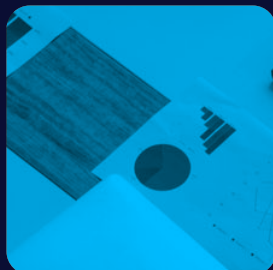
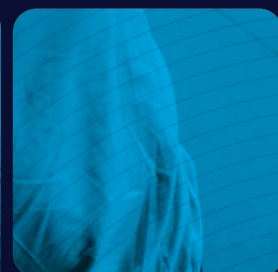


SOVOS



The IRIS Transition Playbook

How to Decide Whether
to Build or Buy Before
FIRE Is Retired





Executive Summary

The transition from the IRS FIRE system to IRIS is the most significant structural change to U.S. information reporting in forty years. Most organizations still think of this as a format to upgrade or a transport change. It is not.

IRIS introduces real-time validation, mandatory API transmission for bulk filers, annual IRS certification, and higher expectations for data quality and security controls. It also forces organizations into a prolonged period of fragmented compliance, with federal reporting modernizing faster than state reporting.

This playbook is not about selling software. It is about helping tax, IT, and finance leaders understand what breaks, why timelines slip, and how to make a defensible build versus buy decision grounded in operational reality.

A Brief History of FIRE and the IRIS Timeline

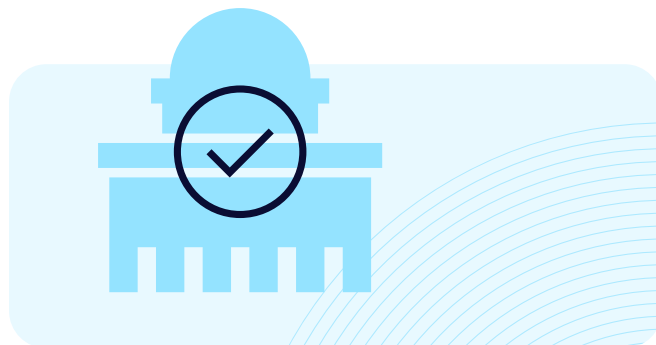
The IRS launched the Filing Information Returns Electronically system, known as FIRE, in the early 1980s. At the time it was a major modernization, moving information returns from paper to electronic submission. When FIRE was first designed, it was built to handle only a few million returns annually.

Over the next four decades, volumes grew steadily as more transactions became reportable; new information returns were created through acts of Congress, and electronic filing became the default.

By the 2024 filing season, the IRS received more than five billion information returns each year through FIRE. With continued growth in reportable payments, deductible transactions, and the addition of new reporting regimes such as digital asset reporting under Form 1099-DA, the IRS is expected to receive close to six billion returns in the 2025 filing season, with volumes continuing to rise rapidly thereafter.

But FIRE was built for a mainframe era. It relies on flat files, batch-only processing, and legacy security models. It cannot support real-time validation, modern cybersecurity controls, or structured data exchange with states and other IRS platforms.

The IRS has formally committed to retiring FIRE and replacing it with the Information Returns Intake System, or IRIS. This means organizations effectively have one remaining live filing season to test, validate, and harden IRIS processes before FIRE is shut down.



Why Most Organizations Are Underestimating IRIS

Most teams still treat IRIS as FIRE 2.0. That framing is the root of missed timelines and blown budgets. This is not a one-time filing change. IRIS permanently changes the operating model for information reporting.

Data must now be structured into rigid XML schemas. Bulk filers must transmit information return data through an API connection. Every year, that connection must be re-certified by the IRS, and that recertification burden applies form by form.

Real-time validation removes the buffer that FIRE's batch model used to provide. Errors that once surfaced days later now appear instantly and must be handled immediately.

At the same time, most states are not modernizing on the same timeline. Federal reporting is moving to IRIS while state reporting remains a patchwork of legacy formats, state portals, and state-specific delivery channels.

The combination of permanent technical change, annual certification, real time feedback, and fragmented state compliance is what most organizations are underestimating.



**This is not a one-time filing change.
IRIS permanently changes the operating
model for information reporting.**

What IRIS Actually Changes in Practice

Under FIRE, information reporting could be treated as a once-a-year batch exercise. Data could be imperfect. Errors could be corrected later. Flat files can be shaped to fit.

IRIS removes much of that flexibility.

Individual name changes and penalty risk

IRIS is XML native. Fields must be structured into rigid schemas, and validations are more precise. But it is important to be clear about what has actually changed. The most significant name format change applies to individual payees, not entities. Individual names must now be separated into true first name, middle initial, and last name fields.

