LANDING.AERO

LANDING.AERO / RESEARCH & PRACTICAL GUIDANCE

Part 135 Operational Control Workflow: Quote to Flight

> LANDING.AERO is a Montreal software studio that builds custom, AI-assisted software for flight operations: operations tools, data integration, workflow automation and dashboards designed around how an airline, charter operator or flight department actually works.

Published by LANDING.AERO · 2026-10-07 · part 135 operational control workflow · part 135 charter operations · charter quote to flight · flight coordination · trip planning · operational control audit trail · charter flight operations software · schedaero · aeroquote

Original article: https://www.landing.aero/articles/en/part-135-quote-to-flight-handoffs

In this article

1. Executive Summary
2. Introduction and Background
3. Key Changes: Keep Commercial and Operational Decisions Separate
4. Canonical Identifiers and Sources of Record
5. Event Ordering and the Quote-to-Flight Handoff
6. Implementation Considerations and Process Changes
7. Buy, Configure or Build at the Remaining Boundary
8. Data Analysis and Evidence
9. Implications and Future Directions
10. Conclusion

Executive Summary

A **Part 135 operational control workflow** should carry an accepted commercial proposal into an operational record without treating acceptance as permission to fly. In the United States, operational control concerns authority over initiating, conducting or terminating a flight. (www.ecfr.gov) The certificate holder retains responsibility for operational control and, where a manual is required under §135.21, must list the name and title of each person it authorizes to exercise operational control. (www.ecfr.gov) (www.ecfr.gov) Authority designations should follow the applicable manual and OpSpecs requirements. (www.faa.gov) The practical integration question is whether those people receive the current trip, leg, aircraft and crew facts, recognize subsequent changes, and leave an attributable decision against the information reviewed.

Public documentation demonstrates why a **quote-to-flight handoff** needs more than a booking status. **Schedaero** distinguishes quotes, quoted trips, scheduled trips, flight legs, crew assignments and duty logs, and explicitly separates trip-level notifications from itinerary-leg changes. (developer.avinodegroup.com) **AeroQuote** describes bookings created from existing quotes or independently, establishing commercial workflow scope without establishing its integration guarantees. (aeroquote.com) These are documented capabilities, not evidence that an operator's installed configuration already maintains an operational-control decision trail.

The proposed design preserves operator-owned trip and leg references, maps external identifiers, records resource revisions separately from event identifiers, and requires human confirmation at operator-defined decision points. CloudEvents distinguishes an event's identity from its subject, while relational constraints can enforce record identity and relationships. (github.com) (www.postgresql.org) Duplicate delivery should produce no duplicate business action; delayed updates should trigger reconciliation rather than silently restore an obsolete state. Amazon documents idempotent processing as a response to repeated message receipt. (docs.aws.amazon.com) An append-only journal can preserve decisions and later corrections, but this is an engineering recommendation, not a newly imposed aviation record requirement. (learn.microsoft.com)

The quantitative evidence describes obligations and documented interface behavior rather than claimed efficiency gains. Avinode documents unsuccessful delivery expiry after **48 hours** and automatic inactivation for zero successful deliveries in a rolling **10-day** window. (developer.avinodegroup.com) Section **135.63** specifies

record-dependent periods, including **30 days** for completed load manifests; these periods do not establish universal event-journal retention. (www.ecfr.gov) Procurement should therefore proceed from a demonstrated boundary: buy a suite when its workflow fits, configure native integration when it carries the required states, and commission a narrow bridge when a verified gap remains. LANDING.AERO's published model is custom software built around an operator's needs. (www.landing.aero) No public evidence reviewed establishes comparable implementation prices, operator-specific entitlements or measured handoff improvements.

Introduction and Background

The charter inquiry usually begins with commercial questions: where the customer wants to travel, when, with which passengers, and under what terms. An operations manager needs a different answer: which current operational record is being considered, what changed, who reviewed it, and who has the relevant authority. Schedaero's documented model makes the distinction tangible: a trip is a container that can hold multiple specific quotes. (developer.avinodegroup.com) The same commercial inquiry can therefore produce alternatives that must not be confused with the scheduled execution record.

