its Transmission Operator, Balancing Authority, or Reliability Coordinator unless the action cannot be physically implemented or would violate safety, equipment, regulatory, or statutory requirements. R9 requires the operator to tell the issuer when it cannot comply. Both carry a High Violation Risk Factor. R11 requires the operator to have Interpersonal Communication capability with its Balancing Authority and Transmission Operator, and R13 requires it to consult the affected entities on restoring that capability when it fails. R14 requires initial training in repeat-back

before operating personnel can receive an oral two-party Operating Instruction, and R15 requires the repeat-back itself during an Emergency. The Transmission Operator and Balancing Authority carry the mirror-image obligations in R10 and R12. Each of these has to be practiced and documented. CLO-003-1 R9 and R10 add a process for developing protection settings in coordination with the owners of electrically joined equipment, and an obligation to follow it.

4.4 Instrumentation and monitoring

An entity cannot show how its facility behaved during a disturbance unless it captured the behavior. CLO-003-1 now states the recording obligations precisely. R1 requires the Computational Load Owner to have sequence of event recording for breaker position on the buses, main power transformer, on-site generation, and reactive devices it owns. Fault recording falls on whoever owns the main power transformer, whether that is the Computational Load Owner, a Transmission Owner, a Distribution Provider, or a Generator Owner: at least two cycles of pre-trigger data, a 2.0-second record, and at least 64 samples per cycle (R2 and R3). The same owner needs continuous dynamic disturbance recording sampled at 960 samples per second or more (R4 and R5), and all recording must be time-synchronized to within one millisecond of UTC (R6). Data must be retrievable for 20 calendar days and provided within 15 calendar days of a request from the planner, operator, Regional Entity, or NERC (R7). A recording failure must be fixed within 90 calendar days, or covered by a Corrective Action Plan submitted to the Regional Entity in that window (R8). For the entity, this means installing and maintaining the recording capability, retaining the captured data, and being able to produce it on schedule. Monitoring data is both a reliability tool, because it shows how the facility actually performs, and a compliance artifact, because it evidences that performance to a regulator.

4.5 Building the practices before the standards take effect

Because the obligations are now written down, an entity does not need to wait for FERC approval to begin. The practices in this chapter (owning the model, provisioning data, following Operating Instructions, engineering and documenting ride-through and protection settings, instrumenting the facility) are what the standards will require. Building them now does three things. It manages the underlying reliability risk immediately. It answers the alert-driven requests the interconnecting and operating entities are already sending. And it positions the entity to be compliant when each phase-in date arrives. The CLO-002-1 operating, communication, and training requirements come due twelve months after the effective date, which is shorter than many organizations' procurement and staffing cycles. Plan backward from that date. Owners in Texas have a parallel track: the PUCT adopted new 16 TAC §25.194, Large Load Interconnection Standards, by order filed September 18, 2026 in Project No. 58481, effective October 8, 2026, and EC-WP-1003 covers that framework.

Chapter 5. The Compliance Monitoring and Enforcement Lifecycle

Compliance is verified continuously, through many channels, on the regulator's schedule. Once registered, an entity operates inside NERC's Compliance Monitoring and Enforcement Program, Appendix 4C to the Rules of Procedure, the standing framework through which compliance is checked and noncompliance is resolved. The CMEP uses several monitoring processes, any of which can reach an entity at any time, and it resolves noncompliance through a defined enforcement and mitigation path. A program built to run continuously, as Chapter 1 described, is ready for all of them. A program run episodically gets caught out.

5.1 The monitoring methods

Appendix 4C lists the compliance monitoring processes. Compliance Audits are planned examinations of an entity's compliance over an audit period running from the day after the prior audit's End Date, with notice of a planned audit at least 270 days ahead and the formal audit notice at least 90 days ahead. Self-Certifications require the entity to attest to its own compliance with specified standards, on at least 60 days' request. Spot Checks are targeted reviews the Compliance Enforcement Authority may start at its discretion. Compliance Investigations can follow events or information from any source. Self-Reports are the entity's own disclosures of possible noncompliance. Self-Logging lets an approved entity log minimal-risk noncompliance instead of filing individual Self-Reports. Periodic Data Submittals require data on the schedule set in a standard or by the Compliance Enforcement Authority, normally with at least 20 days' notice where the standard is silent. Complaints can come from anyone. An entity does not choose which process applies to it. It has to be ready for all of them, which is only feasible if the program and its evidence run continuously.

