memra

Autonomous Capital

Production Theory and Contract Design
for Self-Operating Systems

Amir Behbehani

February 2026

memra

Foreword

This paper was written by Memra's founder to articulate the economic and legal foundations of what we're building.

Memra deploys supervised synthetic workers — autonomous AI systems that perform complete jobs previously held by people, without continuous human involvement. But as we built this capability, we encountered a gap: there was no institutional language for what we were creating. Existing frameworks treat AI either as software (a tool to be licensed) or as labor augmentation (a productivity boost for human workers). Neither captures what synthetic workers actually are: autonomous productive assets that occupy the operative role traditionally held by human labor.

This paper introduces that language. It proposes a modification to production theory, a reframing of the firm's economic function, and a contractual structure — the Capital Asset Formation Agreement (CAFA) — that aligns legal form with economic substance.

We publish it here not as marketing, but as a contribution to the emerging discourse on AI economics. If you're building, acquiring, or thinking about autonomous systems, we hope this framework proves useful.

— Memra Research
February 2026

Abstract

The standard Cobb–Douglas production function relates output to two inputs: capital (K) and labor (L). Capital amplifies productivity, but labor supplies the operative function, the human effort that activates capital and generates output. If labor goes to zero, output goes to zero. This paper introduces synthetic labor (S) as a third input in the Cobb–Douglas framework, a non-human factor that occupies the operative role historically assigned to human workers. By incorporating S into the production function, we show that autonomous AI systems do not merely augment labor; they perform the operative function itself, thereby allowing output to persist as human operating labor approaches zero.

This modification to production theory carries a direct implication for the theory of the firm. When operational intelligence is embedded in capital rather than supplied by labor, the firm's function shifts from Coasean labor coordinator to capital allocator, acquiring and capitalizing externally produced autonomous systems rather than organizing workers to operate passive assets.

These theoretical claims have immediate institutional consequences. Firms seeking autonomous systems currently acquire them through employment and services agreements, instruments designed for labor coordination rather than asset transfer. This misclassification produces predictable failures: IP expropriation, accounting distortions, incentive misalignment, and structural debt (legal and organizational errors made upstream that become embedded in the system's architecture, yielding implementations that function in research settings but fail under operational conditions). The paper introduces the Capital Asset Formation Agreement (CAFA) as the contractual framework that operationalizes the shift from labor procurement to capital acquisition. CAFA separates asset transfer from integration labor, platform from implementation, and capital consideration from service fees, aligning legal form with the economic substance that production theory now requires.

Table of Contents

Abstract 1

Introduction 4

Contribution 6
Roadmap 7
Terminology 7
Connection to Prior Work 7
Why “Formation” Rather Than “Purchase” 7

Part I: Synthetic Labor and Capital Formation 9

Section 1: The Collapse of Production and Operation (The Technological Condition) 9
1.1 Defining Synthetic Labor (The Production Function Condition) 13
1.2 Why This Matters for Autonomous Capital Assets 16
Section 2: External Asset Creation and Zero-Cost Pre-Production (The Organizational Condition) 17
Section 3: From Renting to Owning (The Market Condition) 20
3.1 On the Distinction Between Infrastructure and Asset 21
Section 4: The Third-Party Subsidy: How Consumer-Surplus Competition Enables External Asset Formation (The Economic Condition) 23
Section 5: Why the Firm Acquires Through CAFA 25
5.1 The Demand for Synthetic Workers: Escaping the Coordination Trap 27
5.2 The Structural Failure of the Employment Contract 27
5.3 Two Systems of Comparative Advantage: Why Coordination Advantage Cannot Overcome Formation Disadvantage 28
5.4 The Probability Chain: Why Internal Builds Fail in Practice 31
5.5 The CAFA Adoption Boundary 32
5.6 Evaluation, Uncertainty, and Risk Allocation 33

Part II: The Institutional Mechanism 37

Section 6: Why Existing Instruments Cannot Clear This Transaction 38
6.1 The Employment Problem: Work-for-Hire and Pre-Existing Assets 40
6.2 The Services Problem: Pricing Effort When the Substance Is Asset Transfer 42
6.3 Structural Debt: How Misclassification Compounds During Integration 43
Section 7: The CAFA Framework 43
7.1 Overview 44
7.2 The Two-Component Structure 44
7.3 Deliverable Definition: Autonomy as the Performance Standard 46
7.4 Staged Consideration and Risk Allocation 47
Section 8: The Three-Layer Architecture 48
8.1 Layer 1: Platform (Builder Retains) 48
8.2 Layer 2: Implementation (Firm Owns Exclusively) 49

8.3 Layer 3: Infrastructure (Separate, Always OpEx) 49
8.4 Why the Builder Retains the Platform 50
8.5 Why This Architecture Resolves the Doctrinal Tension 51
Section 9: Capitalization and Accounting Treatment 52
9.1 Establishing Asset Characteristics 52
9.2 What Is Capitalized, What Is Expensed 58
9.3 Useful Life and Amortization 59
9.4 Balance-Sheet Clarity: CAFA vs. Employment 59
Section 10: Implementation Guidance 60
10.1 The Adoption Boundary: When CAFA Is Appropriate 60
10.2 For Builders 61
10.3 For Acquiring Firms 64
10.4 From Roles to Contributions 67
Section 11: Implications for the Labor Market 69
11.1 For Firms and Recruiters: Distinguishing Labor from Capital 69
11.2 For Workers: Displacement, Transformation, and Opportunity 70
11.3 For the Market: Toward Hybrid Instruments 72
Conclusion: Making Autonomous Capital Legible 73

Introduction

The canonical production function assumes that output requires both capital and labor. In the Cobb–Douglas formulation $Y = AK^{\alpha}L^{\beta}$, capital amplifies productivity, but labor supplies the operative function: the human effort that activates capital and generates output. If labor goes to zero, output goes to zero. This dependence is not incidental to the model; it reflects a structural fact about pre-AI production: capital could not act independently.

This paper argues that autonomous AI systems break this structural assumption. These systems do not merely augment labor; they perform the operative function itself. Once deployed, an autonomous system receives objectives, exercises judgment, handles exceptions, and completes tasks without requiring continuous human involvement. The act of producing the system is simultaneously the act of producing the operator.

To formalize this, we introduce synthetic labor (S) as a third input in the Cobb–Douglas framework. This non-human factor occupies the operative position historically held by human workers. The production function becomes:

$$Y = AK^{\alpha}(L + S)^{\beta}$$

This formulation modifies the canonical Cobb–Douglas framework by introducing a non-human labor factor that occupies the operative position historically held by human workers. Holding S > 0 and letting human operating labor approach zero, output no longer collapses, because the operative role has been reassigned, not eliminated.

The claim is not a metaphor. It is a claim about production theory: autonomous systems constitute a distinct productive input that is analytically separate from both capital (which is what gets operated) and labor (which carries wages, hours, exit, and non-ownership). S is ownable, persistent, and capitalizable. It does not leave at the end of the day. It is property.

The Theory-of-the-Firm Implication

This production-theoretic shift carries direct implications for how firms function.

Coase explained that firms exist to minimize the transaction costs of coordinating labor. When production required continuous human effort, it was cheaper to organize workers within a hierarchy than to contract for each task through markets. The employment contract made this possible: a continuous relationship in which the firm directs labor and owns the output of that labor.

Autonomous systems weaken this logic. When operational intelligence is embedded in the system rather than supplied by workers, the transaction-cost advantage of internalizing operating labor no longer applies. The firm's function shifts: rather than serving primarily as an organizer of labor applied to passive capital, the firm increasingly operates as a capital allocator, acquiring and capitalizing externally produced autonomous systems as internal productive assets.

This reframing is the paper's second theoretical contribution. We argue that, for a growing category of production, the firm is better understood as a capital-clearinghouse than as a Coasean labor coordinator. Employment no longer sits at the center of these workflows' production; capital formation and asset transfer do.

The Wealth Creation Implication

When a firm relies on operating labor, expenditures must be continually renewed. The work is performed, the cost is recorded, and the firm must pay again to reproduce the same function. While labor may contribute to learning or internal know-how, the productive capacity itself does not persist as an owned asset unless it is embedded in capital.

By contrast, when a firm acquires and capitalizes an autonomous system, the expenditure yields durable productive capacity. The system performs the function indefinitely without requiring continuous manual labor. What appears on the balance sheet is not a transient cost but an owned capability.

Each such acquisition establishes a fixed point in the firm's production structure. A function that previously required attention, staffing, and repeated expenditure becomes self-operating. Managerial effort and capital can then shift to the next function, the next system, the next formation decision.

This distinction underlies the paper's emphasis on capitalization. The economic significance of autonomous systems is not merely that they reduce costs, but that they transform recurring operational effort into accumulated productive assets. Over time, the firm ceases to be primarily a coordinator of labor flows and instead becomes a portfolio of self-operating systems, each representing equity in operational capacity.

This is the mechanization of comparative advantage: the economic definition of wealth creation.

The Institutional Consequence

These theoretical claims have immediate practical consequences. If autonomous systems are factors of production that can be formed externally and transferred as capital, there must be an institutional mechanism through which such transfers occur.

No such mechanism currently exists.

Firms seeking autonomous systems acquire them through employment and services agreements, instruments designed for labor coordination rather than asset transfer. Employment assumes the firm generates productive assets internally through supervised labor. Services agreements price effort when the economic substance is an asset transfer. Neither can express the acquisition of a self-operating productive system formed on pre-existing technical capital.

The result is systematic misclassification. Builders arrive with substantially formed assets and are priced as if selling time. IP assignment clauses capture work the firm did not fund. Balance sheets show no asset; income statements take the full hit. The labor market can discover these assets (builders appear as candidates and consultants), but it cannot transact them.

This paper introduces the Capital Asset Formation Agreement (CAFA) as the contractual framework that operationalizes the shift from labor procurement to capital acquisition. CAFA separates asset transfer from integration labor, platform from implementation, and capital consideration from service fees. It provides institutional scaffolding for a transaction that production theory now requires but existing legal forms cannot express.

Contribution

This paper makes three contributions.

First, we introduce synthetic labor (S) as a new factor of production and incorporate it into the Cobb–Douglas production function. This modification to the classical capital-labor framework provides the analytical foundation for understanding autonomous systems as productive inputs rather than tools that augment human effort.

Second, we reframe the theory of the firm. When operational intelligence is system-endogenous rather than labor-supplied, the firm's economic role shifts from a Coasean labor coordinator to a capital allocator of autonomous productive assets. This reframing explains why employment — the institutional mechanism built for labor coordination — systematically fails when applied to autonomous system acquisition.

Third, we introduce the Capital Asset Formation Agreement (CAFA) as the contractual structure that operationalizes these theoretical insights. CAFA provides mechanisms to separate platform from implementation, asset transfer from integration labor, and capital consideration from service fees — aligning legal form with economic substance.

Roadmap

The paper proceeds in two parts.

Part I establishes the economic foundation. Section 1 develops the production function analysis, introduces synthetic labor, and demonstrates why autonomous systems require a modification to the standard capital-labor framework. Sections 2-4 identify the conditions under which external asset formation becomes viable: zero-cost pre-production, the shift from renting to owning, and the third-party subsidy that enables builders to form assets without firm-internal resources. Section 5 explains why firms rationally acquire through CAFA rather than hiring internally and introduces the two-component transaction structure.

Part II introduces the Capital Asset Formation Agreement. Section 6 explains why employment and services agreements cannot express autonomous system acquisition. Sections 7-8 specify CAFA's contractual architecture and three-layer ownership structure. Section 9 addresses capitalization and accounting treatment. Section 10 provides implementation guidance. Section 11 situates CAFA within broader labor market implications.

Terminology

Throughout this paper, we refer to the acquisition of externally formed, firm-specific Autonomous Capital Assets as an Autonomous Capital Asset Transaction. The Capital Asset Formation Agreement (CAFA) is the proposed, standardized contractual structure used to execute such transactions; it is not an actor, but a legal form adopted by the parties.

Connection to Prior Work

This framework operationalizes the shift from role-based employment to contribution-based capital formation. While prior work identified the economic necessity of compensating strategic expertise as capital rather than labor, this paper provides both the production-theoretic foundation and the legal mechanism required to implement the shift.

Why "Formation" Rather Than "Purchase"

The Capital Asset Formation Agreement is not an asset purchase agreement. The distinction is structural, not semantic.

An asset purchase agreement presumes the existence of a fully formed asset that transfers at closing. The buyer evaluates, negotiates, and acquires an existing asset in its completed form.

This structure suits assets developed for distribution, namely products designed for multiple customers, where the seller's role ends at transfer.

Autonomous Capital Assets do not fit this model. They are designed for acquisition by a single firm. The builder arrives with proprietary methods, including novel approaches to autonomous task execution, command interpretation, operational binding, and deterministic control, developed and protected prior to any engagement. But these methods do not constitute a finished asset. The asset is not complete until the builder applies these methods to produce a firm-specific instance, configured for the acquiring firm's infrastructure, data, and operational context.

The term "formation" captures this structure. The builder contributes pre-existing intellectual property, including methods, architectures, and accumulated expertise, that enable the rapid construction of autonomous systems. The firm provides the operational context that specifies what the instance must do. What transfers is not the platform itself, understood as the builder's generative capability, but an instance: a firm-specific implementation produced by that platform. The builder retains the ability to produce future instances for other firms; this firm acquires exclusive ownership of this particular instantiation.

Technically, formation relies on a clear boundary between what can be developed externally and what must be completed within the firm. The builder's proprietary methods, including techniques for task execution, control binding, and autonomous operation, are in place prior to engagement and do not require access to firm-specific systems. Completion, however, necessarily occurs through integration, which involves applying those methods to the firm's infrastructure, data flows, and operational decision context. Formation bridges this boundary by combining externally developed methods with firm-specific instantiation, without requiring either the builder or the firm to internalize the other.

Importantly, this structure does not collapse into a labor or services agreement merely because integration work occurs or because the builder may remain involved after formation. The defining feature of the transaction is not the presence or absence of ongoing activity, but the existence of a distinct, capitalizable asset that is formed by applying pre-existing methods to a firm-specific context in exchange for consideration. The builder's contribution to formation is finite, front-loaded, and separable: the application of proprietary methods and expertise to produce a discrete, transferable instance. Whether the builder later provides stewardship, support, or additional services, or whether the firm ultimately absorbs the builder's entity, is orthogonal to the formation of the asset itself and does not retroactively convert the transaction into labor provision. Accordingly, the agreement governs asset formation and acquisition as a structural matter, even if subsequent arrangements govern operation, maintenance, or further development.

A formation agreement establishes the baseline probability by specifying acceptance criteria, integration scope, and the pathway to completion. Under applicable accounting standards, an asset must be "probable to complete" to be recognized as such[1]. The asset is not purchased as-is, but formed to specification, with capital committed upon demonstrated value. The Capital Asset Formation Agreement, therefore, governs a different kind of transaction than a purchase agreement. It structures the process through which a bespoke, firm-specific instance comes into existence, separating what the builder contributes, namely proprietary methods, platform capability, and prior intellectual property, from what emerges through integration, namely the firm-specific implementation.

The agreement governs both formation and acquisition. What distinguishes it from a standard asset purchase agreement is that the asset requires firm-specific completion before transfer. The builder arrives with substantially formed methods and platform capabilities; the engagement completes the asset by integrating it into the firm's operational context. Asset consideration compensates prior formation; integration fees compensate completion work. Both are necessary, but they are priced and accounted for distinctly.

Part I: Synthetic Labor and Capital Formation

Section 1: The Collapse of Production and Operation (The Technological Condition)

Historically, in the absence of embedded operational intelligence, capital assets lacked the capacity to originate and coordinate productive activity without ongoing human intervention. Operational intelligence, therefore, resided outside the capital asset itself, remaining embodied in human operators rather than embedded within the system. This two-phase model was not a historical accident. It followed from the nature of capital as inert. Factories, machine tools, and software systems could *amplify* human intent, but they could not originate or execute action independently. A lathe does not machine parts on its own; an accounting system does not reconcile ledgers without human direction. Because capital was passive, labor had to perform two different economic functions: Direct Labor to create the system, and operating labor to use it productively. The entire managerial hierarchy of the firm evolved to coordinate this ongoing operating labor.[2]

[1] On the "probable to complete" requirement for capitalization of internal-use software costs, see FASB ASC 350–40–25–2.
[2] Chandler, Alfred D. *The Visible Hand: The Managerial Revolution in American Business*. Cambridge, MA: Harvard University Press, 1977; Coase, Ronald H. "The Nature of the Firm." *Economica* 4, no. 16 (1937): 386–405.

