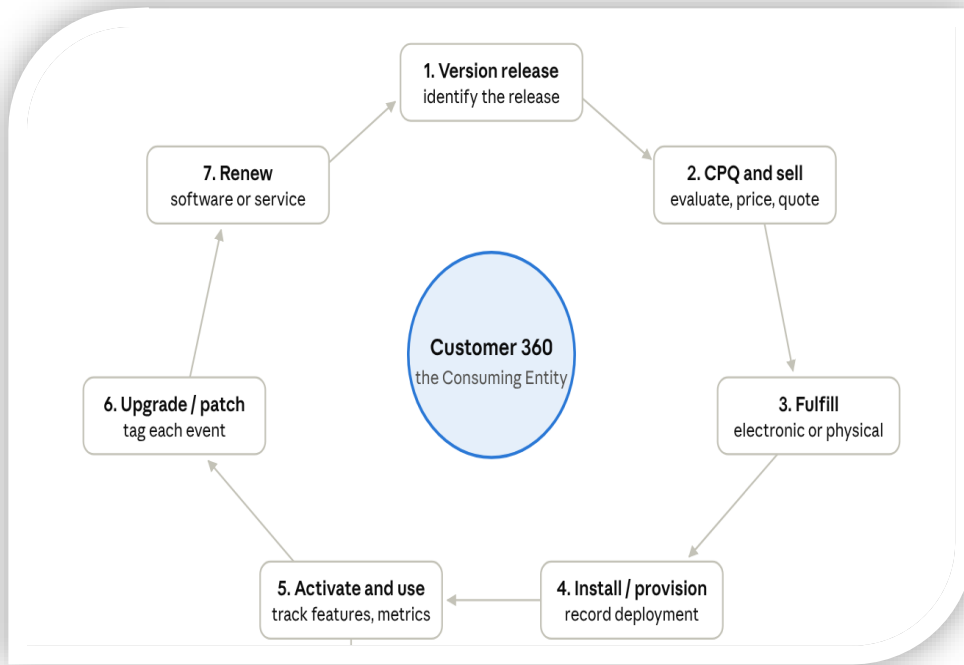


Enterprise Architecture

M&A Integration Playbook



David Wright Sr –
Enterprise Architect

Contents

1. Purpose of this Playbook	4
1.1 The value of a single integration framework	4
1.2 How to use this playbook	4
2. M&A from the EA Perspective	5
2.1 Overview	5
2.2 Strategic Integration Approaches	5
2.3 The Enterprise Reference Architecture	6
2.4 The Commerce Model	6
2.5 Fundamental Process of Software/Service Monetization	7
2.6 Summary	10
3. Data-Centric Enterprise Architecture	11
3.1 The Layered View	11
3.2 The End-to-End (E2E) View	12
3.3 Business Layer E2E	14
3.4 Security Layer E2E	21
3.5 Data Layer E2E	21
3.6 Application Layer E2E	26
3.7 Technology Layer Overview	27
4. The Integration Analysis Process	29
4.1 Introduction	29
4.2 Process Gap Analysis	29
4.3 Data Analysis	31
4.4 Application Analysis	32
4.5 Technology Analysis	32
5. The Integration Execution Process	34
5.1 Customer-Journey Driven Use-Case Integration	34
5.2 Phased Integration	34
5.3 Acquired Technology Integration and Capability Migration	35
5.4 Worked Examples of Use-Case Driven Integration	36
Appendix A. Master Data Configuration	38
A.1 Rights BDO Attribute Definitions	38



A.2 Example Rights/SKU NPI Process	41
Appendix B. Product Go-to-Market Scenarios.....	43
B.1 Segmentation.....	43
B.2 Customer Experience	43
B.3 Route to Market.....	44
B.4 Business Operation Application Sequences	45
Appendix C. App Access File: Fulfillable Item JSON	47



1. Purpose of this Playbook

This playbook gives an M&A integration team one repeatable, Enterprise Architecture (EA) led method for integrating an acquired business into an enterprise that sells software, content and services. It defines the reference models, analysis steps and phased execution approach needed to take an acquisition target from due diligence to full operation on the Enterprise Business Platform (EBP).

Throughout, **the Enterprise** is the acquiring organization, **the Acquisition Target** (or Target) is the business being acquired, and the **Enterprise Business Platform (EBP)** is the Enterprise's set of back-office and customer-facing subsystems that run the commercial life cycle.

1.1 The value of a single integration framework

A single EA M&A integration framework is:

- Aligned with, and driven by, the end-to-end (E2E) Customer Journey
- Usable for any acquired business model
- Usable for any existing or new go-to-market (GTM) model
- Usable for any existing or new business unit
- Supportive of the Enterprise's international business compliance program
- Usable for any prioritized business-capability integration sequence
- Built on reusable M&A integration process patterns
- Built on Enterprise-standard Business Data Objects (BDOs)
- Built on reusable Enterprise Architecture patterns
- Built on reusable M&A integration technology

1.2 How to use this playbook

Part	Use it to	When
2. M&A from the EA Perspective	Choose an integration strategy and understand the commerce and rights model	Strategy and due diligence
3. Data-Centric Enterprise Architecture	Analyze the Target layer by layer across the E2E life cycle	Due diligence and planning
4. The Integration Analysis Process	Run use-case driven gap analysis of process, data, application and technology	Planning
5. The Integration Execution Process	Integrate and migrate in phases, each executed, tested and released E2E	Execution
Appendices	Configure rights and SKU master data; understand GTM scenarios and transaction sequences	Execution and operation



2. M&A from the EA Perspective

2.1 Overview

An acquisition triggers an integration exercise with two goals for the Enterprise after the merger:

1. The Target's products, services and operations are integrated with the Enterprise's, so that internal processes and tools are, as far as possible, single solutions for each capability needed to trade with partners and customers.
2. The Target's customer buying and user experience improves across the whole Customer Journey by conforming to one consistent Enterprise look and feel, whatever the journey stage or product mix.

The integration strategy depends on the strategic and business purpose of the deal:

- **IP-only acquisition.** If the deal exists to acquire intellectual property, operations may not need to merge at all.
- **Like-for-like business.** If the Target sells similar products through the same GTM scenarios, using capabilities the Enterprise already runs, the strategy is to migrate the Target's information onto the Enterprise infrastructure. The Enterprise operating infrastructure is then "wrapped around" the Target's product and service platforms through well-defined, architecturally abstracted touch points. Those touch points are refined with every deal, so each successive integration takes less effort.
- **Unlike business.** If the Target's products and GTM scenarios differ significantly, wrapping still applies. The GTM may require re-engineered or new business capabilities, but the fundamental Value Chain stays the same. If the processes, information and applications along that chain are well documented, the touch points hold; the abstractions are extended rather than replaced.

2.2 Strategic Integration Approaches

The integration model is built on the Customer Journey laid across the Product/Service Life Cycle. That cycle is the foundation of the Enterprise Value Chain, which in turn rests on the capabilities that manage the Enterprise's information, expressed as Business Data Objects. A **Business Data Object (BDO)** is an abstraction describing a business concept such as customer, sales order or order line item.

- At one extreme, the Target uses like capabilities and applications across the same Value Chain. Integration is then mostly BDO analysis and migration of master and transactional data to the Enterprise infrastructure.
- At the other extreme, BDOs are unlike or unmatched, so migration is far larger. It is accompanied by new or changed capabilities for new business processes, GTMs and product or service types.



2.3 The Enterprise Reference Architecture

Few enterprises start with a complete E2E understanding and a coherent reference model; most build them incrementally. Each integration exercise therefore evolves, refines and documents the reference model further.

With each integration the touch points between new or modified capabilities and the reference EA are standardized as far as possible. The aim is repeatable processes, standard logical information structures (BDOs) and maximum reuse of APIs.

2.4 The Commerce Model

The Enterprise sells "Product" through many GTM models to "Consumers" of that Product. In this playbook, Product can mean:

1. **A tangible product:** a physical item that may be perceived by sense or touch.
2. **An intangible product:** a license or entitlement to use software, content such as digital art or documents, or the capability of a tangible product.
3. **A service:** support, anything-as-a-service (*aaS), consulting.

A software and services enterprise trades mainly in types 2 and 3. In legal terms the Product remains owned by the Enterprise; what is transferred to another party through a business transaction is an entitlement to the capability the Product delivers.

The Enterprise sells rights to use, never the IP itself

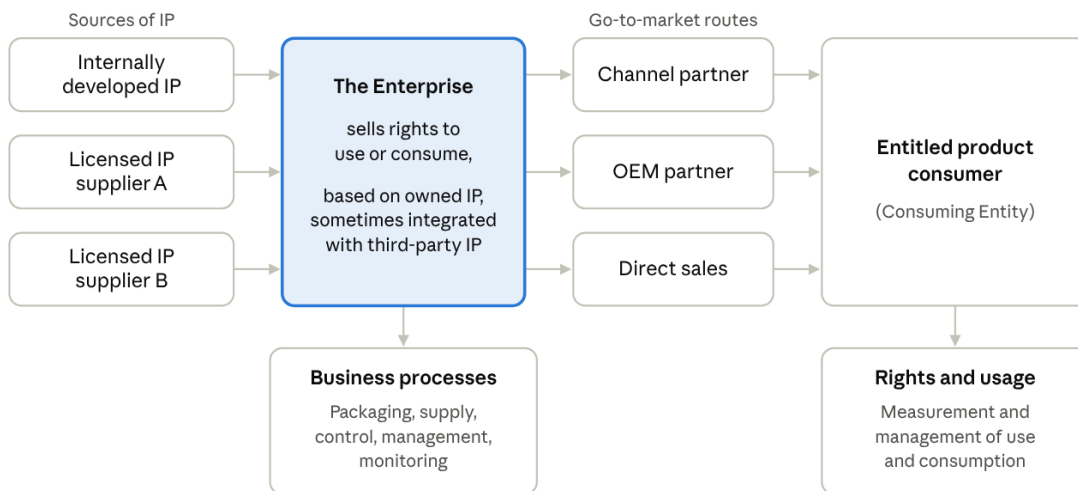


Figure 1: Software/service monetization chain of commerce

IP from internal development and licensed suppliers is packaged into rights, which reach the consumer through channel, OEM and direct routes.

Because the asset transferred to the consumer, the **Consuming Entity**, is a license for use, these Products are best described as **Entitled Products**.



A sale transfers rights to use; ownership stays with the Enterprise



Figure 2: Entitled product/service transfer-of-asset commerce model

The Enterprise is obliged to supply the rights purchased; the end user is obliged to comply with the rights' restrictions.

2.5 Fundamental Process of Software/Service Monetization

When a sale is made through any GTM path, a license passes from the Enterprise to the Consuming Entity in exchange for money. The Consuming Entity acquires a license to use or consume the capabilities of the Product; title stays with the Enterprise. Capabilities might include:

- The capability offered by on-premise software
- The capability offered by Software as a Service (SaaS), or any *aaS
- Access to content owned by the Enterprise, such as stock photos, images and video

The license defines how those capabilities may be used, with specific rights and restrictions. A typical digital-content license, for example, lets the buyer use an asset worldwide and in perpetuity across print, presentations, broadcast, web and social media, but not redistribute it so others can use it unlicensed. Restrictions then vary by license tier:

License tier	Copies or views	Web views	Email, mobile, social, broadcast use	Merchandise or products for resale	Other limits
Standard	Up to a fixed cap (e.g. 500,000)	Unlimited	Allowed	Not allowed	None
Enhanced (premium assets)	Unlimited	Unlimited	Allowed	Not allowed	None
Extended	Unlimited	Unlimited	Allowed	Allowed	Purchased as an upgrade to Standard
Editorial use only	Unlimited	Unlimited	News and events only	Not allowed	No advertising or commercial use; no editing

A rights model can therefore be complex, particularly once it is placed in the supply chain between the Enterprise and the Consuming Entity.

2.5.1 Definition of Product/Service Rights (the Asset)

Characteristics of Rights. A Right, or set of Rights, has these characteristics:

- A Provider Entity (the Enterprise) owns an "Object of Rights" (the Product) whose capabilities may be provided to Consumer Entities.
- There may be one or many product capabilities, each a feature with a metric over some period, for which rights are granted or restricted.
- An instance of a Right to Use or Consume is granted by the Provider Entity to a Consumer Entity and becomes the Consumer Entity's property.
- The Consumer Entity, as license owner, has the rights and restrictions defined in the Rights to consume or manage the capabilities of the Object of Rights.
- The Provider Entity must provide every Consumer Entity with the capabilities defined in the Rights it issued, but is never obliged to relinquish ownership of the Object of Rights.
- All Rights carry complete provenance, including origin and any title or ownership transfer.

The accounting definition of an asset applies: *"a resource controlled by an individual or enterprise as a result of past events and from which future economic benefits are expected to flow to the individual or enterprise."*

Structure of the License/Entitlement BDO. The logical model below follows the ISO/IEC 19770-3 entitlement standard and sets the entitlement beside the usage or consumption of the capabilities it grants.