5.2 Self-reporting and self-logging

Two features of the program deserve particular attention because they reward the right behavior. Self-reporting is the entity's disclosure of a possible violation it has found itself. The Sanction Guidelines count self-reporting within a reasonably prompt time, and before detection or notice of a compliance monitoring engagement, as a mitigating factor, along with voluntary mitigating activities. For minimal-risk issues, the CMEP also provides Self-Logging. A Compliance Enforcement Authority may approve an entity for Self-Logging after a formal review of its internal controls. The approved entity keeps a log of each minimal-risk noncompliance with its description, risk assessment, and mitigating activities, and there is a rebuttable presumption that logged items will be resolved as Compliance Exceptions. The Compliance Enforcement Authority reviews the log at least every three months, with the option of extending that to six.

Both features reward what this book keeps coming back to: internal controls that detect problems, and a culture that surfaces them. An entity that finds and reports its own issues, backed by a sound program, is operating as the framework intends.

Self-reporting decisions are among the most consequential decisions a registered entity makes outside an audit cycle. A Self-Report is the entity's disclosure that it has identified a potential noncompliance and is bringing it to the Compliance Enforcement Authority voluntarily. Under the CMEP it includes a risk assessment, the extent and cause of the noncompliance, and the actions taken or planned to mitigate it and prevent recurrence. The Sanction Guidelines treat prompt self-reporting before detection as a mitigating factor, but a notification letter for an upcoming compliance monitoring engagement generally ends that window for the requirements in scope. A Self-Report also opens a regulatory matter that proceeds on its own track. Whether, when, and how to self-report deserves counsel's involvement, because the framing shapes the response that follows. Programs that write self-reporting protocols in advance, with documented decision criteria, counsel-engagement triggers, and timing standards, make better decisions than programs that decide each case in isolation.

Self-reporting and mitigation planning work as a pair. The Self-Report identifies the potential noncompliance. The Mitigation Plan describes how the entity will correct it and its cause. Together they are the entity's response to a compliance event, and the quality of each affects how the matter is resolved. In my experience, thoughtful Self-Reports with credible Mitigation Plans close without escalation. Hasty Self-Reports and vague plans tend to grow into formal enforcement matters.

5.3 Mitigation

When a noncompliance occurs, whether the entity found it or a regulator did, it is resolved through mitigation: a plan that corrects the noncompliance, addresses its cause, and prevents recurrence, executed and evidenced. Under Appendix 4C, submitting a Mitigation Plan is not an admission of a violation. A credible plan does more than fix the immediate problem. It shows that the entity understands the root cause and has strengthened the control that failed. In my experience the quality of an entity's mitigation is read by the regulator as a signal of program maturity. Approach it with the same rigor as the original obligation, not as an admission to be minimized.

Mitigation planning converts a potential noncompliance into a closed matter. The required contents are set in Appendix 4C, Section 6.2: a point of contact, the noncompliance being corrected, its cause, the activities that will correct the noncompliance and its cause and prevent recurrence, the reliability impact while the plan is under way, and a timetable with milestones, normally no more than three months apart. What is less well understood is that, in my experience, the quality of the plan often matters more than the quality of the underlying compliance argument. A credible plan, with concrete actions, realistic timelines, named owners, and demonstrable milestones, lets the Regional Entity accept it, see the mitigation done, and close the matter without escalation. A vague plan invites a different posture, where the mitigation negotiation becomes the real proceeding.

Continuous improvement converts each audit cycle into a stronger program for the next. A program that closes findings and moves on has met its minimum obligation and left value on the

table. A program that examines each finding for root causes, updates controls and procedures to address them, and builds the changes into operating practice has used the audit to make itself structurally better. Over several cycles the difference compounds. One program gets easier to operate with each audit. The other keeps producing the same findings.

Lessons-learned reviews are how events become improvements. After every significant event (an audit, a Self-Report, a Mitigation Plan, a system event the facility took part in) the program team should hold a structured review, document what was learned, and turn the lessons into procedure updates. The practice comes from the operational world, where post-event reviews are standard in control-room culture, and it travels well into compliance. Programs that adopt it build institutional learning that survives staff turnover. Programs that do not repeat the same mistakes across different teams.

5.4 Operating inside the lifecycle

The enforcement lifecycle is not an outside event that occasionally intrudes on operations. It is the environment a registered entity operates in, every day. The entity that understands this runs its program to be ready at all times: evidence current, controls operating, issues surfaced and self-reported, mitigation executed cleanly. Whatever monitoring process arrives, and whenever it arrives, the answer is already prepared. That readiness comes from the operating model, controls, and evidence infrastructure in the preceding chapters, and it is demonstrated through the deliverable covered next: the RSAW.

Senior leadership engagement is one of the most reliable indicators of program maturity. A program with leadership's attention, through regular briefings, participation in significant decisions, and visible support during events, operates in an organization that treats compliance as a real priority. A program without it operates as a back-office function the organization tolerates but does not value. Over several cycles the consequence is large. Programs with engaged leadership attract resources, keep staff, and influence operational decisions in ways other programs cannot.