This report examines **Part 135 charter operations workflow**, **charter quote to flight workflow** and the data boundary between sales and operational control. The **Part 135 trip planning process** supplies the changing facts; **Part 135 flight coordination handoffs** carry them to the appropriate people. It addresses directors of operations, chief pilots, flight coordinators and their information technology teams. It is a systems analysis of handoffs, rather than an operational procedure for authorizing a particular flight. The FAA explains that an operator's scope is authorized through its **Operations Specifications**, commonly called **OpSpecs**. (www.faa.gov) The applicable manual, OpSpecs and actual assigned authority must govern any workflow implementation.

A **source of record** means the system or controlled process designated to author a particular fact. A **webhook** is a notification sent to another application when something changes. A webhook receipt is evidence of message handling, whereas a human acknowledgment records a person's response to the relevant business or operational information. Google's messaging documentation similarly distinguishes message delivery from subscriber acknowledgment, although its service contract cannot be attributed to aviation vendors. (docs.cloud.google.com)

As of October 2026, this analysis uses opened regulatory sources, vendor documentation and official engineering references. Historical product announcements are treated as historical: Schedaero's API announcement dates to **June 4, 2018**. (schedaero.com) Current entitlement and contract terms still require operator-specific verification.

No operator dataset was supplied. The worked trip, identifiers, revision labels and acceptance tests below are **hypothetical design examples**, not observed production performance. The central purchasing question is testable: can the existing system preserve the relevant identity, current state and human decision through every change the operator actually handles?

Key Changes: Keep Commercial and Operational Decisions Separate

Change the state model, not the allocation of authority

The **Part 135 flight authorization process** must reflect the operator's controlled procedures and designated authority. Its **Part 135 operational control responsibilities** remain the starting point for the application design.

The “changes” here are proposed process and data-design changes, not claims of a recent amendment. **14 CFR 135.77** makes the certificate holder responsible and requires the manual to list each authorized person's name and title. (www.ecfr.gov) Cornell's independently opened reproduction corroborates the naming requirement. (www.law.cornell.edu) A quote acceptance, payment event, salesperson action or scheduling-system status should therefore remain a commercial or administrative fact unless the operator's governing procedures assign it a particular operational meaning.

The system should make separate questions visible:

- **Commercial acceptance:** Which proposal and terms did the customer accept?
- **Scheduling:** Which operational trip and legs represent the intended execution?
- **Review:** Which current aircraft, crew and trip facts were presented?
- **Authority:** Which authorized person made the relevant decision?
- **Change response:** Which later edits require renewed attention under the operator's procedures?
- **Evidence:** Can a reviewer reconstruct the information and decision without relying on today's screen?

These are proposed application distinctions. They do not prescribe a universal release procedure or imply that every charter operator has an airline-style dispatch release. The regulation's definition turns on authority over initiating, conducting or terminating a flight. (www.ecfr.gov) Software should represent the operator's actual allocation of that authority.

Link permissions to controlled procedures

14 CFR 135.23 requires the manual to identify management duties, responsibilities and authority, including people authorized to exercise operational control. (www.ecfr.gov) The access model should reference the operator's maintained designation rather than infer authority from a job title, a software administrator role or the ability to edit a schedule.

FAA certification guidance identifies named management positions for a Standard Part 135 applicant. (www.faa.gov) That management structure is relevant context, but it does not justify a rule that every scheduling action automatically belongs to a particular management title. The implementation team should document how the operator's manual maps people, functions and application permissions.

Useful implementation checkpoints are:

- **Role provenance:** Identify the controlled source for authorized-person designations.
- **Effective dates:** Preserve when a person's application authority became effective or ended.
- **Decision scope:** Bind acknowledgment to the trip or leg state actually reviewed.

- **Delegation evidence:** Follow the operator's documented authority model.
- **Interface vocabulary:** Define what “accepted,” “scheduled,” “reviewed” and any vendor “released” status mean locally.
- **Change ownership:** Assign responsibility for returning a changed record to the appropriate person.

Section **135.21** generally requires a current procedures-and-policies manual, subject to its stated exceptions and authorized deviations. (www.ecfr.gov) Its accessibility provisions also matter when people perform assigned duties. (www.ecfr.gov) Manual access and evidence capture should be designed together: users need to see the applicable procedure while the journal identifies the revision consulted. The electronic manual's revision-date requirement provides a specific regulatory reference for that distinction. (www.ecfr.gov) An integration should point users to the controlled procedure revision rather than embed a second, independently maintained operational manual.

Canonical Identifiers and Sources of Record

Preserve identity while allowing facts to change

A **canonical identifier** is an operator-owned stable reference used to relate records across systems. It should survive changes in dates, crew assignment, aircraft assignment and commercial terms when those changes still concern the same underlying trip. A new quote alternative or genuinely new leg may need its own reference. The operator must define those identity rules before selecting an adapter.

