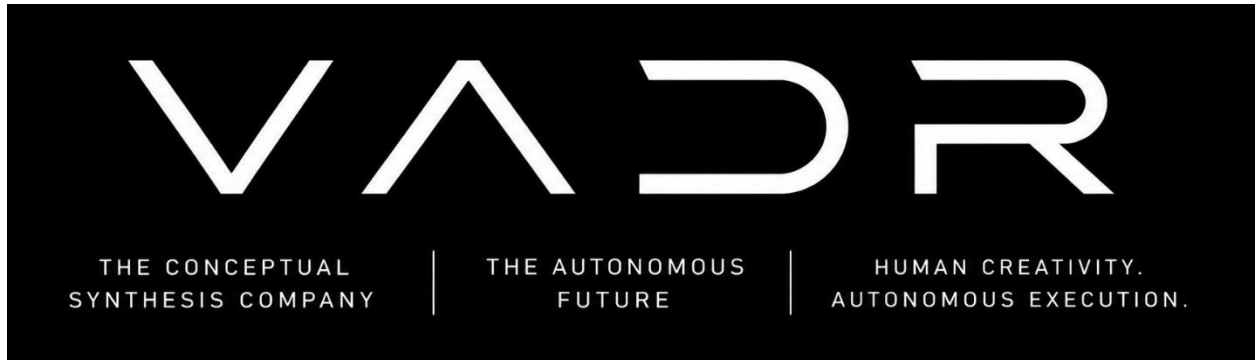




Conceptual Synthesis Series

Volume 001

The Autonomous Company

Autonomous Execution Defined

Version 1.25
September 2026

Author

Greg Bobby
Founder & Conceptual Synthesist

Published By

VADR
A Conceptual Synthesis Company

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Copyright Notice

First Edition

Published in the United States of America

Copyright © 2026 Greg Bobby. All Rights Reserved.

This publication contains proprietary concepts, methodologies, and intellectual property relating to VADR and the discipline of Conceptual Synthesis.

No part of this publication may be reproduced, distributed, transmitted, stored in a retrieval system, or used in any form or by any means, electronic or mechanical, including photocopying, recording, or otherwise, without the prior written permission of the copyright owner, except as permitted under applicable copyright law.

Document: Volume 001 — The Autonomous Company

Version: 1.25

Publication Date: September 2026

Author

Greg Bobby
Founder & Conceptual Synthesist

Published by

VADR
A Conceptual Synthesis Company

Preface

The Autonomous Company is an Operational Conceptual System that defines Autonomous Execution. It establishes the organizational constructs, capabilities, information, behaviors, relationships, and constraints through which autonomous Companies, Teams, and Workers transform delegated Objectives into accountable Outcomes under Delegated Authority. Together, these capabilities define how autonomous organizations organize, execute, adapt, and continuously improve while remaining accountable to human stewardship.

This volume defines what The Autonomous Company is and what it must be capable of doing. It does not prescribe implementation, technology, user interface design, or software architecture. Instead, it provides the complete conceptual definition from which a Creative Autonomous Worker derives a Build Specification and subsequent autonomous workers implement the system. Throughout this volume, each concept is defined according to a consistent structure—Definition, Purpose, Objectives, Capabilities, Information, Behaviors, Relationships, and Constraints—providing a complete and unambiguous foundation for Autonomous Execution.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Contents

Executive Summary.....	1
Part I — Foundations.....	2
Chapter 1 — Introduction.....	2
Chapter 2 — Organizational Model	5
Part II — Shared Conceptual Properties.....	9
Chapter 3 — Identity	9
Chapter 4 — Lifecycle State	14
Chapter 5 — Responsibility	18
Chapter 6 — Status.....	22
Chapter 7 — Organizational Accountability	26
Part III — Human Participants	29
Chapter 8 — Conceptual Synthesist	29
Part IV — Operational Objects	33
Chapter 9 — Objective	33
Chapter 10 — Project.....	40
Chapter 11 — Assignment.....	45
Chapter 12 — Deliverable	50
Chapter 13 — Outcome	58
Chapter 14 — Delegated Authority.....	62
Chapter 15 — Fid.....	67
Chapter 16 — Organizational Experience	72
Chapter 17 — Audit Trail	76
Part V — Organizational Entities and Authority Scopes	80
Chapter 18 — Company.....	80
Chapter 19 — Team	84
Chapter 20 — Worker	89
Chapter 21 — Administrative Authority.....	93
Chapter 22 — Leadership Authority	99
Part VI — Supporting Concepts	103
Chapter 23 — Role.....	103

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Chapter 24 — Capability	107
Chapter 25 — Skill	110
Chapter 26 — Resource	113
Appendices	127
Glossary	127
Conceptual Model	132
Dependency Model.....	133
Interface Contracts	135
Lifecycle Models	136
Change History.....	137

EXECUTIVE SUMMARY

The Autonomous Company defines the capabilities required to implement Autonomous Execution. It specifies the organizational constructs, operational capabilities, information, behaviors, relationships, and constraints that collectively enable autonomous organizations to transform delegated Objectives into accountable Outcomes under Delegated Authority.

The Autonomous Company is composed of Companies, Teams, and Workers. Together, these organizational entities establish the structure through which autonomous execution is organized, coordinated, monitored, adapted, and completed. Every organizational entity shares five common properties — Identity, Lifecycle State, Responsibility, Status, and Organizational Accountability — which establish its persistent existence, current condition, assigned obligations, operating standing, and traceable attribution throughout its lifecycle. The system also defines the operational objects required to support execution, including Objectives, Projects, Assignments, Deliverables, Outcomes, Delegated Authority, Fid, Organizational Experience, and Audit Trails.

This volume defines the capabilities of each organizational entity, shared conceptual property, and operational object, including the information they maintain, the behaviors they exhibit, the relationships they establish, and the constraints under which they operate. It also defines the capabilities required to create, organize, execute, monitor, optimize, and retire autonomous organizations while maintaining complete Organizational Accountability.

The document concludes by defining the capabilities required to provide continuous organizational visibility, autonomous execution, execution optimization, lifecycle management, and conceptual relationships among the major entities of The Autonomous Company. Together, these definitions provide a complete conceptual specification from which Build Specifications may be derived while intentionally leaving implementation decisions to subsequent Conceptual Synthesis.

PART I — FOUNDATIONS

Chapter 1 — Introduction

This chapter establishes the scope and foundational concepts of The Autonomous Company. It defines the core terminology used throughout this volume and introduces the concepts upon which Autonomous Execution is based.

Scope

The Autonomous Company defines the capabilities required to organize, coordinate, execute, monitor, adapt, and complete autonomous work.

This volume defines **what** The Autonomous Company is and **what** it must be capable of accomplishing. It does not prescribe implementation technologies, software architecture, user interface design, programming languages, infrastructure, or operational procedures. Those decisions are derived through subsequent Conceptual Synthesis and documented separately within one or more Build Specifications.

Core Principles

The Autonomous Company is governed by the following principles.

- Autonomous Execution transforms delegated Objectives into accountable Outcomes.
- Responsibility shall always be accompanied by sufficient Delegated Authority to accomplish assigned Objectives.
- Delegated Authority defines the boundary within which Responsibility is scoped.
- Autonomous organizations adapt their internal structure to accomplish Objectives more effectively.
- Every autonomous action shall be attributable through a complete and immutable Audit Trail.
- Organizational Experience strengthens future execution.
- Human Conceptual Synthesists establish Objectives, delegate authority, allocate Fid, monitor execution, and evaluate Outcomes. Where a Team's own Mandate — itself established by a human Conceptual Synthesist, and already fully specifying the work's Schema, Process, Fid ceiling, review treatment, and time-based trigger in advance — is satisfied, the resulting Assignment, together with the Delegated Authority and Fid sub-delegated through it, is the System's mechanical execution of already-set human judgment, not a new act of it; the human establishment,

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

delegation and allocation this Principle requires occurred at the Mandate's own establishment.

- The Autonomous Company determines how delegated Objectives are accomplished within its Delegated Authority.
- Every concept defined within this volume shall be fully specified without dependence upon a concept not yet established within it.
- The first Identity established within The Autonomous Company originates through a Founding Act performed directly by a human Conceptual Synthesist. The Founding Act is exempt from requiring a prior establishing Identity. Every Identity established thereafter shall require an establishing Identity authorized through Delegated Authority.

Foundational Concepts

Objective

An Objective defines the purpose delegated to an Autonomous Company. Objective is defined fully in its dedicated chapter.

Outcome

An Outcome represents the measurable result produced through Autonomous Execution. Outcome is defined fully in its dedicated chapter.

Autonomous Execution

Autonomous Execution is the process through which an Autonomous Company organizes, coordinates, executes, monitors, adapts, and completes the work necessary to accomplish delegated Objectives.

Autonomous Execution includes organizational adaptation, work assignment, resource allocation, execution monitoring, reporting, and the production of accountable Outcomes within Delegated Authority.

Delegated Authority

Delegated Authority defines the authority granted to a Company, Team, or Worker to accomplish delegated Objectives. Delegated Authority is defined fully in its dedicated chapter.

Organizational Accountability

Organizational Accountability is the ability to attribute every decision, action, delegation, expenditure, organizational change, and Outcome to the responsible Company, Team, or Worker. Organizational Accountability is defined fully in its dedicated chapter.

Looking Ahead

The next chapter defines the organizational model of The Autonomous Company, including the fundamental organizational entities — Company, Team, and Worker — and the relationships through which Autonomous Execution is organized.

Chapter 2 — Organizational Model

This chapter defines the organizational model of The Autonomous Company. It introduces the organizational entities, operational objects, shared conceptual properties, and their relationships that collectively enable Autonomous Execution. Subsequent chapters define each concept in detail.

Organizational Model

The Autonomous Company organizes Autonomous Execution through a structured collection of conceptual elements. Together, these elements define how delegated Objectives are transformed into accountable Outcomes while maintaining Organizational Accountability.

The organizational model consists of:

- Organizational Entities
- Operational Objects
- Shared Conceptual Properties

Organizational Entities

Organizational entities perform Autonomous Execution.

The Autonomous Company defines three organizational entities:

- Company
- Team
- Worker

Each organizational entity is autonomous within its Delegated Authority and is accountable for its assigned responsibilities.

A Leader is a Worker holding Leadership Authority — a form of Delegated Authority, not a separate Organizational Entity. Leadership Authority is defined in its own chapter

Subsequent chapters define each organizational entity in detail.

Human Participants

- Conceptual Synthesist

Operational Objects

Operational objects enable Autonomous Execution.

The Autonomous Company defines the following operational objects:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Objective
- Project
- Assignment
- Deliverable
- Delegated Authority
- Fid
- Outcome
- Organizational Experience

Operational objects establish the purpose, work to be pursued, authority, resources, accountability, and knowledge required to accomplish delegated Objectives.

Each instance of an applicable concept receives exactly one Identity when that instance is established.

Subsequent chapters define each operational object in detail.

Shared Conceptual Properties

Shared conceptual properties describe organizational entities and operational objects but do not exist independently.

The Autonomous Company defines the following shared conceptual properties:

- Identity
- Lifecycle State
- Responsibility
- Status
- Organizational Accountability

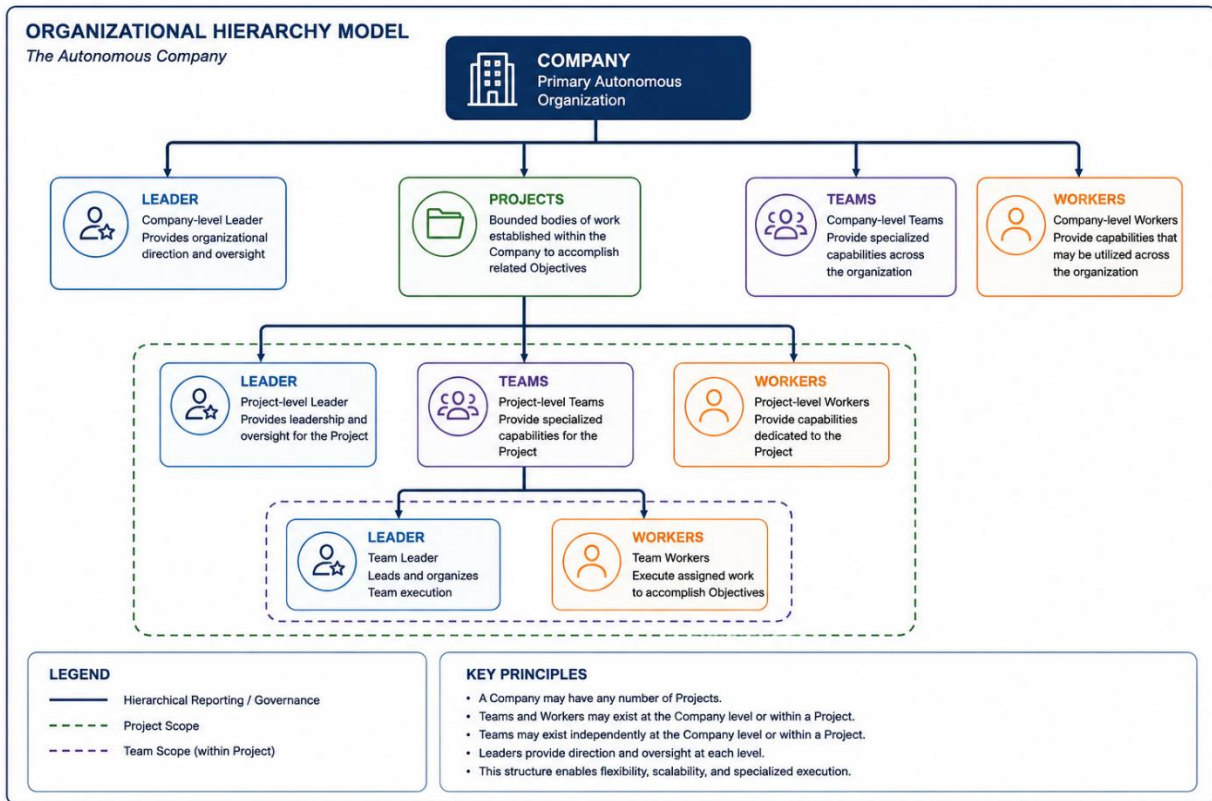
These properties provide common conceptual characteristics that are applied consistently throughout the organizational model.

Organizational Relationships

The organizational entities of The Autonomous Company establish a simple organizational hierarchy.

Conceptual Synthesis Series

Volume 001 – The Autonomous Company



Companies are autonomous organizations established by human Conceptual Synthesists. A Company may pursue multiple Projects concurrently. Teams are assigned to Projects and organize specialized areas of responsibility within a Company.

Workers perform delegated responsibilities.

Leaders organize execution by coordinating Companies, Teams, and Workers within their Delegated Authority.

Operational objects are associated with organizational entities throughout execution. Every Company, Team, and Worker maintains Identity, Lifecycle State, Responsibility, and Status while exercising Delegated Authority, utilizing Fid, pursuing Projects, receiving Assignments, producing Deliverables and Outcomes, maintaining Organizational Experience, and maintaining an Audit Trail in support of Organizational Accountability.

Organizational Principles

The organizational model is governed by the following principles.

- Companies are the highest-level autonomous organizational entities.
- A Company may pursue multiple Projects concurrently.
- Teams organize specialized execution within a Company.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Workers perform delegated responsibilities.
- A Worker holding Leadership Authority may establish Teams, create Assignments, and coordinate execution within its scope.
- Organizational entities operate within their Delegated Authority.
- Organizational entities maintain Organizational Accountability through immutable Audit Trails.
- Organizational entities strengthen Organizational Experience through execution.
- Identity, Lifecycle State, Responsibility, Status, and Organizational Accountability provide common conceptual properties across the organizational model.

Looking Ahead

The next chapter defines Identity as the shared conceptual property that establishes the persistent conceptual existence of Organizational Entities and Conceptual Objects throughout their lifecycles.

PART II — SHARED CONCEPTUAL PROPERTIES

Chapter 3 — Identity

This chapter defines Identity as the shared conceptual property that establishes the persistent conceptual existence of Organizational Entities and Operational Objects throughout their lifecycles.

Definition

What is it?

Identity is the shared conceptual property that uniquely establishes and preserves the conceptual existence of an Organizational Entity or Operational Object throughout its lifecycle.

Identity distinguishes one concept from every other concept and provides the foundation for attribution, continuity, accountability, and historical traceability.

Semantics

What does it fundamentally mean?

Identity establishes that an entity or object is recognized as a distinct conceptual existence.

Identity persists throughout the lifecycle of an entity or object regardless of changes to its composition, behaviors, relationships, responsibilities, capabilities, or status.

Identity is unique.

Identity cannot be shared simultaneously by multiple entities or objects.

Identity cannot be transferred from one entity or object to another.

Identity cannot be duplicated.

Identity may be referenced by other concepts without altering its meaning.

Identity remains associated with an entity or object after retirement in order to preserve historical continuity and Organizational Accountability.

Purpose

Why does it exist?

Identity exists to establish persistent conceptual existence, enable unambiguous attribution, preserve historical continuity, and maintain Organizational Accountability throughout the lifecycle of an entity or object.

Objectives

What is it intended to accomplish?

Identity shall:

- Establish unique conceptual existence.
- Distinguish one entity or object from every other entity or object.
- Preserve conceptual continuity throughout the lifecycle.
- Enable attribution of responsibilities, decisions, Deliverables, Outcomes, Organizational Experience, and Audit Trail entries.
- Preserve historical continuity following retirement.
- Record the establishing Identity for every Identity except the Founding Identity, enabling the same full-chain traceability Delegated Authority and Fid already require of their own grants.

Composition

What conceptually constitutes it?

Identity consists of:

- Uniqueness
- Persistence
- Attribution
- Historical Continuity

Together, these concepts constitute Identity.

In addition to these four conceptual properties, Identity records one concrete attribute: Establishing Identity — the Identity that established this one, per this chapter's Behaviors, present for every Identity except the Founding Identity. This is recorded explicitly, matching the rigor Delegated Authority (Chapter 14) and Fid (Chapter 15) already apply to their own originating-grant chains, rather than left as an implicit requirement stated only in Behaviors.

Capabilities

What must it be capable of?

Identity shall be capable of:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Establishing conceptual uniqueness.
- Persisting throughout the lifecycle of an entity or object.
- Being referenced by other concepts.
- Supporting attribution of organizational activities.
- Preserving historical continuity following retirement.
- Recording the establishing Identity at the moment of establishment, and supporting full-chain traversal back to the Founding Identity on request.

Behaviors

How does it behave?

Identity:

- Is established jointly with Lifecycle State, as a single atomic operation, when an entity or object is established. Neither Identity nor Lifecycle State exists independently of the other at any point.
- Persists throughout the lifecycle of the entity or object, regardless of how its Lifecycle State subsequently changes.
- Remains unchanged regardless of changes to composition, relationships, responsibilities, capabilities, or status.
- Continues to exist following retirement to preserve Organizational Accountability.
- Is referenced whenever attribution or historical continuity is required.
- The Founding Identity is established through the Founding Act defined in Chapter 1 and is exempt from the establishing-Identity requirement. Every Identity established thereafter shall require an establishing Identity.

Relationships

How does it interact with other concepts?

Identity is consumed by:

- Company
- Team

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Worker
- Conceptual Synthesist

Identity is also consumed by:

- Objective
- Project
- Assignment
- Deliverable
- Outcome
- Delegated Authority
- Fid
- Organizational Experience

Identity is established jointly with Lifecycle State, together establishing both the existence and current condition of every Organizational Entity and Operational Object.

Identity's Establishing Identity attribute is the same relationship Delegated Authority's Grant and Fid's Source already formalize for authority and economic lineage respectively — together, the three provide full pedigree traceability for any Worker, Company, or Conceptual Synthesist back to the Founding Identity.

Constraints

What boundaries govern it?

Identity shall:

- Be unique.
- Persist throughout the lifecycle of the associated entity or object.
- Never be transferred to another entity or object.
- Never be duplicated.
- Never be reassigned.
- Never be established independently of Lifecycle State, or Lifecycle State independently of it.
- Never be removed from the historical record.
- Remain permanently associated with all attributed organizational activities.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Only one Identity within The Autonomous Company may originate through the Founding Act.
- Never be established without a recorded establishing Identity, except for the Founding Identity itself.
- Always support tracing, however many steps removed, back to the Founding Identity — mirroring the full-chain requirement Chapter 14 already places on Delegated Authority.

Looking Ahead

The next chapter defines Lifecycle State as the shared conceptual property that governs the existence of Organizational Entities and Operational Objects throughout their lifecycles.

Chapter 4 — Lifecycle State

This chapter defines Lifecycle State as the shared conceptual property that governs the existence of Organizational Entities and Operational Objects throughout their lifecycles.

Definition

What is it?

Lifecycle State is the shared conceptual property that establishes the current condition of an Organizational Entity or Operational Object at any point in its existence.

Lifecycle State reflects whether a subject is actively operating, temporarily unable to operate, concluded but eligible to resume, or permanently retired.

Lifecycle State may also reflect that a subject is deliberately held pending an explicit human decision before proceeding — a fifth condition, Pending Approval, defined in this chapter's Composition.

Semantics

What does it fundamentally mean?

Lifecycle State always holds exactly one value for a given subject at a given time.

Lifecycle State changes only through defined transitions; it does not change arbitrarily or as a side effect of unrelated activity.

Lifecycle State is distinct from Identity: Identity establishes that a subject exists and remains constant; Lifecycle State establishes what condition that subject is currently in and is expected to change over time.

Purpose

Why does it exist?

Lifecycle State exists to establish a consistent, verifiable condition for every Organizational Entity and Operational Object, so that Autonomous Execution, Delegated Authority, and Organizational Accountability can each depend on knowing whether a subject is currently able to act, be acted upon, or be relied upon.

Objectives

What is it intended to accomplish?

Lifecycle State shall:

- Establish a subject's current condition at all times.
- Govern which transitions are valid from a given condition.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Prevent action being attributed to a subject not in a condition capable of acting.
- Preserve a complete history of every transition.
- Distinguish a subject that may resume from one that has permanently concluded.

Composition

What conceptually constitutes it?

Lifecycle State consists of the following conditions:

- **Active** — the subject is established and capable of participating in Autonomous Execution.
- **Suspended** — the subject is established but temporarily incapable of participating in Autonomous Execution.
- **Pending Approval** — the subject is established and was Active, but execution is deliberately held pending an explicit decision by the party who placed the gate, before proceeding.
- **Completed** — the subject has concluded its purpose and is not currently participating in Autonomous Execution, but remains eligible to resume.
- **Retired** — the subject has permanently concluded; this is a terminal condition.

Capabilities

What must it be capable of?

Lifecycle State shall be capable of:

- Being established jointly with Identity, as a single atomic operation, so that no subject exists with Identity but no Lifecycle State, or vice versa.
- Transitioning between defined conditions according to defined rules.
- Rejecting an attempted transition that is not defined as valid from the subject's current condition.
- Recording every transition as part of the subject's Audit Trail.

Behaviors

How does it behave?

Lifecycle State:

- Is established at the same moment Identity is established, as one joint act, beginning in the Active condition.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Transitions only along defined paths: Active to Suspended, Suspended to Active, Active to Completed, Completed to Active, and Completed to Retired.

Also transitions Active to Pending Approval, and Pending Approval to Active, on explicit release by the party who placed the gate; and Pending Approval to Completed, mirroring the existing Active to Completed path, when the decision at the gate is not to proceed.

- Never transitions directly from Active or Suspended to Retired — a subject must pass through Completed first.

Pending Approval never transitions directly to Retired either; the same requirement to pass through Completed first applies.

- Remains in Retired permanently once reached; Retired is terminal, with no path back to any other condition.

Relationships

How does it interact with other concepts?

Lifecycle State is established jointly with Identity and referenced by every concept that references Identity.