**These changes may sound cosmetic.
They are not.**

When filers submit Forms 1099 with invalid name and TIN combinations, the IRS issues CP2100 backup withholding notices. That legal process requires filers to send specific letters to the payee, request a new Form W-9, and potentially begin backup withholding 24 percent on future payments until corrected information is received.

If incorrect data continues to be filed, that same population of records later becomes the basis for proposed penalty notices on Notice 972CG under Sections 6721 and 6722 for filing returns with incorrect information. To have those penalties abated, filers must prove they met the statutory due diligence and reasonable cause standards. That means showing they originally collected names and TINs in accordance with legal requirements and that they properly requested updated information after receiving CP2100 and or 972CG notices.

This is why the individual name field change in IRIS is not just a technical mapping issue. If filers restructure name data incorrectly, they can inadvertently create millions of dollars of penalty exposure by generating large volumes of artificial name and TIN mismatches.

The filing model and security model have changed

Bulk filers must transmit through a system-to-system API connection. There is no file upload option. That integration must be certified by the IRS and re-certified every year.

This is a complete break from how most organizations file today.

Under FIRE, whether filers took files directly from a core system, consolidated files across multiple cores, or built internal aggregation platforms, all those models relied on the same basic construct.

Everyone was coding to the same publicly available FIRE flat file specification, and anyone could obtain a FIRE Transmitter Control Code. As long as the file matched the published layout and basic validations, it could be uploaded and accepted.

Under IRIS, the operating logic changes. Flat file workflows must be rebuilt into structured XML, and organizations must complete IRS security and identity onboarding before they can even receive the IRIS XML schemas.

That onboarding typically requires registering the business with the IRS, applying for an EFIN, and requesting a Transmitter Control Code for A2A filing in IRIS. Only after that process is completed does the IRS release IRIS formats and schemas into the organization's IRS account mailbox.

At that point, the filer must code the XML, build the A2A integration, and apply to the IRS to begin testing. Once the IRS approves test transmissions, the filer can move into production.

And that entire cycle does not happen once. Under IRIS, the file format, the API connection, and the security posture must be retested and recertified every year, and for every return type the organization wants to issue.

This is also a fundamental security change. IRIS is not a passive file drop system. The IRS validates who generated and transmitted the file, not just whether the file matches the XML schema. Approval to transmit is tied to a specific technical environment, security controls, and a validation framework.

That means organizations cannot simply take a file generated by a third-party system and pass it through an approved A2A connection. Files must be generated, validated, and transmitted within the same certified environment.

Real time validations flush out long tolerated errors

Validations now happen in real time. Records are evaluated on intake, not days later. Where data does not make sense in required boxes, entire files can be rejected rather than partially accepted.

That means long standing data practices that were technically wrong but tolerated under FIRE will be flushed out by IRIS validations. Under FIRE, those errors were often silently accepted. Under IRIS, they become immediate filing failures.

State divergence is not a dual format problem

State reporting will diverge from federal reporting for several years, and the gap is already visible in practice.

Fewer than five states have actually converted to IRIS for the 2025 filing season. Many more states are signaling that they may convert for Tax Year 2026 filings, and several states still do not know when or whether they will migrate at all.

This divide has already surfaced in the 2025 filing season. The 2025 Forms 1099-NEC reporting golden parachute payments and the new Form 1099-DA were accepted by the IRS only through IRIS. Because most states were not ready for IRIS, many of them required those same returns to be submitted on paper, through state-specific CSV formats, or via secure file transfer or secure email of PDFs.

This is not just a dual-format problem. It is an operational disruption problem.

Each state migration introduces new exceptions, edge cases, and manual workarounds across form types, boxes, deadlines, and transmission channels.

CF/SF is not a compliance strategy

Relying on the IRS Combined Federal State Filing Program, known as CF/SF, for state compliance is also not prudent and creates real risk.

CF/SF is a limited data sharing program. Filers include additional state reporting fields in their federal submission, and the IRS forwards that data to states that claim to participate. In theory, this allows a filer to satisfy both federal and state reporting obligations through a single transmission.

In practice, the CF/SF participation list is not a reliable indicator of state compliance.

First, many states listed as CF/SF participants still require businesses to submit information returns directly through state portals or other state-specific channels. In those states, CF/SF forwarding does not satisfy the legal filing obligation.

1

Second, CF/SF does not support all forty-one information return types. Some states require returns that CF/SF does not accept or transmit at all, which creates reporting gaps.

