

Run the planning properly before anything gets built

Who to get in the room, how to capture what they know, and what has to come out the other side.

CommerceWeave · Complimentary

The method, the 24-section checklist, and the day-one access list

HOW TO USE THIS

Run it as a live document, not a questionnaire

When somebody quotes a number for an integrated ecommerce project before doing any discovery, that number is not really a number. It is a range, and it is usually wide enough that you would make a different decision at one end of it than at the other.

Discovery is not for producing a document. It is for narrowing that range — improving the reliability and the probability of success, so the work can be delivered against a budget and a timeline that resemble the original estimate, or so the people who need to know are updated early enough to decide about it.

HOW TO WORK THE CHECKLIST

Assign each section an owner on day one. An item is closed when you hold the artifact or a written answer from the named person — **not** when somebody described it in a meeting. Anything still open at the discovery readout goes into the risk register with a date and an owner, and gets quoted as a range rather than a number.

It is deliberately exhaustive. You will not complete every line on every engagement; you *will* be asked, at some point, why nobody asked about kit components or exemption certificates — and the answer must never be "it wasn't on the list."

PART 1

The method

Five steps, in order. The checklist in Part 2 is what you run inside step two and step four.

Who is in the room

Five roles are worth having on an integrated ecommerce project: somebody who owns the storefront and knows how customers actually buy today, somebody who owns the ERP, and the sales, marketing and back-office people whose work the integration passes through. This decides most of the quality of what comes out.

How to capture what they know

Record the screen rather than taking notes, wherever there is no PII or other sensitive data, and have each person walk their own part: how a customer finds a product, gets a price and places an order; how that order lands in the ERP, gets picked and shipped, how the invoice is raised and how payment gets applied. Recording beats notes for a specific reason — when people narrate a process they walk the happy path and skip the exceptions, and the exceptions are where the work is.

How to catalogue it

Every object, in both directions. Every field that carries real meaning. Every rule, and every exception somebody demonstrated without naming. That catalogue is what the refined estimate gets built from, because every object in it is a separate interface with its own failure modes.

How to validate it — the step that gets skipped

Take the catalogue back to the same people and walk it line by line. Two things reliably surface: the real state of the data, and what the environment can actually do — which API surfaces exist on that specific ERP version, what the rate limits are, whether a test environment exists with representative data in it.

What has to come out the other side

Not a document for its own sake. A sequence: the interfaces in the order they get built, what each one depends on, and which data has to be cleaned before which interface can be trusted.

PART 2

Owner codes

CODE	OWNER ROLE
SP	Executive sponsor
PM	Engagement/project manager (delivery side)
ERP	ERP administrator / functional lead
IT	IT director / enterprise architect
SEC	Security reviewer / CISO delegate
FIN	Finance / controller
SALES	Sales leadership
CS	Customer service / inside sales
MKT	Marketing / digital lead
MDM	Master data owner
LOG	Warehouse / logistics
ARCH	Solution architect (delivery side)
3P	Third-party / incumbent vendor / agency

Status convention: - [] open · - [x] closed with artifact in hand · annotate (N/A – reason) when deliberately skipped, never delete a line.

PART 3

The discovery checklist

Twenty-four sections. Work them in whatever order the engagement allows, but assign every section an owner before the first session.

1. Engagement setup and stakeholder identification

[] Obtain the signed SOW, MSA and any change-order history from prior phases. **Owner:** PM. - [] Confirm in writing whether the scope document on the table is a **feasibility assessment or a scope baseline**. **Owner:** PM/SP. **Why it matters:** a feasibility document that nobody demoted gets silently promoted to a contract the moment marketing picks a date off it. - [] Name the executive sponsor and confirm they hold budget authority for data remediation, not just for build. **Owner:** SP. **Why it matters:** remediation is on the critical path and is almost never in the vendor's scope; if no one can fund it, it will not happen. - [] Name a single client-side decision-maker for scope arbitration and document their delegation limits. - [] Obtain a stakeholder map with names, titles, email, phone and time zone for every role in the legend above. - [] Identify roles with **no named holder** (commonly: master data owner, PIM owner). **Owner:** PM. **Why it matters:** an unowned domain becomes an unowned remediation task and then a slipped date. - [] Confirm the ERP administrator's weekly availability in hours, agreed by their manager, not by them. **Owner:** SP. - [] Establish the meeting cadence, the standing invitee list, and who chairs. - [] Agree the RAID log location and who may add to it. - [] Agree the document repository, naming convention and version-control approach for all discovery artifacts. - []

Establish the escalation path with names and response-time expectations at each tier. - [] Confirm communication constraints: shared Slack/Teams channel permitted? Email-only? Client VPN required for document access? - [] Identify all incumbent vendors and agencies, what they still host or control, and their contractual end dates. **Owner:** IT. **Why it matters:** cutover night depends on someone at the incumbent answering the phone about DNS. - [] Determine whether the incumbent is being replaced by this project and whether they know that. - [] Obtain the prior project's post-mortem or lessons-learned document, if one exists. - [] Confirm whether a competing or parallel internal initiative touches the same systems (ERP upgrade, WMS replacement, rebrand). - [] Agree the discovery exit criteria and readout date in writing. - [] Confirm who signs off the discovery deliverable and what "signed off" entitles the project to do next.

2. Business context, drivers and success criteria

[] Document the business case in the sponsor's own words, including the number that justified the spend. - [] Obtain revenue by channel for the last 24 months (direct sales, inside sales, distributor, existing web, EDI, marketplace). - [] Obtain order volume and average order value by channel and by month, to expose seasonality. **Owner:** FIN. **Why it matters:** peak-period volume drives integration throughput sizing and rules out certain batch designs. - [] Quantify the current cost of manual order entry: FTEs, hours per order, error rate, rework cost. - [] Identify the top three business outcomes the sponsor will be measured on, with target values and measurement dates. - [] Confirm the baseline for each target metric and where that baseline is measured today. - [] Establish the target for online order share at 6, 12 and 24 months post-launch. - [] Determine whether the goal is **new customer acquisition or existing-customer self-service migration**. **Why it matters:** these produce almost entirely different requirement sets — SEO, guest checkout and merchandising versus entitlement, contract pricing and account hierarchy. - [] Identify the competitive trigger, if any (a competitor launched, a customer demanded a portal, a large account threatened to leave). - [] Document any customer commitments already made, in writing, that constrain scope or date. **Owner:** SALES. - [] Confirm whether channel conflict has been addressed as a business decision. **Owner:** SP. **Why it matters:** if reps believe the site cannibalises their commission, adoption fails regardless of build quality; this is a compensation decision, not a requirement. - [] Determine the commission treatment for online orders, per rep and per account. - [] Identify which customer segments are in scope for phase 1 and which are explicitly excluded. - [] Identify geographies, legal entities and brands in scope, and those explicitly excluded. - [] Confirm the languages and currencies required at launch versus later. - [] Obtain the written "not in phase 1" list and confirm the sponsor has read it. - [] Establish how success will be reported to the board and on what cadence.