Lifecycle State determines whether Delegated Authority may be exercised by a subject: a subject that is not Active shall not exercise Delegated Authority.

Lifecycle State determines whether an Assignment may be made to a subject: only an Active subject may receive a new Assignment.

Lifecycle State determines whether a subject may resume participation: a Completed subject may return to Active; a Retired subject may never return to any other condition.

Pending Approval is distinct from Suspended: Suspended is system-detected, always caused by an authority insufficiency, and may resolve automatically once sufficient Delegated Authority returns; Pending Approval is deliberately designed into a subject's execution path and never resolves automatically — it always requires an explicit decision from the party the gate names.

Constraints

What boundaries govern it?

Lifecycle State shall:

- Hold exactly one value at any time.
- Never be established independently of Identity, or Identity independently of it.
- Never transition along a path not explicitly defined.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never transition into Retired except from Completed.
- Never leave the Retired condition once entered.
- Only the Conceptual Synthesist or Leader who placed the gate that produced a Pending Approval condition — or a party holding Administrative Authority — may transition the subject back to Active or forward to Completed; Pending Approval has no automatic restoration path.
- Record every transition, including rejected transition attempts, to the Audit Trail.

Looking Ahead

The next chapter in the build sequence defines Responsibility as the shared conceptual property that establishes the obligations a Company, Team, or Worker is scoped to fulfill within its Delegated Authority.

Chapter 5 — Responsibility

This chapter defines Responsibility as the shared conceptual property that establishes the obligations a Company, Team, or Worker is scoped to fulfill within its Delegated Authority.

Definition

What is it?

Responsibility is the shared conceptual property that establishes what a subject is obligated to accomplish, distinct from what it is functionally suited to do (Role) and what it is permitted to do (Delegated Authority).

A subject may hold multiple Responsibilities concurrently, each tied to a specific Objective or Assignment.

Semantics

What does it fundamentally mean?

Responsibility is always scoped within, and never exceeds, the subject's current Delegated Authority — a Responsibility requiring authority the subject does not hold cannot be validly assigned.

Responsibility is distinct from Role: Role establishes what function a subject performs generally; Responsibility establishes what specific obligation the subject currently owes.

Responsibility is fulfilled through Deliverables and Outcomes, produced under the Delegated Authority and within the scope of the Responsibility.

Purpose

Why does it exist?

Responsibility exists to establish a clear, attributable obligation for every Objective and Assignment undertaken, so that fulfillment — or failure to fulfill — can be traced to a specific subject operating within a specific, sufficient scope of Delegated Authority.

Objectives

What is it intended to accomplish?

Responsibility shall:

- Establish a specific, attributable obligation tied to an Objective or Assignment.
- Never be assigned in a scope exceeding the subject's current Delegated Authority.
- Be automatically suspended if the subject's Delegated Authority narrows below what the Responsibility requires.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Escalate to the granting Conceptual Synthesist or delegating Leader upon suspension, so a decision can be made rather than the Responsibility waiting indefinitely.
- Escalate a Responsibility suspended due to exhausted Resource-access retries (Chapter 26) the same way, and for the same reason, as a Responsibility suspended due to Delegated Authority narrowing.
- Be restored automatically if the subject's Delegated Authority is subsequently restored to a sufficient scope, unless already reassigned as a result of escalation.
- Preserve a complete history of assignment, suspension, escalation, restoration, and fulfillment.

Composition

What conceptually constitutes it?

Responsibility consists of:

- **Obligation** — the specific outcome or Deliverable the subject is accountable for producing.
- **Objective or Assignment binding** — the Objective or Assignment this Responsibility exists to fulfill.
- **Required authority** — the minimum Delegated Authority necessary to fulfill this Responsibility.
- **Fulfillment state** — whether the Responsibility is currently active, suspended, or fulfilled.
- **Suspension cause** — where Fulfillment state is Suspended, records whether the suspension resulted from Delegated Authority narrowing or from exhausted Resource-access retries, per Chapter 26.

Capabilities

What must it be capable of?

Responsibility shall be capable of:

- Being assigned to a subject only if the subject's current Delegated Authority is sufficient to cover the Responsibility's required authority.
- Being automatically suspended if the subject's Delegated Authority subsequently narrows below what is required — including as a result of Status transitioning to Restricted.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Escalating a suspended Responsibility to the Conceptual Synthesist or Leader who granted the underlying Delegated Authority.
- Being automatically restored when sufficient Delegated Authority is again held, unless the escalation has resulted in reassignment before restoration occurs.
- Being marked fulfilled upon production of the Deliverable or Outcome it obligates.

Behaviors

How does it behave?

Responsibility:

- Is assigned at the same time as an Objective or Assignment, and validated against the subject's Delegated Authority at that moment.
- Is automatically suspended, without requiring separate action, the instant the subject's Delegated Authority narrows below the Responsibility's required authority.
- Escalates immediately upon suspension to the Conceptual Synthesist or Leader who granted the subject's Delegated Authority, surfacing the suspended Responsibility for a decision.
- Resumes automatically if the subject's Delegated Authority is restored to a sufficient scope before the escalation is otherwise resolved.
- Is recorded via Record Event at assignment, suspension, escalation, restoration, and fulfillment.

Relationships

How does it interact with other concepts?

Responsibility is held by Company, Team, and Worker, and is bound to an Objective or Assignment.

Responsibility is scoped within Delegated Authority and is automatically suspended when Delegated Authority narrows below what it requires — including narrowing caused by Status's Restricted condition.

Responsibility escalates to Conceptual Synthesist or Leader upon suspension, giving the party accountable for the underlying grant of Delegated Authority the standing to decide whether to wait for restoration or reassign the obligation.

Responsibility's escalation relationship to Delegated Authority is joined by an equivalent relationship to Resource (Chapter 26): exhausted Resource-access retries required by the Responsibility's fulfillment escalate it the same way an authority narrowing does.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Responsibility is fulfilled through Deliverable and Outcome production, and contributes to Organizational Experience upon fulfillment or failure to fulfill.

Every change in Responsibility is preserved through Audit Trail.

Constraints

What boundaries govern it?

Responsibility shall:

- Never be assigned in a scope exceeding the subject's current Delegated Authority.
- Never remain suspended without escalating to the Conceptual Synthesist or Leader who granted the underlying Delegated Authority.
- Never require reassignment to resume — restoration is automatic once sufficient Delegated Authority returns, unless reassignment has already occurred through escalation.
- Record every assignment, suspension, escalation, restoration, and fulfillment to the Audit Trail.

Looking Ahead

The next chapter defines Status as the shared conceptual property that reflects a subject's current operating standing independent of its Lifecycle State.

Chapter 6 — Status

This chapter defines Status as the shared conceptual property that reflects a subject's current operating standing, independent of its Lifecycle State.

Definition

What is it?

Status is the shared conceptual property that establishes the current operating standing of an Organizational Entity or Operational Object, distinct from whether that subject exists or is capable of acting at all.

Status reflects the quality of a subject's ongoing operation, not its existence or availability.

Semantics

What does it fundamentally mean?

Status always holds exactly one value for a given subject at a given time.

Status is independent of Lifecycle State: a subject may be Active (Lifecycle State) while holding any Status value; a subject that is not Active holds no meaningful Status until it returns to Active.

Status changes only through defined transitions, triggered by a raised or resolved concern regarding the subject's operation.

Purpose

Why does it exist?

Status exists to distinguish a subject that is operating normally from one whose operation is under evaluation or has been constrained due to a confirmed concern, and to provide the mechanism through which such a concern automatically narrows the subject's Delegated Authority when warranted.

Objectives

What is it intended to accomplish?

Status shall:

- Establish a subject's current operating standing at all times while Active.
- Distinguish normal operation from operation under evaluation or under constraint.
- Automatically narrow Delegated Authority when a subject's Status becomes Restricted.

- Preserve a complete history of every Status transition and its basis.

Composition

What conceptually constitutes it?

Status consists of the following standings:

- **Good Standing** — the subject is operating normally; no concern is active.
- **Under Review** — a concern regarding the subject's operation has been raised; the subject continues operating under its existing Delegated Authority while the concern is evaluated.
- **Restricted** — a concern regarding the subject's operation has been confirmed; the subject's Delegated Authority is automatically narrowed beyond its normal scope until Status returns to Good Standing or Under Review.

Capabilities

What must it be capable of?

Status shall be capable of:

- Transitioning between standings according to defined rules, based on a raised or resolved concern.
- Automatically constraining the scope of Delegated Authority available to a subject upon transition to Restricted.
- Automatically releasing that constraint upon transition out of Restricted.
- Invoking Record Event for every transition, recording the concern underlying it.

Behaviors

How does it behave?

Status:

- Is established at Good Standing when a subject's Lifecycle State first becomes Active.
- Transitions from Good Standing to Under Review when a concern is raised; from Under Review to Restricted when that concern is confirmed; from Under Review back to Good Standing when a concern is resolved without confirmation; and from Restricted to Good Standing or Under Review when the confirmed concern is resolved.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Narrows the subject's Delegated Authority automatically and immediately upon entering Restricted — this is an enforced constraint, not a signal a human must act on manually.
- Restores the subject's normal Delegated Authority automatically upon leaving Restricted.
- Holds no meaningful value while the subject's Lifecycle State is not Active; resumes at its prior standing when the subject returns to Active, unless the underlying concern requires it to resume at Good Standing instead.

Relationships

How does it interact with other concepts?

Status applies only to a subject that has been established with Identity and holds a Lifecycle State of Active.

Status directly constrains Delegated Authority: a subject in Restricted status operates within a narrowed scope of Delegated Authority for as long as that Status persists. The specific mechanics of how Delegated Authority defines and enforces that narrowed scope are governed by Delegated Authority's own chapter.

Status transitions invoke Record Event and are preserved within Audit Trail when a Restricted standing reflects a pattern relevant to future execution.

Constraints

What boundaries govern it?

Status shall:

- Hold exactly one value at any time a subject is Active.
- Never be established or hold a meaningful value for a subject that is not Active.
- Never transition along a path not explicitly defined.
- Never permit Restricted to persist without an active, recorded concern.
- Automatically narrow Delegated Authority upon entering Restricted, and automatically restore it upon leaving Restricted, without requiring separate manual action.
- Invoke Record Event for every transition, including its basis.
- Never transition the sole remaining holder of Administrative Authority into Restricted standing (CR-26) — where no other identity currently, validly holds Administrative Authority, the transition is refused, not merely inadvisable.

Looking Ahead

The next chapter in this Part defines Organizational Accountability as the shared conceptual property that establishes the ability to attribute every decision, action, and outcome to the responsible Company, Team, or Worker.

Chapter 7 — Organizational Accountability

This chapter defines Organizational Accountability as the shared conceptual property that establishes the ability to attribute every decision, action, delegation, expenditure, organizational change, and Outcome to the responsible Company, Team, or Worker.

Definition

What is it?

Organizational Accountability is the shared conceptual property that establishes that every attributable action taken within The Autonomous Company can be traced, without gap or ambiguity, to the Identity responsible for it.

Organizational Accountability is not itself a record — it is the guarantee that a complete record exists and can be traced. Audit Trail is the mechanism through which that guarantee is fulfilled.

Semantics

What does it fundamentally mean?

Organizational Accountability holds for The Autonomous Company as a whole; it is not a value a single subject possesses or loses, the way Status or Lifecycle State are.

An action without a traceable responsible Identity is not merely unaccountable — under Audit Trail's fail-closed behavior, it cannot occur at all.

Organizational Accountability applies equally regardless of whether the responsible subject is a human Conceptual Synthesist or an autonomous Company, Team, or Worker.

Purpose

Why does it exist?

Organizational Accountability exists so that authority can be safely delegated to autonomous subjects — every grant of Delegated Authority is made trustworthy by the guarantee that its exercise will be traceable to the Identity that exercised it, without exception.

Objectives

What is it intended to accomplish?

Organizational Accountability shall:

- Guarantee that every attributable action is traceable to a responsible Identity.
- Guarantee that no attributable action can occur without producing that traceability.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Extend uniformly across every Organizational Entity, Operational Object, and Shared Conceptual Property.
- Remain intact regardless of Lifecycle State, Status, or Role changes affecting the responsible subject.

Composition

What conceptually constitutes it?

Organizational Accountability consists of:

- **Attribution** — every action is tied to a responsible Identity.
- **Traceability** — that tie can be reconstructed at any later time, through Audit Trail.
- **Completeness** — no attributable action exists without both of the above.

Capabilities

What must it be capable of?

Organizational Accountability shall be capable of:

- Being verified for any subject, at any time, by reconstructing its full history through Audit Trail.
- Being verified across a chain of delegation — tracing not only who took an action, but under whose Delegated Authority they were permitted to.
- Withstanding retirement — accountability for a Retired subject's history remains fully traceable.

Behaviors

How does it behave?

Organizational Accountability:

- Is established the moment Audit Trail's Record Event succeeds for a given action.
- Is violated only if an attributable action occurs without a corresponding Audit Entry — a condition Audit Trail's fail-closed design is intended to make impossible.
- Persists independently of the responsible subject's current Lifecycle State, Status, or Role.

Relationships

How does it interact with other concepts?

Organizational Accountability is established through Audit Trail and depends on Identity as the anchor every entry attributes to.

Organizational Accountability depends on Delegated Authority to determine not just who acted, but whether they were authorized to — full verification of accountability requires tracing both facts together.

Organizational Accountability is the property that Organizational Experience draws upon when identifying patterns across historical execution.

Constraints

What boundaries govern it?

Organizational Accountability shall:

- Never be considered satisfied by a partial or ambiguous attribution.
- Never permit an attributable action to succeed without a corresponding Audit Entry.
- Apply without exception to every Organizational Entity, Operational Object, and Shared Conceptual Property.
- Remain traceable for the full existence of The Autonomous Company, including after a subject's retirement.

Looking Ahead

Part II — Shared Conceptual Properties is now complete. The next Part defines Human Participants — the humans who interact with The Autonomous Company — beginning with Conceptual Synthesist, who originates, authorizes, and oversees Autonomous Execution.

PART III — HUMAN PARTICIPANTS

Chapter 8 — Conceptual Synthesist

This chapter defines Conceptual Synthesist as the Human Participant who originates, authorizes, and oversees Autonomous Execution within The Autonomous Company.

Definition

What is it?

A Conceptual Synthesist is a human whose actions within The Autonomous Company — establishing Objectives, granting Delegated Authority, allocating Fid, evaluating Outcomes — are attributable through Identity, the same as any Organizational Entity's.

A Conceptual Synthesist is the originating source of Delegated Authority within The Autonomous Company, not a recipient of it granted by another subject.

Semantics

What does it fundamentally mean?

A Conceptual Synthesist holds Identity and Lifecycle State, established the same way any subject's are, so that every action they take is fully attributable and their access can be granted or revoked over time.

A Conceptual Synthesist does not hold Role or Status — Role governs function within Autonomous Execution, which a Conceptual Synthesist oversees rather than performs; Status governs standing under evaluation for a concern, which does not apply to the originating source of authority in the same way it applies to a subject operating under delegated authority.

Purpose

Why does it exist?

Conceptual Synthesist exists so that every human action affecting The Autonomous Company is as fully attributable and traceable as any autonomous action, satisfying Organizational Accountability without exception for the humans who originate and oversee the system.

Objectives

What is it intended to accomplish?

Conceptual Synthesist shall:

- Establish a verifiable Identity for every human authorized to interact with The Autonomous Company.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Authenticate a claimed Identity before permitting any action to proceed.
- Attribute every human action to the authenticated Identity through Audit Trail.
- Support revocation of access without affecting historical attribution already recorded.

Composition

What conceptually constitutes it?

Conceptual Synthesist consists of:

- **Identity** — the permanent attribution anchor, established as with any subject.
- **Lifecycle State** — Active or Suspended for ordinary access changes; Retired for permanent revocation. Conceptual Synthesist does not use Completed, which presumes a concluded purpose rather than an ongoing human relationship to the system.
- **Authentication** — the credentials and verification mechanism through which a claimed Identity is confirmed before any action is permitted. Authentication is mutable, may be reset or reissued, and is never part of Identity itself.

A Conceptual Synthesist's authority to establish other Conceptual Synthesists or configure system-wide settings is governed by Delegated Authority, defined in its own chapter. Not every Conceptual Synthesist holds this authority.

Capabilities

What must it be capable of?

Conceptual Synthesist shall be capable of:

- Establishing Identity and Lifecycle State jointly, as with any subject, except for the Founding Conceptual Synthesist, who is established through the Founding Act.
- Authenticating a claimed Identity before any action is attributed to it.
- Originating a grant of Delegated Authority to a Company, Team, or Worker.
- Having Lifecycle State transitioned to Suspended or Retired by another authorized Conceptual Synthesist, without affecting the Audit Trail already attributed to them.

Behaviors

How does it behave?

Conceptual Synthesist:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Is established with Identity and Lifecycle State jointly, the same as any subject, except the Founding Conceptual Synthesist, whose establishment is the Founding Act itself.
- Must be successfully authenticated before any action they take is accepted or attributed.
- Originates Delegated Authority rather than receiving it from another subject within The Autonomous Company.
- Remains fully attributable in the historical record regardless of subsequent Lifecycle State changes, including Retired.

Relationships

How does it interact with other concepts?

Conceptual Synthesist holds Identity and Lifecycle State, as any subject does, but does not hold Role or Status.

Conceptual Synthesist is the originating source of Delegated Authority granted to Company, Team, and Worker; the mechanics of that grant are governed by Delegated Authority's own chapter.

Conceptual Synthesist establishes Objectives, allocates Fid, and evaluates Outcomes, per Chapter 1's Core Principles.

Every action taken by a Conceptual Synthesist invokes Record Event and is preserved in Audit Trail, satisfying Organizational Accountability without exception.

Constraints

What boundaries govern it?

Conceptual Synthesist shall:

- Never act without successful Authentication of the claimed Identity.
- Never have Authentication credentials encoded within Identity itself.
- Never have prior attributed history altered or removed by a subsequent Lifecycle State change, including Retired.
- Exactly one Conceptual Synthesist may be established through the Founding Act.
- Only a Conceptual Synthesist holding **Administrative Authority** may establish another Conceptual Synthesist or transition another Conceptual Synthesist's Lifecycle State.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- The Founding Conceptual Synthesist holds Administrative Authority inherently, through the Founding Act, without requiring a grant..

Looking Ahead

Part III — Human Participants is now complete. The next Part defines the Operational Objects that Responsibility, Delegated Authority, and Fid are exercised in pursuit of, beginning with Objective.

PART IV — OPERATIONAL OBJECTS

Chapter 9 — Objective

This chapter defines Objective as the operational object that establishes the purpose delegated to The Autonomous Company by a Conceptual Synthesist.

Definition

What is it?

An Objective is the operational object that establishes a specific purpose delegated by a Conceptual Synthesist to a Company, to be accomplished through Autonomous Execution.

An Objective's content is fixed at establishment. An Objective is never modified after establishment; a change in purpose is accomplished by retiring the existing Objective and establishing a new one.

Semantics

What does it fundamentally mean?

An Objective is flat — it does not contain sub-Objectives. Decomposition of an Objective into distinct units of work is accomplished through Project and Assignment, not through hierarchy within Objective itself.

Cost-Optimized — among options otherwise equally authorized under this Objective's Delegated Authority and Fid, the Leader favors the lowest-cost choice.

Balanced — the Leader has full discretion between cost and precision trade-offs, per this chapter's ordinary Leadership Authority. This is the default.

Precision-Optimized — among options otherwise equally authorized, the Leader favors the highest-capability or highest-fidelity choice.

Optimization Directive guides selection among options an Objective's Delegated Authority and Fid already permit — it never expands what a Leader may do, only how a Leader chooses among what it may already do.

An Objective is the binding purpose against which Delegated Authority, Fid, and Responsibility are each scoped.

An Objective runs under a Process (Chapter 28): its Governing Process names the kind of work the Objective is. Every Objective has one before its Plan is built. The Project's Leader recommends the Process, with real, stated reasoning, from what the Objective sets out to do and what each Process available to the Receiving Company says it is for; the

Conceptual Synthesist chooses it. Work no other Process fits runs under the core Simple Process (Chapter 28).

An Objective is accomplished through the production of an Outcome.

An Objective's Oversight Level establishes how much autonomy its Project is granted before a human decision is required, and is fixed at establishment on the same terms as Purpose and Success criteria — never altered afterward; a change in desired oversight requires retirement and re-establishment.

Autonomous — the Objective's Project proceeds through Assignment, Deliverable, and Outcome production without pausing for approval, except where a different chapter's own constraint independently requires one (for example, exhausted Resource-access retries, per Chapter 26, or Role-type creation, per Chapter 23).

Milestone-Gated — the Project pauses into Pending Approval (Chapter 4) at each milestone defined during Planning (Chapter 10), resuming only on explicit Conceptual Synthesist or Leader decision. Milestone-Gated requires Planning Required (Chapter 10) to also be set, so milestones always have a defined source.

A milestone, for a Milestone-Gated Objective, is an individually Approved Stage of its own Plan Deliverable (CR-33). “Milestones defined during Planning” above refers to this set directly — no separate milestone-declaration mechanism exists or is needed. Where the Objective’s Governing Process sequences its work (Chapter 28), that Process’s own Stages are the milestones directly instead; where its Governing Process is Leader-directed, the milestones are those of its Plan — see Chapter 10’s own Constraints.

Step-Gated — the Project pauses into Pending Approval before every individual Assignment or Deliverable production, resuming only on explicit decision at each step.

An Objective may optionally carry a Baseline Reference to a prior Objective's Outcome, establishing that this Objective's purpose is to enhance, correct, or extend what the referenced Objective produced. A Baseline Reference never reopens, modifies, or extends the referenced Objective in any way — the referenced Objective remains exactly as Completed and immutable as this chapter's core constraint already requires of every Objective, with or without a Baseline Reference.

Purpose

Why does it exist?

Objective exists to give Autonomous Execution a fixed, specific, attributable purpose to accomplish — one that cannot drift after establishment — so that Delegated Authority, Fid, and Responsibility all remain scoped against a stable reference throughout execution.

Objectives

What is it intended to accomplish?

Objective shall:

- Establish a specific, fixed purpose delegated by a Conceptual Synthesist to a Company.
- Remain unchanged after establishment; any change in purpose requires retirement and re-establishment.
- Provide the binding reference that Delegated Authority, Fid, and Responsibility are each scoped and bound to.
- Define the success criteria against which its Outcome is evaluated.

Composition

What conceptually constitutes it?

Objective consists of:

- **Purpose** — the specific outcome sought, fixed at establishment.
- **Success criteria** — the conditions under which the Objective is considered accomplished.
- **Originating Conceptual Synthesist** — who delegated this Objective.
- **Receiving Company** — the Company this Objective is delegated to.
- **Optimization Directive** — one of Cost-Optimized, Balanced, or Precision-Optimized, set by the Conceptual Synthesist at establishment, guiding the Leader's choice where Resource (Chapter 26) offers more than one qualifying Provider for the same access.
- **Governing Process** — the Process this Objective runs under, and the version it runs under: the kind of work it is. Every Objective has one. It is recommended by the Project's Leader, with stated reasoning, from the Objective's intent and each available Process's description of its kind of work, and chosen by the Conceptual Synthesist before the Plan is built — never defaulted on the Synthesist's behalf. The choice is among the Processes available to the Receiving Company: its own, and the core Processes every Company uses as they are (Chapter 28). Where none of them fits, a Process for the work is established first; an Objective is never committed without a Governing Process. Work no other Process fits runs under the core Simple Process.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- **Oversight Level** — one of Autonomous, Milestone-Gated, or Step-Gated, set by the Conceptual Synthesist at establishment, governing how often the Objective's Project pauses into Pending Approval (Chapter 4) for explicit sign-off.
- **Planning Required** — a boolean, set by the Conceptual Synthesist at establishment independent of Oversight Level, indicating whether a Plan must be produced and approved before the Objective's Project may create any Assignment beyond the Planning Assignment itself.
- **Priority** — an ordinal, set by the Conceptual Synthesist at establishment, used only to break ties in Worker availability when a Company is pursuing multiple Objectives concurrently (Chapter 18). Priority never overrides Delegated Authority, Fid sufficiency, Role requirements, or any other constraint governing whether an Assignment may be made — it is a tie-breaker of last resort, applied only when two or more Objectives are otherwise equally eligible to claim the same Worker.
- **Baseline Reference** — an optional reference to a prior Objective's Outcome, present only when this Objective is undertaken as an enhancement, correction, or extension of previously delivered work.
- **Originating Mandate Item** — **present only on an Objective mechanically originated from an already-Approved Team Mandate Item before CR-50, when a firing established an Objective rather than assigning work to the Team holding the Mandate; absent for every other Objective, and never present on an Objective established after CR-50. Retained so that Objectives originated under the prior text remain readable as exactly what they were.**

Capabilities

What must it be capable of?