Usage mirrors the entitlement it is consumed against

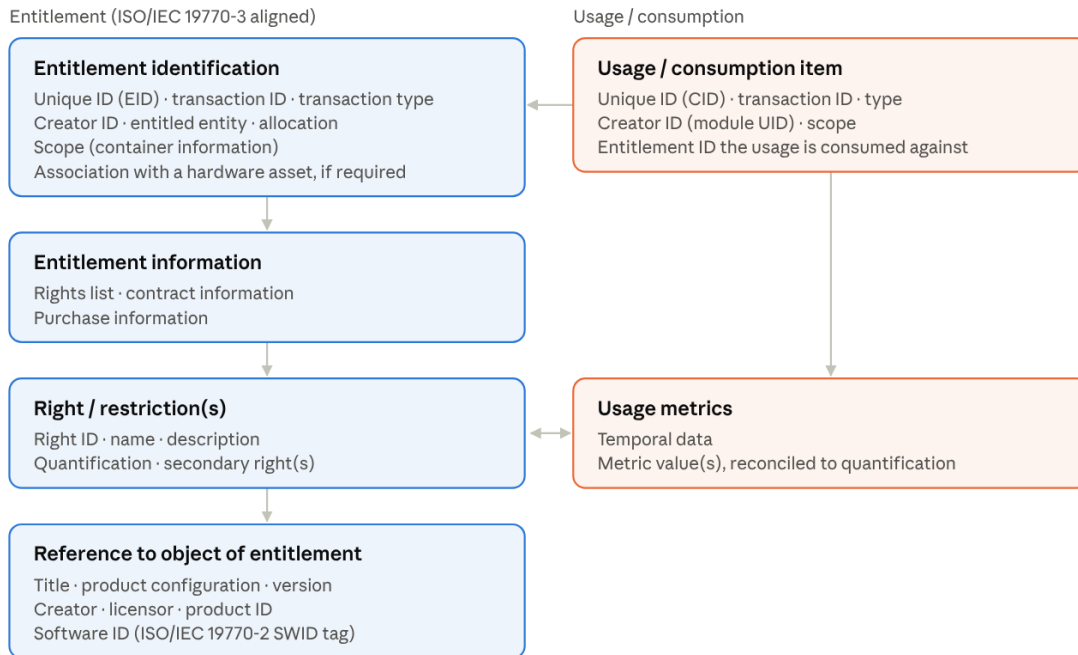


Figure 3: Attributes of a license/entitlement and usage

Each usage item points back to the entitlement ID it consumes, and its metrics are reconciled against the right's quantification.

2.5.2 Impact of Rights on M&A Processes

Rights encapsulate part of "the Offering". Understanding the Target's GTM scenarios and its complete range of offerings across all GTMs is therefore essential to configuring and supporting them across every life cycle in the Enterprise process structure and infrastructure.

Data migration must carry reference data and metadata describing rights models, SKU structures, bundling, fulfillment provisioning and usage metering and billing. Without it, the destination infrastructure cannot support what may be new concepts in those domains (see Appendix A).

2.5.3 Definition of the Consuming Entity (the Legal Owner of a Rights Asset)

After distribution through one or more GTM models, Rights are acquired by the eventual consumer of the capabilities. The Consuming Entity may be an individual (for example, a buyer of consumer desktop software) or a business, from small business to enterprise. Because it must be qualified to hold an asset, the legal holder of Rights must be a person or a legal entity.

The Consuming Entity owns the Rights but may manage and allocate them to others. A corporation may allocate Rights to individuals, roles or departments, yet it remains legally liable for compliance with the Rights' terms. Therefore:

- A product may require the Consumer to perform specific operational roles to manage or use the services it provides.
- The Consumer Entity may delegate specific roles to entities the Enterprise has authorized, as the product's services or consumables require.
- The Rights owner may have the right to split or allocate sub-components of its Rights and grant them onward to other Consumer Entities.
- Those Consumer Entities may or may not receive title to the sub-component Rights, depending on the business arrangement in the original granting transaction.
- Roles act within the bounds of the Rights and their role authorization, which may, for instance, prevent visibility of Rights managed or consumed by other roles. Ownership of all Rights stays with the Consumer Entity and is not transferred to a role or delegate.

2.6 Summary

GTMs, rights offerings and usage metering and billing are the heart of an entitled-product business. Analyzing all three is the precondition for integrating the Target's products, services and operations successfully.



3. Data-Centric Enterprise Architecture

To avoid silos and a domain-centric architecture, the M&A EA practice analyzes, designs and builds the reference architecture on a two-dimensional matrix: architectural **layers** crossed with the **E2E Customer Journey/Value Chain**.

3.1 The Layered View

Business data objects sit at the centre of the EA stack

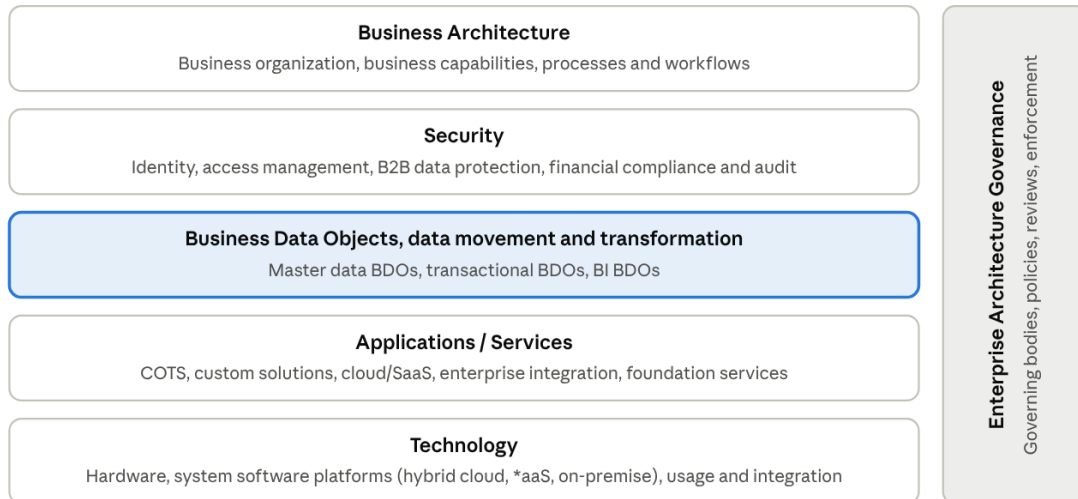


Figure 4: Information-centric EA analytical stack

- **Business Architecture** sits at the top: the organization, capabilities and processes or workflows that run the business.
- **Security Architecture** is an integral layer, not an add-on. EA and information security both manage risk, so they need one way of thinking, solutioning and governance. This layer covers identity management (people and process), access management, compliance and audit.
- **Information (BDOs and data flows)**. An enterprise has three major assets: employees, IP and information, and information is the business of EA. BDOs describe business concepts (customer, sales order, order line item) independently of the tables that physically store them, which eases translating the business layer into the IT and cloud capabilities that support it.
- **Applications/Services** supply the raw business capabilities, orchestrated to manage and manipulate BDOs.
- **Technology** sits at the bottom. The ability to replace, upgrade, scale or repurpose technology must be preserved as technology changes.
- **EA Governance** spans all layers: governing bodies, policies, reviews and enforcement.

In an acquisition each layer of the Target is transformed into the corresponding layer of the Enterprise target architecture, with BDOs and data flows as the central path.

Each layer of the Target transforms into the matching Enterprise layer

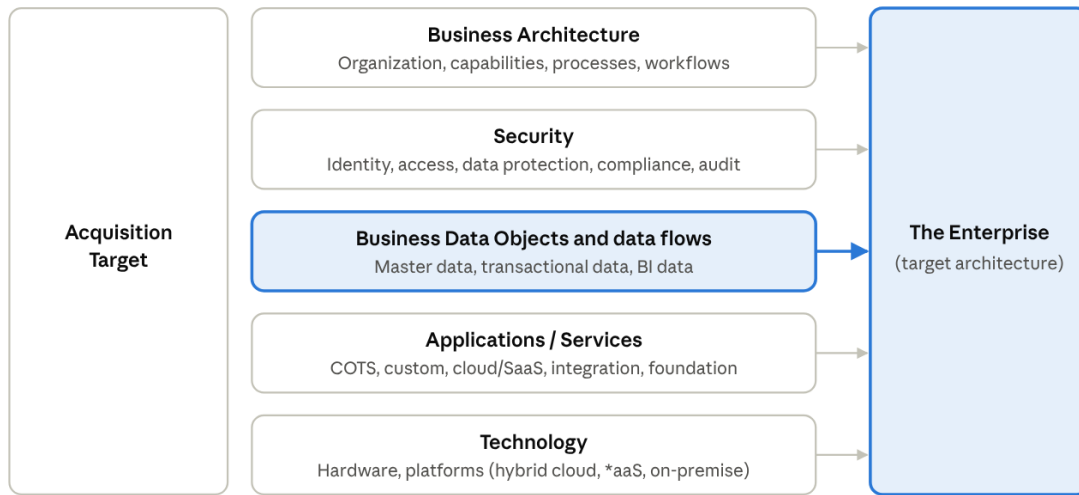


Figure 5: Multi-tiered M&A transformation

3.2 The End-to-End (E2E) View

The "inside-out" product life cycle lies alongside the "outside-in" Customer Journey. Together they form a **Customer 360** view, based on ISO SAM/ITAM standards for software asset and entitlement management.

Seven life cycle steps circle the customer

Product life cycle (inside out) laid on the Customer Journey (outside in); based on ISO SAM/ITAM standards

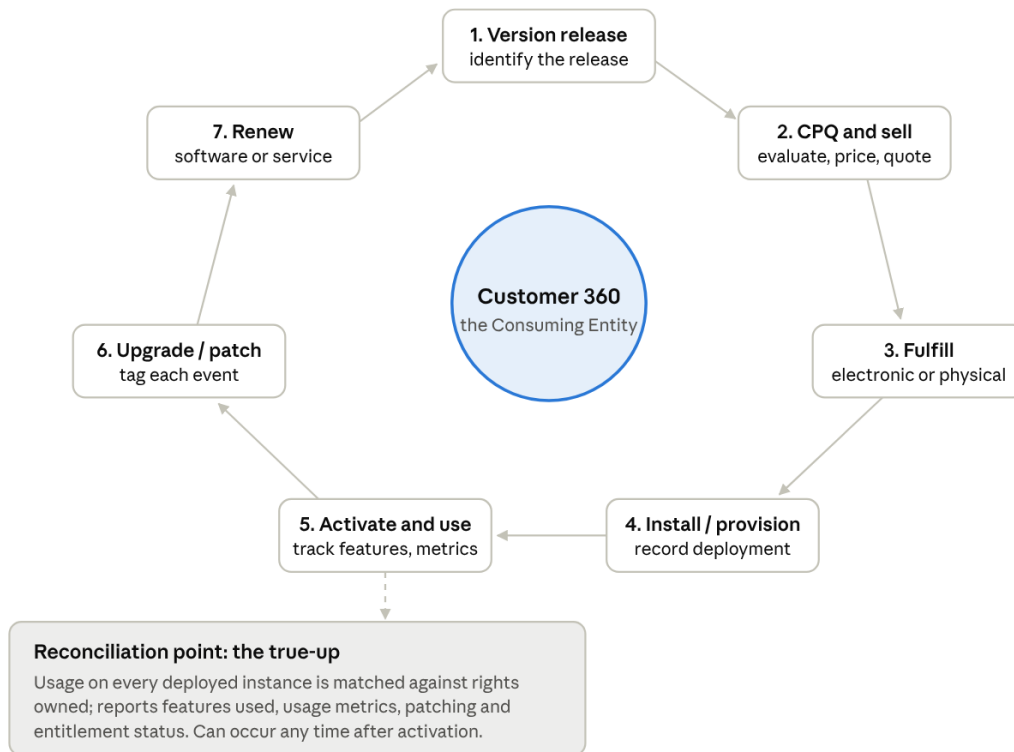


Figure 6: Life cycle integrated with the Customer Journey

- 1. Version release.** Initial release, then incremental releases or replacement by a new product that continues the same capabilities. In M&A, an acquired product offered under license must be defined and released within the Enterprise GTM framework.
- 2. CPQ to sell.** From initial lead (perhaps through try-and-buy), through request for quote, configuration and quote, deal negotiation where needed, to order submission into Enterprise order management. Each rights model (perpetual, subscription, pay-per-use, *aaS, meter type) must be tied to a unique item (SKU) here, so every transaction generates a uniquely identifiable Rights record managed across its whole life cycle.
- 3. Fulfill product and rights.** The Enterprise, or an OEM, can inject an identifier into each fulfilled shipment, tying it back to its business transaction and Rights.
- 4. Install or provision.** Installers customize each installation with installation-specific information (time, date, authorization) in addition to publisher data, per ISO/IEC 19770-2. Provisioning is often product-dependent and rarely a single process.
- 5. Activate and use.** Activation, indexed to the Rights' unique identifier, is the first act of usage, per ISO/IEC 19770-3. Products record activation and usage, and may meter incremental usage feature by feature or meter by meter.