3. Current-state systems inventory

[] Obtain a complete list of business systems with vendor, product, version, hosting model and business owner. - [] For each system, record whether it is in scope, adjacent (must be integrated), or out of scope. - [] Obtain the current-state architecture diagram, and confirm its last-updated date. **Why it matters:** an architecture diagram older than 18 months is a hypothesis, not a document. - [] Identify the existing eCommerce platform, its version, its hosting, and who built it. - [] Determine how orders reach the ERP today, step by step, including every human in the loop. - [] Identify every spreadsheet, Access database or shared drive that acts as a system of record. **Owner:** CS. **Why it matters:** these are unmanaged integrations and are invisible on every diagram. - [] Document all existing integrations between systems, including nightly file drops and scheduled tasks nobody maintains. - [] Identify existing middleware/iPaaS, its licence, its administrator and whether the client can modify it. - [] Identify the CRM, its data ownership boundary with the ERP, and which system wins on customer record conflicts. - [] Determine whether a PIM exists, is licensed but unused, or is aspirational. - [] Identify the WMS/OMS, its relationship to the ERP, and where inventory truth actually lives. - [] Identify the CPQ or configurator, whether it is native to the ERP or third-party, and whether its output is quotable online. - [] Identify the tax engine, address-validation service and shipping-rating service in use. - [] Identify the payment gateway, processor and merchant acquirer, plus any stored-card vault. - [] Identify the marketing automation, email

service provider and analytics stack. - [] Identify customer-service tooling and any existing customer portal. - [] Obtain the licence renewal calendar for every adjacent system. **Owner:** IT. **Why it matters:** a renewal date inside the project window is either leverage or a landmine. - [] Record for each adjacent system whether a non-production environment exists. - [] Identify any system scheduled for replacement during the project window.

4. ERP deep-dive

[] Confirm exact ERP product, edition and version/build number (for example Infor CSI 10.x vs CloudSuite Industrial multi-tenant; Epicor Prophet 21 vs Kinetic; Dynamics 365 F&O vs Business Central). **Owner:** ERP. **Why it matters:** integration surface differs materially between editions of the same product name, and multi-tenant editions forbid the customisations single-tenant ones allow. - [] Confirm deployment model: on-premises, hosted single-tenant, vendor cloud multi-tenant. - [] Confirm the upgrade/patch calendar for the next 18 months, including forced vendor upgrades. **Why it matters:** a mandatory multi-tenant upgrade mid-build can invalidate an integration approach with no notice. - [] Obtain the list of ERP modules licensed and which are actually in use. - [] Obtain an inventory of customisations: custom tables, custom fields, modified forms, event handlers, user-defined fields, custom stored procedures. - [] Determine which customisations affect order entry, pricing, item master or customer master specifically. **Owner:** ERP. - [] Identify any third-party ERP add-ons or ISV bolt-ons and whether they intercept order processing. - [] Document the available integration surface: REST/SOAP APIs, ION/IDO (Infor), OData, RESTlets/SuiteTalk (NetSuite), Data Management Framework (D365), OData/BAPI/IDoc (SAP), contract-based APIs (Acumatica), Service Connect/BOD (Epicor/Sage). - [] Confirm whether direct database read access is available, permitted, and supported. **Why it matters:** direct SQL is fast to build and is the single most common cause of an upgrade breaking a live integration. - [] Obtain written API documentation with version and date. - [] Confirm rate limits, concurrency limits, governance/usage quotas and payload size limits. - [] Make a **live authenticated call against the client's own sandbox** for at least one read and one write operation. **Owner:** ARCH. **Why it matters:** this single test resolves more estimate variance than any document review; a described API and an observed API are different systems. - [] Confirm a sandbox exists, is refreshed from production, and record its refresh cadence and last refresh date. - [] Confirm who can request a sandbox refresh and the lead time. - [] Confirm whether sandbox data is masked and whether that masking breaks pricing or customer tests. - [] Obtain administrator or integration-level credentials for the sandbox, in the delivery team's name. - [] Confirm the ERP licensing model for integration users: named user, concurrent, API-call-metered, or per-transaction. **Owner:** ERP/FIN. **Why it matters:** some ERPs bill integration traffic as licensed users; this can add five figures annually and is routinely discovered after go-live. - [] Obtain a written quote or licence position for the integration user(s) required. - [] Confirm whether the ERP vendor's professional services must perform any part of the integration contractually. - [] Identify the ERP support partner and their responsiveness, including SLA. - [] Determine the ERP's transactional behaviour on partial failure: does a failed order line roll back the order? - [] Identify order-entry validation rules enforced by the ERP that the storefront must replicate or respect. - [] Confirm which fields are mandatory on order creation and which have defaults. - [] Determine ERP behaviour for orders created via API versus UI — are the same workflows, defaults and events triggered? **Why it matters:** API-created orders frequently bypass a workflow the business assumes always runs. - [] Identify any batch/nightly ERP processes and their windows, during which integration must not write. - [] Confirm the ERP's timezone configuration and date-handling conventions. - [] Obtain a production data volume profile: item count, customer count, open order count, annual order line count. - [] Confirm database size and growth rate if extract-based integration is contemplated. - [] Identify the disaster-recovery posture and RTO/RPO for the ERP.

5. Product and catalog data

[] Obtain the total active SKU count, and separately the count intended for the storefront in phase 1. - [] Obtain a full item master export in CSV with every column, not a curated sample. **Owner:** MDM/ERP. - [] Determine the count of items with a customer-facing description distinct from the internal description. **Why it matters:** internal descriptions

such as BRKT, STL, 3/8, ZP are unsellable and unsearchable; the gap between item count and described-item count is a content workstream. - [] Measure attribute coverage: for the top 10 attributes, the percentage of in-scope SKUs populated. - [] Determine where attributes live today: ERP UDFs, spreadsheets, PDF spec sheets, supplier portals, or nowhere. - [] Obtain the current product taxonomy/category structure and its depth. - [] Confirm whether the taxonomy is customer-facing language or internal/accounting language. **Why it matters:** ERP product lines are usually a financial reporting structure and make a poor navigation tree. - [] Determine whether any industry-standard classification is required (UNSPSC, ETIM, eCI@ss, NAICS) by any customer. - [] Obtain the image inventory: count of SKUs with at least one image, image resolution, file naming convention, storage location. - [] Confirm image rights and whether supplier-provided imagery may be used online. **Owner:** MKT. **Why it matters:** manufacturer imagery is frequently licensed for print or channel use only. - [] Inventory non-image assets: spec sheets, CAD/BIM files, MSDS/SDS, installation guides, videos, certificates. - [] Determine whether SDS documents must be current-version-controlled for compliance. **Owner:** SEC/MKT. - [] Obtain a sample of 20 items end-to-end with every attribute, asset and description as it exists today. - [] Determine the base unit of measure and all alternate UoMs per item. - [] Confirm UoM conversion factors and whether they are per-item or global. - [] Determine selling UoM versus stocking UoM versus pricing UoM per item. **Why it matters:** priced-per-foot, sold-per-reel is a classic source of order-value errors that survive UAT. - [] Identify items sold in fixed pack quantities, minimum order quantities and order multiples. - [] Obtain the cross-reference data: customer part numbers, manufacturer part numbers, competitor cross-references, UPC/GTIN, supplier part numbers. - [] Determine where customer part numbers are stored and whether they are complete. **Why it matters:** B2B buyers search by their own part number; missing cross-references produce zero-result searches that read as "your site doesn't have it." - [] Obtain supersession and replacement chains, including multi-hop supersessions. - [] Identify discontinued, obsolete and phase-out items and the policy for displaying them. - [] Confirm lifecycle status codes and their meanings, verified against actual usage rather than documentation. - [] Identify kit, bundle and assembly items and how they are structured in the ERP. - [] Determine whether kits are exploded at order entry, at pick, or never. - [] Determine whether kit component availability must be checked online. - [] Identify BOM structures relevant to service parts and whether BOMs are usable for parts lookup. **Owner:** ERP. **Why it matters:** engineering BOMs are usually structured for manufacture, not for a customer finding the right filter for their machine. - [] Identify configurable/made-to-order products and the configurator that governs them. - [] Determine whether configured products require a quote or can be priced and ordered directly. - [] Identify serialised and lot-controlled items and whether serial/lot must be visible to customers. - [] Identify hazardous materials, export-controlled items and items with shipping restrictions. - [] Identify items restricted by customer, region, licence or certification. - [] Determine the item-creation process and its typical lead time. **Why it matters:** if a new SKU takes three weeks to reach the site, merchandising cannot function. - [] Confirm who owns item data quality after launch and what their day job is.