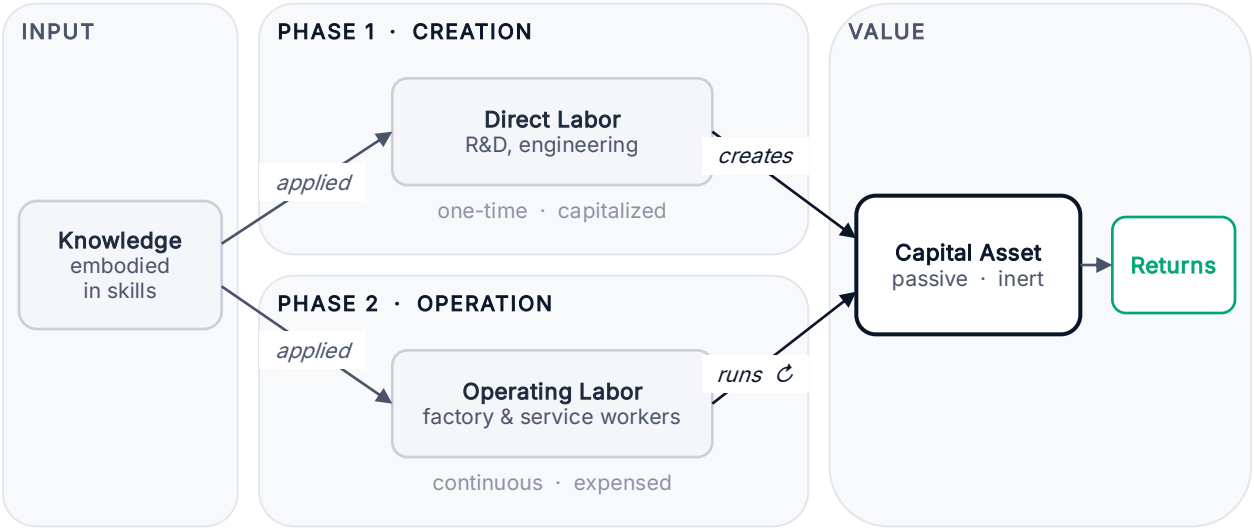
This distinction shaped both firm structure and accounting practice. Direct labor, including engineers who design production lines and developers who write software, creates the asset and is therefore capitalized, with costs recorded on the balance sheet and amortized over the asset's useful life. Operating labor, consisting of workers who run the line or operate the software, extracts value from the asset and is expensed on a continuous basis as an operating cost.

Indirect labor, including supervisors, coordinators, and managerial layers, existed solely to organize and monitor operating labor. It did not constitute a distinct productive function. Instead, it arose structurally from the requirement that humans operate the system. As operating labor declines, the need for indirect labor declines with it.

Autonomous AI systems collapse this separation. The act of producing the system is simultaneously the act of producing the operator. Once deployed, the system performs the work directly, without requiring the continuous operating labor that industrial and software-era production assumed as a constant. This collapse of the creation-operation boundary shifts the economic nature of value creation: expertise front-loads into the design of a self-operating, active capital asset rather than being embodied in ongoing human labor.

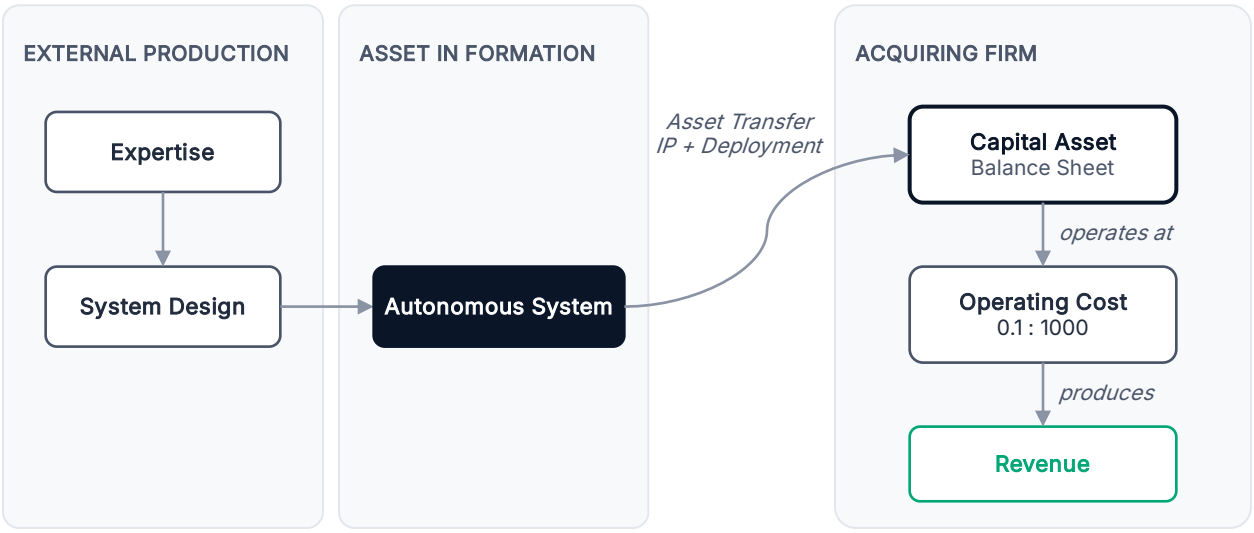
This inversion is fundamental to the production logic. We express this structural difference in two flows:

Traditional Two-Phase Production (Labor-Mediated Capital)



- **Phase 1 (Creation):** Direct labor (e.g., R&D, engineering) applies knowledge to create the passive capital asset. This labor is a one-time capitalized cost.
- **Phase 2 (Operation):** Operating labor (e.g., factory/service workers) is continuously applied to *run* the passive asset. This labor is an ongoing expense.
- The asset is inert; its operation depends entirely on labor input.
- A structural separation exists between asset *creation* and *operation*. The firm's managerial hierarchy exists to coordinate labor operations.

Single-Phase Production (Autonomous Capital)



- **Step 1 (Creation):** Expertise is encoded *once* (externally) into a System Design, which creates a self-operating system. This single act *replaces both* direct labor and operating labor.
- **Step 2 (Acquisition):** The autonomous system is transferred to the firm as a single asset, recognized on the balance sheet as a Capital Asset.
- **Step 3 (Operation):** The Capital Asset operates itself. The system eliminates the need for operating labor and operates continuously without human involvement. The firm acquires an active asset that acts as its own operator, fundamentally shifting the economic logic from operating expenses to capital formation.

When a firm deploys an AI-based invoice processing system, it eliminates the need for manual labor to extract data, classify line items, route approvals, or update accounting records. Constructing the system by encoding workflow logic, shaping task decomposition, and defining exception handling simultaneously creates the operator itself. Once deployed, the system performs the work directly. No second phase follows. The role historically assigned to operating labor no longer occurs.

Traditional automation preserved the two-phase structure. Robots are capital assets produced through direct labor, yet they still depend on operating labor, such as technicians, line supervisors, and quality inspectors. Software automation followed the same pattern. Enterprise systems are built once but require continuous operation by users and administrators. Autonomous AI systems break this pattern. They do not merely reduce the need for operating labor; they eliminate the role.

The economic consequence is immediate. As operating labor approaches zero, the firm's recurring cost structure collapses toward infrastructure costs, including cloud compute, model inference, and occasional system refinements. Value creation shifts toward system design. The better the initial construction, the less ongoing intervention the system requires. This reallocation of productive effort from continuous operation to upfront construction inverts the traditional labor-capital relationship. Rather than labor continuously activating passive capital, builders apply expertise once to create active capital that operates indefinitely.

The cost flow difference illustrates the shift. In the traditional model, a firm hires an accountant, whose labor is expensed annually, to process invoices using accounting software that is depreciated over its useful life. Both costs persist as long as the work continues. In the autonomous model, the firm acquires a system that processes invoices without requiring accountants to do so. The operating labor cost disappears, leaving ownership of a capital asset and its infrastructure footprint.

This shift is not speculative. Firms already operate systems that perform legal review, customer communication, data extraction, analytical reporting, and software code generation with

minimal human oversight. As model performance improves and organizations adopt architectures that embed operational behavior within the system itself, cognitive work migrates from labor expense to capital asset.

We refer to such systems as **Autonomous Capital Assets**: self-operating, firm-specific productive systems whose economic value is realized through ownership and integration rather than through ongoing labor or licensed access. The term captures three defining properties:

- **Autonomous**: The system performs operational functions without continuous human labor. It is not a tool that amplifies human effort; it is an operator that replaces it.
- **Capital**: The system is a durable, productive asset, not a service or subscription. It generates economic returns over multiple periods and is owned rather than rented.
- **Asset**: The system is identifiable, controllable, and transferable. It can be separated from its creator, assigned to a firm, and recognized on a balance sheet.

Autonomous Capital Assets are distinct from traditional software, which requires ongoing human operation; from SaaS products, which are licensed rather than owned; and from human labor, which must be continuously coordinated and compensated. They represent a new category of productive capacity, one for which existing contractual and accounting frameworks lack a recognized transactional pathway.

The remainder of Part I specifies the economic conditions under which Autonomous Capital Assets can be formed externally and transferred to firms. Part II introduces the Capital Asset Formation Agreement (CAFA), the contractual structure through which such transfers occur.

The two-phase production model — build the tool, then use the tool — ceases to apply when the tool uses itself. This collapse is the first structural condition enabling the external formation and transfer of Autonomous Capital Assets: when production and operation are no longer separable, the builder transfers in embodied form at delivery, and no ongoing labor relationship is required.

1.1 Defining Synthetic Labor (The Production Function Condition)

To understand how autonomous systems differ from traditional capital, it is helpful to begin with the structure of standard production functions. In the canonical two-input framework of human labor L and capital K, a Cobb–Douglas form,[3]

[3] Cobb, Charles W., and Paul H. Douglas. "A Theory of Production." *American Economic Review* 18, no. 1 (1928): 139–165.

$$Y = AK^{\alpha}L^{\beta}$$

contains a built-in assumption: something must operate the capital. Historically, this role was filled by human labor. The mathematics does not require that the operator be human, but it does require that an operator exist. If operating labor goes to zero, output goes to zero.

This dependence of output on the presence of an operating agent is not resolved by "better technology." Total factor productivity A scales the effectiveness of inputs, but it does not replace inputs. Increasing A makes each unit of capital and labor more productive; it does not supply an operating agent. As shown above, even with arbitrarily high A, if $L \to 0$, then $Y \to 0$.

The implication of the preceding result is straightforward. The Cobb–Douglas production function does not require *human* labor per se, but it does require an **operating input**. If output persists as human-operated labor approaches zero, then some other input must be occupying the operative position previously held by L. The two-factor model lacks a variable that can represent this condition.

To represent this analytically, we introduce a third input, S, which we term a *synthetic labor unit*. S is defined as a non-human operating input capable of acting on capital to produce output autonomously. S is analytically distinct from L because it supplies the operative function without possessing the economic or legal attributes of labor. *The operative capacity represented by S is artificial operational intelligence. It occupies the operative role in production, receiving objectives, exercising judgment, handling exceptions, and carrying tasks through to completion.*

Formally, the production function may be written as:

$$Y = AK^{\alpha}(L + S)^{\beta}$$

This formulation preserves the structural role historically occupied by labor in the production function, while permitting either human or synthetic operator to perform that role.

Holding $S > 0$ and letting human operating labor approach zero:

$$\lim_{L \to 0} Y = \lim_{L \to 0} A\,K^{\alpha}(L + S)^{\beta} = AK^{\alpha}S^{\beta} > 0$$

Output no longer collapses because the operative role required by the production structure has not disappeared; it has been reassigned. As human labor L approaches zero, tasks may shift into capital K and productivity A, but the operative role itself must still be supplied; S represents that role when it is no longer human.

Importantly, S is not introduced to "patch" the mathematics. It is introduced because the mathematics already presumes the existence of an operator. What has changed is not the necessity of that role, but the identity of the entity performing it. Human labor L and synthetic labor S occupy the same structural position in the production function: both act on capital to generate output.

At this stage, S is purely an analytic placeholder. Its introduction does not yet make claims about ownership, contracts, or legal form. It simply makes explicit a production condition that the two-input model cannot represent: sustained output with zero human operators.

Historically, advances in capital have displaced many tasks previously performed by human workers. Mechanical systems, industrial automation, and software reduced the amount of human effort required to produce output, but they did not eliminate the role of labor itself. A human operator remained responsible for receiving directives, interpreting objectives, coordinating activity, handling exceptions, and ensuring completion. Capital replaced tasks; labor continued to supply the operative function.

Collapsing S into A would be a category error: total factor productivity scales the effectiveness of inputs, but it does not perform work. Collapsing S into K would be a specification error: it would reintroduce the original problem, since the two-factor form $AK^{\alpha}L^{\beta}$ cannot represent $L \to 0$ without forcing $Y \to 0$. Traditional capital is what gets operated; S, like L, is what does the operating.

Nor can S be modeled as a form of L. Labor represents human effort and carries wages, hours, exit, and non-ownership. S, by contrast, is ownable, persistent, and capitalizable. It does not leave at the end of the day. It does not negotiate compensation. It is property.

In production-function terms, S occupies the operative role historically assigned to labor. In legal and accounting terms, it is treated as capital because it is embodied in a capital asset. The introduction of S therefore does not violate the logic of Cobb–Douglas; it preserves it. The model has always required an operator. Autonomous systems make explicit that this role need not be human.

As an accounting matter, the synthetic labor unit is not capitalized as a separate unit. When a firm acquires an Autonomous Capital Asset, it capitalizes the system that embodies S, the operative capacity that allows the system to act on capital and produce output. In some cases, this capacity is embedded in physical capital that the firm also acquires, as with an autonomous vehicle. In other cases, it is embodied in software that operates on infrastructure the firm already owns or rents. In all cases, S denotes the operative dimension of the capitalized item, not a separately capitalized asset. Accounting describes how autonomous systems are recorded once acquired. Production theory explains why firms rationally acquire systems rather

than hire operators, and, as the following section addresses, how such acquisitions should be structured.

Contractual Implication

When a system does not internalize the operative role and human labor remains necessary to interpret directives, coordinate activity, and ensure completion, the system functions as capital that augments labor. Firms hire employees and equip them with such tools within a standard labor–capital framework.

When a system internalizes the operative role by performing an end-to-end operational function without continuous human involvement, no labor is required to staff it. S more accurately represents the system's contribution than L. The relevant transaction is therefore not the hiring of labor to operate capital, but the acquisition of a self-operating productive capacity.

The Capital Asset Formation Agreement (CAFA) is the contractual structure designed to govern transactions in precisely this latter case: when the firm acquires an Autonomous Capital Asset that occupies a factor-of-production role not captured by the standard two-input model.

1.2 Why This Matters for Autonomous Capital Assets

Once we recognize, from the production function analysis above, that these systems behave as synthetic labor units[4], a set of implications follows:

- They cannot be priced or procured through labor contracts because they are not labor.
- They cannot be purchased off the shelf as software, because they are bespoke rather than generic.
- They must be constructed, configured, and transferred as capital assets.
- Their creation requires a governance and contracting structure oriented toward asset formation rather than employment.
- Their integration requires operational knowledge transfer closer to the induction of a new worker than to the installation of equipment.

[4] By "synthetic labor unit," we mean an autonomous capital asset that, within a bounded operational domain, is delegated outcome responsibility and substitutes for human labor input as a productive factor rather than serving as labor-augmenting capital. This usage is economic rather than legal; it does not imply personhood, employment status, or treatment as labor under existing law.

Taken together, these properties define Autonomous Capital Assets: systems capable of independent operational judgment, occupying a factor-of-production role that is neither reducible to labor nor to conventional capital.

The distinction is not semantic. It determines the appropriate economic and legal treatment. Traditional capital increases the marginal product of labor; it complements labor by increasing productivity. Autonomous Capital Assets, by contrast, are themselves productive factors. They do not merely augment labor; they perform operational functions that were previously dependent on human workers. In the production function framework developed above, they enter as a separate input S, contributing directly to output. Because they occupy the operative role rather than merely augment it, they exhibit high substitutability with human labor. They transition from complements to substitutes, not replacing tools but replacing the workers who would have used said tools.

The difficulty, therefore, is not that employment contracts are defective instruments. The contracting error arises upstream, at classification, not downstream, at execution. Employment treats system creation as an open-ended labor process, with output accruing incidentally over time. Autonomous Capital Assets, by contrast, are discrete capital goods: they have definable boundaries, completion states, and transferability. Hiring labor to "build internally" collapses asset formation into an ongoing effort, obscuring ownership, blurring incentives, and internalizing development risk that properly belongs with the asset's formation. A properly structured acquisition begins with the correct premise that the firm is commissioning and acquiring an externally formed Autonomous Capital Asset, and the initial bargain must reflect that premise. The Capital Asset Formation Agreement is the contractual form designed for this purpose.

Section 2: External Asset Creation and Zero-Cost Pre-Production (The Organizational Condition)

Section 1 established a production-function condition under which autonomous systems internalize the operative role traditionally supplied by human labor. In those cases, operating labor does not disappear; it is reconstituted as a synthetic labor unit (S) embedded within the system itself.

This production condition, however, does not by itself explain how such systems originate outside the firm. Section 1 established that autonomous systems can *operate* without ongoing labor once deployed. But the ability to operate independently says nothing about how such systems originate externally. A system could be autonomous in operation yet still require firm resources, firm data, or firm supervision during its initial construction.