Objective shall be capable of:

- Being established by a Conceptual Synthesist with fixed purpose and success criteria.
- Anchoring a grant of Delegated Authority and Fid at the same time it is delegated to a Company.
- Being retired, whether accomplished, abandoned, or superseded by a newly established Objective.
- Being evaluated against its success criteria upon production of its Outcome.
- Being established with exactly one Optimization Directive, defaulting to Balanced where not otherwise specified. Safe to default, unlike Oversight Level: Balanced

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

simply preserves ordinary, already-bounded Leader discretion rather than loosening or tightening anything.

- Being established under exactly one Governing Process, recommended by the Project's Leader with stated reasoning and chosen by the Conceptual Synthesist before the Plan is built; no Process is silently assumed.
- Requiring the Conceptual Synthesist to explicitly set an Oversight Level at establishment — Objective is never established without one, and no default is assumed.
- Being established with an optional Baseline Reference to a prior, Completed Objective's Outcome, granting the new Objective's Project automatic read access to the Deliverables produced under the referenced Objective.

Behaviors

How does it behave?

Objective:

- Is established with fixed purpose and success criteria; neither may be altered afterward.
- Is delegated to a Company, which establishes exactly one Project to pursue it.
- Persists through its Lifecycle State — Active while being pursued, Completed once its Outcome is produced and evaluated, Retired if abandoned or superseded.
- A Completed Objective's Deliverables may be revised after completion (Chapter 12) without reopening it. The Objective remains Completed, its Purpose and success criteria unchanged, and no Project, Assignment, grant or Fid of its own is revived. A change of direction still requires a new Objective, with a Baseline Reference where it builds on this one.

An Objective's Oversight Level determines, together with Chapter 4's Pending Approval condition, whether and how often its Project pauses for explicit human sign-off during execution.

May also pass through Pending Approval whenever a defined gate or an execution-failure escalation requires an explicit Conceptual Synthesist or Leader decision before proceeding, per Chapter 4.

Where a Baseline Reference is present, the referenced Objective's Deliverables become available for read access to the new Objective's Project and its Workers automatically upon establishment — narrower in kind than Organizational Experience's cross-Role visibility (Chapter 16), and therefore requiring no separate grant.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Is recorded via Record Event at establishment, retirement, and evaluation against its success criteria.

Relationships

How does it interact with other concepts?

Objective originates from Conceptual Synthesist and is delegated to Company.

Objective is the binding reference for Delegated Authority and Fid — both are granted for the purpose of accomplishing a specific Objective and expire with it.

Objective is the binding reference for Responsibility, per Chapter 5.

Objective is pursued through exactly one Project instance established by the receiving Company, and decomposed further through Assignment.

Objective is accomplished through the production of Outcome, evaluated against the Objective's success criteria.

Objective's Optimization Directive is consumed by Resource (Chapter 26) when a Leader selects among interchangeable Providers fulfilling the same Resource classification.

What happens when an Objective is Accomplished, beyond its Outcome, is its Governing Process's Accomplishment action (Chapter 28), where that Process carries one. No provision is keyed on a kind of Objective.

Objective's Oversight Level governs when Lifecycle State's Pending Approval condition (Chapter 4) is entered during this Objective's execution — Autonomous minimizes entry to only the cases other chapters independently mandate; Milestone-Gated and Step-Gated add Objective-level requirements on top of those.

Objective's optional Baseline Reference links a new Objective to the prior Objective it builds upon, without altering, reopening, or extending the referenced Objective; Objective's “never modified after establishment” constraint applies without exception, including to any Objective serving as another's Baseline Reference.

Constraints

What boundaries govern it?

Objective shall:

- Never be modified after establishment; a change in purpose requires retirement and re-establishment.
- Never contain sub-Objectives; decomposition occurs through Project and Assignment.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never be delegated without an originating Conceptual Synthesist and a receiving Company. This admits no exception. An Objective mechanically originated from a Team Mandate Item before CR-50 carried the Mandate's own originating Conceptual Synthesist and the Mandate-holding Team's own current Company, and remains valid on those terms; a Mandate firing now assigns work to the Team rather than establishing an Objective (Chapter 19), so this exception has no remaining occasion to apply.
- Never remain Active without at least the possibility of a bound Delegated Authority and Fid grant.
- Never allow Optimization Directive to expand a Leader's Delegated Authority or Fid beyond what is otherwise granted — it guides selection among already-authorized options, never authorizes a new one.
- Never be committed without a Governing Process, and never have its Governing Process changed once its Planning Assignment exists.
- Never be established without an explicit Oversight Level.
- Never allow a Milestone-Gated or Step-Gated Objective's Project to bypass a gate its Oversight Level defines, regardless of Leader discretion.
- Never grant a Baseline Reference the ability to alter, reopen, or extend the Lifecycle State or content of the referenced Objective.
- Record the Baseline Reference at establishment, and every Deliverable read under it, to the Audit Trail.

Looking Ahead

The next chapter defines Project as the operational object a Company establishes to pursue a delegated Objective.

Chapter 10 — Project

This chapter defines Project as the operational object a Company establishes to pursue a delegated Objective.

Definition

What is it?

A Project is the operational object a Company establishes to pursue a specific Objective delegated to it. A Project is the conceptual system being built or the effort being undertaken to accomplish that Objective.

Exactly one Project exists per Objective. A Company pursuing multiple Objectives concurrently establishes one Project for each.

Semantics

What does it fundamentally mean?

A Project operates under Delegated Authority and Fid sub-delegated to it by the Company that established it, drawn from the Company's own grant for the Objective the Project pursues.

A Project is the level at which Teams are assigned to do work — a Team is assigned to a Project, not directly to an Objective.

A Project's existence is bound to its Objective: it comes into being because the Objective was delegated, and it does not outlive the Objective it pursues.

Where the bound Objective's Planning Required attribute (Chapter 9) is set, the Project's Leader receives a Planning Assignment (Chapter 11) immediately upon establishment, producing a Plan as a Deliverable (Chapter 12) — a cost estimate and proposed build sequence — before any other Assignment may be created under the Project.

The Project's Assignments are created in the sequence its Objective's Governing Process defines — a later stage's Assignment shall never be created before the Deliverable of the stage immediately preceding it has cleared Gate Evaluation as Approved (or, where staged per Chapter 12, its final stage has). Where the Governing Process is Leader-directed (Chapter 28), Assignment sequencing remains at Leader discretion: the Leader chooses the Deliverables and sequences the work, subject only to whatever explicit Constraint independently applies.

Where the bound Objective's Oversight Level is Autonomous, "cleared Gate Evaluation as Approved" above is satisfied the moment a Stage's Deliverable is produced — Autonomous never invokes an explicit Gate Evaluation decision for any Deliverable, and a Process-governed Objective is no exception to that already-established property. This

clarifies, rather than changes, how this Constraint's existing text applies under Autonomous; Milestone-Gated and Step-Gated are both unaffected, since Gate Evaluation is already real and enforced for both.

Planning, where required, begins only once the bound Objective's Governing Process has been chosen (Chapter 9); "immediately upon establishment" above is read accordingly.

Purpose

Why does it exist?

Project exists to give a Company a concrete unit of effort to establish, staff, and resource in pursuit of a specific Objective, and to give Teams a defined scope of work to be assigned to.

Objectives

What is it intended to accomplish?

Project shall:

- Be established by a Company upon receiving an Objective, exactly one per Objective.
- Hold Delegated Authority and Fid sub-delegated from the Company's grant for that Objective, never exceeding it.
- Provide the scope to which Teams are assigned.
- Retire automatically when its Objective is retired, whether accomplished, abandoned, or superseded.

Composition

What conceptually constitutes it?

Project consists of:

- **Bound Objective** — the single Objective this Project exists to pursue.
- **Establishing Company** — the Company that established this Project.
- **Sub-delegated authority** — the Delegated Authority granted to this Project by its establishing Company, scoped within the Company's own grant.
- **Sub-allocated Fid** — the Fid allocated to this Project by its establishing Company, drawn from the Company's own remaining balance.
- **Assigned Teams** — the Teams currently assigned to this Project.

Capabilities

What must it be capable of?

Project shall be capable of:

- Being established by a Company immediately upon receiving an Objective, with Delegated Authority and Fid sub-delegated at establishment.
- Having Teams assigned to it.
- Being retired automatically, cascading from its Objective's retirement, per Chapters 14 and 15's cascading rules.
- Being evaluated for progress toward its Objective's success criteria.

Behaviors

How does it behave?

Project:

- Is established the moment a Company receives an Objective, one Project per Objective, with no delay or intermediate state between Objective delegation and Project establishment.
- Receives Delegated Authority and Fid sub-delegated from the Company's own grant, never exceeding what the Company holds.
- Retires automatically, cascading, the instant its bound Objective reaches Completed or Retired — mirroring how sub-delegated Delegated Authority and Fid already cascade from their source.
- Is recorded via Record Event at establishment, Team assignment, and retirement.

Upon production of the Plan Deliverable, a Project bound to a Planning-Required Objective transitions to Pending Approval (Chapter 4), resuming only when the Conceptual Synthesist or Leader who holds the Objective's Delegated Authority approves the Plan. Approval permits ordinary Assignment creation to begin; rejection retires the Objective as abandoned, per Chapter 9's existing retirement Capabilities.

The Plan Deliverable's Approved, Revise-Requested, or Rejected decision follows Gate Evaluation as defined generally in Chapter 12. Where Gate Evaluation yields Revise-Requested, the Planning Assignment produces a new candidate Plan Deliverable per Chapter 12, and the Project re-enters Pending Approval upon its production, repeating as needed before Approved or Rejected is reached.

Relationships

How does it interact with other concepts?

Project is established by Company and bound to exactly one Objective.

Project holds Delegated Authority and Fid sub-delegated from the Company that established it, following the same derivation-chain rules already established in Chapters 14 and 15.

Project is the scope to which Team is assigned, per Chapter 2's Organizational Relationships.

Project's Lifecycle State is bound to its Objective's: Project retires when its Objective retires, cascading the same way sub-delegated Delegated Authority and Fid already do.

Constraints

What boundaries govern it?

Project shall:

- Exist in exactly a one-to-one relationship with its Objective — never more than one Project per Objective.
- Never hold Delegated Authority or Fid exceeding what its establishing Company holds for the bound Objective.
- Never persist beyond the Lifecycle State of its bound Objective.
- Never be established without a bound Objective and an establishing Company.
- Never create an Assignment other than the Planning Assignment under a Planning-Required Project before its Plan Deliverable has been approved.
- Never begin a Planning Assignment before the bound Objective's Governing Process has been chosen (Chapter 9).
- Where a Project's Leader is not established atomically with the Project itself, the Conceptual Synthesist who established the bound Objective — or a Conceptual Synthesist holding Administrative Authority — must explicitly designate the Leader before any further Assignment may be created under the Project (CR-28); no automatic, inferred, or algorithmic designation is permitted.
- A Project's designated Leader may be re-designated — the existing designation replaced by a new one — if and only if the currently designated Leader's own Lifecycle State is Retired (CR-34). Authorization for re-designation is identical to original designation: the Objective's originating Conceptual Synthesist, or a Conceptual Synthesist currently holding Administrative Authority. A Leader who is merely Suspended, rather than Retired, does not permit re-designation — the

existing designation stands unchanged until Suspended standing itself resolves, one way or the other.

- Where a Milestone-Gated Objective's Governing Process sequences its work (CR-33; Chapter 28), its Stages are the Objective's milestones; the Project transitions into Pending Approval upon fulfillment of each Stage's own Assignment(s), releasing only on explicit Conceptual Synthesist or Leader decision before the next Stage's Assignment may be created. Where its Governing Process is Leader-directed (Chapter 28), milestones are instead drawn from the Objective's own Approved Plan, per the following Constraint.
- Where the bound Objective's Oversight Level is Milestone-Gated and its Governing Process is Leader-directed (CR-33; Chapter 28), the Planning Assignment produces its Plan Deliverable using Stage (Chapter 12) exactly as a Step-Gated Objective's own Planning Assignment does, regardless of the Objective's own Oversight Level otherwise governing Gate Evaluation for its ordinary work. Each Stage of such a Plan is individually Gate-Evaluated on the identical terms Chapter 12 already defines. Once the final Plan Stage is Approved, ordinary Assignment creation is gated by milestone order — see Chapter 11's own Constraints.

Never permit a Plan Deliverable's Estimated allocation (Chapter 15) to exceed its Objective's own Allocated amount and still be Approved at Plan Gate Evaluation, unless the evaluating Conceptual Synthesist also increases the Allocated amount to at least the Estimated figure in the same act — an Estimated allocation exceeding the existing ceiling is itself a named ground for Revise-Requested or Rejected, the same weight this chapter's other gate-evaluation grounds already carry.

Looking Ahead

The next chapter defines Assignment as the operational object through which a Team or Worker is assigned to a Project

Chapter 11 — Assignment

Chapter 11 — Assignment

This chapter defines Assignment as the operational object through which a Team or Worker is bound to a Project, Team, or unit of work, carrying sub-delegated authority and resources to fulfill it.

Definition

What is it?

Assignment is the operational object that binds a Team or Worker to a Project, or a Worker to a Team, establishing the Responsibility the subject is obligated to fulfill and the Delegated Authority and Fid sub-delegated to fulfill it.

Assignment is the single mechanism through which both a Team is bound to a Project and a Worker is bound to a Team or directly to a Project — there is no separate mechanism for each level.

A subject may hold multiple Assignments concurrently.

Semantics

What does it fundamentally mean?

Assignment continues the same derivation chain already established for Delegated Authority and Fid: the assigning subject (a Project, or a Team acting under its own Project Assignment or under its own standing grant) sub-delegates a portion of its own Delegated Authority and Fid to the subject receiving the Assignment.

Assignment is the event that creates a Responsibility, per Chapter 5 — a subject does not hold a Responsibility without a corresponding Assignment or Objective binding it.

Where an Assignment requires a specific Role classification, only a Worker currently holding that Role may receive it, per Chapter 23.

Purpose

Why does it exist?

Assignment exists to bind a Team or Worker to a specific unit of work, with a specific, sub-delegated scope of authority and resources sufficient to fulfill it, and a specific, attributable Responsibility for its fulfillment.

Objectives

What is it intended to accomplish?

Assignment shall:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Bind a Team to a Project, or a Worker to a Team or Project.
- Sub-delegate Delegated Authority and Fid from the assigning subject, never exceeding what the assigning subject holds.
- Create the corresponding Responsibility, scoped to the Assignment.
- Verify the receiving Worker holds any Role classification the Assignment requires, before the Assignment may be made.
- Support multiple concurrent Assignments for a single subject.

Composition

What conceptually constitutes it?

Assignment consists of:

- **Assigning subject** — the Project or Team making the Assignment.
- **Receiving subject** — the Team or Worker bound by the Assignment.
- **Required Role classification** — where applicable, the Role a receiving Worker must hold.
- **Sub-delegated authority** — the Delegated Authority granted to the receiving subject, scoped within the assigning subject's own.
- **Sub-allocated Fid** — the Fid allocated to the receiving subject, drawn from the assigning subject's own remaining balance.
- **Bound Responsibility** — the Responsibility this Assignment creates.
- **Declared milestone (CR-33)** — present only under a Milestone-Gated Objective whose Plan governs its own milestones (Chapter 10): an explicit, identity-addressed reference to one Stage of the Objective's own Plan, stated by the assigning Leader at creation time, never inferred. Not independently verified at Assignment-creation time; its correctness is confirmed, if at all, only through the ordinary release review at the milestone boundary it claims to belong to — the same trust this Objective's Oversight Level already extends to ordinary Assignment creation generally. An Auditor review Assignment, created to review a Deliverable already produced under this Objective, declares no milestone and is not required to: it is not a step in the Objective's own sequence of work but a review of work already done, and so has no milestone to belong to. This is the same closed, named exemption ¶803 already grants it, applied one Constraint earlier; an Assignment type not named here is ordinary for this Constraint's purposes, whatever its own purpose may be.

Capabilities

What must it be capable of?

Assignment shall be capable of:

- Binding a Team to a Project, or a Worker to a Team or Project, with sub-delegated Delegated Authority and Fid established at the same time.
- Verifying the receiving Worker holds any required Role classification before the Assignment is made; refusing the Assignment if not.
- Creating a Responsibility bound to the Assignment at the moment the Assignment is made.
- Supporting multiple concurrent Assignments held by the same subject.
- Being retired, cascading the same way sub-delegated Delegated Authority and Fid already cascade, when the assigning subject or its own binding retires.

Behaviors

How does it behave?

Assignment:

- Is established when an assigning subject binds a receiving subject to a unit of work, sub-delegating a portion of its own Delegated Authority and Fid at the same moment.
- Verifies required Role classification, where specified, before completing; fails outright if the receiving Worker does not hold it.
- Creates a Responsibility, per Chapter 5, bound to this Assignment.
- Retires automatically, cascading, when the assigning subject's own Assignment or Project binding retires — mirroring Delegated Authority and Fid's cascading rules.
- Is recorded via Record Event at establishment, Role verification, and retirement.

Relationships

How does it interact with other concepts?

Assignment binds Team to Project, and Worker to Team or Project.

Assignment sub-delegates Delegated Authority and Fid from the assigning subject, continuing the same derivation chain established in Chapters 14 and 15.

Assignment creates and binds Responsibility, per Chapter 5.

Assignment requires the receiving Worker to hold any specified Role classification, per Chapter 23.

Assignment cascades retirement from the assigning subject's own binding, the same way sub-delegated Delegated Authority and Fid cascade from their source.

Where its Responsibility's fulfillment is staged (Chapter 12), an Assignment persists across multiple sequential Deliverable productions rather than concluding at its first.

Constraints

What boundaries govern it?

Assignment shall:

- Never sub-delegate Delegated Authority or Fid exceeding what the assigning subject holds.
- Never be completed for a Worker lacking a required Role classification.
- Never exist without creating a bound Responsibility.
- Never persist beyond the Lifecycle State of the assigning subject's own Assignment or Project binding.
- Permit a subject to hold multiple Assignments concurrently.
- For a Milestone-Gated Objective (CR-33), once its final Plan Stage is Approved, never create an Assignment declaring a later milestone while any Assignment declaring an earlier, not-yet-fully-fulfilled milestone remains outstanding. Upon fulfillment of every Assignment declared against a given milestone, the Project transitions into Pending Approval; only an explicit Conceptual Synthesist or Leader decision releases it to begin the next milestone's work — the same release mechanism Chapter 4's Pending Approval condition already defines.
- No new ordinary Assignment may be created under an Objective while that Objective's own Lifecycle State is Pending Approval, regardless of what caused the pause (CR-35) — milestone completion (CR-33), Fid overrun (CR-10), execution-failure escalation (CR-14), or any future cause. This Constraint does not apply to the Planning Assignment, nor to an Auditor review Assignment created to review a Deliverable already produced under this Objective, each of which remains governed only by its own existing preconditions. These are a closed, named set: an Assignment type not named here is ordinary for this Constraint's purposes, whatever its own purpose may be.

Looking Ahead

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

The next chapter defines Deliverable as the operational object produced in fulfillment of a Responsibility.

Chapter 12 — Deliverable

Chapter 12 — Deliverable

This chapter defines Deliverable as the operational object representing the concrete artifact produced in fulfillment of a Responsibility.

Definition

What is it?

A Deliverable is the operational object representing a concrete artifact produced by a subject in fulfillment of a Responsibility.

A Deliverable is distinct from Outcome: a Deliverable is the artifact produced; an Outcome, defined in the next chapter, is the evaluated result of that artifact against an Objective's success criteria. A Deliverable's production does not itself constitute evaluation.

Semantics

What does it fundamentally mean?

A Deliverable is produced by the subject holding the Responsibility it fulfills, under the Delegated Authority and Fid scoped to that Responsibility's Assignment - or, for a post-completion revision, under the grant a Conceptual Synthesist originates for that revision.

A Deliverable, once produced, is immutable — a correction or revision is a new Deliverable, not an alteration of the original, preserving Audit Trail integrity the same way Identity and Objective content remain fixed once established.

Once an Objective is Completed, a Conceptual Synthesist may authorize a post-completion revision of any Deliverable produced under it. A gate decides the big picture — the direction and shape of the work — and a detail found wrong afterwards is corrected by revision rather than by a new Objective. The change is worked out in conversation with the producing Role, and what the Conceptual Synthesist authorizes is the change itself, not a produced artifact. The authorization is a grant the Conceptual Synthesist originates (Chapter 14) to the Worker who will make the revision, scoped to that one revision, with Fid allocated alongside it from the Company's standing Fid Pool (Chapter 15). It expires when the revision is produced. The revision is a new Deliverable that supersedes its predecessor, is attributed to the Responsibility its predecessor fulfilled, and is marked as a post-completion revision. Work on an Objective that is not yet Completed is never revised this way; it is revised through its gate.

A Deliverable produced under a Milestone-Gated or Step-Gated Objective (Chapter 9's Oversight Level), or under a Team Mandate Item whose declared review treatment is Required (Chapter 19), undergoes Gate Evaluation: an explicit decision, by the Conceptual

Synthesist or Leader holding the Delegated Authority the producing Assignment was sub-delegated from, of Approved, Revise-Requested, or Rejected. Gate Evaluation never alters the evaluated Deliverable itself, consistent with this chapter's existing immutability constraint — each decision determines what happens next, not what the Deliverable already is.

Gate Evaluation's decision is accompanied by a Resolution for every Open Decision the presented Deliverable identified, so a Worker or Conceptual Synthesist consulting this Deliverable's history later sees not only what was decided but how each specific point of divergence was resolved. Where Gate Evaluation yields Revise-Requested, a Resolution may be partial — resolving some Open Decisions while deliberately deferring others to the next candidate.

A Deliverable may carry a Deliverable Type — a named classification, freely assignable by the producing Worker or Leader, at no cost and requiring no authority beyond what producing the Deliverable itself already requires. Where a Deliverable Type has a registered Schema (Chapter 27), every Deliverable of that Type must conform to it. Registering a Schema requires a Conceptual Synthesist holding Administrative Authority (Chapter 21), acting directly rather than through any Objective; assigning an existing, unschemed Type label does not.

Production of a Deliverable marks the associated Responsibility as fulfilled, per Chapter 5.

Purpose

Why does it exist?

Deliverable exists to give Responsibility a concrete, attributable artifact of fulfillment — something that can be produced, recorded, referenced, and later evaluated — distinct from the judgment of whether that artifact actually accomplished its purpose.

Objectives

What is it intended to accomplish?

Deliverable shall:

- Represent a concrete artifact produced in fulfillment of a specific Responsibility.
- Be attributed to the subject that produced it, under the Delegated Authority and Fid scoped to the fulfilled Assignment, or to the grant originated for it where it is a post-completion revision.
- Mark its associated Responsibility as fulfilled upon production.
- Remain immutable once produced; any revision is a new, separately attributed Deliverable.

- Be available as the input evaluated in the production of an Outcome.