2

Third, CF/SF uses routing logic that does not reflect how states actually assert filing jurisdiction. For example, if a payer issues a Form 1099-NEC to a Utah resident who performed services in California, both Utah and California generally require that return to be filed. Under CF/SF, the IRS would forward that record only to Utah based on the payee's address, leaving California unfilled.

3

For any state that requires direct filing, for any form type CF/SF does not support, or for any scenario where a state asserts jurisdiction outside of CF/SF routing rules, filers must submit returns separately using state-specific formats, due dates, thresholds, and reporting channels.

Organizations should expect this fragmented state environment to persist for multiple years as states migrate on different timelines and with different technical requirements.

This is not a new file format. It is a permanent change to the compliance operating model.



The Five Real Transition Challenges No One Budgets For

1

The first challenge is cross functional dependency failure. Tax, IT, security, data governance, and operations all have skin in this game. If any one of those teams slips, the entire filing season is at risk.

2

The second challenge is data maturity. Anything FIRE tolerated will surface immediately under real time validation. Inconsistent names, weak address hygiene, unclear entity structures, and legacy data shortcuts become filing blockers.

3

The third challenge is IRS dependency risk. Certification timelines are not fully under the filer's control. Schema changes reset approval clocks. Form specific validation behavior changes late in the cycle.

4

The fourth challenge is the annual retesting burden. This is not a one-season project. Every year brings new certification, new schemas, and new validation logic.

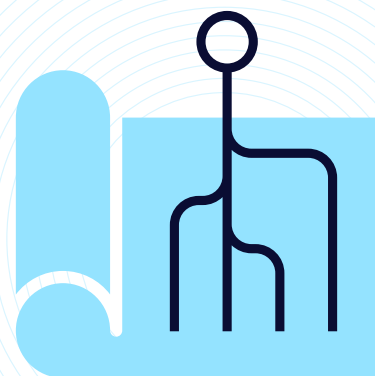
5

The final challenge is state divergence. Federal reporting modernizes first. State reporting lags behind. Most organizations must run parallel compliance models with constantly shifting exceptions.



The Four Forces That Drive Build Versus Buy Outcomes

Before looking at two common paths organizations are taking, it helps to name the four forces that quietly determine outcomes under IRIS. These show whether organizations model them or not.



1

Time to compliance

Under a build model, realistic development timelines are rarely under six to eighteen months. That does not include IRS testing, validation cycles, and certification, which typically add another three to six months. Late schema changes, form-specific validation failures, and certification resets can push timelines out further.

Under a buy model, vendor solutions are typically production-ready within weeks, with implementation timelines ranging from four to twelve weeks depending on complexity. That difference in time to first filing capability is often decisive when deadlines are fixed.

The practical question is simple. What happens if you miss your go-live date for IRIS.

2

Total cost of ownership

Build strategies often appear cheaper on paper because there is no upfront licensing. In reality, development cost is only the first layer.

True total cost includes engineering salaries, opportunity cost of pulling developers off other priorities, IRS certification cycles, annual retesting, schema updates, security patches, bug fixes, infrastructure, and the long-term cost of maintaining specialized compliance engineers.

Those costs are rarely linear. Most organizations experience cost expansion as IRIS evolves; new form types are added, and validation behavior changes. Over a multi-year horizon, total cost of ownership for a build path is often two to three times higher than initial estimates.

Buy strategies introduce predictable licensing and implementation fees, plus transparent subscription or maintenance costs. While those fees are visible and budgeable, they replace a long tail of internal development and compliance engineering costs that are otherwise difficult to forecast or control.

3

Scalability and flexibility

Build paths offer theoretical customization. In practice, every edge case, volume increase, new form type, or new jurisdiction introduces incremental development and architectural change.

As volumes grow and reporting regimes expand, internal systems often require re-architecture, performance tuning, and additional infrastructure investment. Technical debt accumulates, and flexibility comes at the cost of increasing complexity and slower change cycles.

Buy paths typically include built-in scalability for volume growth, pre-tested support for new forms and jurisdictions, and vendor-managed infrastructure scaling. Customization is more limited, but many vendors provide flexibility through configuration layers and APIs without requiring structural rebuilds.

The tradeoff is controlled evolution versus perpetual engineering.

4

Maintenance and regulatory change ownership

Under a build model, internal teams must monitor IRS guidance, schema updates, validation of rule changes, and filing program updates. They must implement changes continuously and retest every year for every type of form.

They also carry full responsibility for compliance accuracy, certification failures, missed changes, and filing season disruptions.