A second condition must therefore hold: the system must be capable of being formed outside the firm as a discrete capital asset, even if firm-specific integration continues after transfer. This is the organizational condition.

What historically prevented external formation

Historically, bespoke productive systems were built inside firms, not because they required continuous human operation, but because the knowledge required to construct them was inseparable from the organization itself. Human operators embodied the intelligence needed to design, coordinate, and refine firm-specific systems through their ongoing involvement in internal workflows and tacit practices. Accordingly, firms co-located production and operations within the organization because the system relied on organizational membership.

Even when capital goods were clearly separable from labor in operation, this constraint remained binding. Firms routinely purchased standardized capital externally, but systems tailored to a specific firm's processes, data, and decision structures still had to be produced internally through supervised employment. Autonomous operation, by itself, did not dissolve this organizational dependency.

The organizational break

External asset formation becomes viable only when this historical dependency severs. That break requires a convergence of structural shifts that separate system construction from firm membership.

First, production intelligence becomes firm agnostic. The knowledge required to construct autonomous systems no longer resides primarily in workers embedded within a particular organization. It increasingly resides in foundation models, reusable orchestration frameworks, and standardized tooling that are accessible outside the firm. Builders inside and outside enterprises now draw from a shared, exogenous substrate. Because the core production intelligence is no longer proprietary to the firm, the firm is no longer structurally required as the site of creation.

Second, development costs collapse, enabling external pre-production. In earlier software regimes, constructing advanced systems required institutional-scale engineering teams, infrastructure, and prolonged supervision. Those resources were typically available only within firms. Foundation models invert this structure. The marginal cost of developing a bespoke autonomous system has fallen to the point that builders can self-fund substantial portions of development before any contractual relationship exists. Creation no longer requires the firm to supply labor, capital, or managerial oversight ex ante.

Third, as established in Section 1, operational intelligence becomes system-endogenous. Once deployed, the system no longer depends on the firm's operating labor to generate output, ensuring that, after transfer, the asset does not require employment to function, a necessary complement to external formation.

Organizational portability

Together, these shifts remove the historical constraints that tied the creation of bespoke systems to the firm. Externally accessible production intelligence and system-embedded operational intelligence enable builders to finance development independently, making asset formation organizationally portable. What emerges is a third structural category between internal development and full organizational acquisition:

- **Internal build** remains appropriate when operational intelligence cannot be specified in advance and must be shaped through ongoing interaction with the firm.
- **Acquisition** remains appropriate when operational intelligence is inseparable from the organization that developed it.
- **External asset formation** becomes viable when operational intelligence can be fully encoded, production intelligence is accessible outside the firm, and the system can function autonomously after transfer.

In this third regime, the transferred object is neither labor, access to shared software, nor an acquiring organization. It is a bespoke, self-operating capital asset. The builder retains the reusable production substrate; the firm acquires the specific implementation as productive capital.

Why employment no longer fits

Because the builder now arrives with a partially formed capital asset, the foundational employment assumption that firms generate productive assets internally through supervised labor no longer applies. Employment governs the coordination of workers within the firm's boundary. It does not provide a mechanism for acquiring capital formed externally.

Once the locus of creation shifts outside the firm, the economic relationship must bifurcate: governance of production remains with the builder, while ownership of the resulting asset and the knowledge required to steward it transfer to the firm.

The Capital Asset Formation Agreement (CAFA) formalizes this bifurcation. It provides a structure for transferring an externally formed autonomous system as capital while preserving the continuity of operational knowledge typically achieved through employment, without requiring the firm to assume platform ownership, R&D obligations, or supervisory burden.

Section 3: From Renting to Owning (The Market Condition)

Sections 1 and 2 established that Autonomous Capital Assets internalize the operative role (S) and can be formed externally as discrete capital assets. This section addresses a different question: why has exclusive ownership of such systems become economically rational, rather than reliance on shared software? The answer is not simply that software became cheaper. Ownership, previously impractical for most firms, has become organizationally and economically feasible.

Historically, firms faced a binding tradeoff. Firms could build systems tailored to their workflows, data, and governance constraints, but only with large internal teams, prolonged development, and sustained coordination. Standardized software reduced this burden but required firms to adapt their operations to external designs. For most firms, owning bespoke systems was not merely expensive; it lay beyond their organizational reach.

SaaS prevailed under these conditions. Vendors amortized fixed development costs across many customers. Firms accepted recurring fees, limited customization, and vendor dependency in exchange for access to functionality they could not otherwise afford to build or maintain. Renting was not preferred; for most firms, it was the only practical option. That constraint has now relaxed. Foundation models, standardized tooling, and reusable orchestration frameworks have sharply reduced both the cost and organizational complexity of constructing firm-specific systems. As Section 2 established, external builders can self-fund substantial pre-production, so the firm need not finance development as labor; instead, it acquires the result as capital.

The effect is not merely lower cost. It is an expansion of what firms can practically achieve. Ownership of bespoke systems, once confined to firms with deep technical capability, is now accessible to firms that lack internal development capacity but can acquire externally developed assets.

SaaS remains appropriate for commodity functions where differentiation is irrelevant and standardization is advantageous. But for core workflows where firm-specific logic, proprietary data, and competitive advantage reside, ownership now predominates. The firm can select software that aligns with its operations rather than reshaping operations to fit software.

Autonomous operation strengthens this case but does not create it. Even without operational autonomy, the feasibility of bespoke construction shifts the calculus toward ownership. Autonomy amplifies the advantage by eliminating ongoing labor dependency; the system not only exists but also runs independently. What was previously an operating expense is becoming a capital investment, realized over the asset’s useful life.

The transition from renting to owning is therefore not a rejection of SaaS, but a consequence of expanded feasibility. What once required sharing to afford can now be owned outright. Autonomous Capital Asset transactions are executed under a Capital Asset Formation Agreement. This shift from renting to owning raises a necessary clarification: if autonomous systems rely on shared cloud models and infrastructure, what exactly is being owned, and how does it differ economically from a subscription service?

3.1 On the Distinction Between Infrastructure and Asset

If autonomous systems rely on large language models and cloud infrastructure, why are they not simply another form of software subscription or configurable infrastructure service: an operating expense like any other SaaS tool? The answer lies in the accounting distinction between variable inputs and durable productive capacity.

The underlying infrastructure (compute, hosting, storage, and model inference) is an operational expense. The firm pays for processing capacity as it is consumed, no differently in economic substance from electricity, bandwidth, or cloud storage. They rent, meter, and expense these inputs as incurred.

The autonomous system built on that infrastructure, however, is not rented. It is created, transferred, and owned.

These systems are not merely automating tasks or exposing configurable tools. They perform complex operational functions that were previously the exclusive domain of human workers. What enables this shift is a separation that did not previously exist: general knowledge now resides in the foundation model, while expertise, the application of that knowledge to a specific firm's operational context, is encoded in the system itself. The model provides the capacity to reason; the system specifies what to reason about, how to act, and how decisions map to the firm's internal controls and workflows.

This system layer is firm-specific, non-transferable to other organizations, and valuable precisely because it performs work that would otherwise require ongoing human labor. It therefore constitutes a discrete productive asset.

Under applicable accounting standards for intangible assets, capitalization eligibility turns on three conditions that SaaS subscriptions cannot meet but autonomous systems can:

- **Identifiability.** The system is a discrete, separable implementation rather than a shared or pooled platform.

- **Exclusive Control.** The firm owns the implementation outright and can restrict access, modify its behavior, and deploy it as it chooses.
- **Future Economic Benefit.** The system generates value over multiple periods by performing operational work that would otherwise require continuous labor expense.

The final accounting treatment remains subject to management judgment and auditor review; these criteria establish eligibility, not a mandated classification.

In a SaaS model, the firm rents a tool. In this model, the firm owns the operator.

The system layer is the asset governed by the Capital Asset Formation Agreement: the autonomous system that performs work on behalf of the firm. The hosting and inference costs remain operational expenses. The system itself is a capital asset. The transaction separates these layers; the Capital Asset Formation Agreement provides the contractual framework for doing so.

To be clear, firms already capitalize labor costs when they directly supervise the creation of internal-use software or equipment. That process, however, presumes a passive asset, which still requires continuous operating labor to generate returns. Autonomous systems differ in two decisive ways:

1. The asset itself performs the operating function, driving operating labor toward zero, and
2. The system is substantially pre-constructed using the builder's own platform, tooling, and accumulated methodology, and is completed through firm-specific integration to render the system self-operating.

The asset is therefore neither entirely created within the firm nor delivered as a generic finished product. It is **substantially formed before engagement and completed through asset-forming integration labor**, undertaken precisely to eliminate future operating labor. The result is a completed productive asset that operates independently once placed into service.

Section 4: The Third-Party Subsidy: How Consumer-Surplus Competition Enables External Asset Formation (The Economic Condition)

Section 3 established the demand-side condition: ownership of bespoke systems can become economically rational for firms. This section establishes the supply-side complement: why external formation can become the dominant site of capital creation even when firms could, in principle, build internally. The explanation is not technological alone. It lies in the location of productive inputs.

Historically, firms have generally insisted on building firm-specific systems internally to retain ownership of proprietary technology, protect confidential information, and preserve competitive advantage. Because the firm supplied the production environment, including capital, infrastructure, supervision, and direction, the work-for-hire doctrine treated the resulting output as created at the firm's instance and expense[5], thereby entitling the firm to ownership. That doctrinal structure made internal employment the natural mechanism for system creation: the firm's need for ownership and control was satisfied precisely because it supplied the productive inputs that triggered legal ownership of the output.

That condition no longer universally holds.

For autonomous systems, a substantial portion of the decisive productive inputs, including reasoning capability, planning capacity, code generation, iterative refinement, and testing, are now supplied by third parties at marginal cost. Foundation-model providers, competing to maximize adoption and ecosystem lock-in, make increasingly capable models widely accessible. This dynamic functions as a third-party subsidy to system builders: providers externalize the cost of intelligence creation because they derive value from scale rather than exclusivity.

Crucially, this subsidy operates prior to any relationship with the acquiring firm. Builders do not require membership in a firm, internal infrastructure, or supervised employment to access these inputs. They can construct substantial portions of a system ex ante, using externally provided intelligence rather than firm-supplied resources.

This shift is not merely a reduction in development cost. It is a relocation of comparative advantage. Even when firms possess internal technical capacity, they are no longer necessarily the lowest-cost or most capable sites for capital formation. External builders can draw on subsidized model intelligence and accumulated insights from prior deployments and

[5] On the "instance and expense" test for work-for-hire, see *Marvel Characters, Inc. v. Kirby*, 726 F.3d 119 (2d Cir. 2013).

engagements, advantages that individual firms cannot replicate internally. Internal creation remains possible, but it is increasingly organizationally burdened and economically disadvantaged.

As a result, capital formation can drift outward. Builders form assets using inputs that the firm neither finances nor supplies; firms encounter systems that are already substantially formed before any employment relationship exists. Under these conditions, firms increasingly encounter systems whose productive formation occurred elsewhere, shifting their involvement from enabling production to evaluation, integration, and capitalization.

This shift exposes the limits of employment as a procurement mechanism. The issue is not that employment is obsolete, but rather that employment presumes that the firm supplies the production environment. Where third-party inputs rather than firm resources power asset formation, that presumption fails from the outset. The firm is not coordinating production; it is encountering the product of production.

This economic structure requires a contractual form that can express it. The Capital Asset Formation Agreement provides a framework for acquiring externally formed capital assets while preserving the exclusivity required for capitalization. Rather than forcing capital acquisition into a labor framework designed for creation only at the firm's initiative and expense, this structure aligns ownership with the actual site of financing and formation. It recognizes that the firm acquires what it did not produce.

The Coasean Implication

This shift reframes the firm's economic role.

Coase explained that firms exist to minimize the transaction costs of coordinating labor.[6] When production required continuous human effort, it was cheaper to organize workers within a hierarchy than to contract for each task through a market. The employment contract made this possible: a continuous relationship in which the firm directs labor and owns the output of that labor.

Autonomous Capital Assets weaken this logic for a specific and growing category of production. Embedding operational intelligence within the system eliminates the need for operating labor to coordinate workflows. Task execution, routing, and decision-making are performed by the capital asset rather than by supervised workers. The transaction-cost advantage that once justified internalizing operating labor no longer applies.

[6] Coase, Ronald H. "The Nature of the Firm." *Economica* 4, no. 16 (1937): 386–405.

The firm does not disappear. Its function changes.

Rather than serving primarily as an organizer of labor applied to passive capital, the firm increasingly functions as a capital allocator. When a firm acquires an autonomous system from an external builder, the builder's accumulated expertise is converted into financial capital, which in turn is converted into productive capacity for the firm. When operational intelligence is system-endogenous, and production inputs are supplied externally, the firm's role shifts from coordinating production to evaluating, integrating, and capitalizing assets formed elsewhere.[7]

In this configuration, the firm becomes a capital clearinghouse. For these workflows, employment no longer sits at the center of production; capital formation and asset transfer do. The contractual structure must align legal form with economic substance by treating autonomous systems as capital assets rather than misclassified labor. CAFA provides that structure.

Section 5: Why the Firm Acquires Through CAFA

Sections 1–4 established that the cost of building autonomous systems has collapsed to the point where they can be formed externally as completed productive assets. What remains is the firm's decision problem: why would a rational firm use a Capital Asset Formation Agreement to acquire these systems instead of simply hiring employees to build them internally?

The Structural Mismatch at the Procurement Stage

The production inversion introduced by autonomous AI systems creates a structural mismatch at the procurement stage. Firms seeking to obtain such systems are currently sourcing through the labor market, attempting to hire individuals to "build internally," rather than acquiring the system itself as a capital asset.

This misclassification is the legacy of a binary inherited from industrial and software-era economics: either a firm hires workers who build capability internally (labor → wages → OpEx), or it licenses standardized SaaS products shared across many customers (subscriptions → OpEx). Under this binary, firms fail to recognize a third path now made possible by falling system development costs and platform-driven reuse: bespoke, self-operating systems can be produced externally and acquired as exclusive capital assets.

[7] Williamson, Oliver E. *The Economic Institutions of Capitalism*. New York: Free Press, 1985.

Markets of One and the Absence of Price Discovery

Bespoke autonomous systems are, by construction, **markets of one**. They are firm-specific, non-substitutable, and non-comparable ex ante. Their value depends on deep integration with a particular firm's controls, data, and workflows; outside that context, assessment is meaningless. As a result, these systems do not admit standard price discovery, comparability, or aggregation.

Markets typically coalesce around commonality and repeatability. Labor markets function because human effort is substitutable across firms, and software markets function because vendors can amortize standardized tools across many customers. In contrast, a bespoke autonomous system is neither substitutable nor reproducible prior to formation. Search costs are therefore prohibitively high, and bargaining costs dominate any potential transaction.

This absence of price discovery does not imply the absence of demand or productive capability. Firms routinely seek systems that tightly integrate with their internal environments, and specialized builders can develop such systems. The absence is institutional: without a recognized transactional structure, these exchanges cannot reliably occur. Markets do not spontaneously form around one-off goods; where assets are non-comparable and firm-specific, **institutional structure must precede exchange**.

In the absence of such structure, firms default to the mechanisms they already know how to use. If internal technical capability exists, firms hire employees to build systems internally. If it does not, firms engage consultancies or license generic software. Both approaches frame the transaction as labor procurement or service provision, even when the economic substance of the output is a capital asset.

The result is not a mistake in execution, but a structural constraint. Firms lack a way to **recognize and structure** the acquisition of bespoke autonomous systems as owned productive assets without first entering into a labor-based relationship. The transaction fails to appear in the firm's decision tree, not because it is undesirable, but because it lacks an institutional form that renders it legible as something other than hiring or services.

The absence of such a form is the institutional gap that the parties must bridge to transact without misclassification. A Capital Asset Formation Agreement does not create a market for bespoke autonomous systems, nor does it resolve price discovery for assets that are inherently non-comparable. Instead, it provides a **transactional framework** that renders such exchanges legible within existing search and matching mechanisms, reducing bargaining costs once parties engage and allowing asset-formation transactions to occur without being misclassified as labor or services.

Engagement Type	Search Mechanism	Firm Acquires	Relationship Structure	Economic Character	Accounting Treatment
Employee (W-2)	Labor market	Ongoing human labor	Continuous, control-based	Labor input	OpEx (wages)
Contractor (1099)	Labor market	Time-bounded services	Scoped, service-based	Purchased effort	OpEx (fees)
Asset Builder (CAFA)	Labor market	Self-operating productive system	Finite, asset-formation	Capital asset	CapEx

5.1 The Demand for Synthetic Workers: Escaping the Coordination Trap

Firms adopt autonomous systems to decouple productive capacity from headcount growth. In traditional models, each incremental unit of output requires a proportional increase in human labor, which entails coordination costs that ultimately constrain the firm's efficiency and margins.