6. **Upgrade or patch.** As at installation, each upgrade or patch is tagged with event data and can be tied to the entitlement that qualifies it.
7. **Renew.** A renewal transaction issues new Rights, which can be linked to previous renewals and other entitlements. A service may combine SaaS, maintenance and content (security signatures, map or document updates).

The reconciliation point is where usage is matched against Rights owned (the "true-up"). It can occur at any time after activation, and reports features used, usage metrics, patch and upgrade status and entitlement status for every deployed instance.

3.3 Business Layer E2E

During due diligence, map the Target's capability requirements onto the Enterprise capabilities to form a **heat map**: green where the EBP already covers the capability, amber where it needs extension, red where it is missing.

3.3.1 Customer Journey/Value Chain Capability Map

The Level 1 capability map below follows the life cycle and Customer Journey above. Each EBP subsystem supports one or more journey stages.



EBP subsystem	Journey stage	Level 1 capabilities
Offer Management	Innovation and forecasting (NPI)	Promotions; eligibility and qualification; offer definition; redemption campaign admin; offer discovery; add-ons; upgrade/downgrade definition; product (SKU) pricing; available payment methods; offer-to-SKU mapping; redemptions; trials
Marketing	Discover, explore	Campaign planning and tracking; search; funnel and lead management; content management; multi-channel campaign execution; plans and pricing page; content generation; web store; demand generation; marketing and sales enablement; personalization; central unsubscribe
Sales	Evaluate, buy	Account management; quote management; contact and lead management; commerce; opportunity management; customer success; sales performance; professional services; field and training enablement; partner relationship management
Order Management	Buy	Initial purchase; renewal orders; additional purchase; eligibility; invoicing; up-sell and cross-sell; fulfillment (subscription, perpetual, consumable); payment method determination; payment validation; external order API; charge, cancel, tax, checkout
Contract Management	Buy	Customer rights management; rights template management; subscription life cycle management; usage and consumption management; compliance management; legal contract and agreement
Authorization	Implement	Activation; rights delegation; end-user provisioning
Customer Admin	Implement, adopt	User admin; deployment; product rights configuration
Payment Management	Adopt	Measured usage and compliance; usage billing mediation and aggregation; payment collection; revenue recognition; payment instrument; redemptions
Revenue and Financial	Adopt, renew	Revenue management; royalties and rebates; tax management; billing management; financial analytics and reporting
Customer Service	Expand/migrate, renew	Service management; remote support; case management; chat, messaging and telephony; post-purchase support; communities; knowledge base; channels and tools; support integration; add-on sales; renewal sales

3.3.2 Core Capability Map

Beneath the journey-specific subsystems sit common core capabilities shared by every business unit.



Domain	Core capabilities
Business Support	Master data management; operations, portfolio and program management; business operations solution delivery; process management; change control; corporate relationship management; PR communication; field enablement (channel and partners); identity management subsystem
Finance	Planning and performance management; corporate account management; payroll accounting; treasury; internal audit; global audit; tax; global trade
Information Technology	Operations management; hosting services; asset and content management; security services; data management; integration services; BI, reporting and analytics subsystem
Human Resources	Workforce planning and strategy; workforce management; payroll; talent management; HR service delivery
Physical Resources	Space and facilities; maintenance; sustainability and energy; capital projects; real estate portfolio; smart-building IoT
Governance, Legal, Risk and Compliance	Corporate governance; compliance and risk; IP management; litigation and eDiscovery; contract management; remediation; operational and reputational resiliency; legal services

3.3.3 Constituents of the Enterprise Architecture Realm

Every step of the life cycle has actors drawn from the Enterprise, its partners and its customers. Each constituent owns business processes that support specific life cycle steps; together these processes deliver the capabilities above. M&A integration, like any new internal offering, is a special case: its due diligence and integration processes must prove that every life cycle step works for the integrated products and acquired processes.

The matrix below records, for each stakeholder, the journey stages where it runs business processes. A business process is a sequence of actions performed by a specific stakeholder on BDOs to achieve a specific business result. Every stakeholder has processes that support the life cycle, so every stakeholder must be engaged.

Stakeholder	NPI	Mktg	Sales	OM	Contract	Auth	Admin	Pay	Finance	Cust Svc
M&A Office	X	X	X	X	X	X	X	X	X	X
IT	X	X	X	X	X	X	X	X	X	X
Product / Business Unit	X	X	X	X	X	X	X		X	X
Quote-to-Cash Operations		X	X	X	X			X	X	
Finance	X		X	X	X		X	X	X	X
Support	X	X	X	X	X	X	X	X	X	X
Data and Analytics	X	X		X	X	X		X	X	
Sales and Sales Operations	X		X	X	X		X	X	X	X
Marketing	X	X	X	X	X		X	X	X	X
Business Process Engineering		X			X	X		X	X	
Compliance	X	X	X	X			X			X
Brand Protection	X				X	X	X	X	X	
Channel Management	X	X	X	X	X			X	X	X
Systems Engineering	X				X	X	X	X	X	
Professional Services	X	X	X		X		X	X	X	X

Stakeholder	NPI	Mktg	Sales	OM	Contract	Auth	Admin	Pay	Finance	Cust Svc
OEM Partner	X	X	X	X	X	X	X	X	X	X
IP Provider	X	X	X		X		X			X
Channel Partner		X	X	X	X		X	X	X	X
Customer		X	X	X	X	X	X	X	X	X

Requirements analysis then proceeds stakeholder by stakeholder:

1. Define the enterprise BDOs being managed (product, entitlement, product feature component, customer, partner, contact/user, contract, vendor, user, role).
2. Define the governors of each BDO (the combination of data steward and stakeholder).
3. Work with each stakeholder to establish which BDOs, at which life cycle steps, require management by role.
4. Design a process set for each stakeholder and capability role, with explicit hand-on and hand-off points.
5. Use those hand-on and hand-off points to draw swim lanes that show the dependencies across stakeholder roles for E2E business-object management.
6. Determine how these processes shape the integration sequence.

3.3.4 Use-Case Driven Integration

Use cases across the Customer Journey are the basis for both analysis and execution. The main high-level use cases are:

- **Customer acquires Rights** to use or consume an Enterprise product, content or service: a new or existing customer acquires a new term set of Rights (subscription) directly, through a co-seller partner, or through an embedded (OEM) partner.
- **Customer activates** a product or service as permitted by the Rights.
- **Customer uses** a product or service, including events triggered by use outside the Rights' compliance and grace rules; usage is reported across the whole licensing life cycle, before and after renewals and add-ons.
- **Customer renews** subscription Rights that are expiring or expired, through any of the same channels.
- **Customer acquires add-on (or reduces) Rights** for an existing subscription, through any of the same channels.
- **Customer requests support** on a specific product instance covered by one of several contracts.

Integration is executed in phases that match these use cases across the journey.



Each journey use case becomes an integration phase onto the EBP

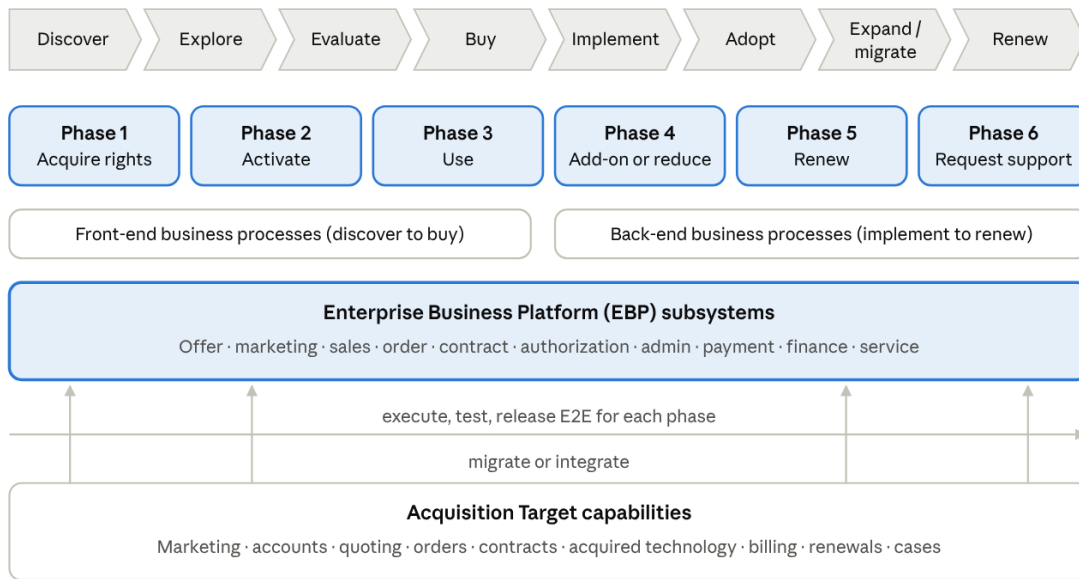


Figure 7: Customer Journey driven use-case integration in phases

Integration is partitioned into two distinct processes: (a) data and, where needed, capability migration; and (b) acquired technology integration. This playbook concentrates on (a), using BDOs for data migration. The reusable architecture for (b) rests on a small number of specific BDOs within the corporate data model (see section 5.3).

Use case analysis. Business processes are sequences of actions performed by specific organizations on business objects to achieve a specific result. They are built from scenarios describing the interaction between the actors (the constituents above) and the business objects.

Example: high-level enterprise sales process. The swim lanes below trace an enterprise deal from opportunity to invoice across Sales, the Deal Desk and Order Management, branching by business unit where product catalogs differ.

An enterprise deal passes from Sales to the Deal Desk to Order Management

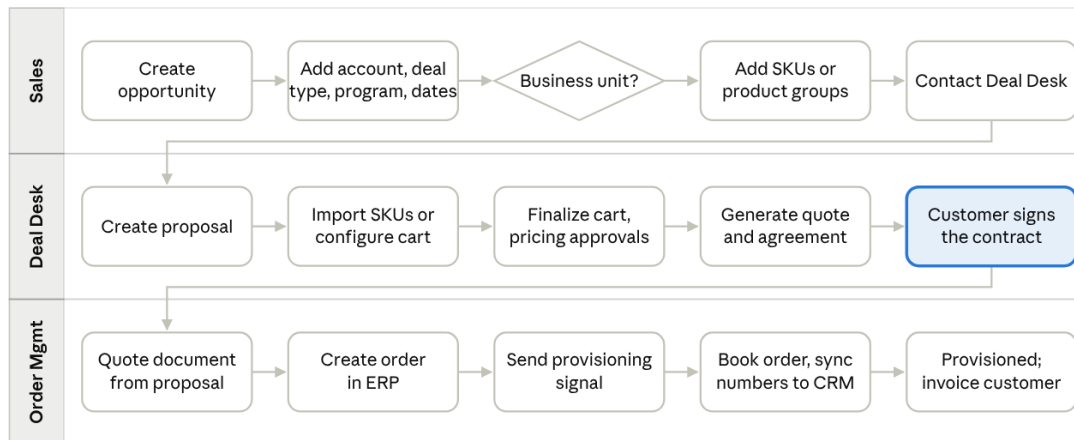


Figure 8: High-level enterprise sales process swim lanes

Sales creates the opportunity and adds products, either as actual SKUs or as high-level product groups, before handing to the Deal Desk. The Deal Desk builds the proposal and cart, triggers pricing approvals and produces the quote, agreement and contract for signature. Order Management converts the signed proposal into an order, sends the provisioning signal, books the order and synchronizes order data back to CRM before invoicing.

3.3.5 Business Process Mapping Templates by Stakeholder

Each stakeholder maps its acquired process set onto the Enterprise process set. The templates below are starting points.

New product introduction (NPI) operational process mapping

Acquired process set	Enterprise process set
Product offer and rights model definition	Governance: Enterprise standard rights models and meters
Product rights analysis	Enterprise standard rights product analysis
Product integration	Enterprise standard rights API sets and product integration across the life cycle
Pricing and licensing	Product rights pricing and licensing (rights SKU management, Appendix A)
Release	Release to production (RTP) of rights with the product, including a conference room pilot (CRP) of every operational process for the version

Finance revenue operational process mapping

Acquired process set	Enterprise process set
Order processing	Order processing
Line item processing	Rights sales order line item processing
Contract management	Contract management
Utility / billing	Utility / billing (standard touch point)
Invoicing	Invoicing
Payment	Payment

Customer self-service process mapping (admin tool usage)

Acquired process set	Enterprise process set
Product configuration	Self-service product rights and end-user configuration
Entitlement management	Self-service rights management
Quoting	Self-service quote management
Purchasing	Self-service purchasing
Deployment	Self-service deployment, patching and upgrade management
Fulfillment	Self-service product and rights fulfillment
Renewal	Self-service rights renewal management
Compliance	Self-service product usage and rights compliance
Trials	Self-service try-and-buy management

Example: E2E process analysis for a pay-per-use model. For a usage-based offering, the analysis follows each stakeholder role from order through usage to revenue.

