SAMPLE

Building the Business Infrastructure of Tomorrow

Building the Operating Backbone of the Enterprise for Tomorrow

Aiman JoeHanz

Free sample: Introduction & Chapter 1

putrabooks.github.io

Buy the full book:

Carousell carousell.sg/p/building-the-business-infrastructure-of-tomorrow-ebook-pdf-1467068281

Gumroad

Ko-fi

ElevenReader

Payhip

Contents

Introduction

Chapter 1: The Business as a System — What Business Infrastructure Actually Is

Chapter 2: The Records Layer — ERP, Ledgers, and the Company's Memory

Chapter 3: The Customer Layer — CRM, Commerce, and the Front of the House

Chapter 4: The People Layer — HR, Payroll, and the Employment Machine

Chapter 5: The Money Layer — Payments, Treasury, and the Flow of Value

Chapter 6: The Integration Layer — APIs, Events, and the Connective Tissue

Chapter 8: The Identity Layer — Access, Trust, and the Security Posture

Chapter 9: The Transformation Problem — Selection, Implementation, and Change

Chapter 10: The Re-Foundation — When the Backbone Learns to Reason

Frequently Asked Questions

Glossary

About this Series

Connect

This free sample contains the Introduction and Chapter 1. The complete book contains the full contents above.

Introduction

Every business, whatever it sells, runs on a second business that no customer ever sees: the business of keeping records, moving money, tracking people, and coordinating work. A bakery keeps this second business in the owner's head and a shoebox of receipts. A multinational keeps it in a lattice of systems — ledgers, registers, pipelines, and contracts — so dense that no single person understands all of it, and so load-bearing that a failure in one corner stops production lines, paychecks, or shipments on another continent. This lattice is business infrastructure: the deliberately designed backbone of records, money, people, data, and integration on which the visible business stands.

The phrase deserves the same treatment this series has given cloud services and the industrial base of machine intelligence. Business infrastructure is not a product category; it is the enterprise's operating substrate, and it has a shape — a layering of records, customer systems, people systems, money systems, connective tissue, data, and identity — that is more durable than any vendor's logo. Understanding that shape is the difference between a company that steers its systems and a company that is steered by them; between a transformation budget that compounds and one that evaporates; between an organization that can adopt machine intelligence deliberately and one that bolts it onto a foundation that was never designed to carry it.

This book describes the backbone layer by layer. It begins with what business infrastructure actually is — the anatomy of a company as a system rather than an org chart — and then walks the great strata: the records layer, where the company's memory lives; the customer layer, where demand is recorded and served; the people layer, where employment becomes process; the money layer, where value moves; the integration layer, where systems speak to each other; the data layer, where the company learns from itself; and the identity layer, where trust is enforced. The final chapters turn to the two questions every reader actually faces: the transformation problem — why changing this backbone is the hardest engineering work in business — and the re-foundation that begins as machine intelligence moves into the systems of record themselves.

As throughout the series, the approach here is structural rather than promotional. Vendor feature lists, market statistics, and adoption percentages are omitted: all three decay faster than print does. Systems re-brand themselves every decade or so; what persists underneath — the layers, the bargains, the ways these systems fail — is what these chapters teach the eye to catch.

Chapter 1: The Business as a System — What Business Infrastructure Actually

Is

Ask a business leader to draw their company and you will usually get an org chart: boxes for people, lines for reporting. Ask the same question of the systems and you get a different picture entirely: the company as a flow of records and obligations, originating in commitments — an order, a hire, a purchase — flowing through approvals and transformations, and settling into the ledgers as facts. Both pictures are true, and the gap between them is the subject of this book: the org chart describes who decides, but the record flow describes what actually happens, and the infrastructure is the machinery that connects the two.

This chapter defines the term, sets out the anatomy that the rest of the book walks layer by layer, and establishes the properties that make business infrastructure a genuine engineering discipline rather than a software shopping list.

Business infrastructure, in the sense these chapters give the term, is the purpose-built assembly of systems and standards by which an organization records what it has done, coordinates what it is doing, and carries out what it has promised. It spans what practitioners call the systems of record — the authoritative stores for money, people, customers, and things — together with the systems of engagement that touch the outside world, the integration plumbing that connects them, and the standards — chart of accounts, document formats, approval thresholds, master-data definitions — that make the pieces operable as one machine.

The definition's key word is authoritative. A company has many spreadsheets that describe its customers; it has exactly one system whose answer is the answer, the place where a customer's balance, a contract's terms, an employee's status are settled and where every other representation defers. Business infrastructure is the construction and maintenance of that authority: which system owns which fact, how the facts move, who may change them, and what happens when two systems disagree. This is why the subject is engineering and not procurement — the difficulty is never the software's features; it is the design of authority itself, an organizational constitution written in configuration tables and interfaces.