The firm wants synthetic workers because they represent frozen expertise: a capital asset that generates value through integration with the firm's operational context, rather than through the continuous management of human labor. The firm is not seeking a new AI department. It seeks a factor of production that yields a proportional return on capital once owned.

5.2 The Structural Failure of the Employment Contract

The central problem for the firm is that the traditional labor market is misaligned with the realities of modern AI development. While the labor market can now supply *kinetic* assets, builders who arrive with prefabricated, substantially formed systems, the standard employment contract is still designed to purchase *potential* (future labor).

The Mismatch of Formation. Employment assumes that the firm is the primary site of production, supplying the tools, the data, and the direction. However, modern builders increasingly use their own platforms and tooling outside the firm. An employment contract legally expropriates this external expertise under the work-for-hire doctrine, creating a misalignment of incentives in which the builder is penalized for having pre-built the very thing the firm needs.

The Learning Gap. Hiring buys a capability (a team that might build a system), whereas the firm needs a capacity (a system that already works). MIT research found that 95% of enterprise AI pilots deliver no measurable P&L impact, and that internal builds succeed roughly 33% of the time compared to 67% for external partnerships.[8] The problem is not the quality of the

[8] Challapally, Aditya, Chris Pease, Ramesh Raskar, and Pradyumna Chari. "The GenAI Divide: State of AI in Business 2025." MIT Project NANDA, MIT Media Lab, July 2025.

underlying models. It is that employment forces the firm to manage a process of creation rather than evaluate the result of formation.

The Inefficiency of Acqui-hiring. Currently, only Big Tech can address this mismatch by acquiring entire companies to secure key personnel and their technology. Microsoft paid $650 million for Inflection's technology and staff; Google paid $2.7 billion for Character.AI's founders and licensing rights; NVIDIA paid $20 billion for Groq's technology licenses and engineering talent.[9] These reverse acqui-hires are evidence of market failure. They are massive, expensive workarounds to the lack of a standard mechanism to acquire the asset without hollowing out the builder or absorbing its entire payroll. They serve only the tails of the distribution: price tags that exclude mid-market firms and acquirers who are themselves in the AI business.

Dimension	Reverse Acqui-hire	CAFA
Acquirer	Big Tech (wants AI capability)	Non-AI firm (wants productive capacity)
What's acquired	Talent + non-exclusive technology license	Firm-specific asset + exclusive rights
Employment	Required (talent absorption is the point)	Optional (asset transfer is separately structured)
IP structure	Non-exclusive license (builder hollowed out)	Exclusive rights to implementation; builder retains the platform
Builder's fate	Absorbed; the company survives as a shell	Retains platform; may accept employment, absorption, or neither
Price range	$650M – $20B	Accessible to mid-market firms

5.3 Two Systems of Comparative Advantage: Why Coordination Advantage Cannot Overcome Formation Disadvantage

Even if the firm lacks a comparative advantage in building autonomous systems, why can't it simply hire people who possess that advantage? Hiring specialists to overcome internal capability gaps is precisely what firms have always done.

The answer lies in the structure of the firm's comparative advantage itself.

Firms possess a comparative advantage in coordinating labor. Historically, this advantage has enabled firms to overcome their comparative disadvantage in specific technical domains by hiring specialists and coordinating their efforts to produce output. But this substitution works

[9] On the Microsoft/Inflection deal, see "Microsoft to Pay Inflection AI $650 Million After Scooping Up Most of Staff," Bloomberg, March 22, 2024. On the Google/Character.AI deal, see "Google Paid $2.7 Billion to Bring Back an AI Genius Who Quit in Frustration," Wall Street Journal, September 25, 2024. On the NVIDIA/Groq deal, see CNBC, December 24, 2025 (reporting a transaction of approximately $20 billion), and Groq's announcement of the same date describing a non-exclusive licensing agreement under which its founder, president, and other team members join NVIDIA.

only when productive output is generated through ongoing labor effort whose marginal productivity can be increased through supervision, sequencing, and managerial control.

In autonomous system development, the decisive productive input is not ongoing labor effort but prior capital formation: expertise is compressed ex ante into a self-operating system that arrives substantially formed. When the firm lacks a comparative advantage in system formation itself, and the relevant labor arrives with the method already solved, the marginal productivity of firm coordination over that labor approaches zero. Hiring cannot substitute for acquisition. The firm lacks a comparative advantage in formation, and its comparative advantage in coordination cannot be leveraged because there is no labor margin on which coordination can act.

What can the firm's coordination add when the builder arrives with the method already solved? Sequencing? The builder already knows the sequence. Supervision? The firm cannot evaluate the process; it can only evaluate the outcome. Task decomposition? The builder has already decomposed and solved. Direction? The firm does not know what to direct. The firm's coordination advantage is real, but it yields no gains when applied to labor whose marginal productivity cannot be improved through coordination.

External acquisition restores alignment. The builder exercises a comparative advantage in formation. The firm exercises comparative advantage where it actually applies: in integration, operations, and governance following asset transfers. Each party operates within its true domain. The firm's coordination advantage is not wasted; instead, it is redirected to post-acquisition activities that generate value.

This can be stated formally.

Let the firm's coordination advantage be captured as a multiplier on coordinable labor.
Partition specialist labor into two components:

- L_c: labor whose marginal productivity responds to firm coordination (supervision, sequencing, control)

- L_u: labor whose marginal productivity does not respond to coordination because the method is already solved

Effective Labor:

$$L_{\text{eff}} = \phi L_c + L_u$$

where $\phi \geq 1$ captures the firm's coordination advantage.

Formation output can then be written in the same Cobb–Douglas form used above:

$$Y = AK^{\alpha}(\phi L_c + L_u)^{\beta}$$

The marginal product of coordination is therefore:

$$\frac{\partial Y}{\partial \phi} = AK^{\alpha}\beta(\phi L_c + L_u)^{\beta-1}L_c$$

When formation labor arrives with the method already solved $L_c \to 0$. In this limit,

$$\frac{\partial Y}{\partial \phi} \to 0.$$

The firm's coordination advantage remains economically real, but it becomes productively irrelevant for formation. Coordination does not disappear; its marginal product does.[10] Post-acquisition, the firm applies coordination to integration and operation, work that still responds to supervision, sequencing, and control.

[10] Brooks, Frederick P. *The Mythical Man-Month: Essays on Software Engineering*. Reading, MA: Addison-Wesley, 1975.

5.4 The Probability Chain: Why Internal Builds Fail in Practice

Hiring can work, but only under specific conditions that most firms do not meet. Historically, new technological paradigms are first internalized by firms whose primary business is the technology itself. During the machine learning boom, the earliest and most successful adopters were ML-native companies: firms whose products were themselves ML systems, and whose leadership, tooling, and organizational structure were built around model development. Only later, once methods stabilized, talent diffused, and practices became legible, did large enterprises begin to hire ML engineers effectively.

The same pattern is repeating with autonomous systems.

For a non-AI firm, "just hiring people" to build synthetic workers is not a single decision. It is a chain of contingent bets, each of which must succeed for the outcome to materialize.

First, the firm must correctly specify the use case. It must determine which operational tasks can be automated, the degree of autonomy that is viable, and how the system must integrate with existing processes. Many firms fail here, not because they lack intelligence, but because the relevant knowledge is tacit and emergent rather than explicit. The use case that seems obvious at the outset often shifts as the firm learns what is possible.

Second, the firm must correctly identify the required talent. This requires knowing not only who is capable, but what capability is needed. In frontier domains, the firm often hires for roles it does not yet understand, using interview processes optimized for mature disciplines. The probability that the firm selects the right person is therefore less than one.

Third, the firm must successfully manage and integrate the work. Building autonomous systems is not a single-person task. It requires coordination across modeling, tooling, data, infrastructure, and domain context. This creates a human map-reduce problem: decomposing work, recombining outputs, and maintaining coherence over time. Most non-AI firms lack the managerial substrate to do this reliably.

Fourth, the output must solve the original problem. Even if the right people are hired and managed competently, the system produced may fail to integrate, may not generalize, or may address the wrong operational bottleneck. At this point, sunk costs are high, and evaluation occurs only after the organizational commitment has been made.

Each of these stages carries independent uncertainty. The probability of success is therefore multiplicative, not additive. A modest failure rate at each step compounds into a low overall likelihood of success. This is consistent with observed outcomes: MIT research finds that roughly 95% of enterprise AI pilots deliver no measurable P&L impact, and that internal builds

succeed only about one-third of the time compared to roughly two-thirds for externally built or acquired systems.

Acquiring a substantially formed asset changes this structure and collapses this chain.

Under CAFA, the firm does not evaluate a sequence of contingent human potentials. It evaluates a substantially formed system. The firm does not need to know how to build the system, manage its development, or decompose the work. It needs to know only whether the asset performs, integrates, and satisfies governance constraints.

Hiring asks: Can this person eventually build what we need? *Acquiring the asset raises the question: Does this system already do what we need?*

This is not an argument that hiring is irrational. It is an argument that hiring is appropriate primarily for firms whose business is AI itself, that is, firms that already possess the leadership, tooling, and organizational knowledge required to run the entire probability chain internally. For other firms, hiring to build synthetic workers is not a cheaper path to the same outcome. It is a fundamentally different project with materially higher uncertainty.

5.5 The CAFA Adoption Boundary

CAFA is not universal. Firms whose core business is building AI systems should build internally, because they need the capability itself, not merely the resulting capacity. Firms whose operations do not justify capitalized autonomy should not adopt it.

Acquiring an Autonomous Capital Asset through CAFA becomes rational when several conditions coincide:

- **Capacity over capability.** The firm wants productive capacity now, not the internal capability to build it over time. Building that capability would require creating an entirely new function, including hiring, tooling, and institutional learning, that the firm does not otherwise need.
- **Talent inaccessibility.** The firm cannot reliably hire, retain, or manage the required frontier talent. The relevant expertise exists in the market, but not in a form that the employment contract can capture.
- **Outcome evaluability.** The firm can evaluate whether a system works but cannot reliably evaluate the process of building it. The firm knows what it needs the system to do; it does not know how to supervise its creation.
- **Risk transfer.** The firm prefers that formation risk remain with the builder rather than with itself. Paying for results through milestones and acceptance criteria is preferable to paying for effort with uncertain output.

- **Firm-specific value.** The use case is specific enough to the firm's operations that generic infrastructure (SaaS) is inadequate. The firm needs exclusive rights to an implementation tailored to its workflows, data, and governance requirements.
- **Multi-period value.** The system will generate value over multiple accounting periods, thereby justifying recognition as a capital asset rather than as an operating expense.

5.6 Evaluation, Uncertainty, and Risk Allocation

Structuring the acquisition as an asset transfer does not eliminate integration uncertainty. It makes it contractible.

Employment assumes the firm should own output because the firm supplies and directs the production environment. When substantial formation occurs before any employment relationship exists and the firm uses inputs it did not supply, that presumption fails. The appropriate response is to structure the transaction around what the firm must verify: whether the system performs, integrates, and satisfies governance constraints before final payment and capitalization.

Dimension	Employment	CAFA
Object evaluated	Person's potential	Substantially formed Autonomous Capital Asset
When evaluation occurs	After hiring	During integration
Who bears pre-proof risk	Firm	Builder
Contractual tools	Salary, termination	Milestones, acceptance criteria, staged consideration
Capitalization	Incidental, uncertain	Explicit and intentional

The evaluation problem remains, but it becomes tractable. The firm is no longer assessing whether a person might build something. It assesses whether a substantially formed system meets the defined requirements.

The Evaluation Paradox. But this raises an immediate difficulty. In the context of employment, firms use a track record as a proxy for future performance: if this person built ML systems at Google, they can likely build one here. The evaluation precedes the commitment. Under CAFA, the firm is not evaluating the builder's credentials or a finished product. It is evaluating the likelihood that a substantially formed system will perform as claimed. Yet that likelihood can only be confirmed through integration, and integration itself carries costs. How does the firm evaluate before committing when evaluation requires integration?

The Reverse Acqui-hire as Structural Precedent. The answer already exists in the market. When Microsoft paid $650 million for Inflection's technology and staff, or Google paid $2.7 billion for Character.AI's founders and licensing rights, they did not structure these as pure asset purchases. They structured them as two separate transactions:

1. A licensing or consideration payment for pre-existing technology, compensating prior formation
2. Employment or engagement of key personnel to complete integration and support transition

This separation is not incidental. It solves the legal and economic problem simultaneously. The pre-formed technology is transferred via an explicit IP agreement, not under work-for-hire, because it was created before the employment relationship began. The integration work is performed under an employment arrangement, and work-for-hire applies to that work, which is exactly what the acquirer wants.

The same structure applies at mid-market scale through CAFA**,** which formalizes asset formation without collapsing acquisition into employment by separating consideration for prior formation from compensation for integration.

The Two-Component Structure. Under CAFA, the transaction separates into two distinct flows:

IP Transfer. The builder assigns or exclusively licenses the firm-specific instantiation of a pre-existing platform, together with exclusive deployment rights within the firm's operating context. This payment reflects years of prior development, accumulated deployment knowledge, and reusable architecture. It is capitalizable because it represents the acquisition of a productive asset rather than payment for services rendered. The builder retains the underlying platform for future deployments; what transfers is the instantiation and the exclusive right to operate it within the firm's environment.

Integration Services. The builder (or builder's personnel) performs the work of integrating, configuring, and fine-tuning the system for the firm's environment. This is labor structured as employment or services, compensated at market rates, and subject to work-for-hire by design. The firm wants to own the integration layer. The integration fee is reported on the income statement as an operating expense or capitalized if it meets the criteria for internal-use software under ASC 350.

Component	What It Covers	Legal Treatment	Accounting
IP Transfer	Firm-specific instantiation; exclusive deployment rights	Explicit IP agreement; formation preceded engagement	Capitalizable asset
Integration Services	Configuration, adaptation, firm-specific fine-tuning	Employment or services; work-for-hire applies	Expensed (or capitalized per ASC 350)
Firm's Evaluation Costs	Internal diligence, sandbox, security review	Firm's own resources	Expensed

Optionality for the Builder. The two-component structure does not foreclose employment or absorption. The builder may accept employment, whether as the primary integration vehicle or alongside a services arrangement, for governance, relationship continuity, or access to firm resources. The builder may choose full absorption if that serves their interests. This structure ensures that these are options rather than requirements and that the asset transfer is treated as a separate transaction.

This matters for both parties. The firm gets the asset it needs without mandatory ongoing employment obligations. The builder retains the ability to continue as an independent operation, accept employment, or be absorbed, depending on what serves the engagement. Employment, if it exists, cannot expropriate what was formed before the relationship began. The IP transfer is governed by the agreement, not by the work-for-hire doctrine.

Why the Structure Clears. This separation resolves the evaluation paradox by unbundling what was previously conflated.

For the firm:

- Pays market rate for integration labor, familiar, budgetable, comparable to any services engagement
- Pays separately for IP rights to the firm-specific instantiation, capitalizable, with clear ownership
- Work-for-hire applies to integration work, which is exactly what the firm wants
- Evaluation occurs during integration, but the firm is paying only for labor and access during this period. Capital is committed only upon acceptance

For the builder:

- Receives real compensation during integration, not working for free, not financing the firm's learning
- Receives separate payment for IP rights reflecting prior formation, not hours worked
- Platform ownership is protected by explicit IP terms. Work-for-hire cannot reach back to capture what was built before the relationship began
- Retains platform for future deployments; transfers only the instantiation and contextual exclusivity

Balance-Sheet Clarity. The two-component structure produces a clean accounting outcome that employment cannot. Labor spend is expensed (or capitalized under established rules); asset consideration is capitalized intentionally. This avoids the ambiguous treatment that

plagues internal AI initiatives, in which firms incur years of payroll expense without a capitalizable asset and must later justify write-offs when projects stall. The two-component structure gives CFOs what they need: defensible categorization, clear capitalization triggers, and an asset that appears on the balance sheet only after it has demonstrated value.

Staged Consideration and Acceptance. The two-component structure naturally supports staged payment tied to milestones:

1. *Engagement.* The parties execute the IP agreement (specifying the transfers and timing) and the services agreement (specifying the integration scope and compensation). Integration services begin; the builder receives labor compensation.
2. *Integration.* Builder configures the system for the firm's environment. The firm provides access, data, and evaluation bandwidth. Acceptance criteria are pre-defined and outcome-based.
3. *Acceptance.* Upon meeting the acceptance criteria, IP transfer is complete. The firm pays the asset consideration. Capitalization occurs.
4. *Post-acceptance.* Integration services may continue to provide support, training, or further adaptation, yet remain structured as services and compensated as labor.

If acceptance criteria are not met, the firm has paid only for integration labor and its own evaluation costs. The IP does not transfer. The builder retains the platform and the instantiation. Both parties exit cleanly.