In a pay-per-use model, metered usage drives billing and revenue

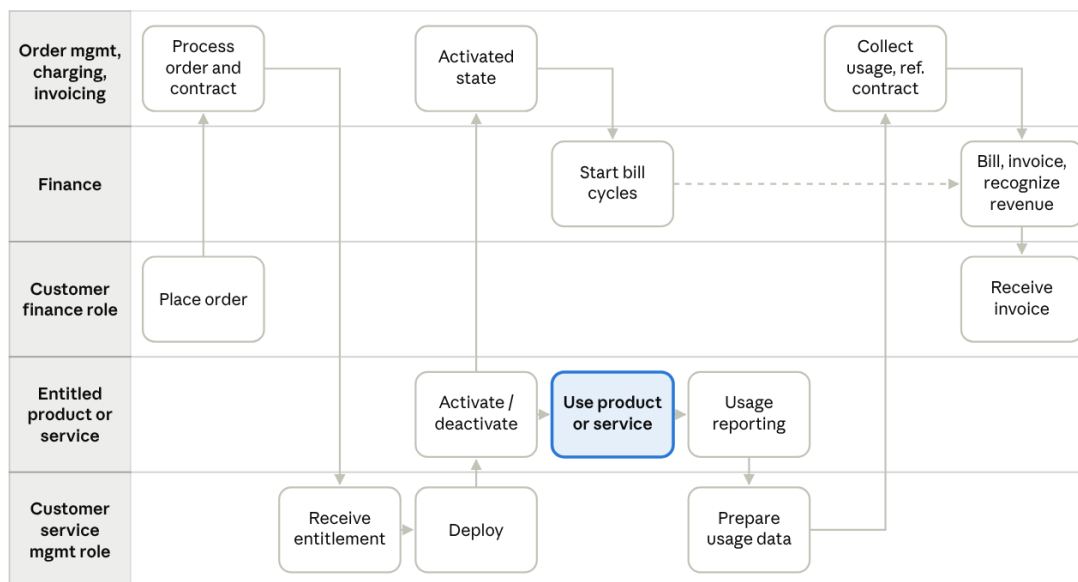


Figure 9: E2E process analysis, pay-per-use model

Revenue recognition then carves out service revenue, product usage revenue, sales commission and any revenue split with partners.

3.4 Security Layer E2E

3.4.1 Overview

The security layer controls who may act on which Rights at every life cycle step. Integration must map the Target's identity stores, roles and access policies onto the Enterprise identity management service and admin console, without interrupting customer access. The pattern differs by GTM segment.

3.4.2 Consumer GTM

A single admin user is enabled at provisioning (life cycle step 4). Depending on the product and licensing model, that admin, with Enterprise support if needed, delegates any other end-user access, to use or to manage the service, through the Enterprise admin console. Admin console access is managed by the Enterprise identity management service.

3.4.3 Mid-Market GTM

A business buys a team license and nominates one or more administrators. Administrators invite named users, who receive identical entitlements, and manage seat assignment and removal. Integration tasks: migrate the Target's team, admin and user records; preserve seat assignments; and map the Target's sign-in method (local accounts, social login, single sign-on) onto the Enterprise identity service.

3.4.4 Enterprise GTM

Enterprises federate identity through their own identity provider (single sign-on) and expect directory synchronization, role-based administration and audit. Rights may be delegated to departments, each with its own administrator who sees only that department's position (Appendix B). Integration tasks: re-establish federation trust with each customer's identity provider; map Target roles to Enterprise admin roles at system, product and department level; carry over audit history; and verify segregation of duties and data-protection obligations in each jurisdiction.

3.5 Data Layer E2E

3.5.1 Data-Centric View

The business layer defines the process flows needed to support the Target's business models and GTMs. The data-centric view builds on that layer and on the application layer beneath it. A data-oriented approach makes objects visible across system agents: data changes drive progress and composition, so data and context are shared between activities and sub-components.

- The Enterprise back office is a set of orchestrated systems, a "System of Systems" (SoS).



- It must rest on a data-centric architecture of **foundation master data** that supports commerce transactions, and **living transactional data** generated by the product and service commerce life cycles.
- It should provide responsive, distributed intelligence by taking computation to the data.

Every stakeholder works on the same foundational data objects

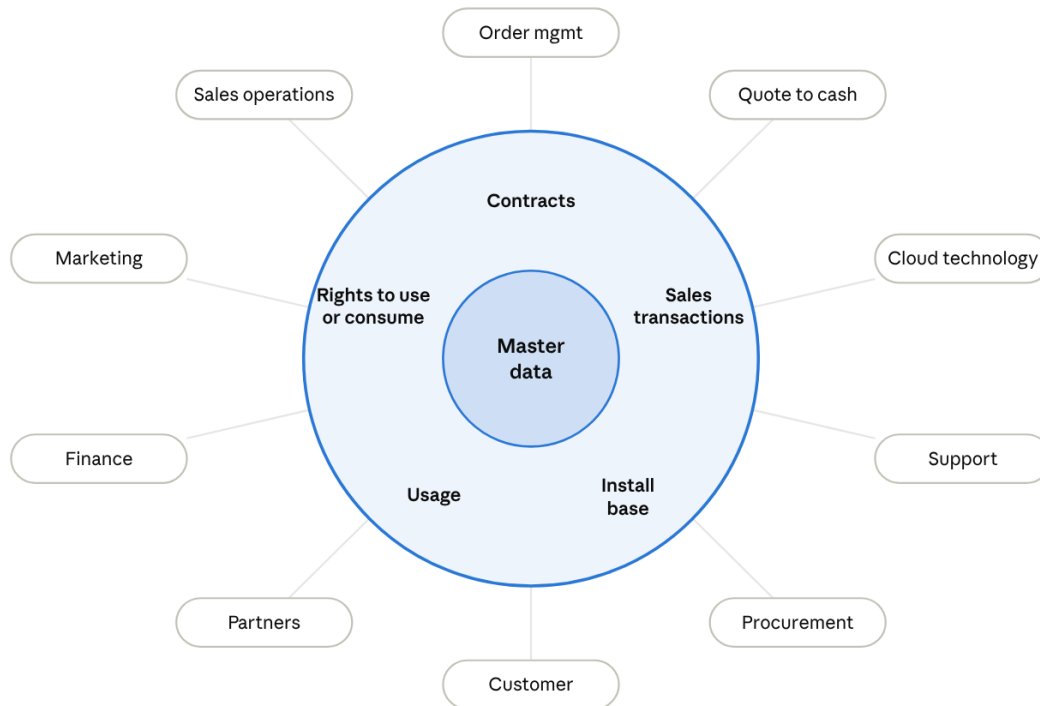


Figure 10: Data-centric view of entitled-product commerce

The foundational BDOs of entitled-product commerce are product, product usage rights, usage against rights, customer, partner, vendor and account.

3.5.2 Business Data Object Overview

A BDO is a containerized business concept (customer, product, sales order, order item) used when designing a logical data model. It may be described by several entities that share a relationship with a root entity; those entities may also relate to entities in other BDOs or outside any BDO.

The BDO is conceptual but has a specific logical structure, designed by analyzing the concept's life cycle: the events and state changes that the business processes impose on it.

Enterprise-standard BDOs give a reference corporate data architecture and accompanying processes. They let the M&A team align an integration quantitatively with business goals for each GTM. They also allow reusable tools for Target process analysis and data migration across many acquisitions, and they isolate integration work from changes in the EBP's physical infrastructure.



The central mechanism for M&A data migration is transformation into Enterprise-standard BDOs. The figure shows how data is extracted, staged, transformed, tested and migrated to the EBP.

Target data is transformed into standard BDOs before it is loaded

Objects may need sequential loading because of inter-object dependencies

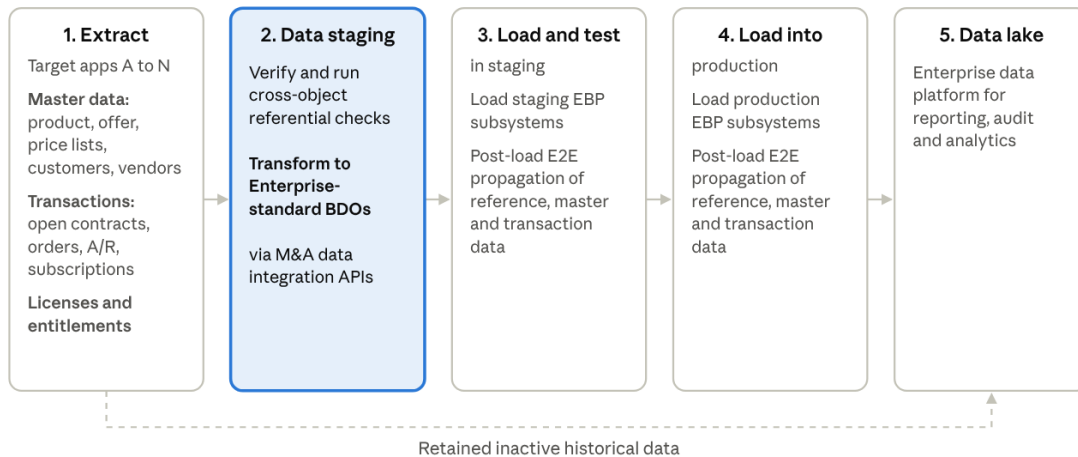


Figure 11: Data migration path from Acquisition Target to EBP

Migration may be **staged**, capability by capability (for example, finance first), or **big bang**, moving the whole Value Chain in one sequence. Either way, analysis, preparation, test and release run in the use-case phases of section 3.3.4.

Target data rarely conforms to the standard BDO models. It is cleansed and extracted, staged, verified for cross-object referential integrity, and then transformed into BDO format. It falls into three classes:

- **Reference data:** the technical master data the Target needed to process master and transactional data through its own (source) infrastructure. Examples: data mappings across source applications; application-to-application lineage transformations; manually administered pick lists in CRM and CPQ tools. These mappings may be reused as is or adapted to the destination.
- **Master data:** data at rest, as defined by the master-data BDOs.
- **Transactional data:** data in motion, as defined by the transactional BDOs, both closed and in flight.

Three classes of Target data each pass through extract, transform, verify

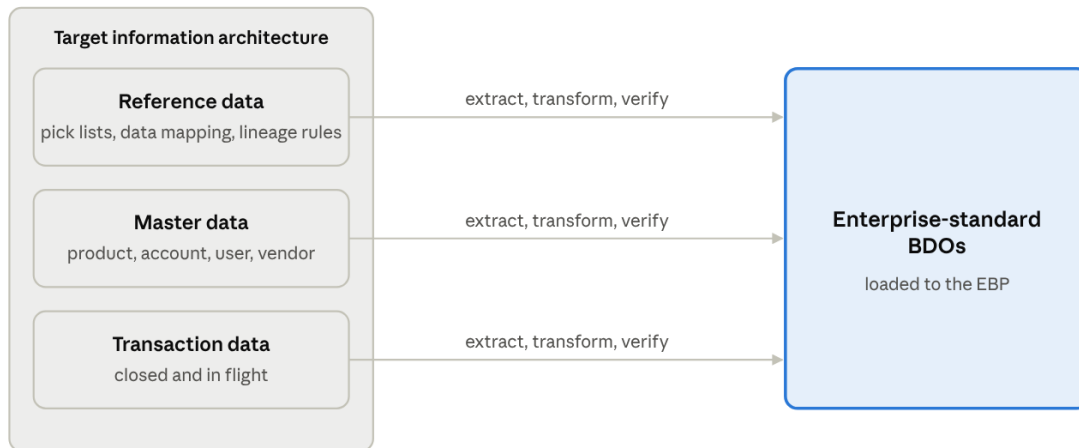


Figure 12: Data migration conceptually

Full automation of every BDO is not a realistic expectation, even in the simplest case. Data volume, mapping complexity and integrity drive the method:

- Where master-data volume is low and mapping complex, manual entry can beat automation.
- Open transactional data (an in-flight subscription or timed Rights) has a finite life. Closed historical data may be of little value to the customer or the Enterprise. It is often cheaper to start new transactions on the EBP and let open Target transactions age out on the legacy infrastructure, retaining inactive history in an archive or data lake.

Comparing the Target's process and data models with the EBP's exposes capabilities the EBP lacks or under-serves. That comparison yields quantitative requirements for changing the EBP to support the combined business.

Data mapping and migration steps (source to destination)

Step	Data in scope	Stage
Source application to EBP application mapping	Reference, master, transaction	Planning
Logical object mapping analysis	Reference, master, transaction	Planning
Physical attribute analysis	Reference, master, transaction	Planning
Integrity analysis	Reference, master, transaction	Planning
Transformation rule design and build	Reference, master, transaction	Development
Load and integration flow simulation testing	Reference, master, transaction	Continuous integration
Data load and post-load data propagation	Reference, master, transaction	Release
User acceptance testing	Reference, master, transaction	Release
Release to production (RTP)	Reference, master, transaction	Concept accepted

3.5.3 Example Business Data Objects

The initial focus is on these BDOs:

Class	BDOs
Master data	Product; pricing/conditions; customer (Consuming Entity); partner; account (sales and customer); vendor; role
Transaction data	Order; order line item; contract; Rights; usage

Product. As defined in section 2.4: tangible, intangible (rights to software, content or capability) or service. The Enterprise retains ownership; the capability is entitled to another party.