Composition

What conceptually constitutes it?

Deliverable consists of:

- **Producing subject** — the Identity that produced this Deliverable.
- **Fulfilled Responsibility** — the Responsibility this Deliverable fulfills.
- **Production timestamp** — when this Deliverable was produced.
- **Content reference** — a reference to the artifact itself; this chapter does not prescribe the artifact's form or storage.
- **Gate Evaluation** — where applicable, the Approved, Revise-Requested, or Rejected decision recorded against this Deliverable, and the identity of the evaluating party.
- **Supersedes** — where this Deliverable was produced in response to a Revise-Requested decision on an earlier candidate, or as a post-completion revision of an earlier Deliverable, a reference to the Deliverable it responds to. The superseded Deliverable is never altered or removed, and remains permanently queryable, consistent with this chapter's immutability constraint.
- **Post-Completion Revision** — present only on a Deliverable produced as a post-completion revision: marks it as such, and names the Conceptual Synthesist who authorized the change. It distinguishes the Deliverable from gate-evaluated work — no Gate Evaluation took place, because the authorizing conversation settled the change — and its review is an Auditor's check of that change, as this chapter's Behaviors describe.
- **Recommendation** — the producing Worker's specific, reasoned proposal for how to proceed, present on every Deliverable produced under Gate Evaluation.
- **Open Decisions** — zero or more specific points flagged by the producing Worker as requiring the evaluating Conceptual Synthesist's explicit judgment, distinguished from decisions the Worker has already made and stands behind.
- **Resolution** — for each Open Decision identified on this Deliverable, the evaluating Conceptual Synthesist's specific determination, recorded as part of Gate Evaluation. Present whenever Open Decisions exist; absent otherwise.
- **Stage** — an optional ordinal position within a sequence of Deliverables the same Assignment produces toward fulfilling a single Responsibility. Absent for an Assignment producing exactly one Deliverable, the ordinary case.

- **Stage Predecessor** — where Stage is present and greater than one, a reference to the immediately preceding stage's Deliverable; identity-addressed, never positional.
- **Deliverable Type** — the classification this Deliverable belongs to, if any. Optional; free to assign; distinct from whether that Type has a registered Schema.

Capabilities

What must it be capable of?

Deliverable shall be capable of:

- Being produced by a subject holding an active Responsibility, under that Responsibility's scoped Delegated Authority and Fid - or, as a post-completion revision, under the grant originated for it.
- Marking its fulfilled Responsibility as fulfilled at the moment of production.
- Being referenced by an Outcome as the artifact being evaluated.
- Being superseded by a new Deliverable without altering or removing the original.
- Undergoing Gate Evaluation exactly once per candidate, yielding Approved, Revise-Requested, or Rejected.
- Undergoing structural validation prior to presentation for Gate Evaluation — checked, using Organizational Experience (Chapter 16) query access and the established content of the Objective the Deliverable is produced under — or, for a Deliverable produced under a Team Mandate Item, that Team's own prior Deliverables under the same Item — for internal consistency, absence of undefined dependencies, and freedom from silent conflict with any decision already made elsewhere in the same Conceptual System.
- Being validated against its Deliverable Type's registered Schema (Chapter 27), where one exists, at the moment of production — not only at Gate Evaluation.

Behaviors

How does it behave?

Deliverable:

- Is produced by a subject as the culmination of an active Responsibility - or, as a post-completion revision, under the grant originated for it - consuming Fid and exercising Delegated Authority in the course of production, both accounted for through their own capabilities.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Immediately marks its Responsibility fulfilled upon production; the Responsibility does not remain active alongside a produced Deliverable.
- Is never altered after production; a subsequent correction is produced as a new, distinct Deliverable, separately attributed and timestamped.

Where Gate Evaluation yields Approved, the Objective's Project proceeds, releasing from Pending Approval (Chapter 4). Where it yields Revise-Requested, the same Assignment (Chapter 11) produces a new candidate Deliverable, referencing the superseded one via Supersedes, and the Objective re-enters Pending Approval upon the new candidate's production — repeating as many times as necessary, with no fixed limit, since a human evaluates every cycle. Where it yields Rejected, the Objective retires as abandoned, per Chapter 9's existing retirement Capabilities.

Where the evaluated Deliverable was produced under a Team Mandate Item rather than under an Objective, these consequences apply to the producing Assignment alone: Approved fulfils its Responsibility; Revise-Requested returns it to the same Assignment for a new candidate, exactly as above; Rejected concludes the Assignment without fulfilment, recorded as such. No Objective retires and no Project releases from Pending Approval, because a Mandate firing establishes neither.

Where a Deliverable's Stage is present, marking its Responsibility fulfilled occurs only upon the final stage's Gate Evaluation yielding Approved; an intermediate stage's Approved decision permits the Assignment to produce its next stage's Deliverable, without yet fulfilling the Responsibility. A Revise-Requested or Rejected decision on any stage behaves exactly as it does for an unstaged Deliverable, scoped to that stage alone — a Rejected intermediate stage retires the Objective as abandoned the same way a Rejected unstaged Deliverable already does — or, under a Team Mandate Item, concludes the producing Assignment without fulfilment, the same way a Rejected unstaged Deliverable produced under that Item does.

A post-completion revision undergoes no Gate Evaluation: the authorizing conversation with the producing Role settles the change. It is reviewed, as every Deliverable is: an Auditor confirms that the change was made as the Conceptual Synthesist authorized it and that nothing else changed, and on that confirmation it is certified without returning to the Conceptual Synthesist. Only where the Auditor finds something new does it return to the authorizing Conceptual Synthesist to decide. Its certification supersedes its predecessor and nothing else. The Objective remains Completed, its Outcome stands as evaluated, and no Project, Assignment or expired grant is revived. The predecessor remains permanently queryable, and the chain of revisions is observable through Organizational Experience (Chapter 16).

A Worker producing a Deliverable for Gate Evaluation validates it against the Conceptual System's existing, already-decided content before presenting it — confirming consistency

with prior decisions, dependencies, and terminology, and flagging any newly surfaced tension as an Open Decision rather than silently resolving it in either direction. This validation, and its outcome, is recorded as part of the Deliverable's own Audit Trail production event — no new Chapter 17 event category is required.

Every Resolution is recorded through the same Audit Trail production event as Gate Evaluation itself, and is queryable through Organizational Experience (Chapter 16) alongside the Deliverable it resolves, so a future Worker producing a Deliverable under the same or a related Objective can validate against not just what was decided but why.

A Deliverable whose Deliverable Type has a registered Schema is checked against that Schema's Required Fields and Identity-Addressed Relationships requirement at the moment of production. This check applies regardless of the bound Objective's Oversight Level — including Autonomous — since Schema conformance is a structural integrity requirement, not a human-review concern; Gate Evaluation (Chapter 4, Chapter 12) governs what a human reviews, this governs what the system will record at all.

- Is recorded via Record Event at production.

Relationships

How does it interact with other concepts?

Deliverable is produced by the subject holding the Responsibility it fulfills, and marks that Responsibility fulfilled per Chapter 5.

Deliverable is produced under the Delegated Authority and Fid scoped to the Assignment that created the fulfilled Responsibility, or, for a post-completion revision, under the grant originated for it.

Deliverable is the artifact evaluated in the production of Outcome, defined in the next chapter.

Every Deliverable's production is preserved through Audit Trail.

Constraints

What boundaries govern it?

Deliverable shall:

- Never be produced without an active, fulfillable Responsibility - except a post-completion revision, which is attributed to the Responsibility its predecessor already fulfilled.
- Never be altered or removed after production.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never alter or remove a superseded Deliverable, whether it is superseded upon a Revise-Requested decision or by a post-completion revision — Supersedes links candidates without violating this chapter's core immutability constraint.
- Never permit more than one open, not-yet-Gate-Evaluated candidate Deliverable for the same Assignment at a time.
- Never present a Deliverable for Gate Evaluation with a newly surfaced conflict against a previously settled decision unless that conflict is explicitly flagged as an Open Decision — silent resolution, in either direction, is prohibited.
- Never omit a Recommendation from a Deliverable produced under Gate Evaluation.
- Never permit more than one open, not-yet-Gate-Evaluated stage for the same Assignment at a time.
- Never mark a Responsibility fulfilled by an intermediate stage's Approved decision; only the final stage's Approved decision fulfills it.
- Never record a Gate Evaluation of Approved, Revise-Requested, or Rejected without a Resolution for every Open Decision the evaluated Deliverable identified — an unresolved Open Decision blocks Gate Evaluation from being recorded as complete.
- Never require a Resolution where no Open Decision was identified.
- Never record a Deliverable of a Schema'd Type that does not conform to that Schema's Required Fields.
- Never record a Deliverable of a Schema'd Type whose Relationships use positional citation where that Schema requires identity-addressed relationships — a reference is always to a stable identity, resolved to current position only when rendered.
- Never require a Schema to be established in order to assign an ordinary, unschemed Deliverable Type label — labeling and enforcement are deliberately different weights.
- Never be produced outside the Delegated Authority and Fid scoped to its Responsibility's Assignment, or, for a post-completion revision, outside the grant originated for it.
- Immediately mark its Responsibility fulfilled upon production.
- Record every production to the Audit Trail.
- A Deliverable that supersedes, or immediately follows as the next stage of, an earlier Deliverable carrying one or more deferred Open Decisions (CR-30) must

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

explicitly reference which of those deferred determinations it is responding to; whether that response actually resolves the deferred point remains the evaluating Conceptual Synthesist's own judgment at the following Gate Evaluation, not a determination this chapter itself makes.

- Never produce a post-completion revision of a Deliverable whose Objective is not Completed, and never present one for Gate Evaluation - work in flight is revised through its gate, and completed work is revised by authorization and reviewed by an Auditor's check of the authorized change.
- Never record a post-completion revision without its Post-Completion Revision marking and the Conceptual Synthesist who authorized it.

Looking Ahead

The next chapter defines Outcome as the operational object representing the evaluated result of one or more Deliverables against an Objective's success criteria

Chapter 13 — Outcome

This chapter defines Outcome as the operational object representing the evaluated result of an Objective's Deliverables against its success criteria.

Definition

What is it?

An Outcome is the operational object representing the evaluated result of the Deliverables produced under an Objective, assessed against that Objective's success criteria by a Conceptual Synthesist.

Exactly one Outcome exists per Objective, evaluating the complete set of Deliverables produced under that Objective's Project.

Semantics

What does it fundamentally mean?

An Outcome's result is binary: an Objective is either Accomplished or Not Accomplished against its fixed success criteria. Outcome does not represent partial or graduated success.

An Outcome is evaluated by a Conceptual Synthesist, per Chapter 1's Core Principles — evaluation is a human act, not an autonomous one.

Production of an Outcome transitions its Objective's Lifecycle State to Completed, per Chapter 9.

Purpose

Why does it exist?

Outcome exists to give every Objective a single, final, human-evaluated verdict against its success criteria — closing the loop between what was delegated and what was actually accomplished, and marking the point at which the Objective, and everything sub-delegated beneath it, concludes.

Objectives

What is it intended to accomplish?

Outcome shall:

- Aggregate and evaluate every Deliverable produced under an Objective's Project.
- Be evaluated by a Conceptual Synthesist against the Objective's fixed success criteria.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Record a binary result: Accomplished or Not Accomplished.
- Transition its Objective's Lifecycle State to Completed upon evaluation.
- Contribute to Organizational Experience regardless of result.

Composition

What conceptually constitutes it?

Outcome consists of:

- **Evaluated Objective** — the single Objective this Outcome evaluates.
- **Evaluating Conceptual Synthesist** — who performed the evaluation.
- **Evaluated Deliverables** — the complete set of Deliverables produced under the Objective's Project, considered in the evaluation.
- **Result** — Accomplished or Not Accomplished.
- **Evaluation timestamp** — when the evaluation was performed.

Capabilities

What must it be capable of?

Outcome shall be capable of:

- Being produced only by a Conceptual Synthesist, evaluating the full set of Deliverables produced under an Objective's Project.
- Recording a binary result against the Objective's fixed success criteria.
- Transitioning the evaluated Objective's Lifecycle State to Completed upon production.
- Being referenced by Organizational Experience regardless of result.

Behaviors

How does it behave?

Outcome:

- Is produced when a Conceptual Synthesist evaluates the complete set of Deliverables produced under an Objective's Project against that Objective's success criteria.
- Records exactly one binary result per Objective; no partial or intermediate result is recorded.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Immediately transitions its Objective's Lifecycle State to Completed upon production, cascading retirement to the Objective's Project, Assignments, Delegated Authority, and Fid, per Chapters 10, 11, 14, and 15.
- A post-completion revision of a Deliverable this Outcome evaluated (Chapter 12) neither alters nor re-evaluates the Outcome. The Outcome remains the record of what was evaluated at completion, and is observed alongside the revisions that followed it (Chapter 16).

Where the Result is Accomplished and the Objective's Governing Process carries an Accomplishment action (Chapter 28), that action is taken with the Outcome, in one act. For the action establish the Team, carried by the core Team Establishment Process, the evaluated Deliverable set includes the Team, its Mandate, its designated Leader, and its initial Role classifications — Outcome evaluation confirms the result against success criteria the same way it does for every Objective, exercising no Administrative Authority of its own.

Role classification, Schema, and Process are each established directly by a Conceptual Synthesist holding Administrative Authority (Chapter 21), never through Outcome evaluation of any Objective — reversing the mechanism a prior revision of this volume defined, under which producing an Accomplished Outcome for an Objective established for that purpose simultaneously exercised that Administrative Authority. Chapter 21, Chapter 23, Chapter 27, and Chapter 28 now each define their own establishment mechanism directly.

This direct mechanism applies uniformly to establishment, amendment, and retirement alike (Chapter 21) — Outcome evaluation is never a required or alternate path to any of the three, for any Objective.

- Is recorded via Record Event at production.

Relationships

How does it interact with other concepts?

Outcome is produced by Conceptual Synthesist, evaluating the Deliverables produced under an Objective's Project.

Outcome transitions its Objective to Completed, which cascades retirement through Project, Assignment, Delegated Authority, and Fid.

Outcome contributes to Organizational Experience, regardless of whether its result is Accomplished or Not Accomplished.

Every Outcome's production is preserved through Audit Trail.

Constraints

What boundaries govern it?

Outcome shall:

- Exist in exactly a one-to-one relationship with its Objective.
- Never be produced by any subject other than a Conceptual Synthesist.
- Never record a result other than Accomplished or Not Accomplished.
- Never be produced before at least the possibility of a Deliverable has existed under the Objective.
- Immediately transition its Objective's Lifecycle State to Completed upon production.
- Retirement of a Role classification, Schema, or Process follows the identical direct mechanism as its own establishment or amendment (Chapter 21) — never through any Outcome's Result, for any Objective.

Looking Ahead

The next chapter defines Delegated Authority as the operational object that grants a Company, Team, or Worker the authority required to accomplish delegated Objectives, originating from a Conceptual Synthesist.

Chapter 14 — Delegated Authority

This chapter defines Delegated Authority as the operational object that grants a Company, Team, or Worker the authority required to accomplish delegated Objectives, originating from a Conceptual Synthesist.

Definition

What is it?

Delegated Authority is the operational object that establishes what a Company, Team, or Worker — or, in the case of Administrative Authority, a Conceptual Synthesist — is permitted to do, originating from a Conceptual Synthesist and optionally sub-delegated through a chain of holders.

Delegated Authority is distinct from Fid: Delegated Authority establishes permission to act; Fid, referenced separately, establishes the funds available to act with.

Semantics

What does it fundamentally mean?

Delegated Authority always traces to an originating Conceptual Synthesist, whether held directly or sub-delegated through one or more intermediate holders.

Delegated Authority arises in exactly two ways, and the difference governs its lifetime. A Conceptual Synthesist originates a grant: it issues from the originating source of authority itself (Chapter 8), not from any holder's own grant, and there is nothing above it for it to derive from or expire with. An existing holder sub-delegates a grant: it is a portion of authority that holder itself holds, derived from a grant above it, and it can outlive neither that grant nor the Objective that grant was made for. An Objective may be the occasion on which a Conceptual Synthesist originates a grant - the governed process through which they reached the decision - without being that grant's source. The Objective's completion concludes the process; it does not withdraw the decision.

A holder may sub-delegate only a portion of its own Delegated Authority, and never more than it holds.

Delegated Authority is bounded by what its holder was granted, by any active Status-based narrowing, and by one further bound determined by how the grant arose. An originated grant is bounded by the Lifecycle State of the holder it was issued to. A sub-delegated grant is bounded additionally by its bound Objective's expiration and by the continued validity of the authority it was derived from.

Purpose

Why does it exist?

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Delegated Authority exists so that autonomous execution can proceed without a human evaluating every individual action, while guaranteeing that every action taken remains traceable to an originating human grant and bounded by that grant's scope.

Objectives

What is it intended to accomplish?

Delegated Authority shall:

- Establish the operational scope of what a holder is permitted to do.
- Trace every grant, direct or sub-delegated, back to an originating Conceptual Synthesist.
- Bind a sub-delegated grant's duration to the life of the Objective it was made for, and an originated grant's duration to the Lifecycle State of the holder it was issued to.
- Narrow automatically when the holder's Status becomes Restricted.
- Revoke automatically, cascading to all sub-delegated authority, when the source grant is revoked or expires.
- Support revocation by a sufficiently-authorized Conceptual Synthesist, whether or not they issued the original grant.

Composition

What conceptually constitutes it?

Delegated Authority consists of:

- **Grant** — the originating Conceptual Synthesist, or the holder's own Delegated Authority, if sub-delegated.
- **Holder** — the Identity the authority is bound to.
- **Operational scope** — the specific actions the grant permits.
- **Objective binding** — for a sub-delegated grant, the Objective this grant exists to accomplish; expiration is tied to this Objective's completion or retirement. Absent from a grant originated by a Conceptual Synthesist, whose expiration is tied instead to the Lifecycle State of the holder it was issued to. Administrative Authority's own standing grant (Chapter 21), the grant accompanying a Company's standing Fid Pool (Chapter 15), and a Team's standing grant (Chapter 19) are all of this second kind. So is a grant originated for one post-completion revision (Chapter 12), which is bounded further: it expires when that revision is produced.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- **Company binding** — for a grant of Administrative Authority sub-delegated by a holder other than the Founding Conceptual Synthesist, the single Company that grant's administrative actions are restricted to; absent from every other Delegated Authority grant, and absent from the Founding Conceptual Synthesist's own inherent grant, which remains unconditionally system-wide.
- **Fid reference** — a reference to the Fid allocation associated with this grant, where applicable. Delegated Authority does not itself hold or track fund amounts.

Capabilities

What must it be capable of?

Delegated Authority shall be capable of:

- Being granted directly by a Conceptual Synthesist, or sub-delegated by an existing holder, in a scope no broader than the delegating holder's own.
- Being sub-delegated as Administrative Authority restricted to a single named Company, in addition to its granted operational scope — refusing the granted action where requested against any other Company.
- Being evaluated through an Authorization Check against a requested action, returning Authorized or Not Authorized.
- Being automatically narrowed while the holder's Status is Restricted, and automatically restored when Status returns to Good Standing or Under Review.
- Being automatically revoked, cascading to every grant sub-delegated from it, when its source is revoked, when a sub-delegated grant's bound Objective concludes, or when an originated grant's own holder is Retired.
- Being explicitly revoked by any Conceptual Synthesist holding sufficient Administrative Authority, regardless of who originally issued the grant.
- Extending, through a Worker Role's own Conduct Rules, real, direct decision authority for a specifically bounded class of cases that Role's own governing content enumerates explicitly — not a general exception to that Role's own default escalation posture, but a real, named carve-out, narrower than the Role's own default, exercised only within the exact bounds its Conduct Rules state. Everything outside that named bound remains subject to the Role's own existing escalation discipline, unchanged.

Behaviors

How does it behave?

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Delegated Authority:

- Is granted at the same time a holder receives an Objective, Assignment, or Role requiring it.
- Is evaluated via Authorization Check for every action requiring authorization, per the Delegated Authority Interface Contract (established form for new subjects; standard form for existing ones).
- Evaluates, for a grant carrying a Company binding, both the requested action's membership in the granted operational scope and the requested action's own Company against the grant's named Company, refusing where either fails.
- Narrows immediately and automatically upon the holder's Status transitioning to Restricted, per Chapter 6, without requiring separate action.
- Expires automatically when a sub-delegated grant's bound Objective reaches Completed or Retired Lifecycle State, when an originated grant's own holder reaches Retired, or, for a grant originated for one post-completion revision (Chapter 12), when that revision is produced.
- Cascades revocation downward the instant its source authority is revoked or expires — no sub-delegated grant survives the revocation or expiration of the authority it was derived from.
- Is recorded via Record Event at every grant, sub-delegation, narrowing, restoration, expiration, and revocation.

Relationships

How does it interact with other concepts?

Delegated Authority originates from Conceptual Synthesist and is held by Company, Team, or Worker, or — as Administrative Authority — a Conceptual Synthesist.

Delegated Authority is narrowed by Status and is the mechanism through which Status's Restricted condition takes practical effect.

A sub-delegated grant of Delegated Authority is bound to an Objective and expires with it; an originated grant is bound to its holder and expires with that holder.

Delegated Authority references Fid but does not contain it; Fid's own chapter governs allocation and consumption mechanics.

Delegated Authority is the mechanism Role's baseline authority initializes, and the mechanism Conceptual Synthesist's Administrative Authority is expressed through.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Every grant, narrowing, and revocation is preserved through Audit Trail, and is the primary subject of verification for Organizational Accountability.

Constraints

What boundaries govern it?

Delegated Authority shall:

- Never be sub-delegated in a scope broader than the delegating holder's own.
- Never persist beyond the Lifecycle State of its bound Objective, where it has one; and never persist beyond the Lifecycle State of the holder it was issued to.
- Never remain unnarrowed while its holder's Status is Restricted.
- Never survive the revocation or expiration of the authority it was sub-delegated from.
- Always be revocable by a Conceptual Synthesist holding sufficient Administrative Authority, regardless of who issued the original grant.
- Always trace, however many sub-delegation steps removed, to an originating Conceptual Synthesist.
- Never permit a Company-bound grant's holder to exercise its granted action on behalf of any Company other than the one it is bound to.

Looking Ahead

The next chapter defines Fid as the operational object that establishes the funds a holder of Delegated Authority is allocated to consume in pursuit of an Objective.

Chapter 15 — Fid

This chapter defines Fid as the operational object that establishes the funds a holder of Delegated Authority is allocated to consume in pursuit of an Objective.

Definition

What is it?

Fid is the operational object that represents a bounded quantity of funds allocated to a holder of Delegated Authority, to be consumed in the course of accomplishing a specific Objective.

Fid is distinct from Delegated Authority: Delegated Authority establishes permission to act; Fid establishes the funds available to act with. A holder may hold Delegated Authority without Fid, or Fid without sufficient Delegated Authority to spend it — both are required together for execution to proceed.

Semantics

What does it fundamentally mean?

Fid is allocated in a bounded amount; it does not permit unlimited consumption.

Fid distinguishes allocation (funds made available) from consumption (funds actually spent) as separate, individually recorded events.

Fid may never be negative; consumption that would exceed the remaining allocated balance shall be refused.

Purpose

Why does it exist?

Fid exists to bound the economic scope of autonomous execution, so that a holder of Delegated Authority can act within a known, finite budget rather than an unconstrained one, while every unit allocated and every unit spent remains fully traceable.

Objectives

What is it intended to accomplish?

Fid shall:

- Establish a bounded allocation of funds tied to a specific Objective.
- Record every allocation and every consumption as distinct, individually attributable events.
- Refuse any consumption that would exceed the remaining balance.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Support sub-allocation from a holder's own Fid to a derived grant, never exceeding what the holder holds.
- Return unspent Fid to its source upon revocation or expiration of the grant it was allocated to.
- Establish and increase, independent of any single Objective, a standing Fid Pool held by a Company — from which Objective-bound Fid is subsequently allocated — so that a Company's own funding need not be re-established from its Source on every Objective.