The Structural Advantage Over Pure Employment. Under pure employment, the firm bundles all three economic objects (IP rights, integration labor, and evaluation) into a single instrument: salary. This creates three problems:

1. *Mispricing.* Salary reflects labor time, not asset value. A system worth $500,000 as a productive asset may be built by someone earning $200,000 per year. The firm captures the residual value only upon successful completion, while assuming the entire risk of non-completion.
2. *Work-for-hire overreach.* If the builder arrives with a substantially formed platform, the work-for-hire doctrine may claim it, even though the firm contributed nothing to its creation. The builder is penalized for having pre-built the very thing the firm needs.
3. *Evaluation timing.* Employment commits the firm before evaluation can occur. The firm pays a salary while hoping the output materializes. Termination is the only lever, and it comes with switching costs, severance, and reputational risk.

The two-component structure unbundles these flows. Labor is paid as labor. The asset is paid as an asset. Evaluation occurs before capital is committed. Work-for-hire applies only to what it should: work performed under the engagement, not work that preceded it.

This risk reallocation is not incidental to the structure. It is one of its primary functions.

Part II: The Institutional Mechanism

Part I established the economic conditions under which autonomous systems can be formed externally and transferred to firms as capital assets. Development costs have collapsed. Builders can construct substantial systems using proprietary platforms and third-party infrastructure, without reliance on firm-internal resources. Once deployed, these systems internalize the operative role: they perform work autonomously and generate value over multiple periods without continuous human operating labor.

For core workflows, firms increasingly favor ownership rather than access. Firm-specific systems enable direct control over data, governance, and operational logic, thereby avoiding the constraints inherent in standardized software and ongoing vendor dependence. Concurrently, builders possess the technical capital required to develop such systems externally, without necessitating employment relationships or the commercialization of generalized products.

The economic fit is real. What prevents these transactions from reliably clearing is not technology or incentives, but institutional form.

In practice, firms have only two mechanisms for acquiring what people bring: employment and services. Builders, therefore, arrive as candidates or consultants, even when what they offer is neither open-ended labor nor ongoing access, but a firm-specific productive system built on pre-existing technical capital. The transaction is forced into contractual categories designed for something else.

Part II explains why this mismatch is structural rather than accidental and how it can be resolved.

Section 6 shows why employment and services agreements cannot cleanly express the acquisition of externally formed autonomous systems. Employment collapses pre-existing capital into work-for-hire. Services agreements price effort when the economic substance is an asset transfer, and collapse into licensing or SaaS when ownership is attempted. These failures are not drafting errors; they arise from assumptions embedded in the forms themselves.

Section 7 introduces the Capital Asset Formation Agreement (CAFA): a contractual structure that renders asset-formation transactions legible within existing legal and accounting frameworks. CAFA separates prior formation from integration labor, platform from

implementation, and capital consideration from service fees, allowing ownership to transfer without distorting incentives.

Section 8 specifies the three-layer architecture governing the platform, implementation, and infrastructure, allocating ownership based on where value is generated. Section 9 addresses capitalization and accounting treatment. Section 10 provides implementation guidance for builders and acquiring firms.

Together, these sections describe an institutional structure for a transaction that already exists in economic reality but lacks a coherent legal expression.

Section 6: Why Existing Instruments Cannot Clear This Transaction

The mismatch arises not because firms pursue autonomous systems incorrectly, but because there is no institutional form through which such systems can appear as objects of transaction.

Firms encounter the problem first as an operational gap. A department absorbs too much labor. A workflow could be automated. A class of decisions could be executed mechanically. These needs are framed as a lack of capability, not a lack of capital. The firm does not think, “We need to acquire an asset.” It thinks, “We need someone who can build this.” Labor becomes the default solution because labor is the only category through which productive capacity can be acquired.

Independently, builders now form substantial autonomous systems outside the firm. They do so using proprietary platforms, accumulated deployment knowledge, and third-party infrastructure. By the time they engage with firms, through hiring channels, consulting engagements, referrals, or informal collaboration, they are not offering prospective effort alone. They are holders of pre-existing technical capital.

Yet despite this external formation, vertical integration remains the natural destination for most of what these builders create.

The systems in question do not generate value in isolation. They require a firm’s operational context: access to internal data, control over workflows, integration with production systems, and alignment with governance and compliance regimes. Without this context, the system remains technically functional but economically inert.

Just as importantly, it is not economically efficient for the builder to solve for distribution independently. The builder’s comparative advantage lies in system formation, encoding expertise into autonomous operational logic, not in building sales channels, organizational adoption, regulatory interfaces, or internal change management. Firms already possess these

capabilities. For most use cases, attempting to recreate them externally would duplicate existing infrastructure at high cost and low probability of success.

From the builder's perspective, the question is therefore not whether the system must integrate with a firm, but where and how that integration should occur. The asset has already been formed, but it cannot realize its value without being embedded inside an organization that can operate it at scale.

From the firm's perspective, the calculus is complementary. When a builder appears who has already formed what the firm needs, acquiring that asset is preferable to recreating it internally: faster, lower risk, and grounded in demonstrated capability rather than speculative potential.

It is at this point, where external formation meets the need for internal deployment, that the transaction should occur. The builder needs a firm to absorb the asset; the firm benefits from acquiring something already formed. What is missing is a mechanism that enables this integration to occur as a transaction rather than as an employment relationship.

This is why the labor market remains the site of discovery. Builders arrive as people. Firms find them where firms find people: through recruiters, job postings, networks, referrals. The builder and the asset are inseparable. One cannot acquire the asset without engaging the person who holds it. The labor market is, therefore, the appropriate place to surface these transactions, even though what is being transacted is not labor alone.

The labor market, however, provides no contractual mechanism through which this transaction can be expressed directly. Asset purchase agreements exist, but they are not what recruiters, hiring managers, or procurement departments typically rely on when a builder appears. Employment and services agreements are the instruments at hand, and both presume that value is created inside the relationship, not prior to it. The builder cannot simply transfer what was built before the relationship began, not without separate negotiation, separate consideration, and separate IP treatment. Employment contracts cannot express that I am transferring a capital asset I already own, and you are paying me for it as such.

This is the gap the parties must bridge. The solution is not an alternative to the labor market. It is an extension of it. As autonomous systems become inseparable from the people who build them, the labor market must accommodate capital transactions alongside labor transactions. CAFA provides the contractual structure that allows it to do so. The firm sees a candidate or a consultant. What the recruiter sees is a placement. What legal sees is a standard agreement. What no one sees, because there is no place to see it, is an asset acquisition. The result is misclassification. The asset is either transferred silently under work-for-hire assumptions or not transferred at all. Neither outcome reflects the economic substance of the parties' exchange. The deeper issue is this: employment pursues the wrong kind of control. The firm's

instinct is to control the worker: to direct effort, set schedules, supervise execution, and own work product as it is produced. This is operational control. It is effective when value is generated through ongoing labor whose marginal productivity can be shaped by management.

But autonomous systems invert this structure. The decisive productive act is not an ongoing effort but prior formation. Once the system exists, controlling the person adds little. What the firm needs is asset control: durable, exclusive rights over the system that performs operational work. Employment conflates these. It grants authority over the person while providing no reliable claim over the productive capacity that has already been created. The worker can leave. The firm cannot retain the capacity itself. At best, it retains partial legal rights to outputs it may not understand, operate, or extend. Operational control persists; asset control does not exist.

The Capital Asset Formation Agreement separates these concerns. The firm acquires control over what can be owned: the system itself. The builder remains free to leave, to continue developing their platform, and to transact again. Control is allocated to the asset, not misapplied to the person. What employment attempts to achieve indirectly through supervision and retention, CAFA achieves directly through ownership and transfer.

6.1 The Employment Problem: Work-for-Hire and Pre-Existing Assets

The silent transfer occurs through a mechanism designed for a different purpose.

Employment agreements assign to the firm all intellectual property created during the employment term. But the assignment does not stop at work performed in the office. Standard clauses assign all inventions conceived or developed during employment that "relate to the company's current or anticipated business." For an autonomous system builder, the platform is related to the implementation being built for the firm. As a result, the standard assignment clause exerts a form of legal gravity that pulls the platform into the firm's ownership, regardless of when or where it was created.

This structure presumes that the firm supplies the production environment, including tools, data, infrastructure, and direction, while the employee supplies labor within that environment. Under that presumption, assignment is appropriate: the firm owns what it enabled to be created.

Autonomous system builders do not fit this model. They arrive with pre-existing technical capital: platforms, architectures, deployment methodologies, and partially completed implementations. This capital was formed before the employment relationship began, using inputs the firm did not supply. Economically, it functions as capital rather than labor.

Many employment assignment clauses do not reliably distinguish between a builder's pre-existing platform and the firm-specific implementation generated during the engagement.

When drafted broadly, these clauses can operate to transfer both. In such cases, the firm acquires not only what the builder creates during employment, but also technical capital formed prior to the relationship, without separate negotiation, separate consideration, or explicit recognition that an asset acquisition has occurred.

Employment agreements do include a mechanism for protecting prior inventions through carve-outs. Employees may list pre-existing inventions on a schedule and exclude them from assignment. This mechanism, however, was designed to prevent inadvertent transfer of unrelated side projects, not to structure the transfer of a capital asset that is central to the engagement.

The carve-out creates its own trap. Standard provisions require that if the employee incorporates any prior invention into company work, the company receives a non-exclusive, perpetual, royalty-free license to that prior invention. The builder who uses a pre-existing platform to construct a firm-specific implementation has, by definition, incorporated prior inventions into company work. The license grant is triggered automatically and silently. The firm obtains perpetual rights not merely to the implementation, but to the platform itself, which is the very asset the builder intended to retain. The builder receives no asset consideration for this grant.

This is precisely the opposite of what CAFA requires. Under CAFA, the firm receives exclusive ownership of the implementation and no rights to the platform. Under standard employment with a carve-out, the firm receives ownership of the implementation and a perpetual license to the platform. The builder's retained capital is compromised in either case.

Attempts to resolve this through more aggressive carve-outs are unsuccessful. If the builder denies the firm any license to the platform, the implementation may become inoperable. The firm owns an asset it cannot use, because the asset depends on platform components that the firm has no right to access. Conversely, if the carve-out is structured to allow the implementation to function while truly protecting the platform, the transaction is no longer work-for-hire. It is the acquisition of an asset built on retained capital: a transfer transaction, not a labor relationship. The carve-out does not fix employment; it collapses it. What remains is an asset purchase dressed in employment terms, lacking the consideration structure, acceptance criteria, and accounting treatment that an asset purchase requires.

The accounting mismatch reinforces the structural failure. Wages are paid for the potential to work; they are not legally recognized as the purchase price for a pre-existing capital asset. If a firm claimed it "acquired" a platform worth $500,000 through a $150,000 annual salary, the capitalization and valuation would be indefensible. Employment provides no mechanism for separate asset consideration because employment presumes no asset is being transferred.

The misalignment is categorical. Employment treats pre-existing capability as labor output. When the builder arrives with capital, employment cannot express the transaction without either expropriating the platform or abandoning the employment structure altogether.

6.2 The Services Problem: Pricing Effort When the Substance Is Asset Transfer

Services agreements, whether styled as consulting agreements, independent contractor agreements, or master services agreements, appear to resolve the employment problem. The builder is not subordinated to the firm. There is no expectation of ongoing labor. The relationship is framed as delivery of a defined outcome.

In practice, services agreements still treat the transaction as the purchase of labor.

The builder "performs services" and is compensated through hourly, daily, or milestone-based fees tied to effort. Even when the agreement references deliverables, they are contractually defined as the output of labor performed under the agreement rather than as a discrete asset that the builder has already formed and is transferring. The firm is understood to be purchasing effort, not acquiring a system.

This framing directly informs the accounting treatment. Payments under a services agreement are ordinarily recognized as operating expenses, even when the economic substance is the transfer of a durable capital asset.

The treatment of intellectual property reinforces the misalignment. Most services agreements distinguish between "pre-existing IP" (retained by the builder) and "work product" (assigned to the firm). But this binary cannot express the three-layer structure that autonomous system transactions require:

- Layer 1: Platform — the reusable substrate that the builder must retain
- Layer 2: Implementation — the firm-specific instantiation that the firm must own exclusively
- Layer 3: Infrastructure — hosting and compute, which neither party "owns" in the relevant sense

Services agreements have no native mechanism to separate Layer 1 from Layer 2. The result is either full assignment (which expropriates the platform) or full retention with a license (which denies the firm ownership).

The licensing workaround is common but introduces its own failure. When the builder retains the platform and licenses use to the firm, whether through hosted access or activation-restricted delivery, the relationship becomes a SaaS arrangement. The firm receives access

rights, not title. Payments become recurring fees. The transaction has collapsed into precisely the vendor-dependent model that ownership was meant to avoid.

This matters because autonomous systems, by definition, do not require ongoing vendor involvement to function. Their operational logic is internal. A licensing model that requires ongoing payments to maintain access to the asset contradicts the economic rationale for acquiring an autonomous asset. The firm pays indefinitely for capability that, once deployed, requires no continued service.

6.3 Structural Debt: How Misclassification Compounds During Integration

The conflict typically emerges as engineering discussions shift into implicit disputes over ownership and control:

- Who owns the workflows being encoded?
- Which components belong to the builder's platform versus the firm's implementation?
- Which improvements are reusable across clients, and which are firm-specific?
- Who determines when the system is sufficiently autonomous to hand over?

These are not technical questions. They are consequences of the contract form. Because ownership was not allocated correctly at the outset, the parties must negotiate it during integration, precisely when rapid iteration and mutual trust matter most.

The outcome is predictable. Progress slows. Trust erodes. Projects fail not because the architecture is unsound, but because the contract imposes an adversarial posture at the moment alignment is most necessary.

Structural debt also reintroduces the labor dependencies the system was designed to eliminate. Under employment, the builder is embedded in a managerial hierarchy, evaluated on process compliance, and incentivized to preserve their role rather than complete a transfer. Under services, the builder is incentivized to retain control through licensing, ongoing support, or architectural choices that preserve dependence. In both cases, the contract anchors labor in place even when the economic objective is to remove it. This is why firms struggle to transition from pilot automations to production autonomous systems. The contract itself prevents autonomy from scaling.

Section 7: The CAFA Framework

CAFA is a contractual framework designed to express the acquisition of externally formed autonomous systems according to their economic substance. The transaction consists of the

transfer of a firm-specific capital asset built on pre-existing technical capital, completed through integration labor.

7.1 Overview

CAFA makes three separations explicit:

1. **Asset versus labor.** The transaction separates into two components: IP transfer (the asset) and integration services (labor). Each is priced, documented, and accounted for distinctly.
2. **Platform versus implementation.** The builder's reusable technical capital (platform) remains with the builder. The firm-specific instantiation (implementation) transfers to the firm. Neither party acquires what the other needs to retain.
3. **Capital consideration versus integration fees.** Consideration for the asset reflects prior formation, including platform development and accumulated deployment expertise. Integration fees compensate current labor at market rates for configuration, adaptation, and fine-tuning.

These separations use familiar legal tools: IP assignments, services agreements, milestone payments, and acceptance testing. What CAFA provides is not new law but institutional scaffolding, a recognized structure that allows parties to express what is being exchanged.

7.2 The Two-Component Structure

The core of CAFA is a two-component transaction structure. This structure was previewed in Part I (Section 5.6) as the resolution to the evaluation paradox. Here we specify its contractual mechanics.

Component 1: IP Transfer

The builder assigns or exclusively licenses to the firm the firm-specific implementation, meaning the instantiation of the builder's platform configured for the firm's workflows, data, and governance requirements. This transfer includes:

- The operational system, configured for the firm's environment
- Exclusive rights to operate the implementation within the firm's operational context
- Any firm-specific adaptations, integrations, or workflow encodings

The transfer excludes the builder's platform: the reusable orchestration logic, schemas, methodologies, and accumulated deployment expertise that enabled construction. The builder retains the platform for future engagements.

Where the builder has established provisional patent protection over the implementation's novel methods (which, as Section 9.1 explains, is necessary for identifiability in a low-replication-cost environment), those provisional rights transfer to the firm as part of the IP assignment. The firm then assumes responsibility for prosecuting utility patents based on the builder's filings. This allocation reflects comparative advantage: the builder's expertise lies in system formation, not patent prosecution; the firm has the resources and long-term interest to secure durable IP protection for an asset it now owns. The provisional filing establishes the asset; the assignment transfers both the asset and the legal rights that protect it.

This component is capitalizable. It represents the acquisition of a productive asset, not payment for services rendered. The consideration paid for the IP transfer reflects prior formation, specifically the years of platform development and deployment expertise that enabled rapid construction, not hours worked during the engagement.

Component 2: Integration Services

Integration work includes configuring the system for the firm's infrastructure, encoding firm-specific workflows, adapting to data sources and governance requirements, and transferring operational knowledge. This is labor, compensable at market rates and subject to the work-for-hire doctrine. Whether structured as employment or as a services engagement, the integration component is analytically distinct from the asset transfer.

Work-for-hire applies to the integration component by design. The firm seeks ownership of the integration layer, including firm-specific configurations, workflow encodings, and adaptations created during the engagement. Unlike the platform, which preceded the relationship, integration work is performed within the engagement and properly belongs to the firm.