6. Customer and account data

[] Obtain a full customer master export including inactive accounts. - [] Document the account hierarchy model: parent/child, corporate/branch, bill-to/ship-to relationships. - [] Determine the maximum hierarchy depth actually in use. - [] Determine how many ship-tos the largest accounts have. **Why it matters:** a 400-ship-to account breaks any UI that assumes a dropdown. - [] Confirm whether ship-tos are ERP records or free-text addresses at order entry. - [] Confirm whether a customer may ship to an address not on file, and who approves it. - [] Obtain the contact data structure: are contacts on the customer record, in the CRM, both, or neither? - [] Determine the count of accounts with at least one valid email address. **Why it matters:** portal invitation campaigns fail silently against stale contact data, and the failure looks like low adoption. - [] Quantify duplicate customer records and the rule that would identify them. - [] Determine the customer-numbering scheme and whether it is meaningful (region, type, salesperson embedded). - [] Obtain credit limit, credit hold status, and aging exposure fields. - [] Confirm payment terms per customer and the full list of terms codes in use. - [] Identify customers on prepay, COD, or credit-card-only status. - [] Identify tax exemption status and where certificates live. - [] Determine the assigned salesperson/territory field and whether it drives pricing or routing. - [] Determine the customer type/class codes and what business rules depend on them. - [] Confirm which

customers are dealers/distributors versus end users, and whether they see different content. - [] Define the portal user provisioning model: self-registration with approval, invitation-only, or bulk-provisioned. - [] Define who approves a new portal user and their SLA. - [] Define the mapping between a portal user and an ERP customer record, including users who span multiple accounts. **Why it matters:** a buyer at a multi-branch customer expects one login and an account switcher; retrofitting this is expensive. - [] Define the role model within a customer account: buyer, approver, admin, view-only, AP clerk. - [] Determine whether the customer administers their own users or the client does. - [] Determine the deprovisioning process when a buyer leaves the customer's employment. **Owner:** SEC. - [] Confirm whether customers require SSO/SAML into the portal. - [] Determine what a guest or unauthenticated visitor may see: pricing, availability, ordering. - [] Confirm whether new-customer online registration and onboarding is in scope, including credit application.

7. Pricing and entitlement

[] Obtain a written description of the complete price determination sequence, in evaluation order. **Owner:** ERP/SALES. **Why it matters:** pricing is the single most common source of post-launch escalations, and almost every client's verbal description of their own pricing is incomplete. - [] Obtain the ERP price list structure: list prices, price books, price codes, effective dates. - [] Identify contract pricing: how contracts are stored, how they are keyed, and how many are active. - [] Identify price matrix rules (customer × product, customer class × product line, and so on) and their count. - [] Obtain quantity break/tier structures and confirm whether breaks are per line or per order. - [] Determine whether pricing is discount-off-list, cost-plus, or absolute price, per rule type. - [] Confirm rounding and decimal precision rules for price and extended amount. - [] Identify promotions, campaign pricing and coupon requirements. - [] Determine whether promotions must be honoured by the ERP or may be storefront-only. **Why it matters:** a storefront-only discount that the ERP recalculates produces an invoice that does not match the order confirmation. - [] Identify rebate programmes, backend rebates and their effect on displayed price. - [] Identify special pricing agreements (SPAs) with suppliers and whether they affect customer price. - [] Confirm currency requirements, exchange-rate source and rate refresh policy. - [] Determine whether multi-currency pricing is maintained per currency or converted. - [] Determine the authoritative pricing engine for the storefront: real-time ERP call, cached, or replicated rules. **Owner:** ARCH. - [] Measure the ERP price-call response time under realistic load, on the sandbox. **Why it matters:** real-time price calls are correct and are frequently too slow for a category page of 48 items; the mitigation is architectural, not a tuning exercise. - [] Define cache invalidation triggers if pricing is cached. - [] Define customer-specific catalog/entitlement rules: which customers may see or buy which items. - [] Determine whether entitlement is inclusive (allow list) or exclusive (deny list) and where it is stored. - [] Determine whether items outside entitlement are hidden entirely or shown as unavailable. - [] Confirm whether price visibility differs from purchase entitlement (can see, cannot buy). - [] Determine whether prices are shown before login, after login only, or as "call for pricing" by segment. - [] Identify the "call for quote" trigger conditions. - [] Confirm minimum margin rules and whether the site may ever display a price below them. - [] Identify who may override price, by how much, and whether that is in scope online. - [] Obtain three real customers' price outputs for the same 20 items, to test rule coverage. **Owner:** ERP. **Why it matters:** this is the single most valuable pricing artifact in discovery and reliably exposes rules nobody mentioned.

8. Inventory and availability

[] Obtain the list of stocking locations, warehouses, sites and legal entities. - [] Confirm whether the ERP is multi-site and whether inventory is visible across sites. - [] Determine which locations' inventory is sellable online. - [] Define the availability calculation to be shown: on-hand, on-hand less allocated, ATP with supply, or a banded indicator. - [] Confirm whether ATP logic exists in the ERP and whether it is exposed via API. **Owner:** ERP. **Why it matters:** the difference between on-hand and true ATP is often the difference between a promise you can keep and one you cannot. - [] Determine allocation and reservation behaviour: does an online order reserve stock, and at what moment? - [] Determine whether the storefront may hold soft reservations during checkout. - [] Define the oversell policy and who

absorbs the cost. - [] Determine the backorder policy: allow, block, partial ship, or customer choice. - [] Confirm whether partial shipment is permitted per customer or per order. - [] Obtain lead-time data per item and its source (fixed field, supplier data, planning calculation). - [] Determine whether lead times are reliable enough to display. **Why it matters:** displaying an unmaintained lead-time field converts a data-quality problem into a customer-service problem. - [] Identify drop-ship items and suppliers, and how drop-ship availability is known. - [] Determine whether supplier inventory feeds exist and their format and cadence. - [] Define the inventory refresh cadence acceptable to the business, in minutes. - [] Determine the peak rate of inventory change to size the sync approach. - [] Confirm whether inventory should be published as exact quantity, threshold bands, or in-stock/out-of-stock. **Owner:** SALES. **Why it matters:** publishing exact quantities to a market that includes competitors is a commercial decision, not a technical one. - [] Define the display rule for items with zero stock but active supply. - [] Define behaviour for non-stock, special-order and made-to-order items. - [] Confirm cycle-count accuracy percentage from the last 12 months. **Owner:** LOG. - [] Determine whether physical inventory or count freezes will occur during the project window. - [] Identify consignment, customer-owned and vendor-managed inventory and its visibility rules.

