FREE SCORED CHECKLIST

The Data Exchange Readiness Checklist

Twenty-eight scored checkpoints across seven sections, for teams moving data between internal systems, cloud platforms and trading partners who need to know which interface will fail first.

Gridwise Technology · gridwisetechnology.com · 28 checkpoints · 56 points · 7 sections

Data exchange is the part of the estate that nobody owns and everybody depends on. Files land on a server at 2am, an API call runs on a schedule set five years ago, a partner sends a feed in a format that was agreed by email, and none of it appears on an architecture diagram. It works until a certificate expires, a field changes width, or a partner switches platforms. This checklist scores how ready your exchanges are for those ordinary events.

HOW TO SCORE

There are **28 checkpoints** in **7 sections of 4**. Score each **0, 1 or 2**. **0** means absent or unknown. **1** means true for some interfaces but not all, or true but never verified. **2** means true for every production interface, verified within the last twelve months, with evidence you could produce. Each section carries a maximum of **8 points** and the checklist a maximum of **56 points**. Record all seven section subtotals. **The lowest-scoring section is your constraint**: in data exchange the weakest layer determines the reliability of every interface that passes through it, so closing it is worth more than improving anything else.

SCORE AGAINST THE WHOLE SET, NOT THE FLAGSHIP

Almost every organization has one well engineered integration built recently by people who cared. Score the others. The interface most likely to cause your next incident is the oldest one, running on a schedule, moving files, and maintained by nobody since the person who built it left.

Section 1: Inventory and ownership (maximum 8)

- [ ] 1. A single register lists every production data exchange, including scheduled file transfers, API integrations, database links, event streams and manual exports.
  Manual exports belong on the list. A monthly spreadsheet emailed to a partner is a data exchange with no controls at all.
- [ ] 2. Each exchange has a named business owner and a named technical owner, and both know they hold the role.
  Test this by asking the named person. Ownership assigned in a document nobody has read is not ownership.
- [ ] 3. Each exchange records what data it carries, its classification, its direction, its frequency, and the systems at both ends.
  This is the field that turns an incident question, 'what did we just send', into a lookup rather than an investigation.
- [ ] 4. The register records the business impact of each exchange failing, and for how long that failure is tolerable.
  Without it, every failed job looks equally urgent at 3am, which means the important one gets the same response as the trivial one.

Section 2: Interface contracts and change (maximum 8)

- [ ] 5. Every exchange has a documented data contract specifying fields, types, formats, required and optional elements, encodings and expected volumes.
  Agreed by email years ago does not count. The contract needs to exist somewhere both parties can find it today.
- [ ] 6. Interfaces are versioned, and a version change follows a defined process rather than a coordinated deployment on a Friday.
  Versioning is what allows the two ends to change at different times, which is the entire point of an interface.
- [ ] 7. A change to a data contract requires notice to consumers, with an agreed notice period and a compatibility window.
  The most common cause of integration failure is an upstream change that was announced internally and not to the consumers.
- [ ] 8. Consumers of each data set are known, so the blast radius of a change can be established before it is made.
  If you cannot list who reads a data set, you cannot change it safely and you probably cannot retire it either.

Section 3: Transport and connectivity (maximum 8)

- [ ] 9. All exchanges use encrypted transport, and any legacy plaintext protocol still in use is documented with a funded removal date.
  Unencrypted file transfer to partners persists far longer than most teams believe. Check what is actually configured, not what is standard.
- [ ] 10. Certificates, keys and their expiry dates are inventoried with alerting far enough ahead to act, and a named owner per certificate.
  Expiry is the single most predictable outage in this entire checklist and the one most frequently missed.
- [ ] 11. Endpoints are restricted to known sources, and any inbound listener exposed to the internet is justified, owned and reviewed.
  Legacy file transfer servers exposed for partner convenience are a recurring root cause of serious incidents.
- [ ] 12. Network paths for exchanges are documented, including any dependency on a single circuit, gateway or third party service.
  Redundant connectivity that both interfaces route through the same appliance is not redundant.