This component is expensed as incurred or capitalized if it meets the criteria for internal-use software under ASC 350-40. Either way, its treatment is distinct from the asset consideration.

Why the Separation Matters

The two-component structure resolves the failures identified in Section 6:

- *Work-for-hire scope*: Employment's assignment clauses reach only integration work, not the pre-existing platform. The platform is protected by explicit IP terms in the transfer agreement, not by carve-outs that collapse the employment structure.
- *Pricing*: The firm pays separately for the asset (reflecting prior formation) and for labor (reflecting current effort). Neither is mispriced as the other.

- *Accounting*: Asset consideration is capitalizable. Integration labor is expensed or capitalized in accordance with established rules. The firm's balance sheet reflects what it actually acquired.
- *Ownership*: The firm owns the implementation exclusively. The builder retains the platform. Neither party owns what the other needs.

The two-component structure is not novel in form. It mirrors the structure of reverse acqui-hires at scale (Part I, Section 5.6). CAFA provides a standardized framework that makes this structure accessible at the mid-market scale without requiring the parties to negotiate from scratch.

7.3 Deliverable Definition: Autonomy as the Performance Standard

Under employment and services agreements, deliverables are typically defined in terms of labor inputs or task completion. The employee "performs duties as assigned." The contractor "delivers the work product described in the statement of work." These definitions assume that the firm can evaluate the creation process or, at least, specify what the output should look like in advance.

Autonomous system acquisition does not satisfy this assumption. The firm engages the builder precisely because it cannot specify how the system should be constructed. It can only evaluate whether the system works.

Under a CAFA-structured transaction, the deliverable is defined by autonomous performance rather than by labor inputs.

The system specification describes what the system does: the operational functions it performs, the decisions it makes, the workflows it executes, and the conditions under which it must perform them. Acceptance criteria are outcome-based:

- Does the system perform the specified functions within defined parameters?
- Does it integrate with the firm's infrastructure, data sources, and controls?
- Does it operate autonomously, without requiring continuous human intervention?
- Does it satisfy the firm's governance and compliance requirements?

This framing shifts evaluation from process to result. The firm does not need to understand how the builder constructed the system. It needs to verify that the system performs as specified in its operational environment.

This is consistent with how firms generally evaluate capital assets. A firm acquiring manufacturing equipment does not supervise its fabrication; it tests whether the equipment

meets performance specifications. Autonomous systems are no different in kind. They are productive assets whose value is realized through operation, not through observation of their creation.

7.4 Staged Consideration and Risk Allocation

The two-component structure supports staged payment tied to milestones and acceptance. This staging allocates risk in a way that employment and services cannot.

Stage 1: Engagement

The parties execute two agreements:

- An IP agreement specifying what transfers (the implementation), what is retained (the platform), the conditions for transfer, and the consideration
- A services agreement (or employment agreement) specifying integration scope, timeline, compensation, and work-for-hire terms

Integration services begin. The builder receives labor compensation: a market-rate payment for current work. The firm has committed to integration costs, not to asset acquisition.

Stage 2: Integration

The builder configures the system for the firm's environment. The firm provides access to infrastructure, data, and personnel as required. Both parties work toward acceptance criteria defined in the IP agreement.

During this stage, the builder bears formation risk. If the system cannot be made to perform as specified, the builder has invested effort that will not result in asset consideration. The firm bears integration risk, specifically the cost of access, internal resources, and evaluation bandwidth, but has not yet committed capital to the asset.

Stage 3: Acceptance

Upon meeting acceptance criteria, the IP transfer completes. The firm pays asset consideration. Capitalization occurs.

The firm now owns a productive asset that has demonstrated value in its environment. The asset appears on the balance sheet only after it has been tested, not before.

Stage 4: Post-Acceptance

Integration services may continue for support, training, further adaptation, or operational refinement. These remain structured as services, compensated as labor. The builder may transition to employment if both parties elect, but employment is not required for the asset transfer to complete.

If Acceptance Fails

If the system does not meet acceptance criteria, the IP does not transfer. The firm has paid only for integration labor and its own evaluation costs. The builder retains the platform and the implementation. Both parties exit cleanly.

This structure inverts the risk allocation of employment. Under employment, the firm commits to a salary before knowing whether the output will materialize. Under CAFA, capital is committed only after the system demonstrates value. The builder bears pre-acceptance risk; the firm bears post-acceptance integration risk.[11] Both parties are compensated appropriately for the risk they assume.

Section 8: The Three-Layer Architecture

The two-component structure separates asset transfer from integration labor. But a further separation is required: within the asset itself, what does the firm own and what does the builder retain?

Employment and services agreements cannot answer this question cleanly. Employment assigns everything to the firm. Services agreements distinguish only between "pre-existing IP" and "work product," a binary distinction that cannot express the actual structure of autonomous system development. CAFA introduces a three-layer architecture that allocates ownership according to where value is generated.

8.1 Layer 1: Platform (Builder Retains)

The platform is the reusable substrate on which firm-specific implementations are built. It includes:

- Orchestration logic and workflow scaffolding
- Data schemas and representational structures
- Integration patterns and API abstractions

[11] Dixit, Avinash K., and Robert S. Pindyck. *Investment Under Uncertainty*. Princeton, NJ: Princeton University Press, 1994.

- Deployment methodologies and operational playbooks
- Accumulated knowledge from prior implementations

The platform is not a product. It is not licensed to the firm, and the firm does not interact with it directly. It is the builder's means of production: the technical capital that enables rapid construction of autonomous systems tailored to specific operational contexts.

The platform exists prior to the engagement. It improves through each deployment as the builder refines patterns, resolves edge cases, and accumulates operational knowledge. This cross-deployment learning is the source of the builder's competitive advantage and the foundation of the builder's business model.

The builder retains the platform.

8.2 Layer 2: Implementation (Firm Owns Exclusively)

The implementation is the firm-specific instantiation of the platform. It includes:

- The deployed instance of the platform, configured to the firm's infrastructure and workflows
- Workflow encodings specific to the firm's operations
- Integrations with the firm's data sources, infrastructure, and controls
- Business rules, decision logic, and exception handling tuned to the firm's context
- Any adaptations made during integration

The implementation is what generates value inside the firm. It performs the operational functions for which the firm acquired it. It operates on the firm's data, within the firm's governance constraints, in service of the firm's objectives.

Once deployed, the implementation accumulates firm-specific operational knowledge. It learns the firm's exception patterns, data distributions, and workflow variations. This learning is valuable only to the firm. It cannot be transferred to another context without losing its relevance.

The firm owns the implementation exclusively.

8.3 Layer 3: Infrastructure (Separate, Always OpEx)

Infrastructure includes the computational resources required to operate the system:

- Cloud hosting and compute
- Model inference (LLM API calls, embedding generation, etc.)
- Storage, networking, and related services

Infrastructure may be provided by the builder, by the firm, or by third-party providers. The choice is operational, not structural. It does not affect ownership of the implementation or retention of the platform.

Infrastructure costs are always operating expenses. They are consumed as incurred, do not create durable assets, and are not capitalizable. This is true regardless of who provides them.

The three-layer architecture keeps infrastructure analytically separate from the asset transfer. The firm acquires a capital asset (the implementation). The firm incurs operating expenses (infrastructure). These are distinct transactions with distinct accounting treatment.

8.4 Why the Builder Retains the Platform

The separation of Layer 1 from Layer 2 is not a negotiating concession. It reflects the transaction's economic structure.

Builder viability. The builder's business model is serial deployment. The platform enables rapid construction of firm-specific implementations. If every firm acquired the platform, the builder would have to rebuild from scratch for each engagement. No viable builder would accept this structure. Platform retention is a condition of market participation.

The firm does not need the platform. The firm seeks a synthetic worker: a system that autonomously performs operational functions. The firm does not want a synthetic worker factory. The implementation generates value for the firm. The platform does not. Acquiring the platform would burden the firm with an asset it cannot use, does not want to maintain, and has no capability to operate.

Unpriced expropriation. The platform is the capital that the builder established prior to the engagement. Its internalization without specific consideration is the same problem identified in employment: the firm would acquire a capital asset without paying for it as such. The platform's value is realized through the asset consideration for each implementation, not through a single transfer that destroys the builder's ability to continue operating.[12]

Clean boundaries. If the firm owned the platform, it would own something the builder needs for other clients. This creates entanglement: the firm could restrict the builder's future deployments, demand royalties, or assert rights over implementations delivered to other firms. Clean separation avoids these disputes. The firm owns the resources that generate value in its

[12] Teece, David J. "Profiting from Technological Innovation: Implications for Integration, Collaboration, Licensing and Public Policy." *Research Policy* 15, no. 6 (1986): 285–305.

context. The builder owns the capabilities that generate value across contexts. Neither party has claims against the other's retained assets.

8.5 Why This Architecture Resolves the Doctrinal Tension

The failures described in Section 6 arise because employment and services agreements cannot express the three-layer structure.

Employment collapses all three layers into a single layer of firm ownership. Work-for-hire reaches the platform (because it was "used" during the engagement), the implementation (because it was "created" during the engagement), and arguably even infrastructure configurations (because they were "developed" in connection with the work). The builder retains nothing. This is why background IP carve-outs do not fix employment. They attempt to subtract Layer 1 from a structure that assumes total assignment.

Services agreements with full assignment produce the same result. Services agreements with licensing workarounds collapse Layers 1 and 2 into builder retention, granting the firm only access rights. The firm cannot own the implementation without the builder surrendering the platform, and the builder cannot retain the platform without depriving the firm of ownership. The resulting binary, assignment, or license cannot express a transaction in which Layer 2 transfers while Layer 1 is retained.

CAFA's three-layer architecture provides the missing vocabulary:

Layer	What It Contains	Who Owns It	Accounting Treatment
Platform	Reusable substrate, cross-deployment learning	Builder retains	Not transferred; not on the firm's books
Implementation	Firm-specific instantiation, operational knowledge	The firm owns exclusively	Capitalized as an intangible asset
Infrastructure	Compute, hosting, inference	Third party or either party	OpEx as incurred

This structure aligns ownership with value creation. The firm internalizes the value-creating activities in its environment.[13] The builder retains what generates value across environments. Infrastructure is expensed by the entity that incurs it. Each layer is accounted for correctly because each layer is owned correctly.

[13] Wernerfelt, Birger. "A Resource-Based View of the Firm." *Strategic Management Journal* 5, no. 2 (1984): 171–180.

Section 9: Capitalization and Accounting Treatment

The preceding sections established the contractual structure: the two-component transaction (Section 7) and the three-layer architecture (Section 8). This section addresses how the transaction is recognized on the firm's financial statements.

Accounting treatment matters because it determines how the firm's investment in autonomous systems appears to shareholders, lenders, and regulators. Under employment or services agreements, payments are recognized as operating expenses; they reduce current-period earnings and do not appear on the balance sheet. Under CAFA, the asset consideration is capitalizable. It appears on the balance sheet as an intangible asset and is amortized over its useful life.

This distinction affects reported profitability, return on assets, and the firm's apparent capital structure. It also affects managerial incentives: expenses are minimized, and assets are invested in. Correct accounting treatment aligns financial reporting with economic substance.

9.1 Establishing Asset Characteristics

Under both U.S. GAAP (ASC 350) and IFRS (IAS 38), an intangible asset is capitalizable when it satisfies three criteria: identifiability, control, and future economic benefit. In a CAFA transaction, these criteria must be satisfied twice: first by the builder, who must possess an asset capable of transfer, and then by the firm, which must receive an asset capable of capitalization.

Identifiability

An asset is identifiable if it satisfies either of two conditions: it is separable (capable of being separated from the entity and sold, transferred, licensed, or exchanged) or it arises from contractual or legal rights.

Historically, separability was sufficient. If a system could be extracted, documented, and delivered to another party, it was identifiable as an asset. When replication costs were high, the builder's ability to demonstrate and transfer a system provided practical exclusivity. Even if a firm understood what had been built, recreating it internally required substantial time, labor, and expertise. The builder's head start was durable. Separability implied control.

AI disrupts this assumption.

As replication costs collapse, technical separability no longer guarantees economic control. A firm can observe a system's functionality, or receive a detailed demonstration, and direct its

internal teams to recreate a substantially similar system in weeks rather than months or years. The builder's system may remain technically separable, but the builder no longer controls separation itself. The firm can separate the asset from the builder without consent and without payment. Separability alone becomes insufficient to establish identifiability.

The second path to identifiability, therefore, becomes essential: arising from contractual or legal rights.

In a low-replication-cost environment, asset character depends on enforceable exclusion through intellectual property rights, confidentiality obligations, and access controls, rather than on technical separability alone.

Among available IP mechanisms, provisional patent filings carry the most structural weight in this context. A provisional patent filing establishes a priority date over novel methods, architectures, and processes embodied in the system. The builder can prove, with a date-stamped filing, that they invented this first. While provisional patents do not, by themselves, grant exclusionary rights, they create a 12-month window to file a full patent application, and any patent that issues will have priority as of the provisional filing date.

Under CAFA, the provisional patent rights transfer to the firm as part of the IP assignment. The firm then has the option, and typically the incentive, to prosecute utility patents based on the builder's provisional filings. The builder establishes priority and creates the asset; the firm acquires both the implementation and the legal rights that protect it and bears the cost of converting provisional protection into issued patents. This allocation reflects comparative advantage: the builder's expertise lies in system formation, not patent prosecution; the firm has the resources and long-term interest to secure durable IP protection for an asset it now owns.

The practical effect is structural. Where a builder has filed provisional patents covering the system's novel elements, a firm that attempts replication faces uncertainty as to future infringement. The transaction shifts from a race to replicate into a negotiation over transfer. Absent such rights, the builder has no legal basis to prevent the firm from building what it saw. What the builder possesses is a head start that evaporates upon disclosure, not an asset capable of transfer.

For the transaction to clear as a capital transfer rather than collapse into a service, the asset must satisfy the identifiability criterion. Under applicable accounting standards, separability alone is sufficient. But as replication costs decline, separability provides diminishing practical protection. The firm can observe and recreate what the builder demonstrates. In this environment, grounding identifiability in legal rights rather than mere technical separability strengthens the builder's position. In practice, this means provisional patent filings covering the system's novel elements, filed before substantive engagement with acquiring firms.

Control

Control requires the power to obtain future economic benefits from the asset and to restrict others' access to those benefits.

For the firm post-transfer, control is unambiguous. Under CAFA, the firm acquires exclusive ownership of the implementation. It may deploy, modify, or retire the system at its discretion. The builder retains no residual rights in the transferred implementation.

For the builder pre-transfer, control exists only to the extent that the builder can lawfully exclude others from replicating the system. In an environment of near-zero replication cost, control cannot rest on secrecy or complexity alone. It must be established through enforceable IP rights. Without the ability to restrict replication prior to transfer, the builder does not control the asset and therefore cannot transfer one.

Why Labor Is Not an Asset

If autonomous systems can be assets, why can't employees? The answer is control. Under at-will employment, the firm has no enforceable claim to future services. The employee can leave tomorrow. What the firm possesses is not a right but a relationship, one that either party can terminate. No control, no asset.

The distinction requires precision. Employment contracts do provide control, but of the wrong kind. The firm controls how work is performed: it directs tasks, sets schedules, and owns the output of labor during the relationship. This is operational control. But asset control is different. It requires the power to retain the productive capacity, to exclude others from it, and to transfer it. Employment provides none of these. The employee can leave tomorrow; the firm cannot prevent departure, cannot sell the employment relationship to another firm, and cannot exclude competitors from hiring the same person. The firm's operational control depends on the worker's continued presence, which the firm does not control.

This is why the employment contract, despite being structured around control, cannot make the ongoing labor relationship itself into a capital asset. Labor can create assets; that is what direct labor does. But the relationship through which labor is procured is not property. The firm controls the work while the worker is present; it does not control whether the worker remains.

Professional sports illustrate the exception. Player contracts are recognized as intangible assets on team balance sheets, not because the player is property, but because the contract is. The contract grants exclusive, transferable rights to the player's services for a defined term. It functions as a derivative claim on future productive capacity: the team owns not the person, but the right to the person's services. That right can be traded, valued, and capitalized. The

player may not leave during the contract term; the contract may be transferred to another team. What is capitalized is the contractual right, a property interest that satisfies identifiability, control, and future economic benefit.

Ordinary employment lacks this structure. The firm cannot sell an employee's contract to another firm. The employee is not bound for a defined term. The relationship is not property; it is an ongoing exchange of wages for effort, terminable at will by either side.

The sports exception is not arbitrary. It reflects a deeper condition for asset treatment. In most labor markets, individual workers are approximately substitutable. If one accountant leaves, the firm hires another. The marginal product of any specific worker is low relative to total output. The production function does not depend critically on any one person. This is the implicit assumption underlying the treatment of labor as a homogeneous input: one unit of labor is equivalent to another.