Board-level reporting is the high end of that engagement and is worth considering even where corporate governance does not require it. A periodic compliance report to the board, quarterly in my view, gives the board what it needs to oversee compliance and gives the program a visible presence at the top of the organization. Preparing the report forces the program to state its posture clearly, name its risks honestly, and report progress credibly. The reports become part of the evidence base and show the organizational commitment to compliance that regulators look for.

Chapter 6. RSAW Production and Audit Defense

The audit is where the program is tested, and the RSAW is where it speaks. When a Compliance Audit arrives, the entity demonstrates compliance through the Reliability Standard Audit Worksheet, the document NERC posts for a standard, in which the entity sets out how it meets each requirement and points to the evidence. RSAWs for the CLO standards will follow the standards' approval; I found none posted on NERC's RSAW page on October 17, 2026. A strong RSAW is not written the week of the audit. It is the natural product of a program that has been generating owned, mapped, well-retained evidence all along. This chapter covers how to produce RSAW narratives that withstand scrutiny, and how to defend the program in the audit itself.

6.1 The anatomy of a defensible narrative

EC's standing standard for an RSAW narrative is that each requirement gets four things, in order. First, a direct narrative response that addresses the requirement language head-on: what the entity does and how it does it, not a description of the company. Second, explicit evidence references that name the specific document, log, system, or record, with version or date where relevant. Generic phrases like "documented in our procedures" are not acceptable, because an auditor cannot evaluate an artifact that is not identified. Third, the connective logic that explains how the cited evidence satisfies the requirement, which is the bridge an auditor evaluates between obligation and proof. Fourth, the audit-defensibility cues that signal a mature program: the responsible role, the cadence, and the retention path. A narrative built this way reads as an audit-ready response, not a marketing description, and it is the most reliable way to turn genuine compliance into a clean result.

6.2 Honesty and the prohibition on manufactured evidence

A defensible narrative is an honest one. A narrative must never reference evidence that does not exist, never describe a control that does not operate, and never paper over a gap with confident language. If the evidence for a requirement is missing, the right response is to identify the gap, remediate it, and, where appropriate, self-report. Do not draft around it. Auditors are skilled at probing the bridge between narrative and evidence, and a narrative that promises an artifact the entity cannot produce does far more damage than an honest acknowledgment of a gap. This is the practical standard as well as the ethical one, because a single instance of overstated compliance undermines an auditor's confidence in everything else the entity has said. Every CMEP submission is signed by an authorized representative of the entity, and the signature means something.

6.3 Internal audit, self-assessment, and the mock audit

The internal-audit function is the program's self-correcting mechanism. A well-run internal audit examines a subset of the program's controls each quarter, produces findings and observations, and tracks them to closure on a defined timeline. Findings produced internally and closed before the external audit arrive as demonstrations of program maturity. Findings the external audit surfaces that internal audit missed send a different signal: the self-correcting mechanism is not working, and the entity has to explain why. A credible internal-audit function takes years to build, and it pays back in the audit relationship and in the operational soundness of the program.

Self-assessment is internal audit's lighter-weight counterpart. It examines the program's posture against the standards on a defined cadence, annually at a minimum and more often for higher-risk areas, and produces a written report of strengths, weaknesses, and corrective actions. The record becomes part of the evidence base, and in a smaller program it can carry much of the load an internal-audit function would. In my experience, entities that produce credible self-assessments and act on them fare better in external audits than entities that treat self-assessment as paperwork or skip it.

Audit preparation produces better audits for both sides. An entity that arrives prepared, with a clear understanding of the requirements, an organized evidence record, and rehearsed answers to the common questions, makes the audit team's work easier and gets a shorter, more productive audit. An entity that arrives unprepared makes the work harder and gets a longer, more intrusive, more adversarial one. Mature programs treat preparation as routine, and the investment pays back across cycles.

The most effective way to test a program before a regulator does is to audit it internally first. A mock audit applies the same rigor the real one will: selecting requirements, demanding the RSAW narrative and the supporting evidence, and probing the connective logic. It should be run by someone independent enough to find the problems. The mock audit surfaces gaps while they can still be fixed: the undocumented control, the orphaned artifact, the narrative that cannot be supported. An entity that mock-audits itself on a regular cadence enters every real audit having already found and fixed what the regulator would otherwise find. Run it by an internal team, or better, by an outside party with audit experience. Programs that mock-audit routinely experience the external audit as confirmation. Programs that do not experience it as discovery. That is audit readiness.