External identifiers should be stored as exact opaque strings. Avinode explicitly says their format is unspecified and may change. (developer.avinodegroup.com) Parsing a trip number out of an external identifier creates an avoidable dependency on undocumented structure. A cross-reference can instead store the issuing system, account, object type, external identifier and canonical reference.

A **Universally Unique Identifier (UUID)** is an available internal-reference format; its standard also recommends avoiding unnecessary parsing. (www.rfc-editor.org) Vendor identifiers and internal identifiers should both be treated as references, although their issuing authorities differ. (www.rfc-editor.org) It is an option, not an aviation requirement. PostgreSQL's primary-key and foreign-key documentation illustrates how uniqueness and referential integrity can protect linked records. (www.postgresql.org) (www.postgresql.org) Those controls do not establish that the underlying operational facts are correct.

Table 1 is a hypothetical source-of-record matrix to be completed against the operator's actual systems.

Fact or record	Proposed authoritative owner	Reference and version to preserve	Required confirmation
Quote and terms	Commercial quoting application	Quote reference, revision and accepted revision	Sales verifies customer acceptance refers to the correct proposal.
Scheduled trip	Operator-designated scheduling system	Canonical trip reference plus vendor cross-reference	Operations confirms which scheduled record executes the accepted proposal.
Flight leg	Designated scheduling or operations system	Stable leg reference, parent trip and leg revision	Relevant person acknowledges consequential itinerary changes.

Fact or record	Proposed authoritative owner	Reference and version to preserve	Required confirmation
Person and crew assignment	Personnel source for identity; scheduling source for assignment	Stable person reference and separate assignment reference	Designated staff verify reassignment under applicable procedures.
Aircraft identity and assignment	Aircraft register for identity; schedule for assignment	Aircraft reference and versioned assignment	Responsible staff assess a swap through the operator's existing process.
Aircraft status inputs	Operator-designated controlled technical source	Source reference, status observation and observation time	Qualified people handle relevant technical judgments.
Operational decision	Controlled operator workflow	Decision reference, actor, scope and reviewed revisions	Authorized person confirms the decision; software records it.
Cancellation	Commercial and operational owners for their respective effects	Cancellation reference, affected objects and acknowledgment status	Each designated owner confirms effects within their responsibility.

The matrix deliberately separates **identity**, **assignment**, **status** and **decision**. A crew member is not an assignment; an aircraft's registration is not a current status observation. Section **135.63** requires individual pilot records with specified information, reinforcing the need to distinguish personnel evidence from a schedule entry. (www.ecfr.gov) The operator should settle ownership field by field instead of making every system editable for every fact.

Separate identity, revision and schema version

An event identifier answers “which notification?” A resource revision answers “which state of the trip or leg?” A schema version answers “which message format?” CloudEvents requires uniqueness for the combination of event source and identifier, and treats the subject as a separate context attribute. (github.com) (github.com) These concepts are useful even when a vendor uses another format.

The cross-reference should retain:

- **Namespace:** Issuing system, operator account and object type; preserve the event producer's context when applying an identity rule. (github.com)
- **Resource reference:** Canonical identity and the external identity supplied by the vendor; relate child records to the correct parent. (www.postgresql.org)
- **Resource revision:** The documented revision, or a clearly labeled local snapshot revision.
- **Event identity:** A delivery/event identifier when the interface supplies one.
- **Schema identity:** The payload contract used to interpret the event. (learn.microsoft.com)
- **Time evidence:** Source occurrence time, receipt time and processing time as distinct fields with explicit UTC or offset representation. (www.rfc-editor.org)

A local snapshot counter must not be presented as a vendor-guaranteed sequence. Similarly, CloudEvents' recommendation to change a schema URI for incompatible schema changes supports schema governance, not a claim about Schedaero's payload contract. (github.com)

Event Ordering and the Quote-to-Flight Handoff

Follow a changing trip through explicit transitions

Schedaero's event catalog documents separate quote, quoted-trip, scheduled-trip, flight-leg, crew-assignment and duty-log notifications, and states that the flight-leg webhook does not report changes to its associated trip. (developer.avinodegroup.com) Subscription coverage should therefore be checked by object and action, including cancellation and assignment changes, rather than by an umbrella label such as “trip integration.”