The backbone that this book walks has seven strata, and stating them up front gives the reader the map. The records layer (Chapter 2) holds the company's

memory: the general ledger and the operational registers — orders, inventory, contracts, assets — that feed it. The customer layer (Chapter 3) is the front of the house: the systems where demand is recorded, relationships are maintained, and commerce happens. The people layer (Chapter 4)

industrializes employment itself: recruitment through payroll through offboarding, a legal process expressed as a data pipeline. The money layer (Chapter 5) is where value moves: payments, collections, treasury, and the reconciliation discipline that keeps the company honest.

The integration layer (Chapter 6) is the connective tissue: the interfaces, events, and canonical models through which the strata speak. The data layer (Chapter 7) is where the company learns from itself: the warehouses, governance, and definitions that turn operational exhaust into self-knowledge. And the identity layer (Chapter 8) is where the machine enforces trust: who may see, who may do, and how the organization proves it.

The strata are not a stack in the technical sense; they are a set of coupled systems, each with its own clock, vendors, and failure modes, held together by the integration layer and governed by the identity layer. What makes them one subject is that they fail together, succeed together, and are bought, replaced, and regretted together.

Four properties distinguish business infrastructure from ordinary software, and each will recur throughout the book. The first is continuity: these systems cannot be paused. The ledger must close, the payroll must run, the orders must ship — which means change happens through migration, coexistence, and cutover choreography, never through the clean-slate restart that greenfield software enjoys. The second is authority: the systems define the organization's official reality, so every design decision — which system wins an argument, whose data model defines "customer" — is an organizational decision expressed in technical form. The third is coupling to law and convention: tax codes, accounting standards, labor law, and payment conventions are wired directly into the configuration, so the infrastructure inherits the surrounding legal system's complexity and its change calendar. The fourth is longevity: these systems outlive their sponsors, their vendors, and frequently their vendors' vendors; the enterprise runs its backbone for decades, which changes every economic calculation from feature evaluation to risk.

From these properties follows the book's central claim: business infrastructure decisions are the least reversible and therefore the most consequential decisions an organization routinely makes. A product launch that fails is a write-off; a backbone that fails is a decade of constraint. The chapters that

follow equip the reader to make those decisions deliberately — and to recognize, in every vendor conversation and transformation proposal, which layer is actually being sold.

Like its predecessors in this series, this book names almost no vendors. The systems of record are known to practitioners by their products, but the products change hands, merge, and rebrand on a schedule unrelated to their function; the functions — the ledger, the register, the integration pattern, the reconciliation — are the durable layer, and they are what the reader needs. Where a structure is inseparable from a named category, the category appears as the practitioners use it: the ERP, the CRM, the HCM, the payment gateway. By the last page, the reader should be able to place any product announcement, any system selection, and any transformation program onto the map unaided — which is the point of a map.

Three audiences hold a stake in this backbone, and the book addresses all three. The operator — the finance leader, the systems owner, the process owner — gets the working anatomy and the failure modes of each layer. The decision-maker — the executive or board member who approves the budgets — gets the economic logic: why these projects cost what they cost, why they overrun, and what distinguishes a re-foundation from a write-off. And the professional whose career runs through these systems — the analyst, the auditor, the engineer, the consultant — gets the structural literacy that turns experience into judgment. The backbone is invisible precisely because it works; this book makes it visible so that its readers can govern it rather than merely inhabit it.

The backbone is older than the computer, and knowing its prehistory explains its present character. The earliest businesses ran on durable records almost immediately: ledgers on clay and paper, tallies and tokens, the double-entry web — because the moment an enterprise outgrew its owner's memory, its facts had to live outside its owner's head. The machinery of the nineteenth and early twentieth centuries — the filing cabinet, the punched card, the tabulating machine — mechanized the same web, and the discipline that ran it, the clerk's profession, was among the first to be industrialized.

The computer did not invent business infrastructure; it inherited it. The first enterprise systems were ledgers and registers moved onto machines, staffed by the same departments, governed by the same conventions, and this inheritance explains two facts that otherwise look odd. First, why the backbone's clocks — the period close, the payroll run, the fiscal year — are accounting conventions wearing software calendars: the machinery was designed around them, not the reverse. Second, why the records layer's

gravity (Chapter 2) persists in an era of endless software choice: the ledger is the oldest continuously operating information system in the enterprise, and everything else arranged itself around it. The reader who understands that business infrastructure is a five-century-old discipline that happened to acquire electronics will stop asking why the software behaves strangely, and start asking — correctly — why the organization does.