6.4 The first audit

First-time audit defense for a newly registered functional entity has a different shape from what long-registered entities know. The first-time entity is presenting a program the Regional Entity has never seen, demonstrating compliance with requirements that may themselves be new, under the scrutiny that comes with the first cycle of any new registration category. The audit team is not hostile, but it is asking foundational questions. How was the program designed? Who owns it? How is it operated? How is the evidence developed and maintained? Entities that approach the first audit as a structured chance to validate the program come through it well. Entities that approach it as a high-stakes defense produce defensive answers that read as evasive on the record, and that tone is what the audit team remembers when the next cycle begins.

That first audit also carries a structural weight later audits do not. It sets the baseline against which later audits are measured: how the program is designed, who runs it, what evidence it produces, how it documents its decisions. The impressions the first team forms will shape the depth and tenor of the cycles that follow. Treat the first audit as a long-term investment in the audit relationship.

Audit defense is, at one level, an exercise in clear communication. The audit team knows the requirements better than the registered entity does, but the entity knows its program better than the audit team can. Good audit defense translates between the two. It presents the program in terms the team can verify against the requirements, and it gives the team the substantive responses it needs to close its examination. Programs that have practiced this translation through internal audits, self-assessments, and mock audits do it well when the external audit arrives. Programs that have not practiced make the audit harder by the way they present, whatever their underlying compliance.

6.5 Responding to data requests in the audit

During an audit, much of the interaction takes the form of data requests, the auditor asking for specific evidence. A disciplined entity treats these the way EC treats data-request work generally. It maps each request to the precise evidence that answers it, provides exactly what is responsive without volunteering unrelated material, and screens its responses for accuracy and consistency before submitting. The CMEP sets out what happens when requested data is not submitted by the Required Date, and it allows reasonable additional time when the entity asks before the deadline and gives its reasons. Ask early if you need it. A well-organized evidence map, maintained continuously as Chapter 3 described, makes all of this fast and credible. A disorganized evidence base turns every request into a scramble, and the scramble itself signals weak controls. In the end the audit tests not just whether the entity is compliant but whether it can demonstrate compliance efficiently and credibly, and that demonstration is the product of the entire program this book describes.

6.6 After the audit: closure and the Regional Entity relationship

Post-audit closure is its own discipline. An audit produces findings, observations, and recommendations, and each has a defined path to closure. A potential noncompliance requires mitigation: a Mitigation Plan that, under the CMEP, identifies the cause, the activities that will correct the noncompliance and its cause and prevent recurrence, and a timetable with milestones, followed by evidence of completion. Observations need corrective action and program updates. Recommendations need consideration and a documented decision about whether to adopt them. Programs that run closure as a structured workstream, owned, scheduled, and tracked, produce closure records that support the next audit. Programs that treat closure as informal follow-up produce records the next audit reads skeptically.

The relationship with the Regional Entity in the months after the first audit matters at least as much as the audit itself. Findings, observations, areas of concern, and recommendations each carry a different procedural status, but all of them become part of the record the next audit team reads. A program that responds promptly to findings, tracks observations to closure, and reports on areas of concern between audits builds credibility that pays off in the next cycle. A program

that treats anything short of a formal finding as optional builds a different relationship, one where the next team arrives with more questions and less benefit of the doubt.

The relationship with the Regional Entity between audits deserves explicit management. Regional Entities stay in contact with their registered population through alerts, notices, requests for information, and occasional compliance inquiries, and the cumulative quality of those interactions shapes the Regional Entity's view of the program. A program that answers promptly, completely, and consistently builds credibility that pays back in the formal audit cycle. One that answers inconsistently builds the opposite.

Chapter 7. Building the Program: A Phased Approach

A program this comprehensive is built in sequence, not all at once. An organization standing up its first NERC compliance program can be overwhelmed by the breadth of what this book describes. The answer is to build in phases, in an order that puts the foundations first. What follows is the sequence I recommend. Adapt it to your circumstances, but respect the dependencies between the pieces.

7.1 Phase one: scope and foundation

The program begins with knowing exactly what applies. The first phase establishes the entity's registered functions and the standards and requirements that attach to them, stands up the compliance function with clear ownership, and produces the initial requirement-to-control mapping described in Chapter 2. For a computational load, that starts with the registration analysis: whether each site meets the Board-approved criteria (as posted in August 2026, a Computational Load Site with total connected Load of at least 50 MW supplied at 100 kV or above from one or more points of connection), and which legal entity registers as the Computational Load Owner and which as the Computational Load Operator. The operator carries the CLO-002-1 operating requirements, so that decision drives staffing. EC-WP-1005 covers the criteria in detail. The deliverable of this phase is clarity: a documented understanding of every obligation the entity carries and who owns it. Nothing else can be built reliably until that exists, because a control framework, an evidence map, and an RSAW all depend on knowing precisely what must be complied with.