Pricing/conditions. Pricing covers prices for customers or vendors and costs for internal accounting. Conditions are the circumstances that apply when a price is calculated: who the customer is, which product, what quantity, on what date. Each factor is master data stored as condition records. The condition technique determines, during order processing, the gross price and the relevant discounts and surcharges.

Customer (Consuming Entity). OEMs, partners and other channel entities may be called customers, but here the customer is the entity that finally uses or consumes the capability defined in the Rights, with the characteristics in section 2.5.3. Customer master data holds what is needed to do business with the customer, such as addresses and payment terms, and controls how transactions post to the sold-to account.

Partner. An organization that sells, resells, embeds or services Enterprise products: distributors, resellers, co-sellers, OEM and embedded partners, IP suppliers and service partners. Partner master data holds the partner type, agreements and programs, price lists and discount tiers, territories, certifications and the customers it may transact for. A partner may hold Rights on its own account and split them to its customers (Appendix B).

Account. The commercial relationship container. A **sales account** groups opportunities, quotes and contacts for selling; a **customer account** groups contracts, billing and payment terms, Rights and users for service. Accounts form hierarchies (global parent, subsidiary, department) that drive pricing, delegation and reporting.

Vendor. A supplier to the Enterprise, including IP suppliers whose licensed technology is embedded in Enterprise products and who may be owed royalties. Vendor master data holds identity, payment terms, tax data and the agreements that govern royalty and pass-through obligations.

Role. A named set of permissions a person or system holds on behalf of an entity: for example system admin, product admin, department admin, end user, partner admin or support delegate. Roles bind users to the Rights they may manage or consume, and carry the visibility limits in section 2.5.3.

Order. A customer's commitment to buy, captured from any channel. It carries the sold-to, ship-to, bill-to and end-user parties, currency, payment terms, channel and references to the quote



and contract. Order state moves from created, through credit and compliance checks, to booked, fulfilled, invoiced and closed.

Order line item. One SKU on an order, with quantity, price conditions, term dates and the Rights template it triggers. The line item is the link between commerce and entitlement: fulfilling it generates the Rights instance, and for add-ons it references the base entitlement it extends.

Contract. The legal agreement that governs one or more orders and Rights: terms and conditions, start and end dates, renewal terms, pricing commitments, and usage or overage terms. Rights instances carry a contract linkage so usage and support can be traced to the governing agreement.

Rights. The entitlement asset, with the characteristics in section 2.5.1 and the structure of the ISO/IEC 19770-3 aligned model (figure in section 2.5.1). Detailed attribute definitions are in Appendix A.

Usage. Usage mirrors aspects of the Rights in structure: each usage or consumption item references the entitlement ID it is consumed against, the deployed instance, the metric and its value over time. Reconciliation of usage against Rights is the true-up described in section 3.2.

3.6 Application Layer E2E

3.6.1 Overview

The figure maps the Customer Journey to EBP subsystem capabilities and on to the application landscape and physical data architecture. A detailed version should be produced for each segment and business unit.

The journey maps down through capabilities and applications to data

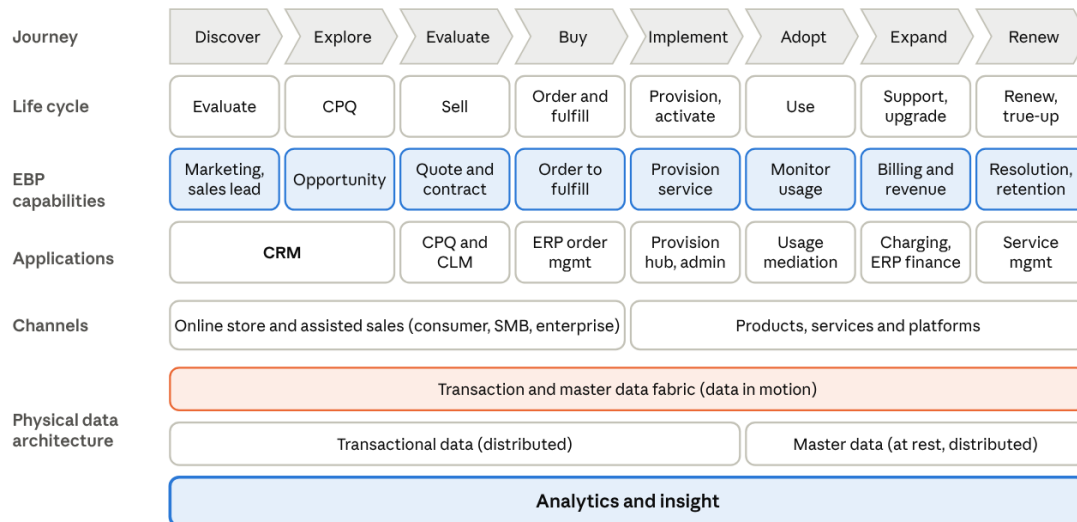


Figure 13: Value Chain/Customer Journey across the EBP subsystem applications

3.6.2 Reference Application Landscape

The figure above shows how application categories support the capabilities along the journey. For each category, record the Enterprise system of record, the Target equivalent and the integration decision (migrate, integrate, retain or retire).

Application category	Role in the life cycle	Key BDOs mastered
Customer relationship management (CRM)	Marketing, leads, opportunities, accounts, contacts, service cases	Account, contact, opportunity
Configure-price-quote (CPQ) and contract lifecycle	Configuration, pricing, quotes, proposals, agreements	Quote, contract
ERP order management and sales & distribution	Orders, credit checks, trade compliance, booking, fulfillment triggers	Order, order line item, SKU, pricing conditions
Convergent charging, mediation and invoicing	Rating of usage, billable items, billing documents, invoices	Provider contract, billable item, consumption item
ERP finance	Contract accounting, receivables, revenue recognition, tax, royalties	Contract account, revenue schedule
Entitlement management service	Rights generation, add-ons, renewals, delegation, revocation	Rights, Rights template
Identity management and admin console	Sign-in, federation, roles, user and seat assignment	User, role
Provisioning hub	Provisioning messages to product platforms, status back to CRM	Provisioning record
eCommerce / online store	Self-service plans, cart, checkout, subscription management	Cart, web order
Procurement (source-to-pay)	Vendor and IP-supplier purchasing and payment	Vendor, purchase order
Data platform (warehouse/lake, BI)	Replication, data models, operational and management reporting	All, as copies

3.7 Technology Layer Overview

The technology layer covers the hardware, system software, hosting platforms (public cloud, private cloud, on-premise) and integration technology under the applications. For each Target workload, decide whether to **rehost** it on Enterprise platforms, **re-platform** it to Enterprise standard services, **retain** it with a defined integration boundary, or **retire** it.



Area	What to assess in the Target	Enterprise target state
Hosting	Cloud providers and accounts, regions, data centers, CDN and storage for product delivery	Consolidated onto Enterprise-approved cloud accounts and regions
Integration	APIs, message queues, file transfers, point-to-point interfaces	Enterprise integration platform, standard BDO-based APIs and events
Identity and access	Identity stores, SSO, license-verification and key services	Enterprise identity service and entitlement management APIs
Data	Databases, warehouses, retention and residency obligations	Enterprise data platform, with residency controls
Delivery	Download, license verification and update flows for desktop and cloud products	Enterprise fulfillment and update infrastructure
Operations	Monitoring, logging, incident management, DevOps toolchain	Enterprise observability and service management standards
Security and compliance	Certifications, penetration tests, open-source and third-party IP inventory	Enterprise security baseline and compliance program

Target platforms keep running behind defined integration boundaries until their use cases have passed through the phases in section 5. Retirement is the last step, once open transactions have aged out.



4. The Integration Analysis Process

4.1 Introduction

Analysis is driven by use cases across the E2E Customer Journey/Value Chain (section 3.3.4). At minimum, analyze: customer acquires Rights; customer activates; customer uses; customer renews; customer acquires add-on (or reduces) Rights; customer requests support. Drill each use case down to lower levels for every GTM the Target uses, and analyze it across the capability map in section 3.3.1.

4.2 Process Gap Analysis

4.2.1 Example: Enterprise Order-to-Invoice Swim Lanes

The swim lanes below follow one enterprise order across applications, from opportunity in CRM to invoice and reporting. Comparing the Target's equivalent flow lane by lane shows where its steps have no Enterprise counterpart.

One enterprise order flows from CRM to invoice and reporting

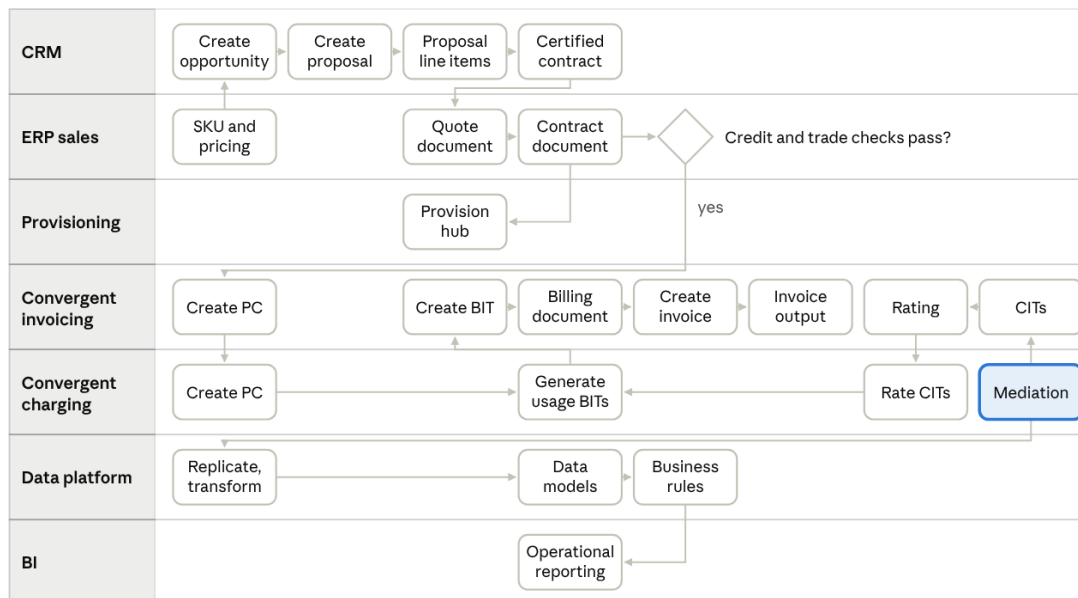


Figure 14: Enterprise order-to-invoice swim lanes across applications

A certified contract in CRM creates a quote document and then a contract document in ERP order management. Once credit and trade-compliance checks clear, a provider contract is created in convergent invoicing and convergent charging. Usage is mediated, rated into consumption items and turned into billable items; billable items roll up into billing documents and invoices. In parallel, source data is replicated to the data platform for operational and management reporting.

Term	Meaning
Provider contract (PC)	The billing-side contract that links a customer's subscription to charging and invoicing
Billable item (BIT)	A priced charge ready for invoicing
Consumption item (CIT)	A metered usage record before rating
Contract account (CA)	The receivables account that invoices post to
Quote / contract document	The ERP sales document types for quote and contract

4.2.2 Capability-Application Mapping

Use the capability map in section 3.3.1 and the application view in section 3.6 to analyze an incoming Target during due diligence. Ask the Target to annotate the capability map with the application, hosting platform and owner for each capability it runs; the result shows at once which capabilities are duplicated, which are missing in the EBP and which are unique to the Target.

Example: acquired desktop software vendor. A typical small software vendor annotates its map with a website and eCommerce store, a third-party payment provider, its own license server, cloud storage and a CDN for downloads, a support desk and a community forum, hosted across two public clouds. Its application flow for product delivery is analyzed as follows.

License verification is the touch point that moves to the Enterprise

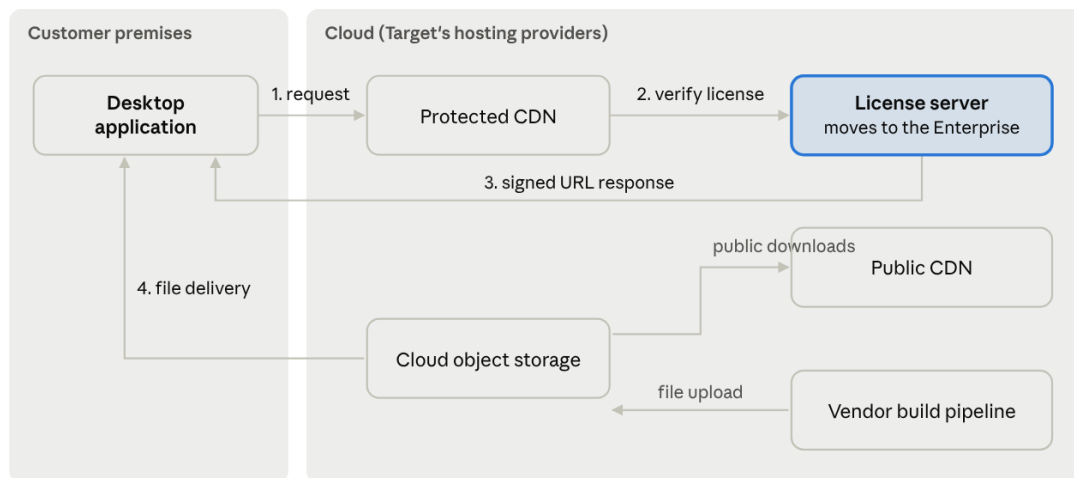


Figure 15: Acquired product delivery and license verification flow

The analysis shows that license verification and download authorization are the touch points that must move to Enterprise entitlement management, while file storage and the CDN can be retained behind that boundary until re-platformed.