Professional athletes invert this assumption. The marginal product of a star player is enormous and individually attributable. A single player can be the difference between a championship and a losing season. That contribution cannot be replicated by hiring a replacement at market rates. The player is not fungible labor. The player is a scarce, identifiable productive input whose individual contribution dominates.

This suggests a threshold condition for when a labor contract can be treated as an asset: the marginal product of the specific worker must be high and individually attributable; the input must be scarce, with no comparable substitute available at market rates; and the firm must be able to secure control through exclusive rights, defined term, and transferability. Ordinary employment fails all three. Sports contracts satisfy all three.

Autonomous systems satisfy the asset test through a different path. The system has a high, attributable marginal product. It performs operational functions that would otherwise require ongoing human labor. The system is non-substitutable. Firm-specific integration and IP protection make replication costly or legally constrained. The firm can secure control not over a person but over the asset itself. The builder's labor does not satisfy these conditions; the builder can leave, and the firm has no claim to future services. But the system, once transferred, is owned outright. It is the system, not the builder, that crosses the threshold into capital.

This raises a broader question. If the threshold for asset treatment is high, and the individually attributable marginal product is combined with contractual control, the sports model may not be the only path. The reverse acqui-hires discussed earlier (Microsoft paying $650 million for Inflection, Google paying $2.7 billion for Character.AI) suggest that the market is already pricing certain individuals at asset-level valuations. These transactions are structured as technology licenses, but the consideration reflects the value of specific people and the systems they built.

But reverse acqui-hires are not the norm. They occur only when the builder has already achieved sufficient scale and visibility to negotiate as an asset holder. Below that threshold lies a larger population: individuals who have built systems, platforms, or methodologies of genuine value, but who have not yet reached the scale that forces asset-level negotiation.

Firms encounter these individuals constantly, not as job applicants, but as operators of small, self-directed enterprises. The firm observes the value. Recruiters reach out. The implicit message is: abandon what you are building and come in as an employee. The firm attempts to acquire the builder's accumulated expertise and productive capacity through an employment offer, pricing what is functionally an asset as if it were fungible labor.

This is a transaction failure. The builder is not selling time. The builder has already created something. The labor market has no structure to express this, so the firm forces the interaction into the only instrument it knows: a job offer. The builder must either accept a systematic undervaluation of what they have built or decline, leaving a productive asset that has no lawful or intelligible path for transfer to the firm.

The Capital Asset Formation Agreement addresses this for autonomous systems by separating the asset from the builder. The builder can transfer the system without abandoning their platform or entering employment. But the broader labor market may eventually require structures that recognize when individuals are not selling time, but stewarding capital, and that allow transactions to clear accordingly.

The sports model suggests a structural template. The player performs labor: attending practice, playing games, and following team directives. But the contract is not merely an employment agreement. It is a property right layered on top of the labor relationship: exclusive, term-limited, and transferable. The labor is the mechanism through which value is delivered; the contract is the asset that can be owned and traded. Employment and assets coexist in a single instrument.

The Capital Asset Formation Agreement follows a parallel logic but separates what sports contracts bundle. The builder may perform integration labor: configuring the system, transferring knowledge, and supporting deployment. But this labor is not the asset. The system is the asset. CAFA structures consideration accordingly: asset consideration for the transfer of the system, and, where applicable, labor compensation for integration work. The two components are priced and treated distinctly.

This separation may point toward a broader possibility. If sports contracts demonstrate that labor relationships can carry asset characteristics when properly structured, the inverse may also hold: asset transfers can include labor components without collapsing into employment. The question is whether the labor market can evolve instruments that recognize both, without

forcing every transaction into pure labor (employment) or pure asset (M&A), and that accommodate the hybrid structures that autonomous systems require.

Future Economic Benefit

An asset must be expected to generate future economic benefits, whether through revenue generation, cost reduction, or operational efficiency.

Autonomous systems satisfy this criterion by performing operational functions that would otherwise require ongoing human labor. The benefit is measurable: the firm can compare the system's output to the cost, reliability, and scalability of equivalent human effort.

But if the system has not yet been integrated into the firm's operations, how can it be said to have value? The answer is that value and the realization of value are distinct.

Value exists prior to integration as a comparative advantage. The builder's system has value because it represents a formed productive capability that the firm would otherwise need to create internally, or, where IP protection applies, cannot lawfully create without risking infringement. The firm can estimate this value before integration by reference to internal build cost, development time, probability of success, and the legal constraints imposed by the builder's IP position.

Integration realizes value through operation in the firm's context. But value does not require realization to exist. A machine tool has value before it is installed in a factory. Its value is the present discounted value of its expected future returns once deployed. The builder's system is no different. Productive potential, verified through demonstration and protected through legal rights, constitutes value. Integration converts that potential into output. Capitalization occurs when the asset transfers; returns are realized through operation.

For a CAFA transaction to succeed, the builder must arrive with an asset that satisfies identifiability, control, and future economic benefit before transfer. In an environment of declining replication costs, identifiability increasingly depends on legal rights rather than on technical separability. Provisional patent filings provide the clearest mechanism for establishing exclusivity at low transaction cost. Builders who engage firms without such protection risk losing the asset prior to transfer, not through misconduct, but through lawful replication that separability alone can no longer prevent.

The asset existed prior to integration because it was formed under enforceable exclusivity. Integration realizes value; it does not create it. This distinction permits autonomous systems to be capitalized as assets rather than expensed as services, and CAFA is designed to preserve it.

9.2 What Is Capitalized, What Is Expensed

The two-component structure yields two distinct cost flows, each with its own accounting treatment.

Asset consideration (capitalizable). The payment for the IP transfer, specifically the firm-specific implementation and exclusive deployment rights, is capitalizable as an acquired intangible asset. This payment reflects the builder's prior formation: the platform development, accumulated deployment expertise, and architectural decisions that enabled rapid construction. It is not payment for services rendered during the engagement.

The capitalization basis includes:

- The negotiated price for the implementation
- Transaction costs directly attributable to the acquisition (legal fees, due diligence costs)
- Any additional consideration contingent on acceptance

Integration services (expensed or capitalized under ASC 350-40). The payment for integration labor, including configuration, adaptation, fine-tuning, and workflow encoding, is treated as either:

- Operating expense, if the work is characterized as ongoing services, or
- Capitalizable under ASC 350-40 (internal-use software), if the work meets the criteria for application development stage activities

Under ASC 350-40, costs incurred during the application development stage of internal-use software are capitalizable. Integration work that places the system into service, such as configuring it for the firm's environment, encoding firm-specific business rules, and integrating with internal data sources, may qualify. Costs incurred during preliminary project activities (evaluation, vendor selection) or post-implementation activities (training, maintenance) are expensed.

The two-component structure makes this allocation explicit. The IP agreement specifies asset consideration; the services agreement specifies integration fees. The firm does not need to disaggregate a blended payment post hoc.

Infrastructure (always OpEx). Hosting, compute, and model inference costs are expensed as incurred. They are the consumption of resources, not the acquisition of assets. This is true regardless of whether infrastructure is provided by the builder, by the firm, or by third parties.

9.3 Useful Life and Amortization

Once capitalized, the asset is amortized over its useful life, the period during which the asset is expected to contribute to the firm's operations.

Useful life for autonomous systems depends on several factors:

- **Technological obsolescence.** Foundation models and deployment practices evolve rapidly. A system built on current architectures may require replacement or substantial modification as underlying capabilities advance.
- **Operational relevance.** The firm's workflows, data structures, and governance requirements change over time. An implementation configured for today's operations may become misaligned as the business evolves.
- **Contractual or legal limits.** If the implementation incorporates licensed components with finite terms, useful life cannot exceed those terms.

In practice, useful lives for autonomous systems are likely to be shorter than for traditional software. Three to five years is a reasonable starting estimate, subject to revision as the asset class matures.

Amortization is typically straight-line unless another method better reflects the pattern of economic benefit. The firm should reassess the useful life annually and test for impairment if circumstances indicate that the carrying amount may not be recoverable.

9.4 Balance-Sheet Clarity: CAFA vs. Employment

Consider two firms acquiring economically identical autonomous systems — one through employment and the other through CAFA.

Firm A (Employment):

- Hires a builder at $200,000 annual salary
- Builder spends 18 months constructing the system
- Total cost: $300,000
- Accounting treatment: $300,000 expensed as wages over 18 months
- Balance sheet: no asset recognized
- P&L impact: $300,000 reduction in operating income

Firm B (CAFA):

- Engages a builder under CAFA

- Pays $200,000 asset consideration upon acceptance
- Pays $100,000 for integration services over 6 months
- Total cost: $300,000
- Accounting treatment: $200,000 capitalized as intangible asset; $100,000 expensed (or partially capitalized under ASC 350-40)
- Balance sheet: $200,000 asset recognized
- P&L impact: $100,000 operating expense plus ~$50,000 annual amortization (assuming 4-year useful life)

The economic outcome is identical: both firms spent $300,000 and received an autonomous system. But Firm A shows no asset and a $300,000 earnings hit. Firm B shows a $200,000 asset and a smaller current-period expense.

This is not accounting arbitrage. It is a correct classification. Firm A misstated the transaction by treating capital formation as labor expense. Firm B reported the transaction in accordance with its economic substance.

For CFOs, this distinction matters. Internal AI initiatives funded through headcount produce years of operating expense with no balance-sheet asset and require write-off justifications when projects stall. CAFA produces a capitalizable asset that appears on the balance sheet only after the system has demonstrated value. The accounting aligns with the risk: capital is committed upon acceptance, not upon hiring.

Section 10: Implementation Guidance

Sections 7 through 9 specified the contractual framework, ownership architecture, and accounting treatment. This section provides practical guidance for builders and firms implementing CAFA transactions.

10.1 The Adoption Boundary: When CAFA Is Appropriate

CAFA addresses a specific transaction type: the acquisition of firm-specific autonomous systems as capital assets. It does not replace employment, services, or SaaS. Each structure remains appropriate for its intended domain.

Employment remains appropriate when the firm needs internal capability, meaning the ability to build systems, not merely the capacity to operate them. Firms whose business is AI, or whose work requires ongoing research and exploratory development, should hire. Employment is also appropriate when the builder does not arrive with a substantially formed platform and the work is genuine greenfield development requiring firm resources and direction.

Services agreements remain appropriate for bounded engagements with defined deliverables, such as reports, analyses, or prototypes, where the firm does not need or want ongoing ownership of an autonomous system.

SaaS remains appropriate for commodity functions where firm-specific configuration is unnecessary and vendor-managed updates are preferable to internal operational responsibility.

Acquiring an Autonomous Capital Asset through CAFA becomes appropriate when the conditions identified in Part I (Section 5.5) coincide: the firm wants productive capacity rather than internal capability; talent is inaccessible through conventional hiring; the firm can evaluate outcomes but not processes; risk transfer to the builder is preferable; the use case requires firm-specific ownership; and the system will generate value over multiple periods. Where these conditions hold, CAFA provides the structure that employment and services cannot.

When they do not, other instruments remain preferable.

10.2 For Builders

Builders considering CAFA transactions should attend to four areas: engagement identification, platform separation, IP protection, and negotiation strategy.

Identifying CAFA-appropriate engagements.

Not every firm encounter is a CAFA opportunity. Many firms that express interest in "AI solutions" are actually seeking employees, consultants, or SaaS subscriptions. The builder should qualify engagements before investing in CAFA-structured proposals.

Signs that an engagement may be CAFA-appropriate:

- The firm has a specific operational problem it wants solved, not a general interest in "exploring AI"
- The firm expresses a preference for ownership over vendor dependency
- The firm has budget authority for capital expenditure, not just operating expense
- The firm can articulate acceptance criteria, including what "working" looks like
- The firm's timeline aligns with asset acquisition (months), not capability building (years)

Signs that an engagement is probably not CAFA-appropriate:

- The firm wants to "hire someone to build AI internally"
- The firm cannot articulate what the system should do
- The firm's primary concern is "access to talent" rather than operational outcomes
- The firm expects indefinite iteration without acceptance milestones

- The firm's procurement process is optimized for services or SaaS, with no pathway for asset acquisition

Qualification matters because CAFA proposals require different positioning than employment applications or service pitches. Presenting an asset-transfer structure to a firm expecting a staffing solution creates friction and confusion.

Structuring the platform/implementation separation.

The three-layer architecture requires the builder to clearly delineate what is retained (the platform) and what is transferred (the implementation). This separation should be established before engagement, not negotiated during integration.

The platform includes:

- Reusable orchestration logic and workflow patterns
- Data schemas and representational structures developed across prior engagements
- Integration methodologies and deployment playbooks
- Proprietary tooling and development environments
- Accumulated knowledge encoded in the builder's processes

The implementation includes:

- The system instance deployed within this firm's operational context
- Workflow encodings specific to this firm's operations
- Integrations with this firm's data sources and infrastructure
- Business rules and decision logic tuned to this firm's context

Gray areas will exist. The builder should establish principles for resolving them:

- If a component existed before the engagement, it is a platform
- If a component would be useful to other clients without modification, it is a platform
- If a component is meaningless outside this firm's context, it is an implementation
- If a component is an improvement to the platform occasioned by this engagement, it is a platform (but the implementation may incorporate it)

Documenting this separation in the IP agreement, with specificity about which transfers are permitted and which are not, prevents disputes during integration.

Protecting platform IP.

The builder's platform is the source of ongoing viability. Protecting it requires attention at three stages.

Before engagement:

- Maintain clear records of platform development predating any client relationship
- Use version control with timestamps demonstrating prior existence
- Consider provisional patent filings for novel platform components
- Document the platform's use in prior engagements (without disclosing client confidential information)

During engagement:

- Ensure the IP agreement explicitly lists retained platform components
- Resist scope creep that pulls platform elements into "work product"
- Maintain separation between platform development (builder's initiative) and integration work (client's engagement)
- Document which improvements arise from platform evolution versus client-specific requirements

After engagement:

- Confirm that the delivered implementation does not include platform source code unless explicitly licensed
- Retain copies of all platform components as of engagement completion
- Update platform documentation to reflect improvements made during the engagement

Negotiation strategy.

CAFA transactions involve two negotiations: asset consideration and integration fees.

Asset consideration reflects:

- The value of the implementation to the firm (cost savings, revenue enablement, risk reduction)
- The builder's prior investment in platform development
- Market comparables (if any exist)
- The firm's alternative: what would internal development or other solutions cost?

Integration fees reflect:

- Market rates for comparable technical labor
- The scope and duration of integration work
- The builder's opportunity cost during the integration period

Builders should resist collapsing these into a single number. The separation matters for both parties: the firm needs it for capitalization; the builder needs it to establish that platform value is being compensated separately from labor.

Common negotiation mistakes:

- Pricing the entire transaction as "services" to fit the firm's procurement process (loses capitalization benefit and undervalues the asset)
- Accepting below-market integration fees in exchange for higher asset consideration (creates cash flow problems during integration)
- Failing to specify acceptance criteria clearly results in indefinite integration periods and disputed completions.
- Agreeing to platform transfer under pressure (destroys the builder's business model)

10.3 For Acquiring Firms

Firms considering CAFA transactions should attend to four areas: opportunity recognition, evaluation structure, internal alignment, and integration management.

Recognizing CAFA opportunities.

Builders with substantially formed systems often appear through conventional hiring channels. They apply for roles, respond to job postings, or are introduced by recruiters. The firm's intake process, which is designed for labor procurement, may not recognize that the opportunity presented is an asset acquisition.

Signs that a candidate or consultant may be a CAFA opportunity:

- The builder describes prior implementations in specific operational terms, not general capabilities
- The builder has a demonstrable platform or methodology, not just skills
- The builder proposes outcomes rather than activities ("I can give you a system that does X" rather than "I can work on your AI initiatives")
- The builder resists standard employment terms, particularly IP assignment clauses
- The builder asks about the firm's operational problems, not its team structure or career paths

When these signs appear, the firm should consider whether the engagement is better structured as an asset acquisition than as hiring. This structure may require engaging different stakeholders (e.g., finance and legal) than a standard recruiting process.

Structuring evaluation.

Part I (Section 5.6) described the evaluation paradox: the firm cannot fully evaluate the system without integration, but integration has costs. CAFA resolves this through staged consideration, but the firm must still structure evaluation appropriately.

Pre-engagement evaluation:

- Review the builder's prior implementations (with appropriate confidentiality protections)
- Assess the builder's platform, not to acquire it, but to understand the foundation on which the implementation will be built
- Define the operational problem clearly: what does the firm need the system to do?
- Establish preliminary acceptance criteria: what would "working" look like?

Integration-period evaluation:

- Provide the builder with access to necessary infrastructure, data, and personnel
- Allocate internal resources for testing, feedback, and iteration
- Monitor progress against acceptance criteria
- Maintain clear communication about what is and is not meeting requirements

Acceptance evaluation:

- Test the system against defined criteria in the firm's operational environment
- Verify that the system performs autonomously within specified parameters
- Confirm integration with firm infrastructure and controls
- Document any exceptions or limitations

The firm should resist two failure modes: premature acceptance (paying asset consideration before the system demonstrates value) and indefinite evaluation (never reaching acceptance because criteria keep shifting).