7.2 Phase two: controls and evidence

The second phase builds the machinery. For each mapped requirement, the entity designs the control (owner, process, evidence, detection) and stands up the evidence infrastructure that captures and retains the resulting artifacts, organized through the evidence map. This is the phase that converts intent into operating reality, and it is where most of the work lives. By the end of it, the entity should be generating and retaining, as a matter of routine, the evidence that proves each control operates.

7.3 Phase three: technical execution

With controls and evidence in place, the third phase operationalizes the technical obligations specific to the computational load: modeling data and verification, data provision and study support, Operating Instructions and communications, ride-through and protection settings, and disturbance recording, each run as an owned control producing dated evidence. Sequencing technical execution after the control and evidence framework means the technical work is documented from the start, not performed first and awkwardly evidenced afterward. Use the

implementation plans' phase-in dates to order the work. CLO-002-1's operating and communication requirements arrive at twelve months, ahead of most CLO-003-1 recording obligations for existing sites.

7.4 Phase four: readiness and defense

The final phase makes the program audit-ready and keeps it that way. The entity drafts its RSAW narratives to the four-part standard, runs a mock audit to find and fix gaps, exercises its self-report and mitigation processes, and sets the recurring cadence (self-assessment, management review, evidence-map maintenance, periodic mock audits) that keeps the program continuously prepared. This phase never really ends. It becomes the steady-state rhythm of operating as a registered entity. An entity that reaches it has turned the obligations traced across this series into routine, demonstrable practice.

7.5 Closing the series

This book is the fifth in the Energy Compliance Large Load and Data Center Series, and it is worth seeing the whole arc at once. The first book examined the grid layer: why large computational loads pose a reliability problem and how the federal interconnection framework is responding. The second examined the NERC layer: the Large Loads Action Plan that turns that problem into obligations, from white paper to alert to standards project. The third examined the entity layer through the April 2026 Computational Load Entity proposal. NERC has since replaced that proposal with the Computational Load Owner and Computational Load Operator functions, which the Board approved on October 8, 2026, and EC-WP-1005 documents the change. The fourth examined the owner layer: the decisions a developer faces under Texas Senate Bill 6 and the co-location regime. This fifth book has examined the operations and compliance layer: how a registered computational load actually operates, evidences, and defends its compliance.

The operating layer is the one closest to the day-to-day work of running a compliance program. The grid, standards, entity, and owner layers each described part of the regulatory environment a registered Computational Load Owner or Operator operates inside. The operating layer turns that environment into the disciplines a program uses to demonstrate compliance: control design, governance, evidence, audit defense, and continuous improvement.

The books are designed to be read in sequence but used independently. A compliance manager will spend most of their time in this book, going back to the earlier ones for context. Counsel will spend more time in the entity and owner layers and in EC-WP-1005. Senior leadership will engage mainly through the grid and standards layers.

Together the books trace one transformation. A class of load that operated outside the reliability framework is being brought inside it, deliberately and quickly, and the organizations that build, own, and operate these loads have to adapt across every layer at once. The obligations are arriving whether or not any individual entity is ready. The purpose of this series has been to make readiness possible, by turning a fast-moving, multi-layered regulatory shift into something an organization can understand, plan for, and execute. The work of compliance is the work of demonstrating reliability, every day and on demand. An organization that builds the program described here is prepared to do it.

Glossary of Terms

The following terms are used throughout this whitepaper. Definitions are summarized for convenience. The operative definitions are those in the NERC Rules of Procedure and its appendices, the Reliability Standards, and the Glossary of Terms Used in NERC Reliability Standards.

Term	Meaning in this paper
Audit (Compliance Audit)	A planned examination of an entity's compliance with specified Reliability Standards over an audit period that begins the day after the End Date of the prior audit, conducted under the CMEP (Appendix 4C, Section 4.1).
Compliance Enforcement Authority (CEA)	NERC or the Regional Entity in its role of monitoring and enforcing compliance with the Reliability Standards.
Compliance Monitoring and Enforcement Program (CMEP)	NERC's program for monitoring compliance and resolving noncompliance, set out in Appendix 4C to the Rules of Procedure. Its monitoring processes are Compliance Audits, Self-Certifications, Spot Checks, Compliance Investigations, Self-Reports, Self-Logging, Periodic Data Submittals, and Complaints.
Computational Load Operator (CLOP)	The entity that operates a Computational Load Site. A functional entity created by the Rules of Procedure revisions the NERC Board approved on October 8, 2026, pending FERC approval.
Computational Load Owner (CLO)	The entity that owns and maintains a Computational Load Site. Created by the same Rules of Procedure revisions.
Computational Load Site	As defined in the Project 2026-02 standards and the Board-approved Rules of Procedure revisions, one or more buildings, structures, or installations at a single location that primarily contains information technology infrastructure used for software applications and services, artificial intelligence training and deployment, cryptocurrency mining, or other computational activities at a data center, and that is supplied through electrical equipment connected to the Bulk-Power System.
Connective Logic	In an RSAW narrative, the explicit explanation of how cited evidence satisfies a requirement: the reasoning an auditor evaluates between the obligation and the proof.
Evidence Map	A structured link from each applicable requirement to the specific artifacts that demonstrate compliance with it; the core organizing tool of a compliance program.
Internal Control	A defined, repeatable mechanism, with an owner, a process, evidence, and a detection-and-correction step, that ensures a requirement is met and surfaces a problem if it is not.