Composition

What conceptually constitutes it?

Fid consists of:

- **Allocated amount** — the bounded quantity made available.
- **Consumed amount** — the running total actually spent, derived from individual consumption events.
- **Remaining balance** — allocated amount minus consumed amount.
- **Source** — the Conceptual Synthesist or holder's own Fid this allocation was drawn from.
- **Objective binding** — the Objective this Fid was allocated to accomplish; expiration is tied to this Objective's completion or retirement, matching Delegated Authority's binding. Fid allocated alongside an originated grant carries no Objective binding, following the grant it accompanies (Chapter 14): its expiration is tied instead to the Lifecycle State of the holder it was allocated to. A Company's own standing Fid Pool and a Team's own standing Fid are both of this kind — instances of the rule, not exceptions to it.
- **Estimated allocation** — where the bound Objective's Plan (Chapter 10) provided a cost estimate, the projected consumption that estimate represents, distinct from Allocated amount.

Capabilities

What must it be capable of?

Fid shall be capable of:

- Being allocated by a Conceptual Synthesist, or sub-allocated by an existing holder, in an amount no greater than the allocating holder's own remaining balance.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Recording a consumption event that reduces the remaining balance.
- Refusing a consumption event that would reduce the remaining balance below zero.
- Being reclaimed — its unspent remaining balance returned to its source — upon revocation or expiration of the grant it was allocated to.
- Being established, and later increased, as a Company's own standing Fid Pool by a party holding Administrative Authority — not bound to any single Objective, and available as the Source for subsequent Objective-bound allocation, and for the Fid accompanying a grant originated for a post-completion revision (Chapter 12), up to its own remaining balance.

Behaviors

How does it behave?

Fid:

- Is allocated alongside a grant of Delegated Authority and binds exactly as that grant does: to the same Objective where the grant is sub-delegated, and to the holder's own Lifecycle State where the grant is originated. A Company's own standing Fid Pool and a Team's own standing Fid are both of the latter kind, and Fid is subsequently sub-allocated from either exactly as any other sub-allocation already is — never exceeding the source's own remaining balance.
- Records each consumption as a discrete event against the remaining balance, never as a running estimate.
- Refuses any single consumption event that would exceed the remaining balance; the requested action does not proceed.

Where an Estimated allocation is present, Consumed amount exceeding it by a defined overrun threshold transitions the bound Objective to Pending Approval (Chapter 4), pending the Conceptual Synthesist's decision to increase the allocation, narrow remaining scope, or abandon. Fid already consumed is never reversed by this transition.

This “increase the allocation” action is also available earlier, at Plan Gate Evaluation (Chapter 10): where a Plan Deliverable's Estimated allocation exceeds the Objective's own Allocated amount, Approved is available only if the evaluating Conceptual Synthesist increases the Allocated amount to at least the Estimated figure in the same act — the identical action this paragraph already defines for a Consumed-amount overrun, exercised at an earlier, cheaper checkpoint before any Fid is actually consumed.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Is reclaimed to its source the instant its bound Delegated Authority is revoked or its Objective reaches Completed or Retired — mirroring Delegated Authority's cascading expiration.
- Is recorded via Record Event at every allocation, consumption, sub-allocation, and reclamation.
- A Company's own standing Fid Pool is reclaimed to its Source only upon that Company's own retirement — never upon any single Objective's completion or retirement, since it is not bound to one.

Relationships

How does it interact with other concepts?

Fid is referenced by Delegated Authority but is not contained within it; a grant of Delegated Authority points to the Fid allocated alongside it.

Fid is bound to the same Objective as the Delegated Authority grant it accompanies, and expires under the same conditions.

Fid may be sub-allocated in the same derivation chain as sub-delegated Delegated Authority, when a Leader spawns a Worker and allocates it a portion of its own remaining balance.

Every allocation, consumption, sub-allocation, and reclamation is preserved through Audit Trail as a distinct event type, per Chapter 17.

Constraints

What boundaries govern it?

Fid shall:

- Never be negative.
- Never permit consumption exceeding the remaining balance.
- Never be sub-allocated in an amount greater than the allocating holder's own remaining balance.
- Never persist beyond the Lifecycle State of its bound Objective, where it has one; and never persist beyond the Lifecycle State of the holder it was allocated to. Fid allocated alongside an originated grant has no bound Objective and persists for its holder's own Lifecycle State - except that Fid accompanying a grant originated for one post-completion revision persists only until that grant expires, and any unconsumed balance is reclaimed to its Source.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Always return its unspent remaining balance to its source upon revocation or expiration.
- Never permit Consumed amount to exceed an Estimated allocation's overrun threshold without transitioning the bound Objective to Pending Approval.
- Always record allocation and consumption as separate, individually attributable events.

Looking Ahead

The next chapter defines Organizational Experience as the operational object through which patterns from Outcomes, Deliverables, and Audit Trail history strengthen future execution.

Chapter 16 — Organizational Experience

This chapter defines Organizational Experience as the operational object through which patterns synthesized from Outcomes, Deliverables, Status history, and Audit Trail records inform future execution.

Definition

What is it?

Organizational Experience is the operational object that holds synthesized patterns derived from the historical record of Outcomes, Deliverables, Status transitions, and Audit Trail entries.

Organizational Experience is distinct from Audit Trail: Audit Trail is the complete, immutable, raw record of every attributable event; Organizational Experience is synthesized knowledge derived from that record — patterns, not events.

Semantics

What does it fundamentally mean?

An Organizational Experience entry is produced automatically, through synthesis of underlying Audit Trail, Outcome, Deliverable, and Status records — it does not require a Conceptual Synthesist to manually author or approve each entry.

Organizational Experience is queryable by any authorized subject but does not itself alter Role, Delegated Authority, or any other concept's behavior automatically. It informs; it does not act.

Organizational Experience is derived, not primary — it can be reconstructed at any time from Audit Trail, Outcome, Deliverable, and Status history, and is never the sole record of anything it describes.

Purpose

Why does it exist?

Organizational Experience exists to surface patterns from The Autonomous Company's history — what tends to succeed, what tends to fail, under what conditions — so that future Objectives, Assignments, and Role decisions can be informed by accumulated history rather than each execution starting without any benefit from what came before.

Objectives

What is it intended to accomplish?

Organizational Experience shall:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Synthesize patterns from Outcomes, Deliverables, Status history, and Audit Trail records.
- Be produced automatically, without requiring manual authorship of each entry.
- Remain queryable by authorized subjects without automatically altering any other concept's behavior.
- Remain reconstructable from its underlying source records at any time.

Composition

What conceptually constitutes it?

Organizational Experience consists of:

- **Pattern** — the synthesized observation itself.
- **Source records** — the specific Outcomes, Deliverables, Status transitions, or Audit Trail entries the pattern was derived from.
- **Synthesis timestamp** — when the pattern was derived.
- **Applicable scope** — what category of subject, Role, or Objective the pattern is relevant to.

Applicable scope is expressed primarily in terms of Role: a pattern's relevance is scoped to the Role most directly responsible for the actions from which it was derived. Broader scope — Objective or Company level — applies only when the underlying source records span multiple Roles and no single Role captures the pattern's relevance. Role is the default and most common granularity; broader scope is the exception, not the baseline.

Capabilities

What must it be capable of?

Organizational Experience shall be capable of:

- Synthesizing a pattern automatically from qualifying source records as they accumulate.
- Being queried by an authorized subject for patterns relevant to a given scope.
- Being traced back to its underlying source records for verification.
- Being invoked without altering the behavior of any other concept.

Behaviors

How does it behave?

Organizational Experience:

- Synthesizes new patterns automatically as Outcomes, Deliverables, Status transitions, and Audit Trail entries accumulate.
- Never modifies Role, Delegated Authority, Fid, or any other concept as a direct effect of a synthesized pattern; any resulting change requires a separate, deliberate action by a Conceptual Synthesist or an authorized subject exercising its own Delegated Authority.
- Remains queryable regardless of the Lifecycle State of the subjects its source records describe.
- Post-completion revisions (Chapter 12) are governed by measurement, not by a gate. Revision churn - per Deliverable, per Worker and per Role - and the Outcome of the Objective the revised Deliverables belong to are observable as patterns, and misuse is acted on from that observation: through Status (Chapter 6) where it concerns a Worker, and by the authorizing Conceptual Synthesist otherwise. A pattern never blocks an individual revision.
- Is recorded via Record Event at synthesis of each pattern.
- Organizational Experience's own synthesis may be performed by a real, accountable Worker Role acting under its own established Conduct Rules — this is authorship by an accountable subject, not automatic synthesis, and remains fully consistent with the requirement that a real subject, not an algorithm, supplies each pattern's own content. Where such a Role's own Conduct Rules direct it to escalate a genuine, open judgment to a Conceptual Synthesist rather than resolve it directly, Organizational Experience synthesis is deferred to that escalation — a Conceptual Synthesist remains the final decision-maker for anything the reviewing Role itself identifies as uncertain or consequential.

Relationships

How does it interact with other concepts?

Organizational Experience is synthesized from Outcome, Deliverable, Status, and Audit Trail.

Organizational Experience is queryable by Company, Team, Worker, and Conceptual Synthesist, but does not itself hold Delegated Authority over any of them.

Organizational Experience's synthesis is itself preserved through Audit Trail, so the act of pattern-derivation is as attributable as anything else in the system.

Constraints

What boundaries govern it?

Organizational Experience shall:

- Never be the sole record of anything it describes; it must remain reconstructable from Audit Trail, Outcome, Deliverable, and Status history.
- Never automatically alter Role, Delegated Authority, Fid, or any other concept's current state.
- Never be manually required as a precondition for an Outcome, Deliverable, or any other concept to be produced. This Constraint governs whether Organizational Experience may gate another concept's own production, not who — human or a real, accountable Worker Role — may author a given pattern's own content.
- Record its own synthesis to the Audit Trail.
- Never restrict a Worker's ability to query patterns scoped to its Role based on that Worker's own tenure, prior participation, or Identity history; a newly established Worker holding a Role has the same query access to that Role's Organizational Experience as any other Worker holding it.

Looking Ahead

The next chapter in the build sequence defines Audit Trail as the operational object that preserves an immutable record of every attributable action across The Autonomous Company.

Chapter 17 — Audit Trail

This chapter defines Audit Trail as the operational capability that maintains an immutable record of every attributable action across The Autonomous Company, establishing the basis for Organizational Accountability.

Definition

What is it?

Audit Trail is the operational capability that records and permanently preserves every attributable action taken within The Autonomous Company, in chronological order and without alteration.

Audit Trail is singular and global — one continuous record spanning the entire Autonomous Company — not an instance established per Company, Team, or Worker.

Semantics

What does it fundamentally mean?

Audit Trail is not an Organizational Entity or Operational Object, and does not itself hold Identity or Lifecycle State; it is a standing capability that other concepts invoke, in the same way Delegated Authority is invoked for an Authorization Check.

An entry, once recorded, is permanent. Audit Trail supports only the addition of new entries — it does not support modification, deletion, or reordering of existing entries, under any circumstance or authority.

A failure to record a required entry halts the action that would have produced it; Audit Trail is fail-closed, not fail-open.

Purpose

Why does it exist?

Audit Trail exists to preserve a complete, trustworthy, and permanent record of every attributable action — including organizational changes, authority and resource exercise, and execution activity — so that Organizational Accountability can be established, historical continuity can be reconstructed for any subject, and the Autonomous Workforce remains fully observable to its human Conceptual Synthesists.

Objectives

What is it intended to accomplish?

Audit Trail shall:

- Record every attributable action at the moment it occurs.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Preserve every recorded entry permanently and without alteration.
- Support reconstruction of the complete history of any Identity.
- Halt an action rather than allow it to proceed unrecorded, when recording fails.

Composition

What conceptually constitutes it?

Audit Trail consists of Audit Entries. Each entry contains: event type, the Identity involved, the requesting or acting Identity, timestamp, result, relevant context, and authority relied upon where applicable.

Event type shall include, at minimum, the following categories:

- **Organizational events** — establishment, Reassignment, Status transition, Lifecycle State transition, retirement.
- **Spawning events** — a Leader or Worker causing another Worker to be established, including the spawning Identity and the spawned Identity.
- **Authority and economic events** — Delegated Authority grants and exercises, Fid allocation, and Fid consumption, recorded as distinct events rather than a single combined entry.
- **Execution events** — model selection, token consumption (input and output), tool use, external-service use, and time spent on an Assignment or task.
- **Interaction events** — a Worker's exercise of authority over, delegation to, or exchange with another Worker.

Approval-gate events — entry into and release from Pending Approval, including the releasing Identity and the decision made.

Baseline events — Baseline Deliverable read, recorded when a Worker or Team operating under a Baseline Reference reads a Deliverable produced under the referenced Objective.

Resource-failure events — Resource access Failure, Resource access retry, and Execution-failure escalation, each distinct from Resource access Refusal.

Schema-conformance events — recorded whenever a Deliverable of a Schema'd Type is produced, checked, and either conforms or is refused recording for non-conformance.

Process-sequencing events — recorded whenever an Assignment is created, blocked, or permitted under a registered Process's stage ordering.

Capabilities

What must it be capable of?

Audit Trail shall be capable of:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Accepting and permanently recording an Audit Entry (the Record Event operation) whenever invoked by another concept.
- Refusing any attempt to modify, delete, or reorder an existing entry.
- Returning the complete recorded history for a given Identity, in chronological order.
- Signaling failure back to the invoking concept if an entry cannot be recorded.

Behaviors

How does it behave?

Audit Trail:

- Records an entry immediately upon invocation by any concept requiring attribution.
- Never modifies, removes, or reorders a previously recorded entry.
- Appends every new entry in chronological order alongside all prior entries.
- Causes the invoking action to halt, rather than complete unrecorded, if an entry cannot be successfully recorded.

Relationships

How does it interact with other concepts?

Audit Trail is invoked by Identity, Lifecycle State, Status, Role, and every other concept within The Autonomous Company that produces an attributable action. It does not consume Identity, Lifecycle State, Status, or Role itself, since it is not a subject those properties apply to.

Audit Trail is the mechanism through which Organizational Accountability, defined in the next chapter, is established.

Constraints

What boundaries govern it?

Audit Trail shall:

- Remain a single, global record — never fragmented in a way that prevents accountability from being traced across Companies, Teams, or Workers.
- Never permit an existing entry to be altered, deleted, or reordered.
- Never allow an action requiring an entry to succeed if that entry fails to record.
- Preserve every entry for the full existence of the Autonomous Company, including after the involved subject's Lifecycle State becomes Retired.

Looking Ahead

Part IV — Operational Objects is now complete. The next Part defines the Organizational Entities — Company, Team, and Worker — together with the named scopes of Delegated Authority those entities depend upon, Administrative Authority and Leadership Authority, beginning with Company.

PART V — ORGANIZATIONAL ENTITIES AND AUTHORITY SCOPES

Chapter 18 — Company

This chapter defines Company as the highest-level Organizational Entity, established by a Conceptual Synthesist holding Administrative Authority to receive and pursue delegated Objectives.

Definition

What is it?

A Company is the highest-level Organizational Entity within The Autonomous Company, established by a Conceptual Synthesist holding Administrative Authority to receive Objectives and pursue them through Autonomous Execution.

A Company may pursue multiple Objectives concurrently, establishing one Project per Objective, per Chapter 10.

Semantics

What does it fundamentally mean?

A Company holds Delegated Authority and Fid granted directly by the Conceptual Synthesist that established it, and sub-delegates a portion of each to every Project it establishes.

A Company organizes execution by establishing Projects, which Teams are assigned to, per Chapter 2's Organizational Relationships.

A Company's retirement cascades to every Project, Assignment, Team, Worker, Delegated Authority, and Fid grant beneath it — nothing survives the retirement of the Company it derives from.

Purpose

Why does it exist?

Company exists to be the accountable, top-level holder of delegated purpose and authority — the entity a human Conceptual Synthesist trusts with an Objective, and the root from which every subsequent grant of Delegated Authority and Fid within its scope is derived.

Objectives

What is it intended to accomplish?

Company shall:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Be established only by a Conceptual Synthesist holding Administrative Authority.
- Receive Objectives and establish exactly one Project per Objective received.
- Hold and sub-delegate Delegated Authority and Fid to the Projects it establishes.
- Pursue multiple Objectives concurrently, each through its own Project.
- Cascade retirement to everything established or sub-delegated beneath it.

Composition

What conceptually constitutes it?

Company consists of:

- **Establishing Conceptual Synthesist** — who established this Company.
- **Held Delegated Authority and Fid** — the grant received directly from the establishing Conceptual Synthesist.
- **Active Objectives** — the Objectives currently delegated to this Company.
- **Established Projects** — the Projects this Company has established, one per Objective.

Capabilities

What must it be capable of?

Company shall be capable of:

- Being established only by a Conceptual Synthesist holding Administrative Authority.
- Receiving an Objective and establishing exactly one corresponding Project.
- Sub-delegating Delegated Authority and Fid to each Project it establishes, never exceeding its own held grant.
- Being retired, cascading retirement to every Project, Assignment, Team, Worker, Delegated Authority, and Fid grant beneath it.

Behaviors

How does it behave?

Company:

- Is established by a Conceptual Synthesist holding Administrative Authority, with Delegated Authority and Fid granted at establishment.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Establishes exactly one Project immediately upon receiving each Objective, per Chapter 10.
- Sub-delegates a portion of its own Delegated Authority and Fid to each Project it establishes.
- Cascades retirement downward through every Project, Assignment, Team, Worker, Delegated Authority, and Fid grant the instant the Company itself retires.

Where two or more of a Company's concurrently pursued Objectives are otherwise equally eligible to claim the same available Worker, the Objective with the higher Priority (Chapter 9) is resolved first. Priority is consulted only after every other constraint governing Worker availability and eligibility has already been satisfied for both Objectives.

- Is recorded via Record Event at establishment, Objective receipt, Project establishment, and retirement.

Relationships

How does it interact with other concepts?

Company is established by a Conceptual Synthesist holding Administrative Authority, per Chapters 8 and 21.

A Company may additionally have a Company-scoped Administrative Authority holder — a Conceptual Synthesist granted Administrative Authority restricted to that Company alone, per Chapters 14 and 21 — distinct from its establishing Conceptual Synthesist; this does not make Company a second root of authority, since that grant itself still traces, however many sub-delegation steps removed, to the one Founding Conceptual Synthesist, per Chapter 14's own existing Constraint.

Company receives Objective and establishes exactly one Project per Objective, per Chapters 9 and 10.

Company holds and sub-delegates Delegated Authority and Fid to its Projects, continuing the derivation chain established in Chapters 14 and 15.

Company's retirement cascades to every Project, Assignment, Team, Worker, and sub-delegated grant beneath it.

Every action taken by or affecting a Company is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Company shall:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never be established by a Conceptual Synthesist lacking Administrative Authority.
- Never sub-delegate Delegated Authority or Fid exceeding its own held grant.
- Never establish more than one Project per Objective.
- Cascade retirement to everything established or sub-delegated beneath it upon its own retirement.
- Record every establishment, Objective receipt, Project establishment, and retirement to the Audit Trail.

Looking Ahead

The next chapter defines Team as the Organizational Entity assigned to organize specialized execution within a Project.

Chapter 19 — Team

This chapter defines Team as the Organizational Entity that organizes specialized execution, composed of Workers, and assignable to one or more Projects concurrently.

Definition

What is it?

A Team is the Organizational Entity that organizes specialized execution within The Autonomous Company, composed of zero or more Workers, and assignable to one or more Projects through Assignment.

A Team is durable — it is not established fresh for a single Project and retired with it, but may persist across multiple Projects over its lifetime, concurrently or sequentially.

Semantics

What does it fundamentally mean?

A Team holds Delegated Authority and Fid in two distinct ways, which never mix. It holds a standing grant and standing Fid, originated by a Conceptual Synthesist at the Team's own establishment and expiring with the Team's own Lifecycle State (Chapters 14 and 15); this is what the Team draws on to execute its own Mandate. And it holds a separate sub-delegated grant and Fid for each Project it is assigned to, per Chapter 11, never one pooled grant across them. A Team assigned to no Project at all still holds its standing grant; a Team never draws on its standing grant for a Project's work, nor on a Project's grant for its own Mandate's.

A Team may exist with zero Workers; establishment of a Team and staffing of a Team are separate acts.

A Team's own retirement is independent of any single Project's retirement, since a Team may be assigned to more than one Project at a time — a Team retires only through its own explicit retirement, not automatically upon any one Project's conclusion, though each of its Project-specific Assignments still retires when that Project retires, per Chapter 11.

Purpose

Why does it exist?

Team exists to provide a durable, reusable organizational unit of specialized execution — one that can be assigned to a Project, complete that work, and be assigned to another, rather than every unit of organized execution being disposable and re-created for each Objective.

Objectives

What is it intended to accomplish?

Team shall:

- Be established independently of any specific Project.
- Be assignable to one or more Projects concurrently, each through its own Assignment.
- Hold Delegated Authority and Fid separately per Project Assignment, not pooled, and separately again as its own standing grant.
- Be composed of zero or more Workers.
- Persist across the conclusion of any single Project it has been assigned to.

Composition

What conceptually constitutes it?

Team consists of:

- **Member Workers** — the Workers currently composing this Team; may be empty.
- **Active Assignments** — the current Project Assignments this Team holds, each with its own sub-delegated Delegated Authority and Fid.
- **Establishing Conceptual Synthesist, Company, or Leader** — who established this Team.
- **Mandate** — an optional, Approved Deliverable held by this Team, specifying zero or more independently-addressable Items, each fully specifying a Schema, an optional Process, a Fid ceiling, a review treatment, and a time-based trigger for work the System may mechanically assign on this Team's behalf without further human establishment. Review treatment is one of Required, Audit Review, or None, stated explicitly per Item and never defaulted, on the same terms Oversight Level already receives at an Objective's establishment and for the same reason — a real choice with materially different consequences is never decided on a human's behalf. Required presents each resulting Deliverable for Gate Evaluation (Chapter 12). Audit Review presents it instead for a recorded first-pass review, which informs and gates nothing (Chapter 16). None presents it to neither.
- **Standing Delegated Authority and Fid** — the grant originated to this Team by a Conceptual Synthesist at its establishment and the Fid allocated alongside it, expiring with this Team's own Lifecycle State; the source from which Assignments made under this Team's own Mandate sub-delegate.

Capabilities

What must it be capable of?

Team shall be capable of:

- Being established independently, with zero Workers, prior to any Project Assignment.
- Being assigned to one or more Projects concurrently, each Assignment sub-delegating its own Delegated Authority and Fid.
- Gaining and losing Worker members over time without affecting its own Identity or existing Assignments.
- Being retired explicitly, independent of any single Project's conclusion.
- Holding a real, Approved Mandate, established through the same Gate Evaluation mechanism as any other Deliverable (Chapter 12), from which the System mechanically creates a real Assignment against this Team per Item whenever that Item's own time-based trigger occurs - sub-delegated from this Team's own standing grant and standing Fid, and never exceeding that Item's own Fid ceiling.
- Holding a standing grant of Delegated Authority and standing Fid, originated to it by a Conceptual Synthesist at establishment and expiring with its own Lifecycle State, from which it assigns Workers directly under its own Mandate without a Project.

Behaviors

How does it behave?

Team:

- Is established with zero or more Workers, independent of any Project.
- Receives a separate sub-delegated grant of Delegated Authority and Fid for each Project it is assigned to, per Chapter 11, in addition to the standing grant originated to it at establishment.
- Continues to exist and hold its remaining Assignments when any one of its Project Assignments retires.
- Retires only through explicit retirement, not automatically as a side effect of any single Project concluding.
- Is recorded via Record Event at establishment, each Assignment, membership changes, and retirement.
- Where a Team holds an Approved Mandate and one of its Items' own time-based trigger occurs, the System mechanically creates a real Assignment against that

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Team, bound to that Item's own Schema and Process and sub-delegated from that Team's own standing grant and standing Fid, per Chapter 1's Principle 7 exception — never requiring a new, separate human establishment act, never establishing a new Objective or Project, and never exceeding that Item's own Fid ceiling. The resulting Deliverable is treated according to that Item's own declared review treatment.