Section 4: Identity, credentials and access (maximum 8)

- [ ] 13. Every exchange authenticates using a dedicated identity scoped to that purpose, not a shared or personal account.
  Integrations running under a departed employee's account are common and fail the moment offboarding is done properly.
- [ ] 14. Credentials, keys and tokens are stored in a managed secret store, not in scripts, configuration files, scheduler definitions or documentation.
  Search your scripts and job definitions for passwords. The result is the finding.
- [ ] 15. Credential rotation is scheduled, tested, and the process is documented so that rotation does not require the person who set it up.
  A credential nobody dares rotate is a credential that will be rotated in an emergency, badly.
- [ ] 16. Permissions at both ends follow least privilege, and partner access is scoped to the specific data set and reviewed at least annually.
  Integration accounts accumulate rights. Review what the account can reach, not what it uses.

Section 5: Data quality, validation and reconciliation (maximum 8)

- [ ] 17. Incoming data is validated against the contract at the boundary, and records that fail validation are quarantined rather than dropped or partially loaded.
  Silent partial loads are worse than a clean failure, because the downstream report looks plausible.
- [ ] 18. Volume and shape checks run on every batch, so an unexpectedly empty or unexpectedly large file raises an alert before it is processed.
  A zero record file that processes successfully is one of the most damaging failure modes in reporting pipelines.
- [ ] 19. Reconciliation is performed between source and destination on a defined cadence, with a documented tolerance and an owner for discrepancies.
  Counts, control totals and checksums. Reconciliation that nobody reviews is logging.
- [ ] 20. Character encoding, date formats, time zones, numeric precision and null handling are specified and tested, particularly for cross border exchanges.
  These produce the errors that survive testing and appear at quarter end. Test with awkward data deliberately.

Section 6: Reliability, monitoring and recovery (maximum 8)

- [ ] 21. Every exchange alerts on failure to a named recipient, and a job that does not run at all also generates an alert.
  Detecting absence is harder than detecting error and matters more. A scheduler that stops firing is silent by nature.
- [ ] 22. Retry behavior is defined and safe, with idempotent processing so that a retry cannot duplicate records.
  Duplicate financial records created by a retry are expensive to unpick and are usually found by a customer.
- [ ] 23. A documented replay or reprocess procedure exists, has been used, and states how far back replay is possible.
  Recovery from a bad batch is a routine event. It should not require an emergency and improvisation.
- [ ] 24. Exchanges have a defined behavior when the far end is unavailable, including how long data is held and what happens when the buffer fills.
  Unbounded queues fail the storage before they fail the interface, and take unrelated systems with them.

Section 7: Governance, compliance and partners (maximum 8)

- [ ] 25. Personal, regulated or contractually restricted data moving through exchanges is identified, and the legal basis and permitted destinations are recorded.
  This includes test data. Production extracts used in test environments are one of the most common exposure routes.
- [ ] 26. Retention is defined for data at rest in transfer locations, including landing zones, archives and error folders, and it is enforced automatically.
  Landing directories quietly become long term stores of sensitive data with weak access controls.
- [ ] 27. Each external partner exchange has a current agreement covering security expectations, incident notification, contacts and change notice.
  Confirm the technical contacts are current. Partner staff turnover breaks more exchanges than technology does.
- [ ] 28. Exchange activity is logged with enough detail to answer what was sent, when, by whom and with what result, and logs are retained for a defined period.
  The retention period should exceed the time it realistically takes for a data problem to be noticed, which is usually longer than the default.