Table 2 follows one trip **(Hypothetical Example)**. Reference letters and revision labels are invented for illustration. Revisions refer to the operator's proposed local snapshots; they do not claim a vendor exposes these exact version fields. The underlying need to keep event-version handling explicit is documented in Microsoft's event-sourcing pattern. (learn.microsoft.com)

Transition	Identity and revision treatment	Proposed owner	Acknowledgment or evidence
Inquiry creates quote Q-A	Retain inquiry link and Q-A revision A	Sales	Record assumptions and requested itinerary.
Quote is revised	Keep Q-A; append revision B	Sales	Acceptance must identify B rather than whichever revision is currently displayed.
Customer accepts	Preserve Q-A/B acceptance evidence	Sales	Record actor, accepted terms and acceptance source.
Scheduled trip is created	Create T-A and legs L-A/L-B; map to Q-A/B	Scheduling	Operations confirms the mapping and operational record to review.
Departure or destination changes	Keep affected leg identity if still the same leg; append its next snapshot	Scheduling	Route change to the relevant reviewer under operator procedures.
Crew is reassigned	Keep person identities; end old assignment and record new assignment	Crew coordination	Record acknowledgment against current assignment and trip/leg state.
Aircraft is swapped	Keep T-A and existing legs; change assignment and re-evaluate affected inputs	Operations and designated technical owners	Relevant staff confirm the new context through established procedures.
Customer cancels	Preserve T-A, affected legs and prior history; append cancellation states	Sales and operations	Record separate commercial and operational acknowledgments; surface remaining dependencies.

The important continuity is the chain from **accepted quote revision** to **scheduled trip**, **affected leg**, **current assignments** and **decision scope**. Identity normally survives a reschedule; the information associated with it changes. Where a split, merge or replacement makes identity ambiguous, the operator should define an explicit lineage relation rather than quietly reuse a reference.

The adapter must not assume that a notification is a complete state record. Schedaero's integration guide says the notification reports that something changed without carrying the actual change. (developer.avinodegroup.com) The receiver should retrieve the documented resource, preserve the permitted

evidence, compare it with its current projection and route consequential differences. A missing required field should create a visible exception, not a fabricated default.

Handle delivery uncertainty without manufacturing certainty

Idempotency means processing the same input again does not create an additional business effect. Amazon's repeated-delivery guidance recommends idempotent applications. (docs.aws.amazon.com) (docs.stripe.com) Stripe independently documents avoiding duplicate event processing by remembering processed event identifiers. (docs.stripe.com) Neither source establishes the delivery contract of an aviation application; together they demonstrate why an adapter should test duplicate inputs rather than assume uniqueness.

A durable receipt and its business processing should be separate states. Avinode recommends saving the webhook notification and responding with **HTTP 200** for successful consumption. (developer.avinodegroup.com) Successful transport acknowledgment should not be shown as a person's acknowledgment of the changed itinerary.

For ordering, Stripe expressly does not guarantee generation-order delivery, while Google Pub/Sub's default delivery has no ordering guarantee. (docs.stripe.com) (docs.cloud.google.com) The aviation vendor's own guarantee must be verified. Without a documented monotonic revision, a receiver should reconcile against the authoritative current resource and mark unresolved ordering questions for review.

A proposed processing sequence is:

- **Receive:** Validate sender, destination and allowed event category.
- **Persist:** Save a durable envelope and enough permitted evidence to recover processing.
- **Deduplicate:** Apply a documented event key, or an explicitly tested alternative. (docs.aws.amazon.com) Schedaero's integration guide requires using the IDs sent in the webhook to retrieve updated information. (developer.avinodegroup.com) For the proposed Schedaero adapter, do not assume a notification's resource ID identifies a unique change or permanently suppress later notifications carrying that ID; retrieve and reconcile the current resource, and prevent duplicate business effects when the retrieved state is unchanged.
- **Retrieve:** Read the authoritative resource when the notification lacks the changed state.
- **Compare:** Detect relevant differences without assuming local arrival order equals source order.
- **Project:** Update the current view with traceable links to evidence.
- **Route:** Send consequential changes to the designated human workflow.
- **Reconcile:** Compare current source records with the local view after gaps or exceptions; verify what retained inputs can be replayed. (docs.cloud.google.com)

A composite database uniqueness constraint can help enforce a deduplication key. (www.postgresql.org) It cannot prove that a poorly selected key represents a unique event.