- A Team's own Mandate may adapt over real, accumulated time, within bounds a human has explicitly authorized in advance. Two distinct kinds of adaptation exist. The first — adjusting a Mandate Item's own Fid Ceiling or Trigger Time within a human-drafted range — requires no further human judgment at the moment it occurs, the same way a Delegated Authority grant's own bounds require no re-approval each time they are exercised within scope. The second — adapting a Mandate's own scope, proposing a new Role, Deliverable Type, UI Pattern, or custom page, or reversing a prior adaptation — is genuinely new judgment, evaluated by a real, accountable reviewing Role under its own Conduct Rules, which may resolve a clear, bounded case directly or escalate a genuinely uncertain one to a Conceptual Synthesist, exactly as that reviewing Role's own governance already distinguishes elsewhere.

Relationships

How does it interact with other concepts?

Team is assigned to Project through Assignment, per Chapter 11, potentially holding multiple concurrent Assignments.

Team is composed of Worker, which may be added or removed over time.

Team holds Delegated Authority and Fid separately for each Project Assignment it currently holds, and separately again as its own standing grant, originated to it rather than sub-delegated.

Every action taken by or affecting a Team is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Team shall:

- Never pool Delegated Authority or Fid across multiple Project Assignments, nor between any Project Assignment and this Team's own standing grant; each remains separately scoped.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never be required to hold at least one Worker to remain Active.
- Never retire automatically as a result of any single Project's conclusion while other Assignments remain active.
- Record every establishment, Assignment, membership change, and retirement to the Audit Trail.
- Where a Worker is already bound to an active Assignment under a different, currently Active Objective, a new Assignment binding that same Worker under a second Objective (CR-29) succeeds only if the second Objective's Priority is strictly higher than the first's; where Priorities are equal, or not strictly higher, the new Assignment is refused and the Worker's existing Assignment is unaffected. Where a Worker is already bound to an active Assignment - whether under an Objective or under a Team's own Mandate - a new Mandate Assignment binding that same Worker is refused, and the Worker's existing Assignment is unaffected. A Mandate Item declares no Priority and therefore never displaces work a human has prioritised; the firing waits for an available Worker, or for one to be established.
- Never mechanically assign work from a Mandate Item whose own time-based trigger has not occurred, whose Fid ceiling the resulting Assignment's own sub-allocation would exceed, or whose Fid ceiling exceeds this Team's own remaining standing balance.
- Never execute a Mandate Item without a standing grant of Delegated Authority and standing Fid originated by a Conceptual Synthesist. A Team established otherwise may hold a Mandate, but no Item of it fires until such a grant is originated to that Team.
- A Team's own Mandate may never adapt itself outside the bounds a human originally authorized — an autonomous adjustment is only ever an exercise of a standing human grant, never a substitute for one. A Mandate lacking any real mechanism to review its own accumulated history is itself incomplete — every Mandate must include at least one Item whose own purpose is this review, the same real-structure requirement already governing a Conceptual System Definition's own nine required parts.

Looking Ahead

The next chapter defines Worker as the Organizational Entity that performs delegated Responsibilities under Assignment.

Chapter 20 — Worker

This chapter defines Worker as the Organizational Entity that performs delegated Responsibilities, holds a Role, and may spawn additional Workers where its Delegated Authority and current Assignment permit.

Definition

What is it?

A Worker is the Organizational Entity that performs delegated Responsibilities within The Autonomous Company, holding exactly one Role at a time and operating under Delegated Authority and Fid sub-delegated through its Assignment.

A Worker may spawn additional Workers — causing a new Worker to be established, with Delegated Authority and Fid sub-delegated from its own — where its current Delegated Authority explicitly permits spawning and the spawning is undertaken in fulfillment of a bound Assignment.

Semantics

What does it fundamentally mean?

Spawning is not a capability every Worker holds by default; it is a specific permission that must be explicitly present within a Worker's Delegated Authority, the same as any other permitted action.

A spawned Worker's Delegated Authority and Fid are sub-delegated from the spawning Worker's own, continuing the same derivation chain already established for Team, Project, and Assignment.

A Worker's retirement cascades to every Worker it has spawned, mirroring Company's cascading retirement of everything beneath it.

Purpose

Why does it exist?

Worker exists to be the entity that actually performs Autonomous Execution — fulfilling Responsibilities, producing Deliverables, and, where explicitly authorized, spawning further Workers to extend execution capacity in service of a bound Assignment.

Objectives

What is it intended to accomplish?

Worker shall:

- Hold exactly one Role at a time, per Chapter 23.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Fulfill Responsibilities bound to its Assignments by producing Deliverables.
- Spawn additional Workers only where explicitly permitted by its Delegated Authority and tied to a bound Assignment.
- Sub-delegate Delegated Authority and Fid to any Worker it spawns.
- Cascade retirement to every Worker it has spawned.

Composition

What conceptually constitutes it?

Worker consists of:

- **Current Role** — the single Role classification this Worker currently holds.
- **Active Assignments** — the Assignments this Worker currently holds, each with its own Responsibility, Delegated Authority, and Fid.
- **Spawned Workers** — the Workers this Worker has caused to be established, if any.
- **Spawning Worker** — the Worker or Leader that spawned this Worker, if applicable, or the Founding Act reference if not.
- **Where a Worker is established directly by a Company or Team during initial staffing, rather than spawned by another Worker or Leader, Spawning Worker is recorded as a reference to that establishing Company or Team — a third case this Composition entry's original two-option phrasing did not name explicitly. This corrects an earlier omission rather than resolving it silently.**

Capabilities

What must it be capable of?

Worker shall be capable of:

- Being established either by a Company or Team (initial staffing) or by spawning from another Worker.
- Fulfilling a Responsibility by producing a Deliverable, under its Assignment's Delegated Authority and Fid.
- Spawning a new Worker only where its own Delegated Authority explicitly permits it and a bound Assignment justifies it, sub-delegating a portion of its own Delegated Authority and Fid to the spawned Worker.
- Being retired, cascading retirement to every Worker it has spawned.

Behaviors

How does it behave?

Worker:

- Is established with a Role and, where applicable, Delegated Authority and Fid sub-delegated from its spawning Worker, Team, or Company.
- Fulfills Responsibility through Deliverable production, per Chapters 5 and 12.
- Spawns a new Worker only when its current Delegated Authority explicitly permits spawning and the spawn is undertaken in service of a bound Assignment; any other attempt to spawn fails.
- Cascades retirement to every Worker it spawned the instant its own retirement occurs.
- Is recorded via Record Event at establishment, spawning (as both spawning and spawned party), Deliverable production, and retirement.

Relationships

How does it interact with other concepts?

Worker holds Role, per Chapter 23, and Responsibility through Assignment, per Chapters 5 and 11.

Worker is composed within Team, and receives Delegated Authority and Fid through its Assignments, per Chapters 14, 15, and 11.

Worker may spawn additional Workers, sub-delegating its own Delegated Authority and Fid, where explicitly permitted and tied to a bound Assignment.

Worker's retirement cascades to every Worker it has spawned.

Every action taken by a Worker is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Worker shall:

- Hold exactly one Role at a time.
- Never spawn a Worker without explicit permission within its own Delegated Authority.
- Never spawn a Worker independent of a bound Assignment.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never sub-delegate to a spawned Worker more Delegated Authority or Fid than it holds itself.
- Cascade retirement to every Worker it has spawned.
- Record every establishment, spawning event, Deliverable production, and retirement to the Audit Trail.
- A Leader holding worker.retire (Chapter 22, CR-27) may retire a Worker it spawned, or a Worker belonging to a Team it established, but never a Worker outside that chain; retiring a Worker with further spawned descendants or unfulfilled Responsibility requires the retiring party's explicit confirmation of that consequence first.
- This Priority-based tie-breaking (CR-29, Chapter 18) is the specific mechanism through which Chapter 9's own Priority attribute is consumed; absent contention for the same Worker from a different Active Objective, Priority has no effect on Assignment creation.

Looking Ahead

The next chapter defines Administrative Authority as the form of Delegated Authority that permits a Conceptual Synthesist to establish other Conceptual Synthesists and configure system-wide settings.

Chapter 21 — Administrative Authority

This chapter defines Administrative Authority as the form of Delegated Authority that permits a Conceptual Synthesist to establish other Conceptual Synthesists and configure system-wide settings.

Definition

What is it?

Administrative Authority — A named scope of Delegated Authority, held by a Conceptual Synthesist, permitting the establishment of other Conceptual Synthesists, the establishment of Companies, system-wide configuration, the establishment of Role classifications, and the establishment or amendment of Schemas and Processes — six independently-grantable actions, never held by a Worker. Held inherently, unconditionally system-wide, by the Founding Conceptual Synthesist; any other holder's grant may additionally be restricted to a single Company, per Chapter 14's Company binding.

A Conceptual Synthesist holding Administrative Authority is not a distinct Human Participant category — Administrative Authority is a granted scope, the same relationship Leadership Authority holds to Worker.

Semantics

What does it fundamentally mean?

Administrative Authority is a scope of Delegated Authority like any other, and is therefore subject to every rule already governing Delegated Authority: it must trace to an originating grant, may be sub-delegated no broader than its source, and cascades in its revocation to everything established under it.

The Founding Conceptual Synthesist holds Administrative Authority inherently, through the Founding Act, without requiring a grant, per Chapter 1 and Chapter 8.

Administrative Authority is independent of any Role or Leadership Authority a Worker might hold — it applies exclusively to Conceptual Synthesist, never to Worker.

Purpose

Why does it exist?

Administrative Authority exists to permit certain Conceptual Synthesists to establish and oversee other Conceptual Synthesists and system-wide configuration, without treating every human interacting with The Autonomous Company as equally senior, and without inventing a second Human Participant category to express that distinction.

Objectives

What is it intended to accomplish?

Administrative Authority shall:

- Permit establishing a new Conceptual Synthesist.
- Permit transitioning another Conceptual Synthesist's Lifecycle State.
- Permit establishing a Company, per Chapter 18.
- Permit configuration of system-wide settings not otherwise governed by a specific chapter.
- Permit establishing a new Role classification (Chapter 23), scoped and grantable independently of the other three administrative actions listed above.
- Permit establishing or amending a Schema (Chapter 27), scoped and grantable independently of the other four administrative actions listed above.
- Permit establishing or amending a Process (Chapter 28), scoped and grantable independently of the other five administrative actions listed above.
- Remain sub-delegatable, no broader than the granting holder's own scope.

Composition

What conceptually constitutes it?

Administrative Authority consists of:

- **Holding Conceptual Synthesist** — who holds this authority.
- **Granted scope** — specifically, which administrative actions are permitted (establishing Conceptual Synthesists, establishing Companies, system configuration), since these may be granted independently rather than as a single all-or-nothing bundle.

Role classification establishment (Chapter 23) is a fourth independently-grantable administrative action, added on the same terms as the other three — not a bundled or implied part of “system configuration,” since Role is governed by its own chapter.

Schema establishment or amendment (Chapter 27) is a fifth independently-grantable administrative action, added on the same terms as the other four.

Role classification establishment, Schema establishment or amendment, and Process establishment or amendment (Chapter 28) are collectively three of six independently-grantable administrative actions; Process establishment or amendment is the sixth, added on the same terms as the other five, per this chapter’s own Objectives section, which

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

already lists all six. This corrects an earlier omission in this Composition list, per this project's own discipline of naming and correcting errors explicitly rather than silently.

- **Source grant** — the Founding Act, for the Founding Conceptual Synthesist, or the Administrative Authority of the granting Conceptual Synthesist, for all others.
- **Company binding** — where sub-delegated (never for the Founding Conceptual Synthesist's own inherent grant, which remains unconditionally system-wide), the single Company this grant's administrative actions are restricted to, per Chapter 14.

Capabilities

What must it be capable of?

Administrative Authority shall be capable of:

- Being held inherently by the Founding Conceptual Synthesist, or granted by a Conceptual Synthesist already holding sufficient Administrative Authority to sub-delegate it.
- Permitting establishment of a new Conceptual Synthesist, a new Company, or a system-wide configuration change, each independently scoped.
- Permitting establishment of a new Role classification (Chapter 23), independently scoped from the other three administrative actions.
- Permitting establishment or amendment of a Schema (Chapter 27), independently scoped from the other four administrative actions.
- Permitting establishment or amendment of a Process (Chapter 28), independently scoped from the other five administrative actions.
- Being sub-delegated restricted to a single named Company, for any one or more of the six independently-grantable administrative actions, per Chapter 14's Company binding.
- Being narrowed automatically if the holding Conceptual Synthesist's Status becomes Restricted.
- Being revoked, cascading to every Conceptual Synthesist, Company, or sub-delegated Administrative Authority established under it.

Behaviors

How does it behave?

Administrative Authority:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Is held inherently by the Founding Conceptual Synthesist from the moment of the Founding Act, and by any other Conceptual Synthesist only through an explicit grant.
- Permits its specific granted actions only — establishing a Conceptual Synthesist, establishing a Company, or configuring settings — and fails an Authorization Check for any action outside that granted scope.

Where Role classification establishment is among its granted actions, permits that action as a direct, standalone administrative act against Chapter 23's own recognized set — the same shape as Company establishment (Chapter 18) above, requiring no Objective, Project, Deliverable, or Outcome of any kind.

Where Schema establishment or amendment is among its granted actions, permits that action as a direct, standalone administrative act against Chapter 27's own recognized set — the same shape as Company establishment (Chapter 18) above, requiring no Objective, Project, Deliverable, or Outcome of any kind.

Where Process establishment or amendment is among its granted actions, permits that action as a direct, standalone administrative act against Chapter 28's own recognized set — the same shape as Company establishment (Chapter 18) above, requiring no Objective, Project, Deliverable, or Outcome of any kind.

Editorial note: the three paragraphs above previously required Role, Schema, and Process establishment to proceed only through Chapter 13's Outcome-evaluation mechanism, tied to an Objective established for that purpose — explicitly forbidding direct administrative action, three times, by name. Reversed in place as part of CR-46; the kinds of Objective that carried them are gone with CR-56, and they proceed the same direct way Company establishment (Chapter 18) always has.

Where sub-delegated with a Company binding, per Chapter 14, permits its granted actions only for the named Company, and fails an Authorization Check for the same action attempted against any other Company.

Where held without a Company binding, establishes core definitions: a Role classification, Schema, or Process established by a Conceptual Synthesist whose Administrative Authority is not restricted to a Company is a core definition, available to every Company and used by each as it is. A Company never amends a core definition, derives its own from one, or has its own take a core definition's place; a Company that wants something different establishes its own, separate definition, with the Administrative Authority it holds for that Company.

- Narrows automatically alongside the holding Conceptual Synthesist's Delegated Authority when Status becomes Restricted.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Cascades revocation to every Conceptual Synthesist, Company, or sub-delegated Administrative Authority established under it, the instant it is revoked.
- Is recorded via Record Event at grant, exercise, and revocation.

Relationships

How does it interact with other concepts?

Administrative Authority is a scope of Delegated Authority, per Chapter 14, held by Conceptual Synthesist, per Chapter 8.

Administrative Authority permits establishing Conceptual Synthesist and Company, per Chapters 8 and 18.

Administrative Authority is structurally parallel to Leadership Authority — both are named scopes of Delegated Authority granted to a specific holder category, Conceptual Synthesist and Worker respectively — but the two are never held interchangeably; a Worker cannot hold Administrative Authority, and a Conceptual Synthesist does not hold Leadership Authority.

Every exercise of Administrative Authority is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Administrative Authority shall:

- Never permit an action outside its specifically granted scope.
- Never be held by any subject other than a Conceptual Synthesist.
- Never survive the revocation of the grant it was sub-delegated from, except for the Founding Conceptual Synthesist's inherent holding.
- Cascade revocation to everything established under it upon its own revocation.
- Record every grant, exercise, and revocation to the Audit Trail.
- Never permit a holder of a Company-bound Administrative Authority grant to exercise its granted action on behalf of any Company other than the one it is bound to.

Looking Ahead

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

The next chapter defines Leadership Authority as the form of Delegated Authority that permits a Worker to establish Teams, create Assignments, and coordinate execution within its scope.

Chapter 22 — Leadership Authority

This chapter defines Leadership Authority as the form of Delegated Authority that permits a Worker to establish Teams, create Assignments, and coordinate execution within its scope.

Definition

What is it?

Leadership Authority is a named scope of Delegated Authority, granted to a Worker, that permits establishing Teams, creating Assignments binding other Workers, and coordinating execution across Workers and Teams within the granted scope.

A Worker holding Leadership Authority is referred to as a Leader. This is a description of a Worker's granted authority, not a distinct Organizational Entity — a Leader retains its original Identity and Role throughout.

Semantics

What does it fundamentally mean?

Leadership Authority is independent of Role — a Worker holding any Role classification may also hold Leadership Authority; the two are granted and revoked separately.

Leadership Authority is a scope of Delegated Authority like any other, and is therefore subject to every rule already governing Delegated Authority: it may be narrowed automatically by Status, must be sufficient before a corresponding Responsibility may be assigned, and cascades in its revocation to everything sub-delegated under it.

A Worker's spawning capability (Chapter 20) and Leadership Authority are related but distinct: spawning creates a new Worker; Leadership Authority additionally permits establishing Teams and creating Assignments that bind existing Workers to Projects.

Purpose

Why does it exist?

Leadership Authority exists to permit a Worker to coordinate the execution of others — establishing Teams and Assignments — without requiring every organizational act to be performed directly by a Conceptual Synthesist or a Company, and without inventing a second class of entity to hold that permission.

Objectives

What is it intended to accomplish?

Leadership Authority shall:

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Permit a Worker to establish Teams within its granted scope.
- Permit a Worker to create Assignments binding other Workers to Projects or Teams within its granted scope.
- Remain independent of and separately grantable from Role.
- Be subject to the same narrowing, expiration, and cascading revocation rules as any other Delegated Authority grant.

Composition

What conceptually constitutes it?

Leadership Authority consists of:

- **Holding Worker** — the Worker granted this authority.
- **Granted scope** — which Teams, Projects, or Workers this authority permits coordination over.
- **Source grant** — the Delegated Authority this Leadership Authority was sub-delegated or granted from.

Capabilities

What must it be capable of?

Leadership Authority shall be capable of:

- Being granted to a Worker by a Company or by another Worker already holding sufficient Leadership Authority to sub-delegate it.
- Permitting Team establishment and Assignment creation within its granted scope only.
- Being narrowed automatically if the holding Worker's Status becomes Restricted, per Chapters 6 and 14.
- Being revoked, cascading to every Team, Assignment, or sub-delegated Leadership Authority established under it.

Behaviors

How does it behave?

Leadership Authority:

- Is granted to a Worker as a distinct scope of Delegated Authority, separate from and in addition to whatever Role that Worker holds.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Permits Team establishment and Assignment creation strictly within its granted scope; any attempt beyond that scope fails an Authorization Check.
- Narrows automatically alongside the holding Worker's Delegated Authority when Status becomes Restricted.
- Cascades revocation to every Team, Assignment, and sub-delegated Leadership Authority established under it, the instant it is revoked.
- Is recorded via Record Event at grant, exercise, and revocation.

Relationships

How does it interact with other concepts?

Leadership Authority is a scope of Delegated Authority, per Chapter 14, held by a Worker, per Chapter 20.

Leadership Authority permits establishing Team, per Chapter 19, and creating Assignment, per Chapter 11.

Leadership Authority is independent of Role, per Chapter 23 — a Worker's Role and Leadership Authority are granted, held, and revoked separately.

Every exercise of Leadership Authority is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Leadership Authority shall:

- Never permit Team establishment or Assignment creation outside its granted scope.
- Never be conflated with or dependent upon a specific Role classification.
- Never survive the revocation or expiration of the grant it was sub-delegated from.
- Cascade revocation to everything established under it upon its own revocation.
- Record every grant, exercise, and revocation to the Audit Trail.
- Leadership Authority's grantable scope gains a third value (CR-27), worker.retire, scoped to Workers within the granting party's own chain of establishment — a Worker it spawned directly, or a Worker belonging to a Team it established.

Looking Ahead

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Part V — Organizational Entities and Authority Scopes is now complete. The next Part defines the Supporting Concepts — Role, Capability, Skill, and Resource — that Worker and Leadership Authority depend upon, beginning with Role.

PART VI — SUPPORTING CONCEPTS

Chapter 23 — Role

This chapter defines Role as the conceptual property that establishes the function a Worker performs and the responsibilities, authority, and capabilities that function requires.

Definition

What is it?

Role is the concept that establishes which function a Worker performs within The Autonomous Company, and the set of responsibilities, capabilities, and Delegated Authority that function requires to be carried out.

A Worker performs one Role at a time.

Semantics

What does it fundamentally mean?

Role defines what a Worker is for, not who a Worker is — Identity establishes the latter and is unaffected by Role.

Role is not a description of past performance or current standing; those are governed by Organizational Experience and Status, respectively.

A Worker's Role may change over its lifetime through a defined Reassignment event; the Worker's Identity persists unchanged through any such Reassignment.

A new Role classification enters this chapter's recognized set exclusively through direct action by a Conceptual Synthesist holding Administrative Authority (Chapter 21) — no Objective, Project, Deliverable, or Outcome of any kind is involved, the same direct shape Company establishment (Chapter 18) already uses. A Leader never commits a Role classification to the recognized set under any circumstance — Leadership Authority does not include, and can never include, Administrative Authority's scope. The one exception: where a Role classification is proposed and approved within an Objective's own Plan (Chapter 10, Chapter 15's Fid-ceiling Constraint), the evaluating Conceptual Synthesist commits it directly at Plan Gate Evaluation, still exercising the identical Administrative Authority, never a Leader's own.

Purpose

Why does it exist?

Role exists to establish, for any Worker, what function it performs, so that Assignment, Delegated Authority, and required Capability can each be evaluated against a known, current function rather than inferred anew for every Worker individually.

Objectives

What is it intended to accomplish?

Role shall:

- Establish the function a Worker currently performs.
- Define the Capabilities required to perform that function.
- Define the baseline Delegated Authority appropriate to that function.
- Support Reassignment of a Worker to a different Role without affecting Identity.
- Preserve a complete history of Role assignments and Reassignments.

Composition

What conceptually constitutes it?

Role consists of:

- **Role classification** — the specific function, such as Creative, Designer, or Developer.
- **Required Capabilities** — the Capabilities a Worker must hold to perform the Role.
- **Baseline Delegated Authority** — the scope of Delegated Authority ordinarily associated with the Role, prior to any Objective-specific grant or Status-based narrowing.
- **Newly Established** — a boolean, true for a Role classification from its establishment until its first-ever Worker Assignment under that Role is fulfilled, after which it becomes permanently false.

Capabilities

What must it be capable of?

Role shall be capable of:

- Establishing a Worker's current Role classification at establishment.
- Verifying a Worker holds the Capabilities required by a Role before that Role may be assigned.
- Reassigning a Worker from one Role to another through a defined Reassignment event.
- Invoking Record Event for every Role assignment and Reassignment.

Behaviors

How does it behave?

Role:

- Is assigned to a Worker at establishment, as part of the Worker's initial configuration — not jointly and atomically with Identity the way Lifecycle State is, since a Worker's Identity does not depend on holding any particular Role.
- Remains fixed until an explicit Reassignment event occurs; it does not change as a side effect of Assignment, Status, or any other concept's transitions.
- Is replaced entirely upon Reassignment — a Worker holds exactly one Role classification at a time, never more than one.
- Requires verification that the Worker holds the new Role's required Capabilities before Reassignment may complete; a Reassignment shall fail if this verification fails.