Under a buy model, the vendor absorbs responsibility for IRS requirement updates, schema changes, validation logic, annual certification, and platform stability. Compliance patches are delivered as part of the service, and regulatory change risk is externalized.

This is not just operational convenience. It is a governance decision about who owns regulatory risk.



Case Study

Two Paths Organizations Are Taking

The decision to rebuild or buy an IRIS-centric solution is substantial and represents a significant investment. The two paths below are meant to highlight the considerations and issues that are not always accounted for up front but can drastically impact readiness.



The Rebuild Path

One large financial institution began with a twelve-month timeline, four engineers, and a budget of roughly one and a half million dollars. Those estimates were not reckless. They were based on a standard internal cost model: fully loaded engineering costs of roughly \$180,000 to \$220,000 per developer per year, plus modest infrastructure and tooling.

What they underestimated was not engineering capability. It was the cumulative friction introduced by IRS onboarding, certification cycles, schema churn, and form-specific validation behavior.

Before a single line of XML could be tested, the organization had to complete IRS security and identity onboarding. Depending on who served as the Responsible Official on the IRS e-Services account, that process alone took months. In some cases, it required background checks, fingerprinting, mailed identity verification documents, and extended waiting periods for approval.

Even where organizations were able to designate a CPA or attorney as the Responsible Official to avoid background checks, the e-Services setup process still took roughly two months to resolve credentials, roles, and account access.

After that, the IRS advises filers to allow at least forty-five days to receive an IRIS Transmitter Control Code. Only once that TCC was issued could the filer receive the XML schemas, begin coding file formats, and build the A2A integration.

That is when the real work began.

Every form type introduced new validation failures. Each schema update forced re-mapping and re-testing. Each certification cycle uncovered new edge cases. Every time a form failed testing, the certification clock effectively reset.

The original four engineer teams doubled. Timelines expanded from twelve months to twenty-two months. Total spend crossed three million dollars once engineering salaries; extended timelines, infrastructure, external consultants, and repeated testing cycles were included.

And none of that produced a finished product.

What it produced was the first iteration of a permanently incomplete compliance system.

Under IRIS, this cycle never ends. Every year brings new schema versions, new validation rules, new form types, and a fresh round of certification testing for every form the organization issues.

That creates permanent technical debt. Each year, changes layer new complexity onto the prior year's logic. Each certification cycle introduces new rework. Each regulatory change forces another round of development and testing.

Their conclusion was simple. Building IRIS was not a one-time development project. It was the creation of a permanent compliance engineering program with open-ended cost, staffing, and risk exposure.

The Buy Path

Another institution also had the technical capability to build. What they did not have was the organizational appetite to own a permanent IRS compliance engineering program.

They concluded that the highest risk of work under IRIS was not file construction. It was everything that happens around it.

Real time validations would now reject records and sometimes entire files. That meant new operational failure modes: missed deadlines, rework cycles, unhappy vendors and customers, and filing season fire drills. Managing rejection behavior, interpreting IRS error responses, and operationalizing rapid remediation became a daily compliance function, not a once-a-year event.

They also recognized that the individual name format change created a new category of legal and financial risk. Poorly handled name restructuring could artificially inflate name and TIN mismatches, trigger CP2100 notices, force backup withholding, and create seven figure penalty exposure under Sections 6721 and 6722.

Mitigating that risk required new data hygiene rules, new exception handling workflows, and new solicitation processes to obtain updated Forms W-9.

IRIS also changed the timing of downstream processes. Instead of waiting months for IRS notices, response files and acknowledgments now arrive immediately. That forced the organization to design new operational workflows to ingest IRIS responses, reconcile records, trigger remediation tasks, and request updated name and TIN information in near real time.

Buying allowed them to focus internal teams on those highest risk business and compliance problems rather than on XML engineering and A2A infrastructure.

At the same time, buying externalized the permanent technical debt created by IRIS.

The vendor absorbed responsibility for schema updates, validation of logic changes, annual certification, security controls, platform stability, and form-by-form testing. As new form types were introduced, validation rules evolved, and filing programs changed; those updates were delivered as part of the service rather than as internal development projects.

That risk transfer mattered even more at the state level.

As states migrate to IRIS on different timelines, organizations must support federal XML alongside legacy state formats, build new state filing formats, maintain legacy formats, and add new delivery channels such as SFTP, portals, secure email, paper, and state-specific APIs.

Buying shifted that long tail transition burden to a third party rather than forcing the organization to build and maintain fifty different compliance pathways in parallel.

Longer term, the institution also viewed this as a scalability decision.