9. Order-to-cash

[] Obtain the complete list of ERP order types and what each means operationally. - [] Determine which order types the storefront will create and which it must display. - [] Obtain a written order lifecycle with every status value and its transition rules. **Owner:** ERP. - [] Map ERP statuses to the customer-facing statuses to be displayed. **Why it matters:** exposing raw ERP status codes to customers generates support calls; the mapping is a business decision that needs an owner. - [] Determine the order-numbering scheme and whether web orders need a distinct series. - [] Confirm whether the web order number or the ERP order number is the customer reference. - [] Determine order-hold conditions: credit, price, availability, first order, manual review. - [] Confirm credit-hold behaviour: is the order created and held, or rejected at entry? - [] Define the customer-facing message for a credit-held order. **Owner:** FIN/CS. **Why it matters:** "your order failed" and "your order is pending review" have very different commercial consequences. - [] Determine who releases holds and their SLA. - [] Identify internal approval workflows triggered by order value, margin, or item. - [] Identify customer-side approval workflows required in the portal (requisition to approver). - [] Determine order minimums: value, quantity, per-line and per-order, and the enforcement point. - [] Identify small-order surcharges, handling fees, fuel surcharges and how they are calculated. - [] Determine freight charging model: prepaid, prepaid and add, collect, third-party billing, free over threshold. - [] Confirm whether freight is estimated at checkout and reconciled at invoice, or charged as quoted. **Why it matters:** the reconciliation model determines whether a payment authorisation must be adjusted before capture. - [] Determine when tax is calculated: at checkout, at invoice, or both, and which is authoritative. - [] Determine order change and cancellation rules: what may a customer change, until when? - [] Determine whether order changes must flow back into the ERP or be handled by CS. - [] Identify blanket orders, releases and scheduled agreements and whether they are in scope. - [] Identify standing orders and subscription/replenishment requirements. - [] Determine quote-to-order flow: are quotes created online, in the ERP, or in CPQ, and how does a quote convert? - [] Confirm quote expiry and price-hold rules. - [] Obtain the invoicing model: per shipment, consolidated, monthly statement. - [] Determine invoice delivery: email, portal, EDI, print, and per-customer preferences. - [] Define the payment application process and how online payments post to AR. **Owner:** FIN. - [] Determine whether customers pay invoices online, pay at order, or both. - [] Determine handling of short-pays, credits, and deductions if invoice payment is in scope. - [] Obtain the current order-to-cash cycle time as a baseline.

10. Fulfilment and shipping

[] Obtain the list of carriers and service levels offered, by warehouse. - [] Confirm carrier account numbers and whether the client's negotiated rates must be used. - [] Determine the rating method: carrier API real-time, rate table, flat rate, percentage of order, or manual quote. - [] Confirm whether a rating engine already exists in the ERP or a shipping system. **Owner:** LOG. - [] Obtain dimensional and weight data coverage per SKU. **Why it matters:** carrier rating

without dims and weights is impossible; this is a frequent, late, and unwelcome discovery. - [] Measure the percentage of in-scope SKUs with valid weight and dimensions. - [] Determine packaging and cartonisation logic: does an algorithm exist, or does the packer decide? - [] Determine whether multi-carton and multi-package shipments must be rated accurately at checkout. - [] Identify oversized, overweight and freight-only items and their flagging in the item master. - [] Obtain freight class (NMFC) data coverage for LTL items. - [] Determine LTL rating approach: LTL carrier APIs, a broker/TMS, or "call for quote". **Why it matters:** designing LTL as parcel is a launch-blocking error in equipment and industrial businesses. - [] Identify accessorial requirements: liftgate, residential, inside delivery, appointment, limited access. - [] Determine how accessories are selected and charged online. - [] Confirm will-call/counter pickup requirements, locations and hours. - [] Determine whether pickup requires readiness notification and how it is triggered. - [] Confirm order cut-off times per warehouse and their effect on displayed delivery estimates. - [] Determine ship-complete versus ship-as-available default and whether the customer may choose. - [] Identify hazmat shipping restrictions and required documentation. - [] Identify international shipping, incoterms, customs documentation and denied-party screening requirements. **Owner:** LOG/FIN. - [] Determine the source of tracking numbers and the system that first knows them. - [] Define required notifications: order confirmation, ship confirmation, delivery, exception, backorder — with sender and template owner. - [] Confirm whether tracking is displayed in the portal, emailed, or both. - [] Determine whether delivery-date promises are made and against what data. - [] Confirm the returns shipping process for label generation, if in scope.

11. Returns, RMA, service and warranty

[] Obtain the current RMA process end to end, including who authorises. - [] Determine whether customers may initiate an RMA online in phase 1. - [] Confirm the RMA object in the ERP and whether it can be created via API. - [] Define return eligibility rules: time window, item type, condition, original order required. - [] Determine restocking fee rules and who may waive them. - [] Determine return authorisation SLA and who owns it. **Owner:** CS. - [] Define the credit process: credit memo, replacement, refund to original payment method. - [] Confirm whether refunds to card are supported by the gateway and the ERP's AR process. **Why it matters:** card refunds that the ERP cannot represent become manual finance work at volume. - [] Identify warranty claims requirements and whether they are distinct from returns. - [] Determine whether serial-number lookup is required for warranty validation. - [] Identify installed-base/equipment registry data and whether it exists in usable form. - [] Determine whether service parts must be found by machine model or serial. **Why it matters:** parts-by-machine is the highest-value B2B feature in equipment businesses and depends entirely on BOM data usability. - [] Identify field-service integration requirements (scheduling, work orders, service contracts). - [] Determine whether service contracts affect pricing or entitlement. - [] Identify core charges, exchange units and remanufactured item handling. - [] Determine repair-order visibility requirements in the portal.

12. Payments and finance

[] Confirm the current payment gateway, processor and merchant account provider. - [] Determine which payment methods are required at launch: card, ACH, terms/on-account, PO, wire, digital wallets. - [] Confirm whether "pay on account" against credit terms is the primary method. **Why it matters:** in B2B it usually is, which changes the checkout design and reduces PCI exposure substantially. - [] Determine the authorisation and capture model: auth at order, capture at ship, or auth-and-capture. - [] Confirm authorisation expiry handling for long lead-time orders. **Owner:** FIN. **Why it matters:** an authorisation that expires before shipment on a 6-week lead-time item requires re-auth logic nobody scoped. - [] Determine partial capture and multi-capture support for partial shipments. - [] Confirm surcharge/convenience fee policy for cards and its legality by jurisdiction. - [] Determine stored payment method requirements and the tokenisation vault. - [] Confirm PCI DSS scope target and the SAQ type the client intends to complete. **Owner:** SEC. - [] Determine ACH/bank transfer requirements including micro-deposit verification. - [] Determine how payments post to the ERP: real-time, batch, or manual reconciliation. - [] Identify the cash application

process and whether the integration must supply remittance detail. - [] Determine chargeback and dispute handling ownership. - [] Confirm refund authority and limits. - [] Identify credit application and credit-check requirements for new online customers. - [] Determine whether a third-party B2B credit provider is in use or contemplated. - [] Confirm revenue recognition constraints that affect order or invoice timing. - [] Identify multi-entity/intercompany posting requirements. - [] Confirm the fiscal calendar and any period-close freeze windows. **Why it matters:** close windows constrain cutover dates and test-data creation.