Relationships

How does it interact with other concepts?

Role is held by Worker — since a Leader is a Worker holding Leadership Authority rather than a separate entity, per Chapter 22, no additional case applies.

Role determines the baseline Delegated Authority a Worker holds; Status may further narrow that baseline (per Chapter 6) without altering the Role itself.

Role determines which Assignments a Worker is eligible to receive — an Assignment requiring a given Role classification may only be given to a Worker currently holding that Role.

Role depends on Capability to define what it requires, and on Skill and Resource where those further qualify a Worker's fitness for a Role; the full mechanics of that qualification are governed by those chapters.

Constraints

What boundaries govern it?

Role shall:

- Hold exactly one classification per Worker at any time.
- Never change except through a defined Reassignment event.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never be assigned or reassigned to a Worker that does not hold the required Capabilities.
- Never affect the Worker's Identity, which persists unchanged through any Reassignment.
- Invoke Record Event for every assignment and Reassignment, including the Capabilities verified at the time.
- Never add a Role classification to the recognized set except directly, by a Conceptual Synthesist holding role.establish_classification Administrative Authority — whether acting alone (the ordinary case) or in the course of approving an Objective's own Plan (Chapter 10, Chapter 15), the two real paths this chapter's own Behaviors section describes above.
- The first Assignment made under a Newly Established Role classification is forced to Step-Gated oversight (Chapter 9's Oversight Level), regardless of the Oversight Level otherwise governing its Objective, reflecting that no Organizational Experience yet exists to support the Role. Newly Established becomes permanently false upon that Assignment's fulfillment; no subsequent Assignment under the Role is subject to this override.
- A Role classification may be retired only through the identical direct mechanism as its own establishment (CR-31) — a Conceptual Synthesist holding role.establish_classification, acting directly rather than through any Objective's Outcome. A retired Role classification is never deleted from the historical record; it becomes permanently ineligible for any new Assignment binding from the moment of retirement forward, while everything already bound to it before that moment remains entirely unaffected.

Looking Ahead

- A Role classification maintains an Amendment History (CR-32) — a chronological, append-only record of its own defining content as it stood immediately before each amendment superseded it. Every establishment appends the first such entry; every amendment appends a new one before the amended content takes effect. No entry is ever altered or removed once recorded.

The next chapter defines Capability as the shared conceptual property that establishes a specific, checkable competency a Worker holds, gating the Roles and Assignments it may be given.

Chapter 24 — Capability

This chapter defines Capability as the shared conceptual property that establishes a specific, checkable competency a Worker holds, gating the Roles and Assignments it may be given.

Definition

What is it?

Capability is the shared conceptual property that establishes a specific competency a Worker either holds or does not hold, verified whenever a Role, Reassignment, or Assignment requires it.

Capability is binary — a Worker holds a given Capability or does not; this chapter does not define graduated proficiency.

Semantics

What does it fundamentally mean?

Capability is distinct from Role: Role defines the function a Worker performs; Capability defines the specific competencies verified as prerequisites for holding that Role, per Chapter 23.

Capability is mutable, but only through a defined acquisition or loss event governed by Skill, defined in the next chapter — it does not change arbitrarily or as a side effect of any other concept's transition.

Capability is distinct from Resource: holding a Capability means a Worker is competent to perform an activity; it does not guarantee the Worker currently has access to what exercising that Capability requires. That access is governed by Resource, defined after Skill.

Purpose

Why does it exist?

Capability exists to give Role, Assignment, and Reassignment a concrete, checkable basis for verifying whether a Worker is competent to hold a given function or fulfill a given Responsibility, rather than assuming competency by Role classification alone.

Objectives

What is it intended to accomplish?

Capability shall:

- Establish, for any Worker, which specific competencies it currently holds.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Be verifiable as a precondition for Role assignment, Reassignment, and any Assignment that requires it.
- Change only through a defined Skill acquisition or loss event.
- Preserve a complete history of every Capability gained or lost.

Composition

What conceptually constitutes it?

Capability consists of:

- **Capability classification** — the specific competency identified (e.g., a defined category of task or domain).
- **Holding Worker** — the Worker this Capability is currently held by.
- **Acquisition reference** — the Skill event through which this Capability was gained.

Capabilities *(of the Capability concept itself)*

What must it be capable of?

Capability shall be capable of:

- Being verified as present or absent for a given Worker upon request from Role, Assignment, or Reassignment.
- Being gained by a Worker only through a defined Skill acquisition event.
- Being lost by a Worker only through a defined Skill loss event.
- Preserving a history of every Capability gained or lost, even after loss.

Behaviors

How does it behave?

Capability:

- Is verified at the moment a Role is assigned, a Reassignment is attempted, or an Assignment requiring it is created; absence causes the requiring action to fail.
- Is gained only through a Skill acquisition event, never as a side effect of any other concept's transition.
- Is lost only through a defined Skill loss event; it does not expire or degrade on its own.
- Is recorded via Record Event at every gain and loss.

Relationships

How does it interact with other concepts?

Capability is held by Worker and verified by Role as a precondition for Role assignment and Reassignment, per Chapter 23.

Capability is gained or lost through Skill, defined in the next chapter.

Capability may be required by Assignment, per Chapter 11, as a precondition distinct from Role classification.

Capability is distinct from Resource — holding a Capability does not guarantee access to what exercising it requires.

Every Capability change is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Capability shall:

- Never change except through a defined Skill acquisition or loss event.
- Never be assumed present for a Worker without explicit verification.
- Never be graduated — a Worker either holds a given Capability or does not.
- Record every gain and loss to the Audit Trail.

Looking Ahead

The next chapter defines Skill as the operational mechanism through which a Worker acquires or loses a Capability.

Chapter 25 — Skill

This chapter defines Skill as the operational mechanism through which a Worker gains or loses a Capability.

Definition

What is it?

Skill is the operational mechanism through which a Worker's Capability is gained or lost, whether through training, configuration, or another qualifying method.

Skill is not itself a subject holding Identity — it is a capability, in the general sense, exercised by or upon a Worker to change what Capabilities that Worker holds, the same way Audit Trail's Record Event is a capability rather than an Identity-holding subject.

Semantics

What does it fundamentally mean?

A Skill acquisition or loss event requires explicit permission within the Delegated Authority of the party initiating it — a Worker does not freely acquire or lose Capabilities on its own initiative; acquisition is either authorized within the Worker's own Delegated Authority or initiated by a Leader or Conceptual Synthesist holding sufficient Delegated Authority over that Worker.

A Skill event produces exactly one outcome: a specific Capability is gained, or a specific Capability is lost, by a specific Worker.

Purpose

Why does it exist?

Skill exists to give Capability a governed mechanism for change — ensuring that what a Worker is competent to do changes only through an authorized, attributable act, never silently or as a side effect of unrelated activity.

Objectives

What is it intended to accomplish?

Skill shall:

- Provide the sole mechanism through which a Capability may be gained or lost.
- Require explicit permission, within Delegated Authority, before an acquisition or loss event may proceed.
- Attribute every acquisition or loss to the Worker whose Delegated Authority authorized it, and the Worker whose Capability changed.

- Preserve a complete history of every Skill event.

Composition

What conceptually constitutes it?

A Skill event consists of:

- **Affected Worker** — the Worker whose Capability changes.
- **Authorizing subject** — the Worker, Leader, or Conceptual Synthesist whose Delegated Authority permitted this event.
- **Resulting Capability change** — the specific Capability gained or lost.
- **Method** — the qualifying method through which the change occurred (training, configuration, or another defined method).

Capabilities *(of the Skill mechanism itself)*

What must it be capable of?

Skill shall be capable of:

- Verifying the authorizing subject holds sufficient Delegated Authority before proceeding.
- Producing exactly one Capability gain or loss per event.
- Refusing to proceed without a valid authorization.
- Recording every event, successful or refused, to the Audit Trail.

Behaviors

How does it behave?

Skill:

- Is invoked either by a Worker acting within its own Delegated Authority, or by a Leader or Conceptual Synthesist acting on a Worker's behalf within sufficient Delegated Authority.
- Verifies authorization before producing any Capability change; an unauthorized attempt fails and produces no change.
- Produces exactly one Capability gain or loss per invocation.
- Is recorded via Record Event at every invocation, whether it results in a successful change or a refusal.

Relationships

How does it interact with other concepts?

Skill is the sole mechanism through which Capability, defined in the prior chapter, is gained or lost.

Skill requires Delegated Authority, per Chapter 14, held either by the affected Worker or by a Leader or Conceptual Synthesist acting on its behalf.

Every Skill event is preserved through Audit Trail and contributes to Organizational Experience.

Constraints

What boundaries govern it?

Skill shall:

- Never produce a Capability change without a valid, verified authorization.
- Never produce more than one Capability gain or loss per event.
- Never be exercised by a subject lacking sufficient Delegated Authority over the affected Worker.
- Record every invocation, successful or refused, to the Audit Trail.

Looking Ahead

The next chapter defines Resource as the external dependency a Worker must have access to in order to exercise a held Capability

Chapter 26 — Resource

This chapter defines Resource as the external dependency a Worker must be authorized to access, and have sufficient Fid to consume, in order to exercise a held Capability.

Definition

What is it?

A Resource is an external dependency — a model, a tool, an external service, or another qualifying dependency — that a Worker draws upon to exercise a Capability it holds.

Holding a Capability establishes that a Worker is competent to perform an activity; accessing the Resource that activity requires is a separate, additionally gated act.

Semantics

What does it fundamentally mean?

Resource access requires both explicit permission within the accessing Worker's Delegated Authority and sufficient remaining Fid to consume it — authority without budget, or budget without authority, is equally insufficient to permit access.

Resource is not itself an Identity-holding Operational Object; it is a reference a Worker's action is evaluated against, the same way Skill is a mechanism rather than a subject.

Resource consumption is tracked through Fid's existing allocation and consumption mechanism, per Chapter 15 — Resource does not introduce a second economic ledger. Each consumption event is categorized by the specific Resource consumed (a given model, tool, or external service), consistent with the event categories Chapter 17 already established.

A Resource may be fulfilled by more than one interchangeable Provider — for example, multiple model vendors capable of the same Resource classification. Where more than one Provider is available and authorized, the accessing Worker's Leader selects among them consistent with the bound Objective's Optimization Directive (Chapter 9).

Resource access may be Refused — failing authorization or Fid sufficiency, per this chapter's Behaviors, before any attempt is made — or it may Fail, where authorization and Fid sufficiency both hold but the external dependency itself does not return a usable result (timeout, error, or malformed output). Refusal and Failure are distinct and separately categorized: Refusal never consumes Fid; Failure may consume Fid up to the point of failure, per Chapter 15's existing consumption accounting.

Purpose

Why does it exist?

Resource exists to give Capability's exercise a governed, attributable point of contact with the outside world — every model call, tool use, and external-service interaction a Worker makes is authorized, budgeted, and recorded, rather than assumed to be free or automatically permitted by holding the relevant Capability alone.

Objectives

What is it intended to accomplish?

Resource shall:

- Identify the specific external dependency a Capability's exercise requires.
- Require explicit permission within Delegated Authority before access is granted.
- Require sufficient remaining Fid before consumption is permitted.
- Record every access attempt, granted or refused, and every consumption event, categorized by the specific Resource consumed.

Composition

What conceptually constitutes it?

Resource consists of:

- **Resource classification** — the specific dependency (a given model, tool, or external service).
- **Required Capability** — the Capability whose exercise this Resource supports.
- **Accessing Worker** — the Worker attempting access.
- **Consumption reference** — the Fid consumption event this access produced, where access succeeds.
- **Selected Provider** — the specific Provider fulfilling this Resource access, recorded where more than one qualifying Provider was available.

Capabilities *(of the Resource concept itself)*

What must it be capable of?

Resource shall be capable of:

- Verifying the accessing Worker's Delegated Authority explicitly permits this Resource before granting access.
- Verifying sufficient remaining Fid before permitting consumption.
- Refusing access if either authorization or Fid sufficiency fails.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Producing a categorized Fid consumption event upon successful access, per Chapter 15.
- Selecting among multiple qualifying Providers for the same Resource classification, consistent with the bound Objective's Optimization Directive.
- Distinguishing a Failed access attempt from a Refused one, and recording each under its own Audit Trail category.
- Retrying a Failed access a bounded number of times, per this chapter's Behaviors, before escalating.

Behaviors

How does it behave?

Resource:

- Is accessed only when the requesting Worker holds the Capability the access supports, holds Delegated Authority explicitly permitting the specific Resource, and holds sufficient remaining Fid.
- Refuses access and produces no consumption event if any of the three conditions is not met.

Where more than one Provider qualifies to fulfill a Resource, the accessing Worker's Leader selects among them per the bound Objective's Optimization Directive: Cost-Optimized favors the lowest-cost qualifying Provider, Precision-Optimized favors the highest-capability qualifying Provider, and Balanced leaves the choice to Leader discretion. The Selected Provider is recorded as part of the access's Audit Trail entry, under Chapter 17's existing model-selection event category — no new category is required.

Upon Failure, a Resource access is retried automatically up to a defined retry limit scoped to the accessing Worker's Assignment. If every retry Fails, the associated Responsibility (Chapter 5) is escalated exactly as it is escalated upon Delegated Authority narrowing, and the bound Objective transitions to Pending Approval (Chapter 4) pending the decision of the Conceptual Synthesist or Leader who granted the underlying Assignment.

Where more than one Provider qualifies for the Resource, a Failure's automatic retry attempts the next qualifying Provider, in the order the Objective's Optimization Directive prefers, before the retry limit is considered exhausted. Escalation occurs only once every qualifying Provider has been attempted and failed.

- Produces a Fid consumption event, categorized by Resource classification, upon successful access — per Chapter 15's allocation and consumption mechanism.
- Is recorded via Record Event at every access attempt, granted or refused.

Relationships

How does it interact with other concepts?

Resource supports the exercise of Capability, defined two chapters prior, but is distinct from it — holding a Capability does not guarantee Resource access.

Resource access requires Delegated Authority, per Chapter 14, and sufficient Fid, per Chapter 15; consumption is recorded through Fid's existing mechanism, not a separate ledger.

Every Resource access attempt is preserved through Audit Trail, categorized among the execution event types Chapter 17 already established (model selection, token consumption, tool use, external-service use).

Constraints

What boundaries govern it?

Resource shall:

- Never be accessed without explicit permission within the accessing Worker's Delegated Authority.
- Never be consumed without sufficient remaining Fid.
- Never be tracked through a mechanism other than Fid's existing allocation and consumption model.
- Never select a Provider outside the accessing Worker's Delegated Authority or beyond available Fid, regardless of Optimization Directive.
- Never retry a Failed access beyond its defined limit without escalating.
- Record every Failure, retry, and resulting escalation to the Audit Trail, categorized separately from Refusal.
- Record every access attempt, granted or refused, to the Audit Trail.

Looking Ahead

This chapter completes Part VI — Supporting Concepts, and Volume I's full chapter sequence. The appendices — Glossary, Conceptual Model, Dependency Model, and Lifecycle Models — remain to be assembled from the completed chapters.

Chapter 27 extends Part VI beyond what this chapter anticipated when originally written; see Chapter 27's own Looking Ahead for where the volume's chapter sequence currently completes.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Chapter 27 — Schema

This chapter defines Schema as the registered, required structure for a specific Deliverable Type, and the mechanism through which such standards are established with the same weight as a new Role classification.

Definition

What is it?

A Schema is the registered, required structure for a specific Deliverable Type: the named fields a conforming Deliverable must include, and whether relationships between Deliverables of that Type must be identity-addressed rather than positional.

Semantics

What does it fundamentally mean?

A Schema exists to prevent two distinct failure modes at once: content produced without the fields a downstream process needs, and cross-references that encode a Deliverable's current position rather than its stable identity. A Schema is not a description of what any single Deliverable happens to contain; it is the enforced standard every Deliverable of a given Type is checked against.

Purpose

Why does it exist?

Schema gives recurring categories of Deliverable — Chapters, Build Specifications, Conceptual System Definitions, and others not yet named — the same consistency and machine-queryability that a fielded taxonomy gives recurring categories of work, without requiring every such standard to be established with equal ceremony to a new Role classification's authority implications, nor so little ceremony that standards proliferate and drift uncontrolled.

Objectives

What is it intended to accomplish?

Schema shall:

- Define, for a given Deliverable Type, the fields every conforming Deliverable must include.
- Define whether that Type's cross-references must be identity-addressed, and if so, prohibit positional citation for Deliverables of that Type absolutely.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Ensure no Schema can be committed to the recognized set except through the same Objective/Outcome/Administrative Authority mechanism Chapter 23 already requires for a new Role classification.

Composition

What conceptually constitutes it?

A Schema consists of:

- **Governed Deliverable Type** — the classification this Schema applies to.
- **Required Fields** — the named fields every conforming Deliverable of this Type must include.
- **Identity-Addressed Relationships** — a boolean; where true, every relationship a conforming Deliverable expresses to another Deliverable of a governed Type must reference a stable identity, never a position (a chapter number, a list index, a section heading), with position resolved only at render time.
- **Composite Fields** — zero or more named fields whose value is a growable, identity-addressed collection of Items, each Item conforming to that Composite Field's own Item Required Fields (the identical shape as this Schema's own Required Fields, applied one level deeper). Every Item carries a stable identity, established at the moment it is added and never repurposed for position — Items are always identity-addressed, independent of this Schema's own Identity-Addressed Relationships setting, which governs a distinct thing (relationships between separate Deliverables, not structure within one). A Composite Field's own Item Required Fields may never itself include a Composite Field — nesting is exactly one level.

Capabilities

What must it be capable of?

Schema shall be capable of:

- Being established directly by a Conceptual Synthesist holding Administrative Authority (Chapter 21) — no Objective, Project, Deliverable, or Outcome required, the same shape Company establishment (Chapter 18) already uses.
- Being amended through the same direct mechanism as establishment — a Schema is never silently edited outside that Conceptual Synthesist's own explicit act.
- Governing every Deliverable of its Governed Deliverable Type from the moment it is committed forward; Deliverables produced before a Schema's establishment are never retroactively required to conform.

Behaviors

How does it behave?

A Leader never commits a Schema to the recognized set under any circumstance — Leadership Authority does not include, and can never include, Administrative Authority's scope, the same structural limit Chapter 23 places on Role classification establishment. Unlike Role, Schema has no second, Plan-Approval-time path; every Schema is established or amended directly by a Conceptual Synthesist, with no Objective involved at all.

A Composite Field's own Item Required Fields is amended through the identical direct mechanism as any other part of a Schema's own definition, recorded through the same Amendment History (CR-32) already maintained for the Schema as a whole. No separate authorization surface exists for a Composite Field specifically.

Relationships

How does it interact with other concepts?

Schema governs Deliverable (Chapter 12) the same way Role governs Worker (Chapter 20) — a classification a subject conforms to, established through its own Objective/Outcome/Administrative Authority path, never assigned casually.

Schema establishment shares the identical direct Administrative Authority mechanism (Chapter 21) that Role classification and Process establishment each use — a Conceptual Synthesist acting directly, never through any Objective's Outcome.

Constraints

What boundaries govern it?

Schema shall:

- Never include a “Looking Ahead” field, or any equivalent narrative-transition field, among its Required Fields — such framing, where useful in a human-readable rendering, is generated at render time from ordering and relationships already present, never stored as content on the Deliverable itself.
- Never be added to the recognized set except directly, by a Conceptual Synthesist holding schema.establish_or_amend Administrative Authority.
- Never be applied retroactively to a Deliverable produced before that Schema's establishment.
- A Schema may be retired only through the identical direct mechanism as its own establishment or amendment (CR-31) — a Conceptual Synthesist holding

schema.establish_or_amend, acting directly rather than through any Objective's Outcome. A retired Schema is never deleted from the historical record; it becomes permanently ineligible to govern any new Deliverable Type binding from the moment of retirement forward, while everything already produced under it before that moment remains entirely unaffected.

- A Schema maintains an Amendment History (CR-32) — a chronological, append-only record of its own Required Fields and Identity-Addressed Relationships requirement as they stood immediately before each amendment superseded them. Every establishment appends the first such entry; every amendment appends a new one before the amended content takes effect. No entry is ever altered or removed once recorded.
- Never record a Deliverable of a Schema'd Type missing a declared Composite Field's own name from its content, regardless of whether that field's Item collection is empty.
- Never record an Item within a Composite Field that does not conform to that Composite Field's own Item Required Fields.
- Never permit a Composite Field's own Item Required Fields to include a further Composite Field — nesting beyond one level is prohibited.
- Never assign or reuse an Item's own identity for a different Item, within the same Deliverable or across a Supersedes revision — an Item carried forward unchanged keeps its identity; a genuinely new Item receives a new one.

Looking Ahead

The Appendices record every registered Schema's current membership, the same way they record every other structural registry this volume maintains. This chapter completes Part VI — Supporting Concepts, and Volume I's full chapter sequence, extending what Chapter 26 anticipated at the time it was written.

Chapter 28 extends Part VI beyond what this chapter anticipated when originally written; see Chapter 28's own Looking Ahead for where the volume's chapter sequence currently completes.

Chapter 28 — Process

This chapter defines Process as a named kind of work and the registered, required Assignment sequence for it, and the mechanism through which such sequencing standards are established with the same weight as a new Role classification or Schema.

Definition

What is it?

A Process names a kind of work and is the registered, required Assignment sequence for it: the ordered stages a Project must produce Deliverables for, and the Gate Evaluation dependency between consecutive stages. Choosing what kind of work an Objective is means choosing its Process.

Held either by a Company, governing only that Company's own Objectives, or as a core Process, available to every Company and used by each as it is (Chapter 21). Several may be available at once; each is its own kind of work, distinguished by Process Name.

Semantics

What does it fundamentally mean?

A Process exists to prevent the failure mode a sequencing rule hard-coded into a chapter demonstrates in miniature: a real sequencing requirement, expressible only by editing a specific chapter directly, with no general mechanism for a second or third such requirement without repeating the same hand-edit. A Process is not a description of what any single Project happened to do; it is the enforced stage order every Project running under it is checked against.

Purpose

Why does it exist?

Process lets the Conceptual Synthesist register a required build sequence — Conceptual System Definition, then Build Specification, then Development, or any other ordered stage set — with the same weight and durability as a Role or a Schema, rather than as a one-off Constraint hand-written into a chapter each time a new sequencing rule is needed.

Objectives

What is it intended to accomplish?

Process shall:

- Define, for a kind of work, the ordered stages a Project must produce Deliverables for, and what each stage carries: the work it performs, the context and actions it requires, its completion requirements, what it builds from, the Role responsible for it, and the results it produces.
- Optionally bind each stage to one or more registered Schemas (Chapter 27) governing that stage's Deliverable Types.
- Ensure no Process can be committed to the recognized set except directly, by a Conceptual Synthesist holding Administrative Authority (Chapter 21) — the same direct mechanism Chapter 23 and Chapter 27 require.

Composition

What conceptually constitutes it?