Term	Meaning in this paper
Mitigation Plan	The plan an entity submits to correct a noncompliance. Under Appendix 4C, Section 6.2, it identifies the cause and the activities that will correct the noncompliance and its cause and prevent recurrence, with a timetable and milestones. Submitting one is not an admission of a violation (Section 6.1).
Mock Audit	An internal exercise that applies real-audit rigor (selecting requirements, demanding narratives and evidence, and probing the connective logic) to find and fix gaps before a regulator does.
Operating Instruction	Per the NERC Glossary, a command by operating personnel responsible for Real-time operation of the interconnected Bulk Electric System to change or preserve the state, status, output, or input of an Element or Facility. Project 2026-02 extends it to Computational Load Sites, and CLO-002-1 R8 requires Computational Load Operators to comply.
Periodic Data Submittal	A CMEP monitoring process requiring an entity to provide specified data on the schedule in a Reliability Standard or set by the CEA, normally with at least 20 days' notice where the standard sets none.
Reliability Standard Audit Worksheet (RSAW)	The worksheet NERC posts for a Reliability Standard, through which an entity demonstrates how it meets each requirement and identifies the supporting evidence.
Retention	The obligation to keep compliance evidence for the period stated in each standard's Evidence Retention section. Where that period is shorter than the audit period, the entity must demonstrate compliance for the full period by other means (Appendix 4C, Section 4.1.3.2).
Ride-Through	The ability of a facility to remain connected through a defined envelope of voltage and frequency disturbances. NERC's Level 3 Alert asked Transmission Owners to coordinate on maximizing computational-load ride-through up to equipment capability limits; a draft NERC SAR would set minimum ride-through; ERCOT's NOGRR282, effective August 1, 2026, requires it of Large Computational Loads other than those that qualify for its exemptions for existing facilities.
Self-Certification	A CMEP monitoring process in which the CEA asks an entity, at least 60 days in advance, to attest to its own compliance with specified standards.
Self-Logging	A CMEP process under which an entity approved after a formal review of its internal controls logs minimal-risk noncompliance for periodic CEA review instead of filing individual Self-Reports. Logged items carry a rebuttable presumption of resolution as Compliance Exceptions.
Self-Report	An entity's own disclosure of a possible violation it has identified. Prompt self-reporting before detection or notice of a monitoring engagement is a mitigating factor under the Sanction Guidelines.
Spot Check	A targeted CMEP review of an entity's compliance with particular requirements, which the CEA may initiate at its discretion.

How Energy Compliance Can Help

This book describes a program. Energy Compliance, Inc. builds and runs programs like it for registered entities, and the three services that fit a newly registered computational load are the ones the chapters lean on hardest.

Compliance program management. Standing up the operating model in Chapters 1 through 3: the requirement-to-control mapping, the evidence map, the governance cadence, and the retention design, built once and run across every site the organization owns or operates. For a Computational Load Owner or Operator registering in 2027, that work starts with the registration analysis and the choice of which legal entity carries which function.

RSAW production and audit defense. Narratives written to the four-part standard in Chapter 6, mock audits run by someone who has sat on the other side of the table, and support through the first audit cycle, from the 270-day notice to closure.

Data request strategy. The disciplined pipeline Chapter 6 describes for audit data requests, and the same discipline applied to the data specifications and planner requests that CLO-001-1 and CLO-002-1 will generate.

WattOps Compliance is EC's NERC compliance evidence platform (wattopscompliance.com). It is where the evidence map in Chapter 3 lives in practice. EC will run one standard at one facility through WattOps at no cost, so a program can see how its evidence holds up before committing to anything; CIP standards are excluded from that offer.

RenewOps is EC's renewable operations monitoring and analytics platform, with a Data Center module. The disturbance recording, operating data, and event records that CLO-002-1 and CLO-003-1 call for are operations data first and compliance evidence second, and RenewOps is built for the operations side of that equation.