4.2.3 Online Store GTM

For Targets that sell through a web store, map: plan and pricing pages; cart and checkout; payment methods and payment provider; tax calculation; trial-to-paid conversion; subscription

self-management (upgrade, downgrade, cancel); dunning and payment retry; and refunds. Compare each step with the Enterprise online store and record whether the Target's customers can move to the Enterprise store at renewal, or need a migration offer.

4.3 Data Analysis

4.3.1 Master Data Analysis

New product introduction (NPI) master data. Integration across the Customer Journey (try, buy, use, renew) depends on product, offer and SKU master data being set up in the Enterprise systems before any transaction can flow.

The Enterprise does not "sell SKUs". A SKU is the pricing of an offer and an identifier in the billing and accounting system. The Enterprise manages **offers** (for example, subscription plans), and SKUs are onboarded as the ingredients of an offer. An offer launched across many countries and channels almost always has many SKUs.

An offer is what is shown or quoted to a customer. There are three kinds: base, promo and trial. A base offer must exist before any promo or trial of it can launch.

A base offer must exist before any trial or promo of it

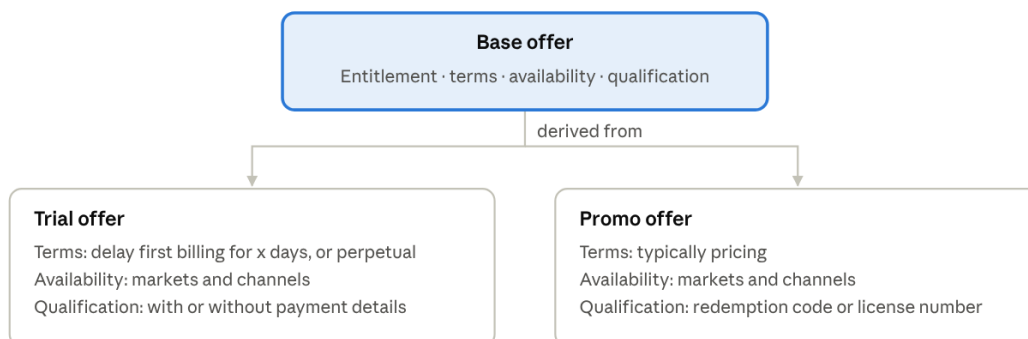


Figure 16: Offer model: base, trial and promo offers

An offer has market availability and terms under which a set of entitlements is fulfilled to qualified customers. Two master-data components generate a **purchasable entitlement**:

- **Product:** the object of entitlement that the end user uses or consumes.
- **Entitlement template:** the base set of terms under which the object of entitlement may be used or consumed.

There may be many entitlement templates per product, because a product can be licensed in many ways and sizes. Collect the Target's existing templates per product early; Appendix A gives the worksheet.



4.4 Application Analysis

For each application category in section 3.6.2, record the Target's system, its owner, contract and renewal date, data volumes, interfaces and the capabilities it serves. Then classify each Target application:

Decision	When it applies
Migrate	An Enterprise system covers the capability; move the Target's data and retire its system
Integrate	The Target system is the acquired technology or holds unique capability; connect it to the EBP through standard BDO interfaces
Retain temporarily	Open transactions or contracts must age out first; freeze changes and set a retirement date
Retire	The capability is no longer needed after integration

4.5 Technology Analysis

4.5.1 Enterprise Direct Order-to-Invoice Landscape

The landscape below is the destination for a Target's direct enterprise business. The numbered steps run from CRM opportunity to invoice; usage enters from product platforms through mediation and charging. Flows that are new for a given integration should be highlighted, and monitoring for them reviewed, because new interfaces are often monitored manually at first.

Direct orders run from CRM through ERP, charging and invoicing to the ledger

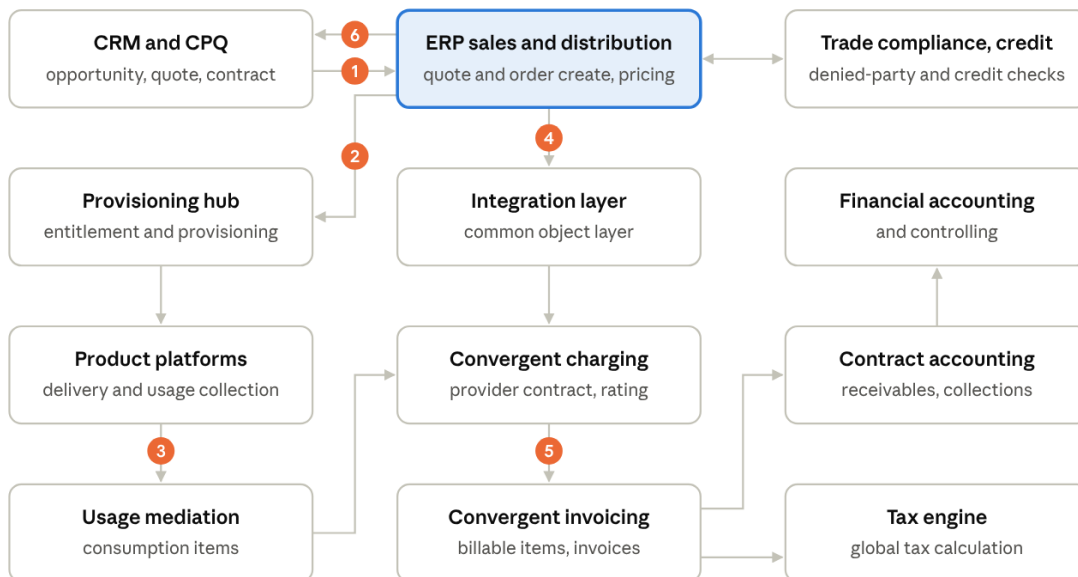


Figure 17: Enterprise direct order-to-invoice landscape

(1) Opportunity to order; (2) provisioning; (3) usage collection and mediation; (4) provider contract creation for charging; (5) billable items to invoice; (6) renewals back to CRM.

4.5.2 Online Store Process Flows

Document the destination online store flow end to end: browse plans, select an offer, create or sign in to an identity, enter payment, calculate tax, place the order, generate the entitlement, provision and activate, then bill each period. Mark where the Target's store differs, such as payment providers, tax engines or trial rules.

4.5.3 Enterprise Indirect (Channel) Order-to-Invoice Landscape

In the indirect landscape, distributors and resellers place orders through a partner portal or EDI. Orders carry the reseller and the end customer; pricing comes from partner price lists and discount programs; Rights are generated for the end customer or for the partner to split onward (Appendix B). Invoicing is to the partner, and point-of-sale reporting feeds partner incentives and revenue recognition.

4.5.4 Enterprise Online Store Order-to-Cash Landscape

The online store landscape runs from web store and cart through payment authorization and capture, order creation in ERP, entitlement generation, recurring billing and collection, to revenue recognition. Payment failures route to dunning and retry, then to suspension of the entitlement if unpaid.



5. The Integration Execution Process

5.1 Customer-Journey Driven Use-Case Integration

5.1.1 Business Operation Use Cases

The static use-case view below shows the main actors and the business operations they take part in. The commerce model of section 2 breaks down into the six use cases of section 3.3.4, plus transitions (upgrade, downgrade, revoke) and solution modification.

Customers, partners and internal teams share one set of operations



Figure 18: High-level business operation use cases

Entitlement management is the hub: orders, fulfillment, revocation, upgrades, downgrades and customer self-service all act on it.

5.2 Phased Integration

Integration runs in the use-case phases shown in section 3.3.4: (1) acquire Rights, (2) activate, (3) use, (4) acquire add-on or reduce, (5) renew, (6) request support. In each phase the Target's capabilities for that use case are migrated or integrated onto the EBP subsystems, then executed, tested and released E2E.

Each phase repeats with increasing transaction complexity, and each release adds a use case to the combined regression suite. Release order need not follow phase numbering: the example

below releases renewal and add-on before metered use, because usage reporting must span renewal and add-on events.

Each release adds a use case; the regression suite grows with it

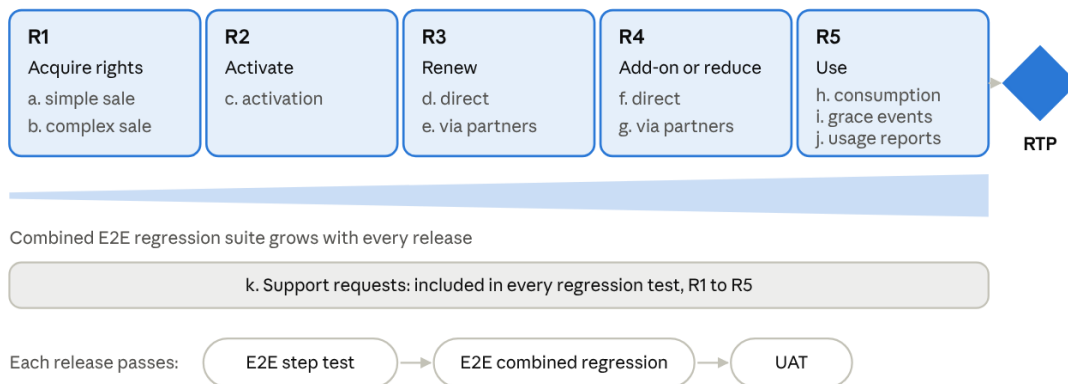


Figure 19: Step-by-step execute, test, release with increasing complexity

Release	Use case added	Scope, simplest first
R1	1. Customer acquires Rights	a. Simplest sales transaction, direct, then via co-seller and embedded (OEM) partner E2E · b. More complex sales (bundles, solutions) across all GTMs
R2	2. Customer activates	c. Activation of a product or service as permitted by the Rights
R3	3. Customer renews	d. Renewal direct · e. Renewal via co-seller and OEM partner
R4	4. Customer acquires add-on or reduces	f. Add-on direct · g. Add-on via co-seller and OEM partner
R5	5. Customer uses	h. Use and consumption · i. Events from use outside compliance and grace rules · j. Usage reported across the Rights life cycle, before and after renewal and add-on
All	6. Customer requests support	k. Support requests, included in each regression test from R1 to R5

Each release passes three test gates: an E2E step test of the new use case; an E2E combined regression test of all use cases released so far; and user acceptance testing (UAT). Release to production (RTP) follows R5.

5.3 Acquired Technology Integration and Capability Migration

Integration has three dimensions: the **E2E dimension** (the Customer Journey and EBP capabilities, left to right), the **acquisition/Enterprise dimension** (Target capabilities below, Enterprise capabilities above) and **time** (the change from Target to Enterprise operation). Most Target capabilities are **migrated** into Enterprise capabilities. The acquired technology itself, the product, is **integrated** through an M&A digital docking station: a reusable layer of Enterprise BDO mappings and APIs at the provisioning, fulfillment and usage touch points. Business process reconciliation happens at the docking station.



Capabilities migrate; the acquired technology integrates

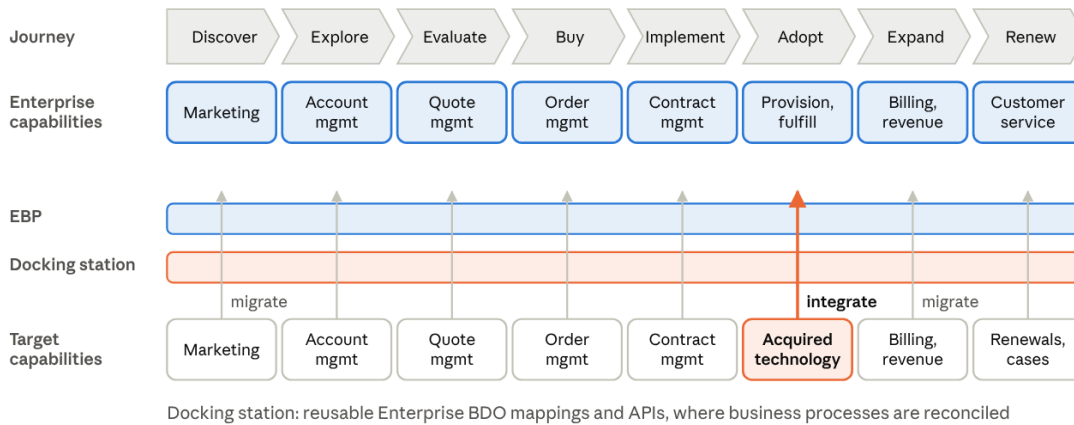


Figure 20: Three-dimensional acquired technology integration and capability migration

5.4 Worked Examples of Use-Case Driven Integration

The examples follow one customer through initial purchase, add-on and renewal of a cloud commerce platform subscription. The plan is priced on the merchant's gross merchandise value (GMV) and provisions three environments (development, staging, production), each with its own Rights list. Other operations to model the same way: solution modification, and the transitions upgrade, downgrade and revoke.