13. Tax and exemption certificates

[] Determine the tax calculation authority: ERP native, third-party engine (Avalara, Vertex, Sovos), or manual. - [] Confirm whether the storefront calls the tax engine directly or via the ERP. **Owner:** ARCH. **Why it matters:** two independent paths to the tax engine will eventually disagree, and the disagreement appears on an invoice. - [] Obtain the list of jurisdictions with nexus and confirm the nexus determination is current. - [] Confirm economic nexus monitoring is in place if selling into new states. - [] Determine product taxability mapping: are tax codes assigned per item, and at what coverage? - [] Measure the percentage of in-scope SKUs with a tax code assigned. - [] Determine the handling of freight taxability, which varies by jurisdiction. - [] Confirm exemption certificate storage: where certificates live today and their expiry tracking. - [] Determine the count of exempt customers and the count with expired certificates. **Owner:** FIN. **Why it matters:** an expired certificate on a live storefront is an audit liability, and discovery is where it becomes visible. - [] Define online certificate capture and validation requirements. - [] Determine behaviour when an exempt customer's certificate is missing or expired at checkout. - [] Confirm VAT/GST requirements, registration numbers and validation for international sales. - [] Confirm reverse-charge and intra-community supply handling if applicable. - [] Determine tax reporting and filing responsibility and whether the integration must supply data. - [] Confirm whether tax is recalculated at invoice and which figure is authoritative. - [] Determine handling of tax on returns and credits.

14. Content, brand and marketing

[] Obtain brand guidelines, logo files and typography licences. - [] Confirm whether a rebrand or visual refresh is in progress. **Why it matters:** a design refresh landing mid-build is a scope event, not a preference. - [] Determine who approves design, and whether that person is the sponsor. - [] Obtain the current site's analytics: sessions, top pages, top searches, conversion rate. - [] Obtain the top 100 landing pages by traffic for redirect planning. - [] Obtain the current URL structure and confirm a redirect map is required. **Owner:** MKT. **Why it matters:** an unplanned URL change on a site with organic traffic is a measurable revenue event. - [] Confirm domain ownership and registrar access. **Owner:** IT. - [] Confirm DNS control and who can make changes on cutover night. - [] Confirm SSL/TLS certificate ownership and renewal dates. - [] Obtain the content inventory: static pages, blog, resource library, case studies. - [] Determine the content migration approach: migrate, rewrite, or drop, page by page. - [] Identify content that must be authored new and who will write it. - [] Determine content authoring capacity in hours per week, by named person. **Why it matters:** content is the most commonly under-resourced workstream and the most common cause of a launch slipping without anyone calling it a slip. - [] Confirm SEO requirements: metadata, structured data, canonicalisation, sitemap, hreflang. - [] Determine accessibility target (WCAG 2.1 AA or 2.2 AA) and whether it is contractual. **Owner:** SEC/MKT. - [] Identify email templates required and the sending domain, SPF/DKIM/DMARC status. - [] Confirm marketing automation integration requirements and the contact sync direction. - [] Identify required tracking pixels, tag manager, consent management and cookie policy. - [] Confirm privacy policy and terms of sale ownership and legal review lead time. - [] Determine multi-brand or multi-site requirements and content sharing between them. - [] Identify translation and localisation requirements and the translation vendor.

15. Search, merchandising and personalisation

[] Obtain the current site's internal search logs, including zero-result queries. **Owner:** MKT. **Why it matters:** the zero-result list is the highest-signal requirements document available, and it is free. - [] Determine the search behaviours

required: part number, description, cross-reference, attribute, natural language. - [] Confirm exact-match part number search must outrank fuzzy relevance. **Why it matters:** B2B buyers typing a part number expect that part, not a relevance-ranked set; getting this wrong destroys credibility immediately. - [] Determine handling of part numbers with dashes, spaces and leading zeros. - [] Determine synonym and abbreviation requirements from the industry vocabulary. - [] Define faceted navigation requirements per category and the attributes that drive them. - [] Confirm attribute data quality is sufficient to support the proposed facets. **Owner:** MDM. - [] Determine the search platform: native, Elastic/OpenSearch, Algolia, Coveo, Bloomreach, or vector-based. - [] Define indexing frequency requirements and the index rebuild window. - [] Determine merchandising requirements: boosts, pins, banners, curated collections. - [] Identify who will merchandise the site after launch and their available hours. - [] Define personalisation requirements: last-ordered, frequently-ordered, recommended, customer-specific landing. - [] Determine whether purchase history is available for recommendations and how far back. - [] Confirm whether AI-driven search or conversational assistance is in scope or a later phase. - [] Determine product comparison and specification-table requirements. - [] Define quick-order, pad-order, CSV upload and barcode-scan entry requirements. **Why it matters:** in replenishment-driven distribution these features drive more revenue than the entire browse experience.

16. Portals and self-service

[] Define the self-service scope: order history, invoices, statements, tracking, quotes, returns, documents. - [] Determine how far back order history must be visible and whether historical data must be migrated. **Owner:** ARCH. **Why it matters:** "all history" versus "24 months" is often the difference between a live ERP query and a data migration project. - [] Confirm whether historical orders placed offline must appear in the portal. - [] Determine reorder-from-history requirements including discontinued item handling. - [] Define saved lists, favourites, shopping lists and shared-list requirements. - [] Define requisition and approval workflow requirements within customer accounts. - [] Determine budget and spend-limit controls per user. - [] Define document access: invoices, packing slips, certificates of conformance, test reports, SDS. - [] Determine whether certificates of analysis or conformance must be retrievable by lot or serial. **Owner:** LOG. **Why it matters:** in regulated manufacturing this is a top-three portal feature and depends on lot traceability data. - [] Define statement and open-AR visibility requirements. - [] Determine whether customers may pay invoices in the portal and by what method. - [] Define user-management self-administration scope for customer admins. - [] Determine notification preferences management. - [] Define account-switching UX for users spanning multiple accounts. - [] Determine whether internal sales reps need an order-on-behalf-of capability. **Owner:** SALES. **Why it matters:** rep-assisted ordering converts the site from a threat into a tool and materially improves adoption. - [] Define impersonation controls and audit requirements for order-on-behalf. - [] Determine mobile requirements: responsive web, native app, or warehouse-floor use cases.