Every layer of the backbone arrives wrapped in vocabulary, and the vocabulary is worth decoding once, because it encodes vendor strategy more often than engineering meaning. The suite is the everything-in-one promise: one vendor's integrated stack for many strata, trading breadth for lock-in. Best-of-breed is the counter-claim: the strongest tool per layer, trading integration effort for capability. The platform is the current synthesis — a core with an ecosystem around it — and the reader should notice that in all three cases the word describes a commercial architecture first and a technical one second: what is being sold is a position in the buyer's organization, and the technical design follows the position.

The same decoder applies to the strata's product names. The enterprise resource system names the records layer from the vendor's ambition; the customer relationship system names the customer layer from the seller's hope; the human capital system renames the personnel department in the language of asset value — and each renaming teaches the same lesson: when a product category's name flatters the buyer's ambitions, the buyer is the product's market, not its user. The reader who runs the vocabulary test on any pitch asks the one question this chapter has equipped them for: which stratum does this actually touch, whose facts does it claim to own, and what did it have to promise the org chart to sound this good? The answers, in the mature buyer, become the requirements.

Every enterprise runs a second backbone beside the one on its architecture diagrams: the spreadsheet estate. The shadow system is business infrastructure's permanent understudy — built wherever the official backbone is too slow, too rigid, or too politically expensive — and it deserves structural treatment rather than the dismissal it usually receives, because it is diagnostic gold. A spreadsheet exists because some process's owner needed authority the official layer would not grant them: the pricing exception, the consolidated view, the model of a decision the systems could not express. The shadow estate is, read correctly, a map of the official backbone's failures — a monument to every place where the organization's real process diverged from its implemented one.

The mature operator treats the shadow system accordingly: as a requirements

backlog, not a compliance violation. Each enduring spreadsheet is a specification, written in the business's own hand, for what the backbone should have done; the operator who harvests those specifications — and either builds the capability or retires the need — retires the shadow system with it. The immature operator declares war on the shadow estate, which fails for the same reason prohibition fails: the shadow exists because a real demand has no legal supply, and suppressing the supply without serving the demand just relocates it. The diagnostic for the whole chapter: what are the ten longest-lived spreadsheets in your company, and what is each one doing that the systems of record will not?

The anatomy earns its keep, as in this series' other volumes, as a reading discipline. When the industry announces something — a product launch, an acquisition, a platform pivot — the reader with the map asks a sequence that converts any headline into structure. First: which stratum does this actually touch? Most "transformation" announcements are engagement-layer reskins with the strata untouched; most "suite" announcements are two vendors' salesforces agreeing on a commission split. Second: whose facts does it claim? A product that wants to own the customer master is a records-layer play wearing a customer-layer logo, and the buyer should price the treaty it is being asked to sign.

Third: what clock does it run on? Announcements about interfaces are quarterly; announcements about the ledger are decadal, and confusing the two is how enterprises buy quick things expensively and slow things cheaply. The reader who runs the sequence on a month of enterprise-software news will find that most of it dissolves on contact with the first question — and that the residue is exactly the material the rest of this book teaches them to evaluate.

Does size change the backbone? The answer is structural and slightly surprising: the layers do not change, but their gravity does. The small enterprise runs the same seven strata the multinational runs — records, customers, people, money, integration, data, identity — but compressed into a single system, a handful of spreadsheets, and the owner's memory; the physics are identical, and only the packaging differs. Growth does not add layers; it separates them. The first separation is usually the ledger's (the accounting outgrows the spreadsheet); the next is the customer layer's (the relationship outgrows the inbox); the money layer's follows (the pipeline outgrows the bank's website); and each separation is a rite of passage that the enterprise either engineers deliberately or acquires as a mess.

The reader running a growing company can therefore use this book as a schedule: the strata described in these chapters arrive in roughly this order,

and the enterprise that recognizes each arrival builds the next backbone on purpose instead of in crisis. Scale, in business infrastructure, is not a size; it is the number of facts the organization can no longer keep in anyone's head.

The backbone's oldest economic question deserves the structural answer: build the systems or buy them? The mature reasoning runs on two axes. The first is differentiation: the strata where the enterprise's process is its advantage — where how it prices, serves, or produces differs from the market's — earn bespoke machinery, because renting a generic process forfeits the advantage; the strata where the process is identical for everyone (payroll, statutory books, identity plumbing) earn products, because building a commodity is paying twice for a standard. The second axis is the maintenance tail, the property this book has priced at every layer: whatever the enterprise builds, it owns forever — with the talent it must retain, the security it must fund, and the roadmap it must staff, on a schedule the enterprise sets and cannot escape.