Implementation Considerations and Process Changes

Build a decision journal with defined retention and access

A **Part 135 operational control audit trail** should preserve the recorded decision and append later corrections or supersession rather than rewrite history. Microsoft's pattern uses append-only storage for an audit trail and compensating events for corrections. (learn.microsoft.com) (learn.microsoft.com) Full event sourcing of the entire sales application is not necessary to achieve a focused decision journal.

The journal should record what was reviewed, by whom, with which authority reference, against which relevant resource snapshots, and what subsequently changed. The National Institute of Standards and Technology's **NIST SP 800-53 Revision 5** audit-content control provides a generic reference for event type, time, location, source, outcome and associated identities. (nvlpubs.nist.gov) This reference is an engineering aid, not a Part 135 adoption requirement.

Keep the distinction between an append-only application interface and storage protection. A database administrator might still alter data unless access and retention controls prevent it. The same NIST edition addresses protection of audit information against unauthorized access, modification and deletion. (nvlpubs.nist.gov) A useful implementation must explain who can append, who can read, who administers storage, and how corrections remain attributable.

Retention requires its own design. Avinode says API contracts can regulate how long returned data may be stored. (developer.avinodegroup.com) The operator should agree which raw payloads, normalized snapshots, references, hashes and decision records may be retained. A hash can help detect changes but cannot reconstruct missing information by itself. Retention of a human decision is not permission to keep every vendor response indefinitely.

Practical journal requirements include:

- **Actor:** Stable identity plus the effective authority reference. (nvlpubs.nist.gov)
- **Scope:** Trip, leg, assignment or other object actually reviewed.
- **Evidence:** Snapshot references and relevant procedure revision.
- **Outcome:** Decision, acknowledgment, exception or supersession.
- **Timing:** Source, receipt and decision timestamps with explicit meaning; use synchronized clocks. (www.rfc-editor.org)
- **Correction:** Append reason and replacement linkage without silently editing the earlier record.
- **Protection:** Separate permissions for operational actions and journal administration. (nvlpubs.nist.gov)
- **Retention:** Apply the approved, record-specific schedule and contractual permissions.

For timestamp interoperability, **Request for Comments (RFC) 3339** defines numeric offsets as local time minus Coordinated Universal Time, or **UTC**. (www.rfc-editor.org) Its synchronization discussion supports avoiding ambiguous local timestamps in exchange records. (www.rfc-editor.org) Preserve an explicit offset or UTC representation, and distinguish clock time from event order.

Test behavior across changes and cancellation

Automated tests for this vendor must respect its published test policy: Avinode prohibits automatic tests from calling either API environment and requires mock substitution. (developer.avinodegroup.com) The worksheet below is therefore a **mocked acceptance-test design (Hypothetical Example)**. Any manual sandbox verification should follow vendor-approved access and use cases.

Each case needs an input fixture, starting state, expected current view, expected journal entries and expected acknowledgment status:

- **Duplicate:** Deliver the same fixture again; retain transport evidence without duplicating assignments or human tasks. (docs.aws.amazon.com)
- **Out of order:** Deliver a newer state before an older notification; the older input must not silently restore obsolete facts. (docs.stripe.com)
- **Missing notification:** Change the authoritative fixture without an event; reconciliation must identify the difference.
- **Missing resource:** Notify an unavailable identifier; retain an exception and do not invent the trip.
- **Leg-only change:** Change a leg with unchanged trip fields; verify the leg path reaches the correct review queue.
- **Two updates to the same leg ID:** In the mocked fixtures, apply two legitimate updates to the same leg ID and deliver a notification after each. Both notifications must trigger retrieval and reconciliation. Then repeat a notification with identical retrieved state; it must not duplicate assignments, human tasks or other business effects.
- **Crew reassignment:** Replace an assignment while retaining person identity; preserve the prior assignment's history.
- **Aircraft swap:** Change the assignment; surface dependent inputs for the operator's established review.
- **Cancellation:** Cancel commercially and operationally in separate fixtures; show which acknowledgment remains outstanding.
- **Late post-cancellation edit:** Record the event without implicitly reactivating the canceled trip.
- **Schema extension:** Add an unknown optional property; preserve compatibility and detect missing mandatory data.