17. B2B procurement, PunchOut and EDI

[] Identify customers currently transacting by EDI and their document sets. - [] Obtain the EDI transaction set list in use (850, 855, 856, 810, 832, 846, 860, 997) and the VAN. - [] Determine whether EDI moves to the new platform or remains where it is. **Owner:** IT. - [] Identify the EDI translator/provider and their change-request lead time and cost. **Why it matters:** EDI partner changes are billed per map and per trading partner and are routinely omitted from budgets. - [] Identify customers requesting PunchOut (cXML or OCI) and their procurement platforms (Ariba, Coupa, Jaggaer, SAP SRM, Oracle). - [] Confirm the PunchOut protocol version and whether Level 2 PunchOut is required. - [] Determine whether purchase orders return via cXML, EDI 850, or email. - [] Determine invoice delivery requirements per procurement network (cXML invoice, e-invoicing mandates). - [] Identify country e-invoicing mandates that apply (Peppol, ZUGFeRD, and similar). **Owner:** FIN. - [] Determine punchout catalog content rules: customer-specific items, contract pricing, UNSPSC coding. - [] Confirm whether static catalogs (CIF/BMEcat) are also required. - [] Determine testing and certification requirements with each procurement platform and their lead time. **Why it matters:** buyer-side procurement teams schedule certification testing on their calendar, not yours; six weeks is common. - [] Identify

marketplace channel requirements (Amazon Business, Grainger, distributor portals) if in scope. - [] Determine supplier-side integration needs: PO transmission, ASN receipt, drop-ship confirmation.

18. Integration architecture and middleware

[] Complete the integration inventory register for every interface (see `integration-inventory-template.md`). - [] For each entity, name the single system of record and record the decision in writing. **Owner:** ARCH. **Why it matters:** an undecided system of record is not an architectural gap, it is a future production incident with a support ticket attached. - [] Confirm the middleware strategy: iPaaS, ESB, platform-native connectors, custom services, or point-to-point. - [] Confirm who hosts, owns and supports the middleware after go-live. - [] Determine whether the client has iPaaS licences already and their entitlement limits. - [] Establish the network path from the storefront to the ERP: VPN, allow-list, private link, gateway. - [] Confirm firewall change lead time and the change-approval calendar. **Owner:** IT. **Why it matters:** this single lead time is the most common cause of a slow start; it is measured in weeks at many industrial clients. - [] Determine authentication approach per interface: OAuth2, API key, mTLS, basic, service account. - [] Determine secret storage and rotation policy and who holds the rotation calendar. - [] Define the error handling model: retry, dead-letter, alert, manual replay. - [] Define idempotency strategy per write interface. **Why it matters:** without idempotency keys, a retried order creation produces duplicate orders in the ERP, and this is discovered in production. - [] Define ordering and sequencing guarantees required per entity. - [] Define the reconciliation approach: how do we prove nothing was lost? - [] Define monitoring, logging and observability requirements and the tooling. - [] Define alerting recipients and after-hours escalation for integration failures. - [] Determine data retention for integration logs and payloads, and PII implications. - [] Determine the throughput and latency targets per interface, tied to the peak volumes from section 2. - [] Confirm environment topology: dev, test, UAT, staging, production, for every system in the chain. **Why it matters:** a full chain requires a non-production instance of *every* system; the missing one is usually the tax engine or the WMS. - [] Confirm test data availability in every non-production environment. - [] Determine whether test orders in the ERP sandbox can be created freely and how they are cleaned up. - [] Confirm whether third-party services have sandbox accounts (payment gateway, carriers, tax). - [] Determine the deployment and release process, including client change-control requirements. - [] Determine rollback approach per interface. - [] Identify any real-time requirement that could be satisfied by near-real-time, and confirm with the business. **Why it matters:** "real time" is usually a synonym for "not overnight" and the distinction is worth significant budget.

19. Security, compliance and infrastructure

[] Obtain the client's security requirements as a written checklist they will assess against at go-live. **Owner:** SEC. **Why it matters:** a security review that first appears in month five with architectural findings is the classic late-stage schedule detonation. - [] Confirm whether a vendor risk assessment or security questionnaire is required and its typical duration. - [] Determine data classification of every entity moving between systems. - [] Confirm PII, PCI and any regulated data in scope. - [] Determine encryption requirements in transit and at rest. - [] Confirm authentication requirements for customers: password policy, MFA, SSO, social login prohibition. - [] Confirm internal user authentication and whether the client's IdP must be used. - [] Determine session, timeout and concurrent-session policies. - [] Confirm audit logging requirements and retention period. - [] Determine whether penetration testing is required, by whom, and when it can be scheduled. **Why it matters:** pen-test slots book out weeks in advance and remediation time must be in the plan, not after it. - [] Confirm the required remediation window between pen-test findings and go-live. - [] Determine WAF, DDoS and bot-management requirements. - [] Confirm hosting region and data-residency constraints. - [] Determine GDPR/CCPA obligations: DSAR process, consent, right to erasure across integrated systems. - [] Identify industry-specific compliance obligations (ITAR/EAR, FDA 21 CFR Part 11, ISO 9001 document control, AS9100). - [] Confirm export-control screening requirements and the screening provider. - [] Determine backup, DR and business-continuity requirements with RTO/RPO per system. - [] Confirm the availability SLA the business expects and whether it is

contractual. - [] Determine the maintenance window and whether the storefront may ever be down. - [] Confirm insurance and liability requirements affecting architecture decisions. - [] Obtain approval to store any client data in the delivery team's environments.

20. Data governance and MDM readiness

[] Confirm whether a data governance function exists, with names. - [] Determine who may create, edit and approve item master records today. - [] Determine who may create and edit customer master records. - [] Obtain the data quality standard, if documented, and confirm whether it is enforced by system or convention. **Why it matters:** a standard enforced only by convention degrades; measure current-state coverage rather than trusting the standard. - [] Establish the attribute governance model for the new catalog: who defines attributes, who populates them. - [] Determine whether a PIM will be introduced and, if so, whether it is in this phase. - [] Define the mastering decision per entity: where each attribute is created and where it is consumed. - [] Identify attributes that must be created new and cannot be sourced from any existing system. - [] Estimate the enrichment effort in records × attributes and confirm who performs it. **Owner:** MDM. - [] Determine whether enrichment will be internal, outsourced, supplier-sourced or AI-assisted. - [] Define the data validation rules the integration will enforce and the rejection behaviour. - [] Determine the deduplication approach and the match rules for customers. - [] Define the data migration scope, cutover approach and reconciliation method. - [] Establish ongoing data stewardship after launch, with named owners and time allocation. **Why it matters:** catalog quality decays without a steward, and the decay is invisible until conversion drops. - [] Complete the data readiness scorecard (`data-readiness-scorecard.md`) and review it with the sponsor.

21. Reporting and analytics

[] Identify the KPIs the business will use to judge the project, mapped to section 2 targets. - [] Determine the reporting tool of record and whether commerce data must land in a warehouse. - [] Confirm existing BI infrastructure and whether the ERP already feeds it. - [] Define web analytics requirements including enhanced eCommerce events. - [] Determine attribution requirements across offline and online orders. **Why it matters:** in B2B most influenced revenue closes offline; attribution that ignores this will understate the project's value to the board. - [] Define the reporting requirement for rep-assisted and rep-influenced online orders. - [] Determine operational reporting needs: integration health, failed orders, exception queues. - [] Define who monitors exception queues daily after launch and their backup. - [] Confirm customer-facing reporting requirements: spend by ship-to, purchase history export, budget tracking. - [] Determine data export requirements and formats requested by customers. - [] Confirm reporting data retention and archival requirements.