As they acquired new companies, launched new products, entered new jurisdictions, and became subject to new tax laws that introduced additional information returns; they did not want to repeat an IRIS build cycle repeatedly.

They concluded that IRIS created permanent technical debt no matter who built it. The only real choice was whether that debt would live internally or be absorbed by a vendor.

For them, buying was not outsourcing for convenience. It was a financially, operationally, and organizationally responsible way to contain permanent compliance risk and preserve internal capacity for higher value work.

The Decision IRIS Forces Organizations to Make

IRIS is not a project with a finish line.

It is a permanent shift in how information returns are created, validated, transmitted, and defended.

Every organization will carry permanent technical debt under IRIS. The only question is where it lives and who is accountable for it during the filing season.

The real risk is not converting a file to XML. The real risk is operational failure: rejected files, missed deadlines, state exceptions, customer and vendor fallout, and penalty exposure created by name and TIN mismatches.

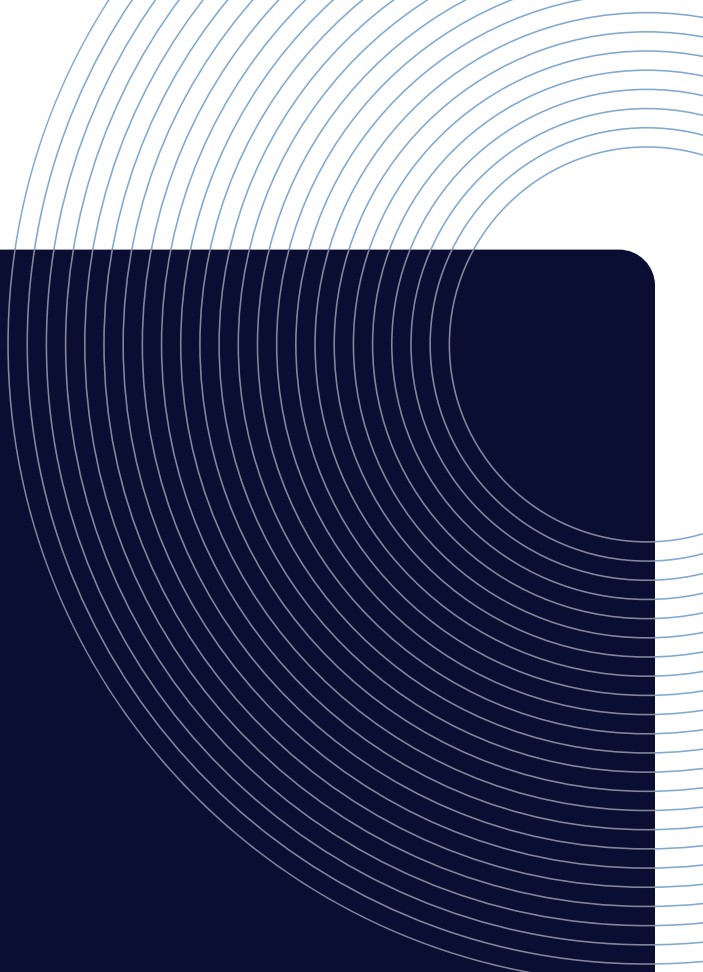
FIRE made information reporting feel like plumbing. IRIS makes it feel like infrastructure.

The organizations that treat IRIS as an operating model change will transition smoothly. The ones that treat it as a format upgrade will not.

Build versus buy is therefore not a software preference. It is a governance decision about who owns regulatory risk, who maintains certification readiness year after year, and who absorbs the long tail of state divergence and new tax law volatility.

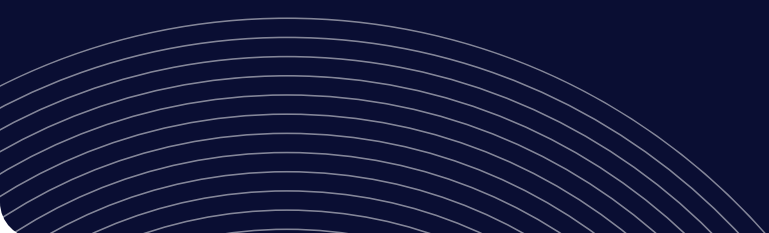


SOVOS



The IRIS Transition Playbook

How to Decide Whether
to Build or Buy Before
FIRE Is Retired



Contact Us: <https://sovos.com/contact-us/>

© 2026 Sovos Compliance, LLC.

SOVOS is a registered trademark of Sovos Compliance, LLC