Avinode states that additional JSON properties may be introduced while existing properties will not change. (developer.avinodegroup.com) That is a documented compatibility statement, not an ordering or resource-revision guarantee. CloudEvents similarly treats schema evolution as a separate concern. (github.com)

The test result should be a reviewable artifact: fixture version, observed result, unresolved exception and responsible owner. A green transport test alone cannot demonstrate that an authorized person received the right operational context.

Buy, Configure or Build at the Remaining Boundary

The **charter flight operations software** procurement decision should start with the installed workflow and a complete transition demonstration. Schedaero's current integrations page offers API connections and existing integrations. (schedaero.com) ForeFlight's current scheduling-integrations page lists Schedaero and describes

scheduled-flight ingestion into flight plans. (ba.foreflight.com) Those documented connections justify evaluating a native path first, while leaving cancellation semantics, decision history and operator-specific enablement to verification.

Table 3 compares implementation routes. It is a decision framework, not a price ranking. **LANDING.AERO** appears because custom flight-operations integration falls within its direct offering; its website describes bridges between legacy systems and modern applications. (www.landing.aero) The brand is an implementation option when a custom boundary exists, rather than an off-the-shelf charter suite.

Route or option	Appropriate condition	Evidence required before selection	Commercial basis to obtain
Single suite, such as Schedaero or AeroQuote	The suite demonstrably covers the operator's sales and operational workflow	End-to-end transition demonstration, access roles, history export and cancellation handling	Operator-specific licenses, configuration, migration and support scope
Native API/webhook configuration	Existing products remain authoritative and documented integration covers the required objects	Enabled subscriptions, permissions, current schemas and tested reconciliation responsibilities	API entitlement, connector scope and operating costs
Narrow custom bridge, including LANDING.AERO	A verified cross-system gap remains after evaluating native coverage	Bounded adapter specification, controlled decision workflow, retention agreement and acceptance results	Scoped build and ongoing ownership agreement

The routes are not mutually exclusive. A suite may remain the commercial and scheduling core while a configured connection serves a flight-planning tool and a small bridge carries an operator-specific acknowledgment. AeroQuote's booking page describes creation from an existing quote or independently. (aeroquote.com) That capability should prompt a workflow demonstration; it should not be extended into an unverified claim about public APIs or audit behavior.

Before estimating a build, the operator should resolve:

- **Access:** Which operations and objects are actually permitted?
- **Coverage:** Which trip, leg, assignment and cancellation transitions are represented?
- **Responsibility:** Who runs reconciliation and resolves exceptions? Distinguish message acknowledgment from the operational response. (docs.cloud.google.com)
- **Authority:** Where is the controlled designation of authorized people maintained? (www.ecfr.gov)
- **Evidence export:** Can the operator reconstruct a decision after a system change? Include its controlled procedure context. (www.ecfr.gov)
- **Ownership:** Who maintains schemas, credentials, mappings and monitoring?

Avinode documents permission-restricted connections associated with a member account. (developer.avinodegroup.com) It also requires implemented use cases to be included in the API subscription. (developer.avinodegroup.com) These requirements make contract and account verification a prerequisite to estimating a production adapter.

LANDING.AERO's about page describes tailored flight-operations software and data-source integration. (www.landing.aero) This first-party description establishes its offering category; it does not establish a specific integration, certification, customer result or cost. The same evidentiary standard should apply to every implementation partner.

Data Analysis and Evidence

The available quantitative evidence is **interface timing, record-specific retention and identifier structure**. It is not an industry benchmark for handoff speed, staffing savings or software return on investment. No operator event log, comparable implementation-cost dataset or independently measured vendor performance comparison was available for this report.

Delivery expiry and subscription health. Avinode states that unsuccessful webhook consumption stops after **48 hours**. (developer.avinodegroup.com) It separately classifies subscriptions with zero successful deliveries in a rolling **10-day** window as unhealthy and inactivates them. (developer.avinodegroup.com) These mechanisms answer different questions: an individual event can age out, while subscription health measures successful deliveries over a longer interval. A proposed monitor should track subscription status, consumption failures and unresolved reconciliation exceptions separately.

Clock tolerance and authentication. Avinode's API Basics says a request timestamp more than **5 minutes** off at receipt is rejected. (developer.avinodegroup.com) Its OAuth documentation gives a default access-token lifetime of **30 minutes**, with optional lifetimes from **1 to 120 minutes**. (developer.avinodegroup.com) These values support testing clock synchronization and credential refresh as integration concerns. They are not operational decision deadlines.