22. Organisational readiness and change management

[] Determine who will own the site operationally after launch and whether that role is filled. - [] Establish the post-launch support model: client-run, vendor-run, or hybrid, with hours. - [] Confirm the customer service team's capacity to handle portal enquiries during launch. - [] Define the training plan for internal users: CS, sales, warehouse, finance. - [] Define the training and onboarding plan for customers, including the migration campaign. **Owner:** MKT/SALES. **Why it matters:** a portal with no customer onboarding campaign reaches roughly the adoption of an unannounced one, which is near zero. - [] Identify the customer communication plan and who signs it off. - [] Identify accounts that will be migrated first and the pilot group. - [] Determine incentives for online ordering, if any, and their finance approval. - [] Assess sales team sentiment toward the project honestly and record it. **Owner:** SALES. - [] Determine whether any role is being eliminated or materially changed by the project, and whether that is known. - [] Identify the internal champions and the internal sceptics by name. - [] Confirm executive visibility and communication cadence to the wider business. - [] Establish the process for handling in-flight change requests during build. - [] Confirm the client team's prior experience with projects of this type.

23. Project constraints

☐ Obtain the target go-live date and the origin of that date. **Why it matters:** a date announced by marketing is a planning input to be managed, not a technical estimate to be defended. - ☐ Determine which is fixed: date, scope or budget. Get the answer from the sponsor, not from the project manager. - ☐ Obtain the approved budget and whether it covers remediation, licences and third-party fees. - ☐ Identify budget items not yet approved and their approval path and duration. - ☐ Obtain the client resource plan: names, roles, allocated hours per week, confirmed by their managers. - ☐ Identify client resource conflicts with other initiatives during the project window. - ☐ Obtain the blackout calendar: month-end and quarter-end close, physical inventory, peak season, trade shows, plant shutdowns, holidays. - ☐ Confirm whether cutover may occur during a fiscal close. **Owner:** FIN. - ☐ Identify external dependencies with fixed lead times: hardware, licences, carrier account setup, merchant account underwriting, procurement network certification. - ☐ Determine the client's procurement process for approving vendor invoices and change orders. - ☐ Confirm contract vehicle for change orders and the approval threshold. - ☐ Identify any regulatory or contractual deadline driving the date. - ☐ Determine the consequence of missing the date, stated plainly. - ☐ Confirm the phasing preference and whether a soft launch or pilot is acceptable. **Why it matters:** a pilot with 10 friendly accounts de-risks more than any amount of additional UAT, and it is a commercial decision.

24. Risks and open questions

☐ Record every unanswered checklist item as a risk with an owner and a target date. - ☐ Record every item answered verbally but not evidenced, marked as unverified. - ☐ Quantify the estimate impact of each open item as a range, in days or dollars. - ☐ Identify the three unknowns with the largest estimate variance and plan specific actions to close them. **Owner:** PM. **Why it matters:** discovery exists to reduce variance; closing three large unknowns beats documenting thirty small ones. - ☐ Identify assumptions being carried into the estimate and publish them explicitly. - ☐ Identify dependencies on third parties who have not yet engaged. - ☐ Record any stakeholder who declined to participate and what that leaves unverified. - ☐ Confirm the contingency percentage applied and the reasoning behind it. - ☐ Agree which risks are owned by the client and which by the delivery team, in writing. - ☐ Schedule the risk review cadence for the build phase.

PART 4

Day-one access request list

Submit these on the day the project is approved, not when the build is ready.

Submit all of these within the first 48 hours, in one consolidated request, to a single named coordinator. Lead times below are typical for mid-market industrial clients; treat anything over five days as a schedule dependency and track it as such.

#	ITEM	TYPE	REQUEST FROM	TYPICAL LEAD TIME	BLOCKS
1	ERP sandbox URL, credentials, integration user	Credential	ERP	3–15 days	All ERP validation, feasibility of every write interface
2	ERP production read-only account (for profiling)	Credential	ERP/SEC	5–20 days	Data profiling, readiness scorecard
3	ERP API documentation and version	Document	ERP	1–3 days	Integration design
4	Full item master export (all columns, CSV)	Export	MDM/ERP	2–5 days	Sections 5, 20, scorecard
5	Customer master export including ship-tos and contacts	Export	MDM/ERP	2–5 days	Sections 6, 20, scorecard
6	Price list, contract and matrix export	Export	ERP/FIN	3–10 days	Section 7, pricing architecture
7	Price output for 3 named customers × 20 items	Export	ERP	3–7 days	Pricing rule validation
8	Inventory snapshot by location with allocation fields	Export	ERP/LOG	2–5 days	Section 8
9	12 months of order header and line history	Export	ERP/FIN	3–7 days	Volume sizing, personalisation, phasing
10	Open order and backorder snapshot	Export	ERP	2–4 days	Cutover planning
11	Item dimensions and weight coverage report	Export	ERP/LOG	2–5 days	Shipping feasibility
12	Freight class (NMFC) coverage report	Export	LOG	3–7 days	LTL rating feasibility
13	Tax code assignment coverage report	Export	FIN	2–5 days	Tax design
14	Exemption certificate register with expiry dates	Export	FIN	3–10 days	Tax and compliance risk
15	Current site analytics access (GA4 or equivalent)	Credential	MKT	1–3 days	Baseline, redirect plan
16	Internal site search logs including zero-result queries	Export	MKT	2–5 days	Search and taxonomy design
17	Current site admin/CMS access	Credential	MKT/3P	3–10 days	Content migration scope
18	DNS/registrar access confirmation (who, not the password)	Confirmation	IT	1–5 days	Cutover
19	Existing architecture diagram and network topology	Document	IT	1–5 days	Integration architecture

#	ITEM	TYPE	REQUEST FROM	TYPICAL LEAD TIME	BLOCKS
20	VPN/allow-list access for the delivery team	Credential	IT/SEC	5–20 days	All connectivity testing
21	Firewall change request process and calendar	Document	IT	1–3 days	Connectivity planning
22	Security requirements checklist / vendor questionnaire	Document	SEC	3–10 days	Architecture, go-live gate
23	Middleware/iPaaS tenant access	Credential	IT	3–10 days	Integration build
24	CRM sandbox and API credentials	Credential	IT/SALES	3–10 days	CRM interfaces
25	PIM/DAM access and asset inventory	Credential	MKT/MDM	2–7 days	Content workstream
26	WMS/OMS access or documentation	Credential	LOG/IT	5–15 days	Fulfilment interfaces
27	Tax engine sandbox and company profile	Credential	FIN	3–10 days	Tax interface
28	Payment gateway sandbox and merchant ID	Credential	FIN	5–15 days	Checkout build
29	Carrier accounts and API credentials	Credential	LOG	5–15 days	Rating build
30	EDI provider contact, transaction list, map inventory	Document	IT/FIN	3–10 days	EDI scope and cost
31	Brand guidelines and logo assets	Document	MKT	1–3 days	Design
32	Product image library access and sample files	Export	MKT	2–7 days	Content workstream
33	Sample of 20 fully populated products (all data, all assets)	Export	MDM	3–7 days	Catalog design proof
34	List of ERP customisations affecting order/item/customer	Document	ERP	3–10 days	Integration risk
35	ERP upgrade and patch calendar (18 months)	Document	ERP/IT	1–5 days	Schedule risk
36	Blackout calendar (close, inventory, peak, shutdown)	Document	FIN/LOG	1–5 days	Cutover date
37	Client resource plan with confirmed hours	Document	SP/PM	3–10 days	Plan feasibility
38	Signed SOW, prior change orders, incumbent contracts	Document	PM/SP	1–5 days	Scope baseline
39	Existing integration documentation and source code access	Document	IT/3P	5–20 days	Reuse assessment

#	ITEM	TYPE	REQUEST FROM	TYPICAL LEAD TIME	BLOCKS
40	Named contact and SLA at each third-party vendor	Document	IT	2–7 days	Dependency management

Track this table as a live artifact. Report weekly on items not yet received, with the number of days outstanding and the workstream blocked. An access item outstanding for more than two weeks is a project issue and should be escalated to the sponsor by name.