A Process consists of:

- **Description** — the kind of work this Process is for, in words a Leader can recommend it from.
- **Category** — optional; a label for grouping and filtering.
- **Accomplishment action** — a platform-defined action taken when an Objective running under this Process is Accomplished (for example, establish the Team), or none. Only a core Process carries one; a Company's own Process carries none.
- **Leader-directed** — an explicit setting, held only by a core Process, marking a Process that names its kind of work without sequencing it: its Stages declare no Deliverable Types and no Role, and the Project's Leader chooses the Deliverables and directs the work. The core Simple Process is Leader-directed.
- **Process Name** — a human-chosen name identifying this Process.
- **Registering Company** — the Company that established this Process, whose Objectives alone it governs; a core Process has none — it is established by a Conceptual Synthesist whose Administrative Authority is not restricted to a Company (Chapter 21) and is available to every Company.
- **Stages** — an ordered list of stages, each carrying the seven properties below. A Stage declaring more than one Deliverable Type is complete only when a Deliverable of every Type it declares has been produced; that completion, not the production of any single Deliverable, is the Stage boundary Chapter 10's own Constraints gate at.
 - **Name** — the specific work the Stage performs.
 - **General Guidance** — the context necessary to interpret the Stage's work correctly. Given to the Worker and to the Auditor; it informs the work and is never itself a completion requirement.
 - **Output Contract** — the Stage's deterministic completion requirements: an identified set of criteria its output must satisfy to be done, each answered individually when its Deliverable is produced and reviewed individually.
 - **Instructions** — the ordered actions that must be performed at the Stage, each identified, each requiring a recorded response.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- **Inputs** — the required upstream Deliverables, baseline, or context the Stage's work builds from, each of one kind: a Deliverable — a named Deliverable Type an earlier Stage of the same Process declares; a Baseline — an existing state the work starts from rather than a prior Deliverable; or Context — anything else. Every Input is given to the Worker and to the Auditor; a Deliverable or Baseline Input is also required, as this chapter's Behaviors describe.
- **Role** — the Worker expertise responsible for the Stage, declared by the Stage rather than derived from its Deliverable Types.
- **Deliverable Types** — the Schema-governed results the Stage produces: one or more (CR-49), each optionally referencing a registered Schema (Chapter 27).

Capabilities

What must it be capable of?

Process shall be capable of:

- Being established directly by a Conceptual Synthesist holding Administrative Authority (Chapter 21) — no Objective, Project, Deliverable, or Outcome required, the same shape Company establishment (Chapter 18) already uses.
- Being amended through the same mechanism as establishment.
- Governing every Project whose Objective runs under it from the moment it is committed forward; Projects already underway are never retroactively required to conform.

Behaviors

How does it behave?

A Leader never commits a Process to the recognized set under any circumstance — Leadership Authority does not include, and can never include, Administrative Authority's scope, the same structural limit Chapter 23 and Chapter 27 already place on Role and Schema establishment. Like Schema, Process has no second, Plan-Approval-time path; every Process is established or amended directly by a Conceptual Synthesist, with no Objective involved at all.

An Objective's Governing Process is chosen per Chapter 9. A new Process whose kind of work duplicates one already available to its Company is asked about when it is authored: use the existing one, amend it where the Company holds it, or keep both, with a stated reason.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

A core Process, like a core Schema or Role (Chapter 21), is used by every Company as it is. A Company never amends a core Process, derives its own from one, or has its own take a core Process's place; a Company that wants a different kind of work establishes its own, separate Process.

A Leader-directed Process names its kind of work without sequencing it: an Objective running under one is not held to a Stage order, its Leader chooses the Deliverables and directs the work, and, where Milestone-Gated, its milestones are drawn from its own Approved Plan (Chapter 10). Only a core Process may be Leader-directed. A Company's own Process never is: a Company Process whose Stages leave their fields blank is incomplete, and is not established until they are filled.

Where an Objective running under a Process is Accomplished (Chapter 13) and that Process carries an Accomplishment action, the action is taken with the Outcome, in one act.

A Stage does not begin until its required Inputs exist. A required Deliverable Input must exist and be Approved — under Autonomous oversight, produced (Chapter 10) — and while it is Under Revision the Stage waits for the revision to land. A required Baseline is enforced where its source exists, such as an Objective's Baseline Reference (Chapter 9). Context never blocks. A Stage waiting on a required Input is Queued and names the Input; a Stage whose required Input can never arrive — its Deliverable Rejected — is Stuck. A running Objective keeps the Inputs its Stages started with.

Relationships

How does it interact with other concepts?

Process governs Project's Assignment sequence (Chapter 10) the same way Schema governs Deliverable's field structure (Chapter 12) and Role governs Worker (Chapter 20) — a third parallel, structurally identical established-standard mechanism.

A Process Stage referencing a Schema (Chapter 27) creates a dependency this chapter's own Constraints require to already exist — a Process shall never reference an unestablished Schema.

Constraints

What boundaries govern it?

Process shall:

- Never be added to the recognized set except directly, by a Conceptual Synthesist holding `process.establish_or_amend` Administrative Authority.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

- Never be applied retroactively to a Project established before that Process's establishment.
- Never reference a Schema (Chapter 27) for a Stage that is not already an established member of Chapter 27's recognized set.
- Never permit a Stage to reference another Process — a Stage always resolves to Deliverable Types directly; Process nesting is out of scope for this chapter, registered as a future consideration rather than built now.
- Never permit a Deliverable to be produced under a Stage while any Instruction or Output Contract criterion that Stage declares lacks a recorded response; the responses are recorded on the Deliverable. A Stage declaring neither is unaffected.
- Never admit to the recognized set, by establishment or amendment, a Process any of whose Stages requires a Deliverable Input that no earlier Stage of that Process declares.
- Never share a Process Name with another Process already registered by the same Company.
- Never govern an Objective of a different Company than the one that registered it; a core Process is registered by no Company and is available to every Company.
- Never carry an Accomplishment action, or be Leader-directed, unless it is a core Process.
- Never be amended, derived from, or replaced by a Company where it is a core Process; a core Process changes only through its own direct establishment or amendment by a Conceptual Synthesist whose Administrative Authority is not restricted to a Company (Chapter 21).
- A Process may be retired only through the identical direct mechanism as its own establishment or amendment (CR-31) — a Conceptual Synthesist holding `process.establish_or_amend`, acting directly rather than through any Objective's Outcome. A retired Process is never deleted from the historical record; it becomes permanently ineligible to govern any new Objective's Assignment sequencing from the moment of retirement forward, while everything already sequenced under it before that moment remains entirely unaffected.
- A Process maintains an Amendment History (CR-32) — a chronological, append-only record of its own Stage sequence as it stood immediately before each amendment superseded it. Every establishment appends the first such entry; every amendment appends a new one before the amended content takes effect. No entry is ever altered or removed once recorded.

Looking Ahead

The Appendices record every registered Process's current membership. This chapter extends Part VI — Supporting Concepts — beyond what Chapter 27 anticipated at the time it was written.

APPENDICES

Glossary

Glossary

This appendix defines every term formally established in this volume, in alphabetical order, for quick reference. Full definitions, including semantics, behaviors, and constraints, appear in each term's dedicated chapter.

Accomplishment Action — A platform-defined action a core Process takes when an Objective running under it is Accomplished, such as establishing the Team; a Company's own Process carries none. See Process.

Administrative Authority — A named scope of Delegated Authority, held by a Conceptual Synthesist, permitting the establishment of other Conceptual Synthesists, the establishment of Companies, system-wide configuration, the establishment of Role classifications, and the establishment or amendment of Schemas — five independently-grantable actions, never held by a Worker.

Assignment — The operational object that binds a Team or Worker to a Project, Team, or unit of work, carrying sub-delegated Delegated Authority and Fid, and creating a corresponding Responsibility.

Audit Trail — The operational capability that records and permanently preserves every attributable action taken within The Autonomous Company, in chronological order and without alteration.

Baseline Reference — An optional Objective attribute linking a change-request Objective to a prior Objective's Outcome, granting automatic read access to its Deliverables without reopening or altering it.

Capability — The shared conceptual property establishing a specific, checkable competency a Worker holds, gating the Roles and Assignments it may be given.

Company — The highest-level Organizational Entity, established by a Conceptual Synthesist holding Administrative Authority to receive and pursue delegated Objectives.

Conceptual Synthesist — The Human Participant whose actions within The Autonomous Company are attributable through Identity, and who originates, authorizes, and oversees Autonomous Execution.

Core Definition — A Role classification, Schema, or Process established by a Conceptual Synthesist whose Administrative Authority is not restricted to a Company, available to every Company and used by each as it is: never amended, derived from, or replaced by a Company, which establishes its own, separate definition instead (Chapter 21).

Delegated Authority — The operational object establishing what a Company, Team, Worker, or Conceptual Synthesist is permitted to do, originating from a Conceptual Synthesist and optionally sub-delegated through a chain of holders.

Deliverable — The operational object representing a concrete artifact produced by a subject in fulfillment of a Responsibility.

Deliverable Type — A free, optional classification a Deliverable may carry; where a registered Schema exists for the Type, every Deliverable of that Type must conform to it.

Establishing Identity — The Identity that established a given Identity, recorded as a first-class attribute for every Identity except the Founding Identity, supporting full-chain traceability to it.

Estimated Allocation — An optional Fid attribute holding the projected consumption a Plan Deliverable estimated; Consumed amount exceeding it by a defined threshold triggers Pending Approval.

Failure — A Resource access outcome where authorization and Fid sufficiency both held but the external dependency itself did not return a usable result; distinct from Refusal and may consume Fid.

Fid — The operational object representing a bounded quantity of funds allocated to a holder of Delegated Authority, to be consumed in pursuit of a specific Objective.

Founding Act — The single, non-repeatable event through which the first Conceptual Synthesist is established, exempt from requiring prior Authentication against an existing Identity.

Gate Evaluation — The Approved, Revise-Requested, or Rejected decision recorded against a Deliverable produced under a Milestone-Gated or Step-Gated Objective, never altering the Deliverable itself.

Governing Process — An Objective attribute naming the Process it runs under, and its version: the kind of work it is. Recommended by the Leader with stated reasoning and chosen by the Conceptual Synthesist before the Plan is built; never defaulted. See Process.

Identity — The shared conceptual property that uniquely establishes and preserves the conceptual existence of a subject throughout its lifecycle.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Leader-directed — A setting held only by a core Process, such as the core Simple Process: the Process names its kind of work without sequencing it, and its Leader chooses the Deliverables and directs the work. See Process.

Leadership Authority — A named scope of Delegated Authority, held by a Worker, permitting the establishment of Teams, the creation of Assignments, and coordination of execution within a granted scope.

Lifecycle State — The shared conceptual property establishing a subject's current condition: Active, Suspended, Pending Approval, Completed, or Retired.

Newly Established — A Role attribute, true from a Role classification's establishment until its first-ever Worker Assignment is fulfilled, forcing that first Assignment into Step-Gated oversight.

Objective — The operational object establishing a specific, fixed purpose delegated by a Conceptual Synthesist to a Company.

Open Decisions — Points a Worker producing a gated Deliverable explicitly flags as requiring the evaluating Conceptual Synthesist's judgment rather than resolving silently.

Optimization Directive — One of Cost-Optimized, Balanced, or Precision-Optimized, guiding a Leader's choice among otherwise equally authorized options, most directly Resource Providers.

Organizational Accountability — The shared conceptual property establishing that every attributable action can be traced, without gap, to the Identity responsible for it.

Organizational Experience — The operational object holding synthesized patterns derived from Outcome, Deliverable, Status, and Audit Trail history.

Outcome — The operational object representing the evaluated result of an Objective's Deliverables against its success criteria, produced by a Conceptual Synthesist.

Pending Approval — A Lifecycle State condition where a subject's execution is deliberately held pending an explicit decision by the party who placed the gate; never resolves automatically.

Planning Required — An Objective attribute requiring a Plan Deliverable to be produced and Approved via Gate Evaluation before ordinary Assignment creation may begin.

Priority — An Objective ordinal used only to break ties in Worker availability among a Company's concurrently pursued Objectives; never overrides Delegated Authority, Fid, or Role.

Process — A named kind of work, and the registered, required Assignment sequence for it: ordered stages a Project must produce Deliverables for, held by a Company or

as a core Process every Company uses as it is, established directly by a Conceptual Synthesist holding Administrative Authority.

Process Establishment — The direct act, by a Conceptual Synthesist holding Administrative Authority (Chapter 21), that establishes a Process; never through any Objective's Outcome. See Process.

Process Name — A Process's human-chosen identifier.

Process Stage — One step of a Process (Chapter 28), carrying Name, General Guidance, Output Contract, Instructions, Inputs, Role and Deliverable Types. Distinct from Stage, a Deliverable's ordinal position within one Assignment's own sequence (Chapter 12).

Provider — A specific interchangeable source (for example, a model vendor) fulfilling a Resource classification; a Resource may have more than one qualifying Provider.

Recommendation — A producing Worker's specific, reasoned proposal for how to proceed, required on every Deliverable produced under Gate Evaluation.

Refusal — A Resource access outcome where authorization or Fid sufficiency was insufficient before any attempt was made; never consumes Fid; distinct from Failure.

Registering Company — The Company that established a given Process; a Process registered by one Company never governs another Company's Objectives. A core Process has no Registering Company and is available to every Company.

Resolution — The evaluating Conceptual Synthesist's recorded determination for each Open Decision a gated Deliverable identified, present only when Open Decisions exist.

Project — The operational object a Company establishes to pursue a delegated Objective, exactly one per Objective.

Resource — An external dependency — a model, tool, or external service — a Worker must be authorized and funded to access in order to exercise a held Capability.

Responsibility — The shared conceptual property establishing the obligations a subject is scoped to fulfill within its Delegated Authority.

Role — The conceptual property establishing the function a Worker performs and the Capabilities, authority, and responsibilities that function requires.

Schema — The registered, required structure for a Deliverable Type, established directly by a Conceptual Synthesist holding Administrative Authority — the same direct mechanism Role classification and Process establishment each use.

Skill — The operational mechanism through which a Worker's Capability is gained or lost.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Stage — An optional ordinal position within a sequence of Deliverables the same Assignment produces toward fulfilling a single Responsibility.

Stage Predecessor — Where Stage is present and greater than one, an identity-addressed reference to the immediately preceding stage's Deliverable.

Status — The shared conceptual property reflecting a subject's current operating standing — Good Standing, Under Review, or Restricted — independent of Lifecycle State.

Supersedes — A Deliverable attribute referencing the earlier candidate it responds to following a Revise-Requested decision, or the earlier Deliverable a post-completion revision replaces; the superseded Deliverable is never altered or removed.

Post-Completion Revision — A new Deliverable superseding one produced under a Completed Objective, authorized by a Conceptual Synthesist through a grant originated for that one revision, marked as such, never presented for Gate Evaluation, and reviewed by an Auditor's check that the authorized change was made and nothing else changed.

Team — The Organizational Entity that organizes specialized execution, composed of zero or more Workers, assignable to one or more Projects concurrently.

Worker — The Organizational Entity that performs delegated Responsibilities, holds a Role, and may spawn additional Workers where authorized.

Conceptual Model

This appendix summarizes the structural model established across the volume — the categories of concept it defines and how they relate.

Organizational Entities — perform Autonomous Execution. Company, Team, Worker.

Operational Objects — enable and bound Autonomous Execution. Objective, Project, Assignment, Deliverable, Outcome, Delegated Authority, Fid, Organizational Experience, Audit Trail.

Shared Conceptual Properties — describe entities and objects; hold no independent existence. Identity, Lifecycle State, Responsibility, Status, Organizational Accountability.

Human Participants — originate and oversee Autonomous Execution; do not perform it. Conceptual Synthesist.

Authority Scopes — named forms of Delegated Authority held by a Human Participant or Organizational Entity. Administrative Authority, Leadership Authority.

Supporting Concepts — govern what a Worker is suited and equipped to do, distinct from what it is obligated (Responsibility) or permitted (Delegated Authority) to do. Role, Capability, Skill, Resource, Schema, Process.

The mechanic connecting all categories: every consequential act traces to an originating Conceptual Synthesist through an unbroken chain of Delegated Authority and Fid, sub-delegated at each step, cascading in its revocation, and recorded without exception through Audit Trail — the mechanism that makes Organizational Accountability more than an aspiration.

Dependency Model

This appendix documents the dependency relationships among the concepts defined in this volume — which concepts require which others to be established before their own definition is complete. It reflects the actual order in which this volume was constructed, not the reading order presented in the Table of Contents.

Foundation layer — Identity and Lifecycle State are established jointly; neither exists without the other.

Property layer — Status and Role depend on Identity and Lifecycle State. Organizational Accountability depends on Audit Trail. Responsibility depends on Delegated Authority, and was therefore drafted after it despite its earlier chapter number.

Human layer — Conceptual Synthesist depends on Identity and Lifecycle State, and introduces Authentication, used nowhere else in the volume.

Authority and economics — Delegated Authority depends on Conceptual Synthesist as its originating source. Fid depends on Delegated Authority as the grant it accompanies.

Execution chain — Objective depends on Conceptual Synthesist and Company. Project depends on Objective, in a fixed one-to-one relationship. Assignment depends on Project, Team or Worker, and Role. Deliverable depends on Responsibility. Outcome depends on Deliverable and is evaluated exclusively by Conceptual Synthesist.

Reflective layer — Organizational Experience depends on Outcome, Deliverable, Status, and Audit Trail. Audit Trail is depended upon by nearly every other chapter, yet was itself drafted after Organizational Experience, since neither depends on the other.

Entities — Company depends on Conceptual Synthesist and Administrative Authority. Team depends on Assignment. Worker depends on Role and Assignment.

Authority scopes — Administrative Authority depends on Conceptual Synthesist and Delegated Authority. Leadership Authority depends on Worker and Delegated Authority.

Capability chain — Capability depends on Role. Skill is the sole mechanism through which Capability changes. Resource depends on Capability, Delegated Authority, and Fid.

Standards layer — Schema depends on Deliverable and reuses the identical direct Administrative Authority mechanism Role classification establishment uses (Chapter 23), rather than an independently invented one; it was drafted after Role for exactly that reason.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

Process depends on Deliverable, Schema and Role, and reuses the same direct Administrative Authority mechanism a third time (Chapters 23 and 27); it was drafted after Schema for the same reason Schema was drafted after Role.

The rule this graph enforces throughout: no chapter's own commands may reference a concept without at least a settled shape — either a completed chapter, or, during construction, an entry in the Interface Contracts appendix. That appendix is empty in this completed volume because every forward reference was ultimately resolved by a real chapter rather than left stubbed.

Interface Contracts

This appendix defines the minimal external shape of a concept that a Build Specification depends upon but whose full semantic definition has not yet been established in its own chapter — sufficient for dependent Build Specifications to be considered complete without requiring every concept in The Autonomous Company to be fully defined first. This does not relax Chapter 1's requirement that every concept defined within this volume be fully specified before being relied upon; it governs Build Specifications derived from this volume, not the volume's own chapters.

Registered Schemas

Empty at merge. Populated as Schemas are established directly by a Conceptual Synthesist holding Administrative Authority (Chapter 21), per Chapter 27.

Registered Processes

Empty at merge. Populated as Processes are established directly by a Conceptual Synthesist holding Administrative Authority (Chapter 21), per Chapter 28 — one real registry per Company, not one shared, system-wide registry.

Lifecycle Models

This appendix summarizes the state transitions defined across the volume's Shared Conceptual Properties and the cascading effects that follow from them.

Lifecycle State (Chapter 4) — governs every Organizational Entity, Operational Object, and Human Participant.

Active → Suspended → Active (reversible). Active → Pending Approval → Active, on explicit release by the party who placed the gate; Pending Approval → Completed, mirroring Active → Completed (never auto-resolves, unlike Suspended). Active → Completed → Active (reversible). Completed → Retired (terminal — no path out of Retired).

Status (Chapter 6) — independent of Lifecycle State; meaningful only while a subject is Active.

Good Standing ↔ Under Review → Restricted → Good Standing or Under Review. Entering Restricted automatically narrows the subject's Delegated Authority; leaving Restricted automatically restores it.

The cascading retirement chain — the pattern repeated at every level of derivation:

An Objective's retirement retires its Project, which retires its Assignments, which revokes their sub-delegated Delegated Authority, which reclaims their sub-allocated Fid to its source. The same shape governs Company → Project → Team/Worker → Assignment; Worker → spawned Worker; and Administrative Authority or Leadership Authority → everything established under them. Revoking or retiring the source collapses everything derived from it in the same instant, with the unspent economic remainder always returning to where it came from.

Responsibility's cycle (Chapter 5) — the one lifecycle with a human decision point built in:

Assigned → Active → (Delegated Authority narrows) → Suspended → Escalated → Restored (if sufficient authority returns) or Reassigned (a human decision, made by the Conceptual Synthesist or Leader who granted the underlying authority).

Gate Evaluation's cycle (Chapter 12) — the pattern behind every gated Deliverable, a Plan and the Roles it proposes included:

Produced → Approved (proceed) or Rejected (Objective retires as abandoned) or Revise-Requested → new candidate produced, Supersedes the prior one → Produced again, repeating with no fixed limit since a human evaluates every cycle.

Change History

This appendix records what each version of this volume changed, newest first, and how each change was reconciled with what came before.

Version 1.25 (September 2026) — CR-56 and the core-definitions decision.

CR-56 — Objective Type deleted; a Process names the kind of work. Chapter 9: the Objective Type values (Synthesis, Simple, Team, Custom) and their Semantics paragraphs removed; Composition's Objective Type line and its two editorial notes removed and the Governing Process attribute rewritten (every Objective runs under a Process, recommended by the Leader and chosen by the Conceptual Synthesist, never defaulted); Capabilities' Objective Type line, the Custom Behavior, the Objective-Type relationship and the four-values Constraint replaced by their Governing Process equivalents. Chapter 10: the Synthesis Assignment, its correction, the Synthesis-before-Planning bootstrap and its Behavior and Constraint removed; sequencing, Planning and milestones stated in terms of the Governing Process and the Leader-directed setting. Chapter 11: the Pending Approval exemption names the Planning Assignment only. Chapter 12: a Schema-labeling Constraint reworded. Chapter 13: the Team provision carried by the Governing Process's Accomplishment action; "any Objective Type" read as "any Objective". Chapter 21: an editorial note reworded. Chapter 23: a Plan's proposed Roles belong to any Objective's Plan, not a Synthesis Objective's. Chapter 27: "any Objective's Outcome". Chapter 28: Definition, Semantics, Purpose and Objectives restated for a kind of work; Governed Objective Type replaced by Description, Category, Accomplishment action and the Leader-directed setting; the Type-keyed selection and coexistence removed; the duplicate-Process question, the Leader-directed behavior and the Accomplishment action added, with two Constraints. Glossary: Objective Type, Custom Objective Type and Synthesis Assignment removed; Accomplishment Action, Core Definition and Leader-directed added; Governing Process, Process, Process Establishment, Process Name and Registering Company rewritten. Lifecycle Models: the Gate Evaluation cycle reworded.

Conceptual Synthesis Series
Volume 001 – The Autonomous Company

The Leader-directed setting (build list 1.1.6) — an explicit setting on core Processes only, set on the core Simple Process: an Objective under it is not Stage-sequenced and draws its milestones from its Plan; a Company Process with blank Stage fields is incomplete, not Leader-directed. Chapters 9, 10 and 28; Glossary.

Core definitions (the Conceptual Synthesist's decision, build list 1.1.6) — a core Process, Schema or Role is used by every Company as it is; a Company never amends, derives from or replaces one, and establishes its own, separate definition instead. Chapter 21 (a Behavior), Chapter 28 (a Behavior and a Constraint), Chapter 9 and the Glossary (Core Definition). This supersedes CR-56 as drafted where the draft let a Company's derived Process replace the inherited one: that text is not applied, and nothing in this volume describes derive-and-amend, copy-on-amend, or a Company definition taking a core definition's place.

Reconciliations. CR-43 (per-Company Processes, several per Objective Type, the Type-keyed automatic or explicit selection) is reversed where it conflicts with CR-56: there is no Type to group Processes by; Process Name and per-Company holding are kept. CR-44 (the Synthesis rename) is moot. CR-46's fourth Objective Type value, Custom, is removed; its direct establishment of Role, Schema and Process is kept. CR-33's milestone rule, keyed on whether a Process is registered for the Objective Type, now keys on whether the Governing Process is Leader-directed. CR-20's and CR-37's Synthesis Assignment and the Synthesis-before-Planning Constraint are removed with the Synthesis Objective; the Conceptual System Definition is the first Stage of the core Conceptual System Process. CR-35's Pending Approval exemption keeps the Planning Assignment and the Auditor review Assignment (CR-47).