Test-environment persistence. Avinode says the sandbox receives a fresh database at least **weekly**, deleting test-created or modified data. (developer.avinodegroup.com) Durable test evidence should therefore reside in permitted fixtures and acceptance records, rather than depend on sandbox objects remaining available. This does not change the requirement to mock automated API calls.

Record retention. Section **135.63(b)** specifies at least **6 months** for its referenced aircraft-list records and at least **12 months** for the referenced pilot and required flight-attendant records. (www.ecfr.gov) The section also requires completed load manifests to be kept for at least **30 days**. (www.ecfr.gov) These are different record classes; copying any of these durations into a universal webhook or authorization-journal retention setting would require additional justification.

Identifier capacity. RFC **9562** defines a UUID as **128 bits** and states it requires no central registration process. (www.rfc-editor.org) This supports considering an operator-owned identity namespace, while collision handling, access control and cross-reference correctness remain implementation responsibilities. A long identifier is not proof that a record matches the correct customer request.

For comparison, Stripe documents live-mode webhook retries for up to **three days**. (docs.stripe.com) That different contract demonstrates why generic “webhooks retry” language is insufficient. Each vendor's enabled interface needs its own delivery, recovery and retention agreement.

The operator can collect useful local measurements without inventing external benchmarks:

- **Coverage:** Count relevant source changes and those represented in the receiving system.
- **Acknowledgment age:** Measure elapsed time from a consequential change to the designated acknowledgment.
- **Reconciliation backlog:** Count unresolved differences and record their age.
- **Identity completeness:** Measure records whose required cross-references are present.
- **Evidence completeness:** Measure sampled decisions with actor, scope and reviewed-state references.

These proposed measures are diagnostic, not published results or universal service targets. Report their denominator, observation window, excluded records and clock assumptions before using them to justify a purchase.

Implications and Future Directions

The most useful integration boundary may be smaller than a replacement platform. If sales and scheduling already share correct trip data, the remaining work might be a controlled acknowledgment, an exception queue or a reliable cross-reference to an existing operational system. If identity and state are already complete, a custom bridge adds obligations without resolving a demonstrated gap. The operator should require evidence of the boundary before choosing an implementation route.

The architecture should remain recoverable. Its reference model should avoid dependencies on parseable identifier components. (www.rfc-editor.org) Google's documentation describes retention-supported replay of previously published messages. (docs.cloud.google.com) An operator adapter should verify its own permitted replay source and separately test reconstruction of the current view. Replaying inputs must not repeat commercial acceptance, reactivate a canceled trip or manufacture a fresh human decision.

Software auditability also depends on governance beyond event capture. NIST describes its control catalog as flexible and customizable. (csrc.nist.gov) Its Revision 5 write-once enhancement illustrates stronger storage protection, but choosing that mechanism requires a specific threat, retention and operational assessment. (nvlpubs.nist.gov) The report does not claim that adopting a named security control satisfies aviation requirements.

Future automation should distinguish drafting assistance from authoritative action. An automated summary could help highlight differences, but the reviewed-state references and responsible person's decision should remain explicit. A generated narrative should not replace the underlying records or turn a model's confidence into operational authority. This is a proposed design limit, not an assertion that any reviewed vendor uses such automation.

Manual revision control remains relevant as systems evolve. Section **135.23** requires the last revision date of an electronically accessed manual to be immediately ascertainable. (www.ecfr.gov) A workflow redesign should therefore identify the controlled procedure version that defines its acknowledgments and escalation path.

The next practical step is an operator-owned workshop using a recent anonymized trip, if available. Trace the accepted quote, scheduled trip, leg change, reassignment, swap and cancellation through the actual installed systems. Document the missing evidence and responsible owner. Select the smallest maintainable change

that resolves that demonstrated gap, then test it against the agreed transition set.

Conclusion

A usable Part 135 operational control workflow preserves the distinction between **commercial commitment**, **scheduled execution** and **the certificate holder's decision**. US rules place operational-control responsibility with the certificate holder and require identification of its authorized people. (www.ecfr.gov) The proposed data architecture supports those people by making the current context and subsequent changes traceable.

The hypothetical trip shows why a handoff cannot stop at “booking created.” A revised quote needs an identifiable accepted version. A scheduled trip needs stable relations to its legs. Crew and aircraft changes need attributable assignment history. Cancellation needs explicit downstream handling. The decisive test is whether the operator can reconstruct what changed, who received it and which state the responsible person reviewed.