PART 5

Discovery red flags

Each of these findings should change the estimate, the phasing, or the recommendation. When one appears, record it, quantify it, and put it in the readout — not in a footnote.

RED FLAG	WHAT IT USUALLY MEANS	EFFECT ON ESTIMATE / PLAN	RECOMMENDED RESPONSE
No sandbox exists and none is planned	Every integration must be built and tested against production, or not at all	Adds 3–8 weeks; may make the project infeasible as scoped	Make sandbox provisioning a contractual prerequisite before the build phase starts
ERP is multi-tenant cloud with a forced upgrade in the project window	The integration surface may change mid-build without your consent	Adds 10–20% contingency plus regression testing per upgrade	Get the upgrade calendar in writing; schedule regression windows into the plan explicitly
Heavy ERP customisation in order entry or pricing	Standard connectors will not work; every assumption must be re-verified	Typically 1.5–2x the integration estimate	Re-baseline after inspecting the customisations; refuse to quote firm before that inspection
Direct database writes proposed or already in use	Fast now, breaks at the next upgrade, unsupported by the vendor	Adds long-term risk and near-term rework	Recommend API-based approach; document the risk if overruled and get sign-off
No named master data owner	Nobody will do the remediation	Data workstream has no owner and will slip	Escalate as a staffing finding to the sponsor; do not accept "we'll figure it out"
Item descriptions are internal shorthand across most SKUs	The catalog is not sellable as-is	Content workstream measured in months, not weeks	Score with the readiness scorecard; propose phased SKU release and enrichment workstream
Attribute coverage under 40% on in-scope SKUs	Faceted navigation and search relevance cannot be delivered as promised	Reduces phase 1 scope or adds enrichment cost	Reduce facets to what data supports; quote enrichment as a separate workstream
Imagery coverage under 50%	The site will look unfinished at launch regardless of build quality	Adds photography or supplier-sourcing effort with long lead times	Start the imagery workstream in week one; it has the longest lead time of any content task
No weight or dimension data	Carrier rating cannot be implemented	Removes real-time shipping from scope	Fall back to rate tables or quoted freight for phase 1; start data capture immediately
No NMFC/freight class on heavy items	LTL rating is impossible	Freight-quote-on-request instead of checkout rating	Confirm the business accepts quoted freight; this is a conversion trade-off
Pricing described verbally but not documented, and the sample export disagrees with the description	There are rules nobody remembers	Pricing is the highest-risk area; add 30–50% to that component	Insist on the three-customer price sample; do not design pricing from a verbal description
Contract pricing stored outside the ERP (spreadsheets)	The system of record for revenue-critical data is unmanaged	Adds a data migration and governance workstream	Surface as a business risk beyond the project; propose ERP consolidation
Inventory accuracy below 95% by cycle count	Published availability will be wrong and will be blamed on the website	Requires banded availability or a warehouse workstream	Recommend threshold display; propose inventory accuracy improvement as a prerequisite

RED FLAG	WHAT IT USUALLY MEANS	EFFECT ON ESTIMATE / PLAN	RECOMMENDED RESPONSE
Real-time ERP pricing call exceeds ~300ms under load	Category pages cannot call it per item	Forces a caching or replication architecture	Redesign to cached pricing with defined invalidation; cost the cache layer explicitly
ERP licenses integration traffic per user or per call	Ongoing cost nobody budgeted	Can add five figures annually	Get a written quote before architecture is finalised; it may change the integration pattern
VPN or firewall access lead time exceeds four weeks	Build cannot start on schedule	Directly delays the critical path	Start the request on day one; make it a sponsor-visible milestone
Security review has not started and no requirements checklist exists	Architectural findings will arrive late	The classic month-five detonation	Demand the written requirements in discovery; add a pen-test and remediation window to the plan
No non-production environment for an adjacent system (tax, WMS, gateway)	End-to-end testing is impossible	UAT quality drops; defects move to production	Identify in discovery; either procure the environment or descope the interface
Launch date was announced externally before discovery	Date is fixed, so scope must move, and nobody has said so	Guarantees a scope fight later	Force the fixed/variable decision with the sponsor in writing during discovery
Sponsor cannot fund data remediation	The critical path has no budget	Launch will slip or scope will shrink	Present remediation as a line item with a number; make the trade-off explicit and documented
ERP admin allocated fewer than 4 hours per week	The most load-bearing participant is unavailable	Add 30–40% contingency to integration	Quantify the requirement to the sponsor as a resourcing decision, not a scheduling one
Sales leadership has not addressed channel conflict	Adoption will fail regardless of build quality	The business case does not land	Raise as a decision item for the sponsor with options; do not treat it as a requirement
Incumbent vendor is being replaced and controls DNS or hosting	Cutover depends on a party with no incentive to help	Cutover risk, potential multi-week delay	Put every request in writing, copy the sponsor, log non-responses with dates
No customer onboarding or migration campaign planned	The portal will launch to an empty room	The business case fails post-launch	Make the campaign a named workstream with a named owner before go-live
Historical order data must be migrated "all of it"	A data migration project has been hidden inside a commerce project	Adds weeks and a reconciliation obligation	Negotiate a cut-off (typically 24 months) with live ERP lookup beyond it
Client has never run a project of this type	Governance and decision latency will be the constraint, not technology	Add decision-latency buffer to every milestone	Increase governance cadence; define decision SLAs in the plan
Requirements arrive as a feature list with no journeys	Scope is undefined and will expand to fill the schedule	Scope creep is guaranteed	Convert to journeys and produce an explicit "not in phase 1" list signed by the sponsor

Discovery readout — minimum contents

[] Scope baseline with an explicit "not in phase 1" list, signed by the sponsor. - [] Completed integration inventory register with every row marked verified or unverified. - [] Completed data readiness scorecard with total score, band and remediation workstreams. - [] Estimate expressed as a range with the assumptions that produce each end of it. - [] Red-flag register with quantified impact and owner per item. - [] Access request status with outstanding items and days elapsed. - [] Named owners for every workstream, client-side and delivery-side. - [] Recommended phasing with the reasoning for each boundary. - [] Decision log of everything the client decided during discovery, with dates.

Compiled from delivery work on ERP-integrated commerce programmes. Complimentary — share it inside your organisation freely.