Use case 1: initial purchase. The merchant buys a one-year subscription to the Commerce Platform Pro plan for GMV = \$10M, from 1 June (year 1) to 31 May (year 2). This entitles one instance each of development, staging and production. Every instance carries the same rights:

Right (purchasable entitlement component)	Quantity per instance	Term
Platform instance	1	Jun 1 Y1 – May 31 Y2
CPU-day usage option	25,000 CPU days	Jun 1 Y1 – May 31 Y2
Store views	20	Jun 1 Y1 – May 31 Y2
Storage capacity	1 TB	Jun 1 Y1 – May 31 Y2
Shipping usage option	25,000 shipments	Jun 1 Y1 – May 31 Y2
Analytics Pro instance	1	Jun 1 Y1 – May 31 Y2

Each right carries a Rights ID scoped to its instance (for example `nnn-yyy-1` in development, `nnn-xxx-1` in production, `nnn-zzz-1` in staging), so usage reconciles per environment.

Use case 2: add-on purchase. On 1 November (year 1) the merchant buys 15 additional store views for the production instance. A new add-on entitlement `aaa` (Rights ID `aaa-xxx-5`, quantity 15) is created, referencing the base entitlement, and is **co-terminated** with it on 31 May (year 2). The entitlement container now holds the base purchasable entitlement plus the add-on.



Use case 3: renewal purchase. The merchant renews for one year from 1 June (year 2) to 31 May (year 3). New Rights are issued under renewal entitlement [mmm](#), linked to the previous term. Development and staging keep the original rights; production renews at 35 store views, folding the add-on into the base. The prior-term Rights lists stay in the container as history.



Appendix A. Master Data Configuration

A.1 Rights BDO Attribute Definitions

The business objects managed by this playbook sub-process are the Rights template and the SKU. The sub-process takes them from product conception to product release.

A.1.1 Fulfillable Item (FI) Master Data

A **fulfillable item (FI)** is the individual unit of customer value. It can be modeled three ways:

- **Sold value.** Each FI represents the marketed, sold value, decoupled from how it is delivered. One desktop application sold as a named-user license, a shared-device license, a desktop trial and a regional edition becomes four FIs, even though the user installs the same application. A separate mapping links sellable values to delivered functionality. Offer properties add the target operating systems (Windows, macOS, Linux, mobile, all) and, for perpetual products, the version. All such FIs share the category "deployable rights".
- **Deployed value.** The FI abstracts the application or cloud service and any on-premise builds that contain the functionality; the value is encapsulated in the deployable. These FIs have the category "deployable app".
- **Sold and deployed value.** A combination of the two, much like a boxed product: the offer contains every deliverable explicitly, cloud-service access plus deployable app bits plus the enabled entitled functionality.

Product access has two aspects:

- **Access to product or service functionality.** For an on-premise application, the build the customer packages (optionally), downloads and installs before use, optionally restricted to a version or set of versions. For a cloud service, access to the service (which may include an on-premise component), optionally version-restricted.
- **Rights for use or consumption.** The functionality enabled in the Rights, encoded as predefined scenarios or as individual features that can be sold as add-ons. Functionality is not versioned; a 1:1 relationship with the delivering build is assumed. Quantity may be numeric or abstract, and an unspecified quantity means unlimited. Consumption can be metered per access, per group of applications in the offer (activations per OS user or per machine) or as a pool.

A.1.2 Rights Template Analysis (Rights Master Data)

The Rights template holds the **static** attributes from which Rights instances are generated. Analyze the Target's rights model first in a workbook, even though not every attribute will apply; mark the attributes that must be analyzed.



A license template defines a fulfillable item of a product. A template can be associated with many SKUs, but each SKU has exactly one template. Templates are collections of template lines.

Example: Rights template – fulfillable item worksheet

Template	Qty	Description	Version	License type	Template lines
AVS 4.0	1	Anti-virus scan engine 4.0	4000	Standard	AVS40_PerpetualEnt, AVS40_1YR_Maint
AVS40AV1YR	1	Scan engine 4.0, anti-virus maintenance and content subscription, 1 year	4000	Maintenance	AVS40_1YR_Renewal
AVS 4.0 Function Only	1	Anti-virus scan engine 4.0, license only	4000	Standard	AVS40_PerpetualEnt

Each template line defines a software feature and its attributes (name, number of users, expiration). Lines can be shared by many products through templates.

Example: Rights template – "rights" worksheet

Template line	Description	Qty	Unit	Start	End
AVS40_PerpetualEnt	Perpetual right for AVS 4.0	1	User	Fulfillment date	Perpetual
AVS40_1YR_Maint	Support, upgrade and anti-virus content for the number of users purchased	1	User	Fulfillment date	Fulfillment date + 365 days
AVS40_1YR_Renewal	Renewal: support, upgrade and anti-virus content for the number of users purchased	1	User	License start date	Start + 365 days

These data are then mapped onto the Enterprise's own rights, offer and SKU structures.

A.1.3 Rights Record and Product Instance Setup (Customer Onboarding)

Generating a Rights record from a template requires only a unique fulfillment ID, temporal parameters and any required signatures. Example attributes, based on the ISO/IEC 19770-3 software entitlement (SWEID) tag:

Attribute	Definition
Template name	Handle for the Rights template, associated with a SKU in order management so a SKU transaction forms Rights from the template plus the transaction data
Tag_id	Unique identifier indexing the business transaction that generated the tag: an ID unique in the creator's domain plus the creator's registration ID (regid)
Parent_id	Link to the parent tag(s), back to the original tag; same structure as Tag_id
Root_id	Tag_id of the original tag published by the licensor or service provider
Tag_type	Type of Rights and its relation to related tags: Add, Revoke, Archive, Replacement or Upgrade; with entitlement type Maintenance, Renewal, True-up or License only
Tag_info	Mandatory creation date and optional modification date
Tag_creator	Identity of the tag creator, as in the software ID (SWID) tag



Attribute	Definition
Elements_owner	Elements owned by the creator, which should not be modified without the creator's consent
Right	The heart of the Rights; may repeat (primary, secondary, additional, conditional). Sub-elements below
Contract_linkage	The agreement governing the Rights; may index a business transaction or contract
Rights_timeframe	Temporal information for the whole Rights: start, end or perpetual
Entitled_entity	The end user of the Rights: name, regid, alias and other information
Previously_entitled	Prior owner, for transfers of ownership through divestiture, merger or acquisition

Sub-elements of **Right**:

Sub-element	Definition
Grant	Name of the right, such as install
Quantification (mandatory)	The meter allowed: seats, terabytes stored, cores on a platform
Time_limit	Start and/or end date, GMT, to day resolution
Geo_limit	Countries included or excluded
Language_limit	Languages included or excluded
Platform_limit	Platforms included
Environment_limit	Approved or excluded environments: academic, test, disaster recovery
Vm_limit	Whether virtual environments are approved, and which
Market_limit	Approved or excluded market types
Scope_inclusion	Any positive term that fits no other limit
Scope_exclusion	Any negative term that fits no other limit
Condition	Circumstance under which the right is granted
Additional_attribute	Any neutral term that fits no other limit
Additional_right	A secondary right, which may cascade

A.1.4 SKU

The price item used by ERP to transact a sales order line item. Example attributes:

Attribute	Definition
SKU	Unique identifier
Description	Short display text
creationDate	Date the SKU was created
startDate	Date the SKU becomes available, where a collection filter uses it
endDate	Date the SKU is no longer available, where a collection filter uses it
listPrice	Default price before discounts or promotions
onSale	Whether the item is on sale
salePrice	Price when onSale is true
dynamicAttributes	Additional attributes



A.2 Example Rights/SKU NPI Process

A new rights product moves from rights model to general availability

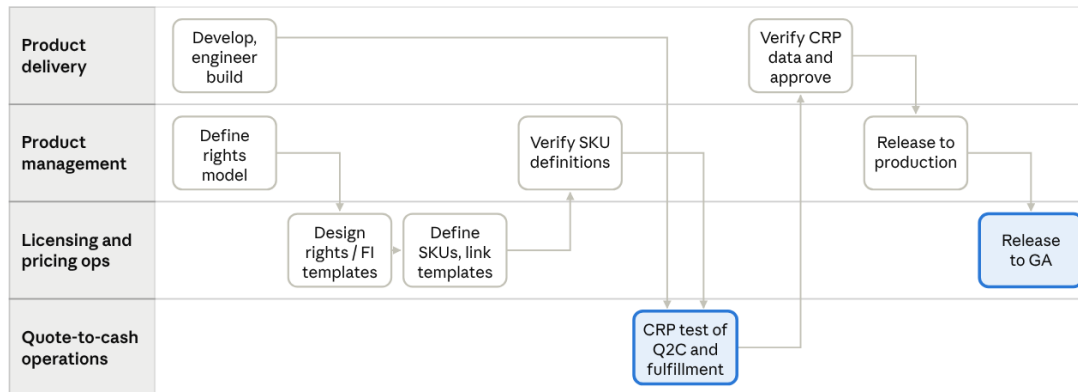


Figure 21: Example rights/SKU NPI process swim lanes

1. **Narrative.** Each swim lane is the entity responsible for the business object's state during that phase of the sub-process.
2. **Define rights model.** The Enterprise maintains a limited set of standard rights models. A model outside that set must pass the governance process for new models before use.
3. **Design rights offerings.** A Rights record comprises features that enable aspects of the product's functionality. Features, and each applicable attribute (Appendix A.1.3), are defined in the offering design.
4. **Design rights templates.** On the current platform, follow the Enterprise's product-onboarding procedure to create the product context and fulfillable items for the new offering. In the target design, templates follow ISO/IEC 19770-3 and each template has a unique ID, its rights "part number".
5. **SKU definition.** A rights part number may have many price points for regional, market (education, SMB, enterprise) and buying-program tiers. Every SKU and price point is defined in ERP.
6. **Associate rights templates and FIs with SKUs.** Each part number is associated with its SKUs, so an order management transaction on any SKU generates the right Rights.
7. **Verify SKU definitions.** Product management, which owns product revenue, signs off that all price points per part number meet the offering criteria.
8. **CRP test of rights and renewals fulfillment.** Conference room pilot testing, usually in staging, covers life cycle steps 4 to 7. Quote-to-fulfillment operations confirms the E2E quote-to-cash process works for a sample of SKUs covering every part number, including upgrades.
9. **Verify CRP test data and approve operations testing.** Product delivery confirms all fulfillment materials (keys, ISO tags) work in the product.

10. **Release rights product to production.** Product management releases to production after signing off all engineering CRP verification.
11. **Release to general availability.** The product may be transacted in ERP once released to production.



Appendix B. Product Go-to-Market Scenarios

B.1 Segmentation

The Enterprise serves three market segments: consumer/individual, small business/mid-market and enterprise. Buying and consumption differ across them and need matching back-office capabilities.

B.2 Customer Experience

Individual. Typically a consumer product licensed per seat and offered as discrete products in families, compared on a feature grid (organization, photo, video benefits and so on). The systems behind these products often descend from the original desktop-software processes; delivery has moved from on-premise to cloud and from perpetual to subscription or timed Rights, but much of the underlying data structure stays the same.

Mid-market. Team products sold to a small business and used by many of its people follow a pattern close to consumer products, mostly licensed per user. Access needs an administrator and permissioned access for delegated users. Capabilities usually do not vary by user, hence the "team" approach, where every delegate gets the same entitlements.

Enterprise. Here the user concept changes. Some offerings (advertising or analytics platforms, for example) are monetized independently of users, serving the enterprise as a whole. They often offer many configurable features, each entitled separately, switched on or off, or with its own rights (time, unit of measure, meter value). The enterprise administrator may delegate capabilities to departments and account for usage by department, with each department admin seeing only its own position.