Score sheet

Section 1, inventory and ownership

out of 8

Section 2, interface contracts and change

out of 8

Section 3, transport and connectivity

out of 8

Section 4, identity, credentials and access

out of 8

Section 5, data quality, validation and reconciliation

out of 8

Section 6, reliability, monitoring and recovery

out of 8

Section 7, governance, compliance and partners

out of 8

Total

out of 56

Lowest section, which is my constraint

Number of exchanges I could not find an owner for

0 to 18	**Undocumented exchange estate** Data is moving in ways the organization cannot fully describe, which makes both incidents and audits unmanageable. Start with Section 1 alone. Build the register, assign owners, and record what each exchange carries and what happens when it stops. Expect the exercise to find interfaces nobody knew were running and at least one that stopped working some time ago without anyone noticing. Do not attempt to redesign anything until the register exists, because you cannot prioritize what you have not listed.
19 to 32	**Known but fragile** The main interfaces are understood and the newer ones are reasonably built, while the older ones run on assumptions. The typical failures in this band are certificate expiry, an upstream field change nobody announced, and a silent job that stopped firing. Three actions give the fastest return: inventory every certificate and key with expiry alerting, add absence alerting so that a job which does not run raises the same alarm as one that fails, and write down the data contract for your three highest impact exchanges. Then rescore.
33 to 45	**Managed** Exchanges are inventoried, owned, monitored and mostly contract driven. Risk at this level shifts to the boundaries: partner side changes, credential rotation nobody wants to attempt, and data sitting in landing and error folders for longer than any policy intends. Focus on validation and reconciliation depth, on making replay a routine tested procedure rather than an emergency one, and on retention enforcement in the places where data pauses rather than the places where it lives.

46 to 56	**Engineered** The exchange estate is governed and resilient, which is uncommon and hard won. Check your honesty before taking the result at face value: any 2 awarded without evidence you could produce today should be a 1. The remaining work is usually consolidation and dependency risk. Look for the number of distinct transfer mechanisms you maintain, the single appliances or services that many exchanges route through, and the partner relationships where your resilience depends entirely on their operating practices rather than your own.

The failure modes worth designing against by name

THESE ACCOUNT FOR THE MAJORITY OF DATA EXCHANGE INCIDENTS AND EVERY ONE OF THEM IS PREVENTABLE AT DESIGN TIME.

Failure	How it shows up	The control that prevents it
Certificate or key expiry	A working interface stops at a precise moment with an obscure error	Inventory with expiry alerting and a named owner per certificate
Silent non-execution	Nobody notices for days because there is no error, only absence	Alert on missing runs, not only on failed runs
Upstream schema change	Fields shift, parsing produces plausible but wrong values	Versioned contracts, notice periods, and validation at the boundary
Empty or truncated file processed as valid	Downstream figures drop and are reported as a business trend	Volume and shape checks before processing
Retry duplication	Duplicate records found later, usually by a customer or in a reconciliation	Idempotent processing keyed on a stable identifier
Credential rotation nobody attempts	One expired secret takes down several interfaces at once	Secrets in a managed store, scheduled and rehearsed rotation
Backlog with no bound	Storage fills and unrelated systems on the same host fail	Defined hold behavior, bounded buffers, and alerting on depth
Partner contact drift	A change is announced to somebody who left, and the first you know is the failure	Annual confirmation of technical contacts on both sides

YOUR NEXT STEP

Find out which interface fails first, before it does

The checkpoints most organizations cannot self-score are the ones that need evidence from the systems themselves: which exchanges genuinely exist, which credentials are embedded in scripts, which certificates expire in the next six months, and which jobs stopped running quietly. The Infrastructure and Cloud Readiness Assessment gathers those facts directly, builds the exchange register for you, and returns a prioritized remediation plan with the constraint section addressed first.

Get the $997 Infrastructure & Cloud Readiness Assessment at gridwisetechnology.com

---

Important. This checklist is a self-assessment aid for internal planning only. It is not a security audit, a data protection assessment, a penetration test or a compliance certification, and a high score does not indicate compliance with any specific regulation or framework. Data protection, retention and cross border transfer obligations vary by jurisdiction and sector. Confirm your own obligations with qualified legal and compliance advisors, and test any change to a production interface before applying it.