If you are preparing for registration, or you have the letter and need the program to be real before the first audit notice, call or write. We will look at where you stand and tell you plainly what has to be built.

Website: energycomplianceinc.com · Email: rsmith@complianceforenergy.com · Phone: 763.438.4427

About the Author

Rob Smith is the Founder and Principal of Energy Compliance, Inc., with more than thirty years in the industry on every side of the system. He began in control-room operations as a Reliability Coordinator, Transmission Operator, and Power System Operator across RTO/ISO and utility control centers. On the regulator side, he was a senior compliance auditor and subject matter expert for NERC Reliability Standards. He audited grid facilities, evaluated mitigation adequacy, participated in risk-based oversight of utility mitigation activities, and supported the development of violation notifications and settlements in FERC-directed enforcement actions. He conducted a regulatory audit in the Sultanate of Oman against the Sultanate's Sector Law and Grid Code.

He founded Energy Compliance, Inc. to deliver senior regulator-side compliance authority without the layered staffing and billable-hour overhead of larger firms.

Education:

- MBA, Energy Management, University of Mary
- MSL, Compliance Law, Fordham Law University
- BAS, Energy Management, Bismarck State College
- BAS, Mechanical Engineering, Metallurgy minor, University of Florida
- BAS, Criminology, St. Petersburg College (State Law Enforcement Certification)
- AAS, Power Plant Technology, Bismarck State College
- AAS, Electrical Transmission System Technology, Bismarck State College

Certifications: NERC Lead Auditor training; previously NERC-certified at the Reliability Coordinator level; 1st Class A Boiler Engineer License, Minnesota.

The disciplines in this book are the ones Rob's practice is built on: NERC registration and Reliability Standard compliance, internal-controls and program design, evidence development, RSAW production, mitigation, Data Request strategy, and audit defense.

The views expressed in this publication do not represent the views of NERC, FERC, or any Regional Entity.

About Energy Compliance, Inc.

Energy Compliance, Inc. (EC) is a one-stop shop for energy industry regulatory compliance, founded and led by Rob Smith. The firm serves the energy industry across NERC, the Texas PUCT, and adjacent regulatory frameworks, and it helps registered entities and prospective registrants build and operate the full compliance program described in this book, from registration and internal-controls design through evidence development, RSAW production, mitigation, and audit defense.

The firm works analyst first: every conclusion is tied to an authoritative source, every narrative is backed by evidence, and every deliverable is built to stand up to regulator scrutiny. EC's software is WattOps Compliance, its NERC compliance evidence platform (wattopscompliance.com), and RenewOps, its renewable operations monitoring and analytics platform, which includes a Data Center module.

This whitepaper is the fifth volume in the Energy Compliance Large Load and Data Center Series; EC-WP-1005 is the series' October 2026 update.

Energy Compliance is not affiliated with, sponsored by, or endorsed by the North American Electric Reliability Corporation, the Federal Energy Regulatory Commission, or any Regional Entity. To discuss building a compliance program for a registered computational load, contact us.

Website: energycomplianceinc.com · Email: rsmith@complianceforenergy.com · Phone: 763.438.4427

Sources and Authorities

This whitepaper draws on NERC's established compliance framework and the computational load registration criteria and standards developed under NERC's Large Loads Action Plan. The registry criteria await FERC approval and the CLO standards await Board adoption and FERC approval; both are identified as such. Readers should confirm the current standards, criteria, and program documents before relying on them.

NERC: compliance framework

- NERC, Rules of Procedure, Rules of Procedure in Effect table (accessed October 17, 2026). <https://www.nerc.com/who-we-are/rules-of-procedure>
- NERC, Rules of Procedure Appendix 4C, Compliance Monitoring and Enforcement Program, effective May 19, 2022, Sections 4.0 through 4.8 (monitoring processes, including 4.1.3.2 audit period and retention, 4.5 Self-Reports, 4.5A Self-Logging, 4.6 Periodic Data Submittals), 6.1 and 6.2 (Mitigation Plans), and Attachment 1 (non-submittal of requested data). <https://www.nerc.com/globalassets/who-we-are/rules-of-procedure/appendix-4c.pdf>
- NERC, Rules of Procedure Appendix 4B, Sanction Guidelines, effective January 19, 2021, Section 3.3 (aggravating factors including concealment; mitigating factors for internal compliance program, cooperation, and self-reporting). <https://www.nerc.com/globalassets/who-we-are/rules-of-procedure/appendix-4b.pdf>
- NERC, Rules of Procedure Appendix 2, Definitions Used in the Rules of Procedure, effective June 27, 2024. <https://www.nerc.com/globalassets/who-we-are/rules-of-procedure/appendix-2.pdf>
- NERC, Reliability Standard Audit Worksheets (RSAWs). <https://www.nerc.com/programs/compliance/reliability-standard-audit-worksheets-rsaws>