Departments each see only their own delegated position

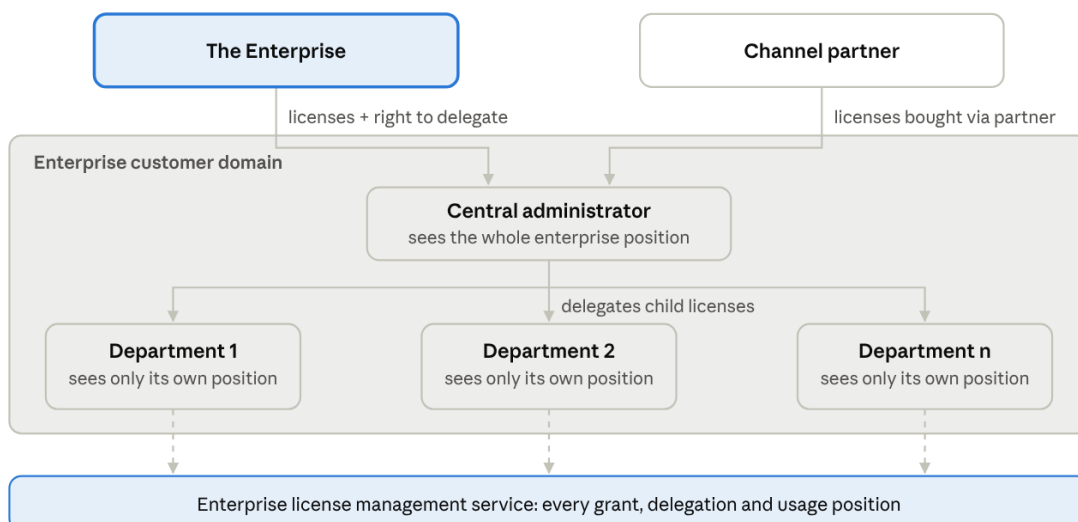


Figure 22: Enterprise departmental rights delegation

Enterprise IT professionals and end users need:

- A systematic view of licensing across the enterprise and at each delegated level
- An understanding of usage and consumption, to judge whether entitlements need remediation
- Automated software asset management (SAM) and entitlement compliance
- The ability to avoid under- or over-procurement, and the cost reduction that follows
- Standard usage measurement across platforms, making heterogeneous environments manageable

Enterprise processes and infrastructure keep evolving, and the M&A team helps drive that evolution through integration requirements. The experience an acquired organization gave its market must be understood, so the Enterprise can at least sustain it and preferably improve on it.

B.3 Route to Market

Products in any segment may be sold and delivered directly or through a channel partner acting for the Enterprise. They may also be sold by a partner that integrates Enterprise products into its own offering and assumes all licensing and entitlement obligations to its customers.

A partner splits licenses across customers who cannot see each other

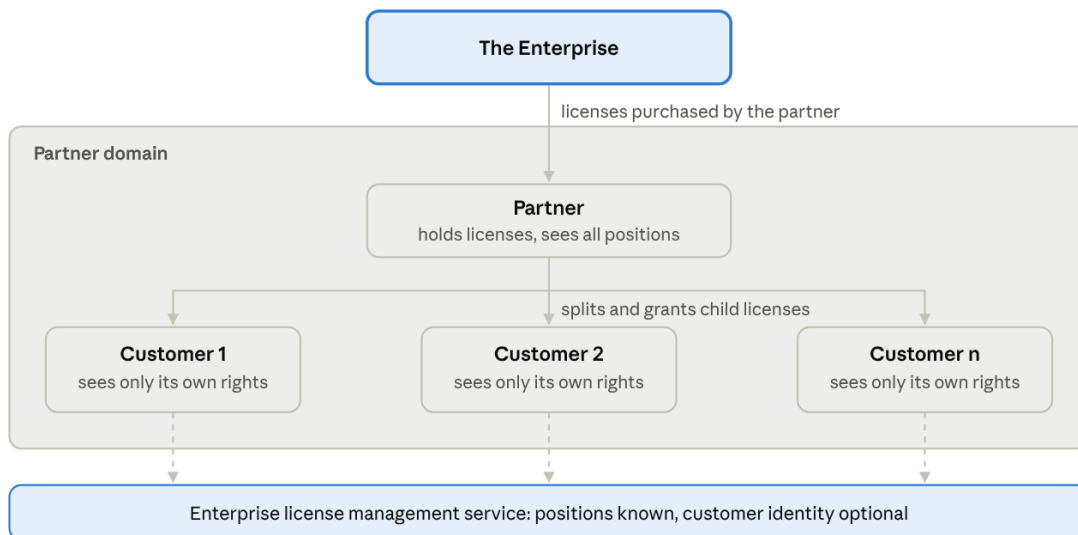


Figure 23: Rights splitting by a partner

The partner's customers are isolated from each other and see only their own Rights positions. The partner sees all positions. The Enterprise can understand each customer's position against usage without necessarily knowing who the customer is. Confirm early whether the EBP supports this pattern; many platforms do not.



B.4 Business Operation Application Sequences

The E2E cross-silo transaction flows must be understood. Where a Target deviates from the standard Enterprise sequences, integration must add new ones.

B.4.1 Initial Purchase

Activation and every usage cycle are recorded against the EUID

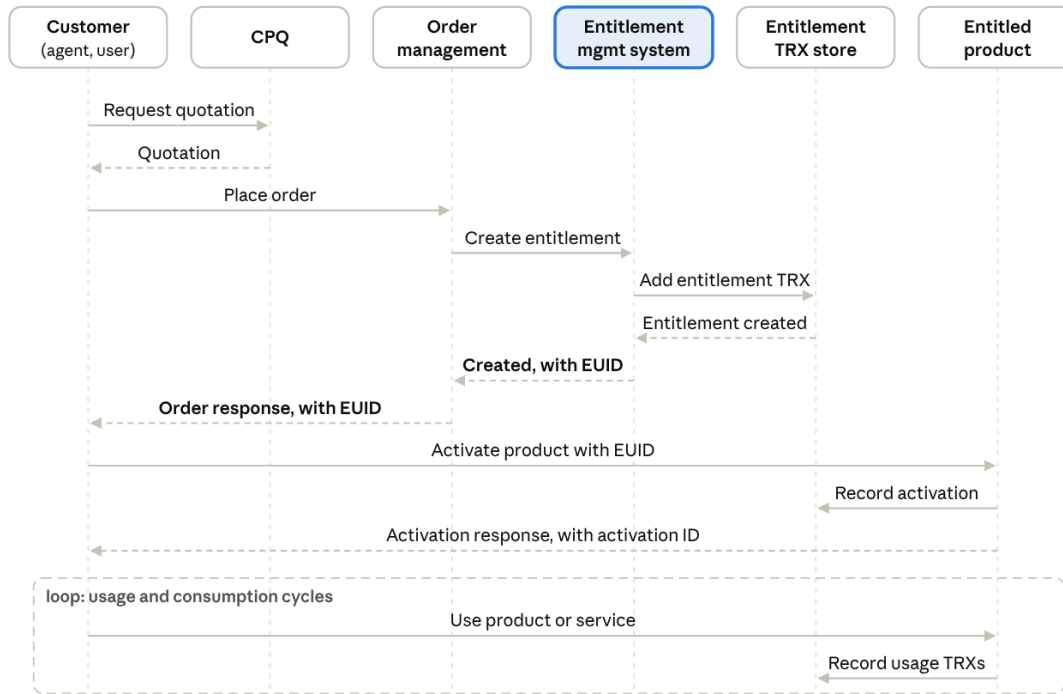


Figure 24: Quote to purchase to usage sequence: initial purchase

The purchasing agent requests and receives a quote from CPQ, then places an order with order management. Order management asks the entitlement management system to create the entitlement, which is recorded in the entitlement transaction store with a unique entitlement ID (EUID). The order response returns the EUID. The end user activates the product with the EUID; activation and every later usage cycle are recorded against it.

B.4.2 Add-On Purchase

Add-ons are checked against the base entitlement before creation

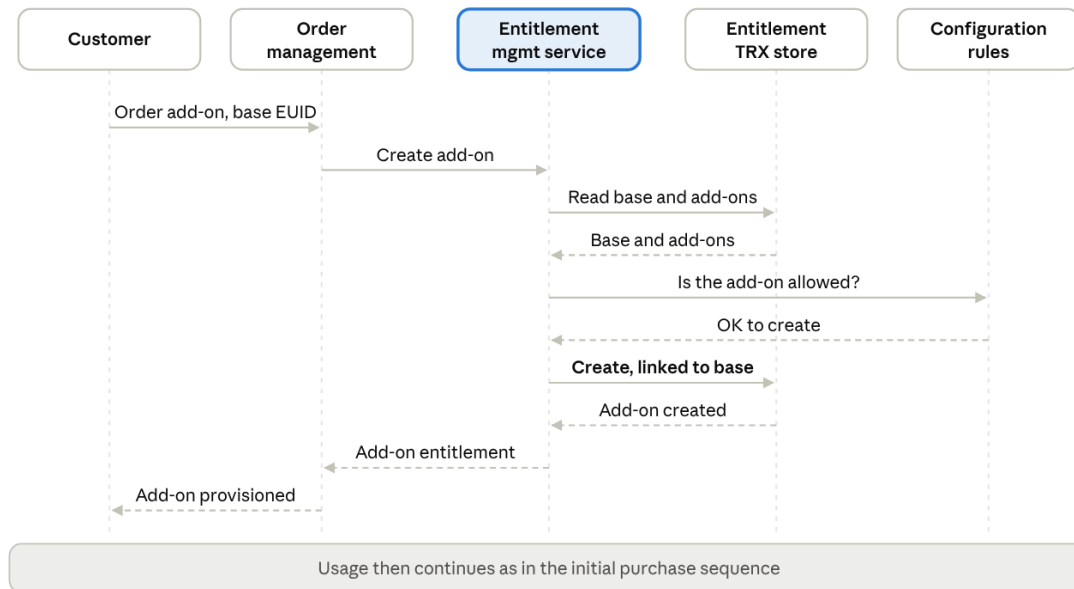


Figure 25: Add-on purchase sequence

The customer orders an add-on, specifying the base entitlement's EUID. Entitlement management reads the base entitlement and any existing add-ons, then checks the add-on against configuration business rules. If valid, it creates the add-on entitlement linked to the base EUID and returns it to the customer. Usage then continues as in the initial purchase sequence.

Appendix C. App Access File: Fulfillable Item JSON

An example product arrangement for a desktop application, expressed as fulfillable items with charging models. Named-user (NU) licenses and trials are delegated to named users; shared-device licenses (SDL) are accessed per organization.

```
{
  "product_arrangement": {
    "code": "sample_simple_app",
    "label": "Sample Simple Any App",
    "fulfillable_items": [
      {
        "id": "ABC121",
        "code": "moonshine_aaa",
        "label": "Moonshine AAA",
        "type": "APPLICATION",
        "fulfillment_configurable": false,
        "delegation_configurable": false,
        "delegation_type": "[NAMED_USER|ORGANIZATION]",
        "overdelegation_allowed": false,
        "charging_model": { "model": "RECURRING", "cap": "UNLIMITED", "overage": "NA" }
      },
      {
        "id": "ABC122",
        "code": "moonshine_features_access",
        "label": "Moonshine Features",
        "type": "FEATURE",
        "fulfillment_configurable": "[false|true]",
        "delegation_configurable": "[false|true]",
        "delegation_type": "[NAMED_USER|ORGANIZATION]",
        "overdelegation_allowed": false,
        "charging_model": {
          "model": "RECURRING",
          "cap": "UNLIMITED",
          "unit": "[NAMED_USER_FEATURES|SHARED_DEVICE_LICENSING|NAMED_USER_TRIAL]",
          "overage": "NA"
        }
      },
      {
        "id": "ABC125",
        "code": "apps_offline_period",
        "label": "Offline Period",
        "type": "QUOTA",
        "fulfillment_configurable": "[false|true]",
        "delegation_configurable": "[false|true]",
        "delegation_type": "NAMED_USER",
        "overdelegation_allowed": false,
        "charging_model": { "model": "RECURRING", "cap": 100, "unit": "days", "overage": "NA" }
      }
    ],
    {
      "id": "ABC126",
      "code": "apps_offline_profiles_count",
      "label": "Offline Activated Machines",
      "type": "QUOTA",
      "fulfillment_configurable": "[false|true]",
      "delegation_configurable": false,
      "delegation_type": "NAMED_USER",
      "overdelegation_allowed": false,
      "charging_model": { "model": "RECURRING", "cap": 2, "unit": "machines_per_user",
"overage": "NA" }
    },
    {
      "id": "ABC127",
      "code": "apps_online_sessions",
```



```

    "label": "Online Machines",
    "type": "QUOTA",
    "fulfillment_configurable": "[false|true]",
    "delegation_configurable": false,
    "delegation_type": "NAMED_USER",
    "overdelegation_allowed": false,
    "charging_model": { "model": "RECURRING", "cap": 2, "unit": "machines_per_user",
"overage": "NA" }
  }
}
}

```

Field	Meaning
code	The effective application, feature list or quota name for the deployable bits
type	APPLICATION (deployable binaries), FEATURE (encoded feature list) or QUOTA (a counted limit)
fulfillment_configurable	Whether the value can be set at fulfillment time
delegation_configurable	Whether the value can be set at delegation or deployment time
delegation_type	Delegation target: named users (NU, trials) or the organization (SDL)
overdelegation_allowed	Whether more can be delegated than was purchased
charging_model.cap	Quantity included in one seat unit; for the offline period, the days an offline profile works before revalidation (0 disables offline use)
charging_model.unit	Unit of consumption, e.g. days or machines per user; calendar-relative units such as anniversary or next billing date can be added