Procurement should follow that evidence. A demonstrated suite workflow can justify buying within the suite. An enabled native connection can justify configuration. A remaining cross-system gap can justify a narrowly specified custom bridge. Public feature descriptions alone do not establish the installed workflow's permissions, event guarantees, decision evidence or recovery behavior.

The acceptance criteria should cover duplicate, missing and delayed notifications alongside itinerary changes, reassignment, swaps and cancellation. Retention and access should be agreed before collecting payloads indefinitely. Local measurements should describe observed coverage and acknowledgment behavior, without borrowing vendor marketing claims as benchmarks.

The result is a reviewable chain of identity, current facts, human confirmation and preserved history. Where a manual is required under §135.21, it must identify the people authorized to exercise operational control. (www.ecfr.gov) (www.ecfr.gov) Authority designations should follow the applicable manual and OpSpecs requirements. (www.faa.gov) Software carries and records that chain; operational authority remains with the people and certificate holder designated through the operator's governing procedures.

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. https://www.ecfr.gov/current/title-14/chapter-I/subchapter-A/part-1/section-1.1
2. https://www.ecfr.gov/current/title-14/chapter-I/subchapter-G/part-135/subpart-B/section-135.77
3. https://www.ecfr.gov/current/title-14/chapter-I/subchapter-G/part-135/subpart-A/section-135.21
4. https://www.faa.gov/licenses_certificates/airline_certification/135_certification/general_info
5. https://developer.avinodegroup.com/docs/schedaero-webhooks
6. https://aeroquote.com/features/bookings-management/
7. https://github.com/cloudevents/spec/blob/main/cloudevents/spec.md?plain=1
8. https://www.postgresql.org/docs/18/ddl-constraints.html

9. https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/standard-queues-at-least-once-delivery.html
10. https://learn.microsoft.com/en-us/azure/architecture/patterns/event-sourcing
11. https://developer.avinodegroup.com/docs/getting-started-webhooks
12. https://www.ecfr.gov/current/title-14/chapter-I/subchapter-G/part-135/subpart-B/section-135.63
13. https://www.landing.aero/
14. https://developer.avinodegroup.com/docs/requests-to-your-quote-management-tool
15. https://docs.cloud.google.com/pubsub/docs/subscription-overview
16. https://schedaero.com/schedaero-launches-apis/
17. https://www.law.cornell.edu/cfr/text/14/135.77
18. https://www.ecfr.gov/current/title-14/chapter-I/subchapter-G/part-135/subpart-A/section-135.23
19. https://www.faa.gov/licenses_certificates/airline_certification/135_certification/general_req
20. https://developer.avinodegroup.com/docs/api-basics
21. https://www.rfc-editor.org/rfc/rfc9562.html
22. https://www.rfc-editor.org/rfc/rfc3339.html
23. https://docs.stripe.com/webhooks
24. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf
25. https://schedaero.com/schedaero-integrations-api/
26. https://ba.foreflight.com/integrations/scheduling
27. https://developer.avinodegroup.com/docs/introduction
28. https://www.landing.aero/about/
29. https://developer.avinodegroup.com/docs/oauth-management
30. https://developer.avinodegroup.com/docs/sandbox
31. https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final

THE PUBLISHER

About LANDING.AERO

LANDING.AERO builds custom software for aviation operations teams at airlines, charter operators and flight departments. It is based in Montreal.

What it builds

Operations tools that reduce manual steps in day-to-day flight operations; integration of existing systems and data sources into one operational picture; workflow automation; real-time operational dashboards; bridges between legacy systems and modern applications; and mobile apps for staff in the field. Each application is designed around the operator's own workflows rather than sold as an off-the-shelf platform.

How it works

Work starts with discovery of the operator's workflows, continues with iterative development alongside the operator's team, and includes integration with the tools already in use and ongoing support.

Its own tools

LANDING.AERO also publishes small tools for aviation professionals, including UTC Alarm, which sets alarms in Zulu time, and NOTAM Notifier, announced for NOTAM change alerts on chosen airports.

Visit

See LANDING.AERO and its projects.

Articles published here explain general practice and cite their primary sources, naming the jurisdiction of any rule they describe. They are not operational, regulatory or legal advice, do not replace an operator's approved procedures or manuals, and requirements change.

Website: https://www.landing.aero

This article is educational. Verify consequential medical, legal, financial, regulatory and technical decisions with appropriate professionals and the cited primary sources. Company services are described separately from the article's research findings.