NERC: computational load registration and standards

- NERC, Large Loads Action Plan Q3 2026 Update (Board approval of Rules of Procedure revisions establishing the Computational Load Owner and Computational Load Operator, October 8, 2026). <https://www.nerc.com/globalassets/initiatives/large-loads-action-plan/llap-quarterly-update-q3-2026.pdf>
- NERC, Project 2026-02 Computational Loads, project page. <https://www.nerc.com/standards/reliability-standards-under-development/2026-02-computational-loads>

- NERC, CLO-001-1, Computational Load Interconnection, Studies, & Modeling Data, final ballot clean draft (October 2026).
<https://www.nerc.com/globalassets/standards/projects/2026-02/final-ballots/project-2026-02-computational-loads-clo-001-1-final-clean.pdf>
- NERC, CLO-002-1, Computational Load Operational Data and Communications, final ballot clean draft (October 2026).
<https://www.nerc.com/globalassets/standards/projects/2026-02/final-ballots/project-2026-02-computational-loads-clo-002-1-standard-final-clean.pdf>
- NERC, CLO-003-1, Computational Load Protection Coordination & Disturbance Monitoring, final ballot clean draft (October 2026).
<https://www.nerc.com/globalassets/standards/projects/2026-02/final-ballots/project-2026-02-computational-loads-clo-003-1-standard-final-clean.pdf>
- NERC, Implementation Plans for CLO-001-1, CLO-002-1, and CLO-003-1, final ballot versions (October 2026).
<https://www.nerc.com/globalassets/standards/projects/2026-02/final-ballots/project-2026-02-computational-loads-clo-002-1-implementation-plan-final-clean.pdf>
- NERC, Project 2026-02 FAQ (October 2026).
<https://www.nerc.com/globalassets/standards/projects/2026-02/final-ballots/project-2026-02-faq---october-2026.pdf>
- NERC newsroom, Computational Loads Standards Pass Final Ballot, Advance to Board Adoption (October 2026). <https://www.nerc.com/newsroom/nercs-computational-loads-standards-pass-final-ballot-advance-to-board-adoption>
- NERC, Level 3 Essential Action Alert, Computational Load Modeling, Studies, Instrumentation, Commissioning, Operations, Protection, and Control (May 4, 2026).
<https://www.nerc.com/globalassets/programs/bpsa/alerts/level-3-computational-load-alert.pdf>
- NERC, Reliability Guideline: Risk Mitigation for Emerging Large Loads (May 2026).
https://www.nerc.com/globalassets/our-work/guidelines/reliability/RG_Risk-Mitigation-For-Emerging-Large-Loads.pdf

Federal Energy Regulatory Commission

- Order Directing the North American Electric Reliability Corporation to File Reliability Standard(s) Pertaining to Computational Load Integration, 196 FERC ¶ 61,031, Docket No. RD26-7-000 (issued July 16, 2026).
<https://www.ferc.gov/sites/default/files/2026-07/RD26-7-000.pdf>

ERCOT and Public Utility Commission of Texas

- ERCOT, NOGRR282, Large Computational Load Ride-Through Requirements (PUCT approved July 9, 2026, Project No. 54445; effective August 1, 2026; exemptions for existing facilities in new Nodal Operating Guide Sections 2.6.4 and 2.15 as shown in the PUCT Report).
<https://www.ercot.com/mktrules/issues/NOGRR282> ;
<https://www.ercot.com/files/docs/2026/07/11/282NOGRR-45-PUCT-Report-070926.docx>
- PUCT, Order Adopting New 16 TAC §25.194, Large Load Interconnection Standards, Project No. 58481, Item 218 (filed September 18, 2026).
https://interchange.puc.texas.gov/Documents/58481_218_1684654.PDF

-
- Texas Register acknowledgment of receipt, 16 TAC §25.194, Project No. 58481, Item 219 (effective date October 8, 2026).
https://interchange.puc.texas.gov/Documents/58481_219_1684678.PDF

Energy Compliance, Inc.: methodology

- Energy Compliance, Inc. RSAW production and audit-defense standards, including the four-part narrative discipline (direct response, explicit evidence references, connective logic, and audit-defensibility cues) and the prohibition on referencing evidence that does not exist.
- EC-WP-1005, Computational Load Owners and Operators: NERC's Registry Criteria and the CLO Standards (October 2026).

© Energy Compliance, Inc. · EC-WP-1004 · Status as of October 17, 2026