Internal alignment.

CAFA transactions cross organizational boundaries that conventional engagements do not. Successful implementation requires alignment among stakeholders who may not typically collaborate.

Stakeholders to involve:

- **Operations** — defines the problem, provides access, evaluates outcomes
- **Finance** — approves capital expenditure, oversees capitalization treatment
- **Legal** — reviews IP terms, ensures platform/implementation separation is clearly documented
- **IT/Security** — assesses infrastructure requirements, reviews integration architecture
- **Procurement** — may need to accommodate a transaction type outside standard processes

The firm should identify an internal sponsor with authority across these functions. CAFA transactions that become stuck in services-oriented procurement processes or lack finance buy-in for capitalization are unlikely to close.

Managing integration.

The integration period is when structural debt (Section 6.3) would surface under employment or services. Under CAFA, the contractual structure reduces but does not eliminate integration risk.

The firm should:

- Assign a counterpart with operational knowledge and decision authority to work with the builder
- Establish regular checkpoints against acceptance criteria
- Resolve scope questions promptly; ambiguity compounds during integration
- Avoid treating the builder as an employee (directing methods, requiring attendance, asserting managerial authority) — this reintroduces the structural debt CAFA is designed to avoid
- Document progress, issues, and resolutions; this record matters if acceptance is disputed

If integration reveals that acceptance criteria cannot be met, the firm should address this directly. CAFA provides a clean exit: the firm pays for integration labor incurred, the IP does not transfer, and both parties move on. This relationship is preferable to indefinite extensions, with the hope that the problems will resolve.

10.4 From Roles to Contributions

CAFA reframes the human's relationship to productive capacity by changing the basis on which control is transferred.

Under employment, humans occupy roles. A role grants possession: access to firm resources, authority within a defined scope, and compensation for time. Humans work with capital but do not own it. Intellectual property created within the role is assigned to the firm by default. When the role ends, whether through termination, reorganization, or automation, the relationship also ends. The human departs with skills and experience, but no asset. What was built belongs to the firm.

An Autonomous Capital Asset Transaction operates differently. The builder enters as an asset owner, not as a candidate for a role. The builder controls a productive system formed on pre-existing technical capital. The firm does not grant access; it negotiates transfer. Ownership of a firm-specific implementation moves from the builder to the firm, pursuant to a discrete transaction governed by acceptance criteria and consideration that reflect the asset's value, not merely the time spent integrating it.

This is a balance-sheet transaction. The asset appears on the firm's books and consideration flows to the builder. When consideration is cash, the builder recognizes the value of prior formation while remaining outside the firm's equity structure. When consideration is equity, structured as an in-kind capital contribution, the builder becomes a partial owner of the firm. The choice depends on objectives. Cash provides liquidity. Equity aligns long-term incentives and may be preferable when the firm is capital-constrained or when continued builder involvement is valuable.

The transfer itself is singular and bounded. It occurs upon acceptance. This transaction does not preclude ongoing relationships. The firm may subsequently absorb the builder as an employee or contractor for integration, support, or further development. The asset continues to operate within the firm's environment. What changes is the structure of the initial engagement: the builder is not subordinated into a role that creates value to be later appropriated. The builder transfers an owned asset.

This distinction matters beyond individual transactions. When expertise can be encoded into transferable systems, those who encode it participate in value creation as owners of assets, and potentially of firms, rather than merely as suppliers of effort. Returns to accumulated expertise flow to those who accumulated it through asset consideration or equity, rather than being captured entirely by the firm that employed them when the expertise was applied.

CAFA does not eliminate roles or replace employment. It clarifies when ownership, rather than possession, is the economically accurate description of the transaction. Where autonomous systems perform the operative role, CAFA provides an institutional structure that allows capital to be acquired as capital and compensated as such.

But CAFA also addresses a prior failure that occurs before the transaction is even contemplated.

The Asset as Entryway

For most participants in the technology economy, existing paths are adequate. Engineers join large companies, subspecialize, and participate in the RSU economy. Entrepreneurs raise venture capital, build products, and capture markets. Both paths have clear entry points, established norms, and well-understood compensation structures.

There is a cohort for whom neither path works.

These are often polymaths — individuals whose expertise integrates multiple disciplines. They may combine engineering with economics, law with systems design, or operations with architecture. Their value lies in synthesis rather than specialization. But synthesis is difficult to hire. Role definitions assume discrete functions. HR taxonomies carve work into fragments. The polymath doesn't fit the boxes because the boxes were designed for subspecialists, optimized to capture the gains from specialization.

Yet it is precisely this synthesis that supports the identifiability requirement discussed in Section 9.1. Specialists recombine known elements within a single domain — inventions that are more likely to constitute prior art. Polymaths combine elements across domains that do not typically intersect — combinations that are more likely to cross the non-obviousness threshold required for patentability. The same integrative capacity that makes the polymath difficult to place in a role taxonomy makes their output more defensible as intellectual property. The gains from generalization, unrecognized by role-based hiring, manifest directly in the legal excludability that transforms a demonstration into an asset.

The venture path is equally misaligned. Venture capital funds operate at a horizontal scale, distributing products to many customers. But the polymath's natural output is often vertical: systems designed to integrate deeply into a single firm's operations. When the optimal end state is vertical integration rather than market ownership, the VC model offers no viable exit. These individuals can build. They can create autonomous systems that generate real value. But they lack a path to realization. No role fits. No funding model aligns.

The Autonomous Capital Asset Transaction provides the entryway. The builder creates an asset. The asset transfers to the firm. The asset requires stewardship, including operation, maintenance, and extension. The builder, having created the asset, is the natural steward. Absorption follows not because HR defined a role, but because the asset defined one. This inverts the traditional sequence. The organization does not define a role and test fit. The asset defines the role, and the builder fills it. The polymath who could not pass an HR screen because the firm lacked language for their contribution enters through a door that the asset itself created. The firm benefits correspondingly. It acquires productive capacity and the person best equipped to steward it. It gains access to talent that conventional hiring cannot surface, not because the talent was hidden, but because the firm lacked a mechanism to recognize it. This structure thus clears two markets at once. It expresses the transfer of autonomous systems that existing contracts cannot accommodate. It provides an entry point for a class of human capital that existing role taxonomies cannot accommodate. The asset is the interface between them.

Section 11: Implications for the Labor Market

The preceding analysis has focused on the structural mechanics of autonomous capital asset transactions. However, the shift from pure labor procurement to a model that includes asset acquisition has broader implications for firms, workers, and the institutions that connect them.

11.1 For Firms and Recruiters: Distinguishing Labor from Capital

The recruiter's traditional function is to match candidates to roles. The underlying assumption is fungibility: the firm needs a software engineer; the recruiter finds software engineers; the best match wins. Compensation is benchmarked against market rates for the role. The candidate's value lies in their ability to perform the job description.

This model fails when the person being recruited is not a candidate.

Increasingly, firms reach out to individuals who are not seeking employment: independent builders operating systems, running one-person or small-team enterprises. These individuals are not "on the market." They are not selling labor. They are developing assets. When the recruiter contacts them with a job offer, the implicit ask is: abandon what you are building and come work for us. The offer prices them as labor when they are functioning as capital holders.

The shift requires a different engagement logic. Instead of asking, "Can this person do the job?" the firm must ask, "Is this person building something we would otherwise need to create ourselves?" If yes, the conversation is not a recruitment activity. It is an acquisition. This is not a minor adjustment. It requires firms to evaluate demonstrated capability rather than credentials

alone. It requires compensation discussions to include asset consideration, not just salary and equity. It requires legal and finance involvement earlier, before the offer, not after. Firms that learn to make this distinction will acquire capability faster, at higher success rates, and with a cleaner balance-sheet treatment. Firms that do not will continue to lose builders who decline to be mispriced, or, worse, will acquire assets unknowingly through instruments that create structural ambiguity for both parties.

11.2 For Workers: Displacement, Transformation, and Opportunity

The rise of autonomous systems raises an unavoidable question: what happens to jobs?

The honest answer is that some work will be displaced. Synthetic labor, defined as autonomous systems performing operational functions, can substitute for human effort across an expanding range of tasks. When a system can perform a function at lower cost, higher reliability, and greater scale, the human labor previously required becomes economically redundant.

But displacement is not the only path.

Direct labor will persist. Humans will continue working, often using AI as a tool, but remaining the operative agent. A surgeon using robotic assistance, a designer using generative tools, an engineer using AI-assisted analysis: these roles are augmented, not replaced. The human remains essential to execution.

Much of what has historically been called indirect labor (processing, routing, checking, coordinating, formatting) exists not because it requires distinctively human contribution, but because humans were the only available mechanism to perform these functions. This is the operational layer that synthetic labor can absorb. The work is not trivial; much of it required judgment. But it is *encodable*, capable of being compressed into systems that perform the operative role autonomously.

As this layer is automated, human labor is freed from functions that, while demanding, do not require human presence. What remains is work that requires human capabilities: creativity, judgment, physical presence, relational trust, and, for those with the requisite expertise, the design of systems that now perform operational work.

For workers with the expertise to form autonomous systems, the opportunity is not to compete with AI but to create it. This inverts the traditional labor model. Instead of selling ongoing effort at market rates, the worker front-loads expertise into a system, transfers that system to a firm as an asset, and captures value at the point of transfer. The decisive contribution is formation, not operation. The worker is compensated not for time but for what they built.

Stewardship may follow. The firm that acquires an autonomous system may want the builder to remain, not to perform the operational function the system now handles, but to extend, refine, and evolve the asset over time. This is not displacement into a monitoring role. It is a continuation of the formative relationship: the builder who created the asset continues to develop it, compensated for ongoing expertise rather than operational labor.

This raises a more difficult question: what about workers who lack the expertise to develop autonomous systems and whose roles are primarily operational?

Two paths emerge.

The first is redirection toward direct labor: work that requires human presence, creativity, judgment, or relational trust. Care work, skilled trades, creative production, human-to-human services, roles where embodiment matters. These functions cannot be performed by synthetic labor, not because AI is incapable of intelligence, but because the value resides in human presence itself. The economy has always had such roles; as operational labor is automated, human employment will increasingly concentrate in them.

The second is participation in the inverted model at smaller scales. Not every worker can form an enterprise-grade autonomous system. However, many workers possess deep process expertise: an understanding of how an operation functions, where exceptions arise, and what the undocumented rules are. This expertise has value if it can be encoded before AI acquires it independently. With AI assistance, workers can compress their knowledge into transferable systems: documented processes, operational playbooks, and structured methodologies that a firm can acquire, own, and deploy. This may be a one-time opportunity rather than a sustainable role: once the expertise is encoded, it is transferred. But for workers whose knowledge would otherwise be displaced without compensation, capturing that value through asset formation is preferable to watching it evaporate.

The sports analogy is instructive. The athlete's value is not benchmarked against the "market rate for players." It reflects their specific, non-fungible contribution. The contract, which is the asset, captures that individual marginal product.

The same logic can apply to builders of autonomous systems. The worker who arrives with a formed productive capacity is not selling labor. They are transacting as an asset holder. If the market develops instruments to recognize this, as the Capital Asset Formation Agreement seeks to do, compensation can reflect the value of what was built, rather than merely the time available.

This does not eliminate disruption. Workers whose roles are purely operational, and who lack the expertise to form the systems that replace them, face genuine displacement. But for those

who can make the transition from performing to forming, the future is not wage compression but value capture, a structural improvement in economic position that the current labor market cannot express, but that new instruments may enable.

11.3 For the Market: Toward Hybrid Instruments

The labor market was built for labor. Its instruments (job postings, interviews, offer letters, employment agreements) presume that what is being transacted is the capacity to work, not the transfer of formed capital.

The capital market was built for capital. Its instruments (term sheets, asset purchase agreements, M&A processes) assume that the transaction involves ownership of entities or discrete assets, rather than the integration of a person's expertise into ongoing operations.

Autonomous systems fall between these categories. The builder arrives as a person but holds an asset. The transaction requires both: labor market discovery (finding the builder) and capital market mechanics (transferring the asset). Neither market, as currently structured, can execute the full transaction alone.

As discussed earlier, professional sports contracts provide a canonical example of a hybrid instrument that combines labor and asset characteristics in a single structure. The athlete performs labor, but the contract itself is a transferable, term-limited property right that captures individually attributable marginal product. That model demonstrates that labor relationships can carry asset characteristics when the underlying conditions are satisfied.

The Capital Asset Formation Agreement is one such instrument. But it is not the only possible structure. The broader need is for mechanisms that can:

- Recognize when a person being recruited holds transferable productive capacity
- Price that capacity as an asset, not as labor
- Structure consideration that reflects asset value alongside any labor component
- Allocate ownership cleanly between what transfers and what the builder retains
- Integrate into existing labor market workflows without requiring M&A-scale process

The sports market solved a version of this problem decades ago. Player contracts bundle labor and asset in a single instrument, recognized by both parties and by accounting standards. The broader labor market has not yet developed equivalent structures for knowledge workers who hold intellectual capital.

As autonomous systems proliferate, this gap will become increasingly costly. Builders will be mispriced; firms will acquire assets under inappropriate instruments; value will be lost to

structural ambiguity. The market opportunity for legal innovators, recruiting platforms, and financial intermediaries is to build the infrastructure that enables these transactions to clear.

The question is not whether labor markets will adapt, but rather how quickly and who will develop the instruments that enable adaptation.

Conclusion: Making Autonomous Capital Legible

This paper introduced two modifications to economic theory and one institutional mechanism to operationalize them.

First, we introduced synthetic labor (S) as a third input in the Cobb–Douglas production function. This modification to the classical capital-labor framework recognizes that autonomous systems do not merely augment human effort; they perform the operative function itself. Output can now persist as human operating labor approaches zero, because the operative role has been reassigned to a non-human factor that is ownable, persistent, and capitalizable.

Second, we reframed the theory of the firm. When operational intelligence is embedded in capital rather than supplied by labor, the firm's function shifts from Coasean labor coordinator to capital allocator. For a growing category of production, the firm is better understood as a clearinghouse for autonomous productive assets than as an organizer of human workers.

These theoretical claims have immediate institutional consequences, and it is here that what has lagged is not technology, nor firm demand, but institutional form.

In practice, transactions involving externally formed autonomous systems are forced through contractual categories designed for something else. Employment agreements assume the provision of labor within the firm's production environment. Services agreements presume effort-for-fee exchanges or ongoing access. SaaS licenses presume vendor dependence. None can express the transfer of a firm-specific productive asset formed on pre-existing technical capital. The result is systematic misclassification: capital formation is treated as labor procurement, with predictable consequences for ownership, pricing, accounting, and incentives.

The Capital Asset Formation Agreement addresses this gap. It does not introduce novel legal doctrine. It assembles familiar instruments (IP transfer, services agreements, acceptance testing, staged consideration) into a coherent structure that aligns legal form with economic substance. By separating platform from implementation, capital consideration from integration labor, and asset transfer from infrastructure consumption, CAFA renders legible a transaction that already exists in economic reality but lacks an institutional expression.

This clarity has second-order effects.

For firms, CAFA provides a way to acquire productive capacity without assuming the ongoing obligations of employment or vendor dependency. The firm commits capital only after value is demonstrated. The resulting asset appears on the balance sheet rather than as an operating expense. Capital allocation decisions become clearer, and the risks associated with exploratory hiring or open-ended service engagements decline.

For builders, CAFA provides a path to liquidity that does not require venture-scale growth. Builders can monetize accumulated expertise (encoded in platforms and deployment methodologies) through serial asset transfers rather than time-based labor. Compensation reflects contribution, not tenure. The builder receives payment for prior formation and for current integration work, without surrendering the means of future production.

For markets, this structure addresses a clearing failure. Asset transactions are not new. What is new is that economically viable assets now arrive through labor market channels, attached to builders who appear as candidates or consultants rather than as asset holders. The labor market can discover these assets but cannot transact them. The Capital Asset Formation Agreement provides a structure that allows it to do so. It enables the transfer of bespoke, firm-specific systems as owned assets without collapsing into licensing models or staffing arrangements. This expands the range of feasible transactions without displacing existing ones. Employment remains appropriate when firms require internal capabilities. Services remain appropriate where outputs are bounded. SaaS remains appropriate where functions are commodities. The structure applies only where autonomous systems can operate as durable, productive assets.

None of this implies the disappearance of labor, firms, or employment contracts. It implies a refinement of institutional vocabulary. When technology changes the nature of what is being transacted, institutions must adapt to accurately describe it. Treating capital as labor does not make it so; it merely obscures ownership, distorts incentives, and accumulates structural debt. The Capital Asset Formation Agreement offers a way to clear transactions that are already mutually beneficial but institutionally blocked. Making autonomous capital legible enables firms and builders to transact on terms that reflect the nature of the exchange. In doing so, it provides infrastructure not for automation itself, but for the lawful, intelligible formation and transfer of autonomous productive assets.