The modern industry has added a third option that dissolves the question's older form: the configurable platform — a purchased core the enterprise extends rather than builds or merely installs. The reader should evaluate the option with the vocabulary test armed: configuration is durable; extension is durable with a treaty (the integration chapter's contracts); modification is debt, compounding at every upgrade. The honest answer for most strata, most of the time, is buy the standard, configure the difference, build only the narrow surfaces where the company genuinely is its process — and treat any proposal to build a payroll as the diagnostic this book has already taught the reader to run.

The infrastructure runs on calendars the enterprise mostly did not choose, and knowing them is knowing the machine's tempo. The fiscal year sets the master cycle: the close's drumbeat, the plan's rhythm, the audit's season. The statutory calendar overlays it: filings, declarations, the payroll's tax deadlines, each jurisdictional obligation with its own clock and its own penalty schedule. The vendor calendar overlays both: upgrade windows, deprecation notices, the software's release cadence — external time arriving in the backbone at the integration layer. And the operational calendar — the business's own seasons, launches, and cycles — must somehow be reconciled with all three, because the enterprise's busiest month is exactly when it can least afford a cutover.

The mature operator maintains a single enterprise calendar — the four clocks overlaid — and governs change against it: no transformation milestone in the close, no upgrade in the peak season, no statutory deadline discovered late. The immature operator keeps the clocks in different departments and meets

them separately, discovering collisions only as incidents. The reader should perform the one-minute diagnostic: ask any enterprise for its change freeze calendar, and note whether the answer names the fiscal close, the statutory obligations, and the commercial peak — or only the first, in which case the enterprise is about to learn the other two calendars exist, on one of their dates.

The backbone is decided in a stack of rooms, and the reader should know the rooms, because each layer of this book has a room where its authority actually sits. At the top, the board's audit and technology committees hold the fiduciary questions: the risk appetite, the spending envelope, the oversight of the transformations. Below them, the executive table holds the operating model: which functions own which strata, what the enterprise's canonical bodies are, and the annual rhythm that governs them. In the middle, the governance bodies do the constitutional work this book has described without always naming its room: the architecture council that owns the canonical model, the data council that owns the definitions, the security committee that owns the posture, the change board that owns the calendar. And at the base, the owning functions run the machinery: the design decisions, the configuration, the queues, and the daily truth the registers certify.

The stack's health is the enterprise's real decision health, and its pathology is the bypass: the decision made in a room that does not own it, usually the vendor conversation that became a commitment before the councils heard of it. The mature enterprise wires its rooms deliberately — every layer of this book maps to a standing body with a name, a calendar, and a quorum — so that when the reader asks "who owns the customer master?", the answer is a room, a person, and a document. The immature enterprise answers with a shrug, and the shrug is the shadow backbone's signature, appearing one more time, at the enterprise's highest altitude.

The chapter's anatomy can be compressed into a single working document — the enterprise's backbone constitution — and stating it here gives the reader a template to carry. One: each canonical fact has exactly one authoritative register, named, owned, and published; all other copies are derived and refreshable. Two: each register's changes run on a governed clock — the close's, the payroll's, the market's — and no clock is allowed to collide with another without a calendar that says so. Three: every interface between registers is a treaty — documented, versioned, tested, and owned by both parties. Four: every act on the backbone is attributable — an identity, a permission, an audit trail — or it does not happen. Five: the whole machine is reconciled — continuously, by design — because the enterprise's honesty is not a virtue but a mechanism.

The reader should notice what the constitution is not: it is not a software list, not an architecture diagram, and not a strategy. It is the set of commitments that survive every vendor, every reorganization, and every technology wave — the five sentences a new chief executive should be handed on day one, and the five questions this book has taught the reader to ask. Enterprises that hold the constitution explicitly can change anything; enterprises that hold it implicitly can change nothing without breaking something they did not know was load-bearing. The chapters that follow are the constitution, expanded — layer by layer, evidence first.

The chapter closes with the argument the whole book will keep making, stated once as economics. A backbone that is not understood is not free; it bills constantly, in the currencies this chapter has named and the rest of the book will price: the reconciliation overtime, the transformation overruns, the audit findings, the shadow estates, the decisions made slowly or wrongly because the facts could not be found or trusted. Against those standing costs, the costs of structural literacy — the reading discipline, the diagnostic questions, the map of the seven strata — are trivial: they are paid once, in attention, and they compound in the opposite direction for as long as the enterprise lives.

Business infrastructure is the enterprise's operating substrate: seven coupled strata — records, customers, people, money, integration, data, identity — that hold the organization's authoritative reality. Its defining properties — continuity, authority, legal coupling, and longevity — make it the least reversible layer of an enterprise's decisions and therefore the one most worth thinking about structurally. The chapters that follow walk the strata in order, from the ledger outward, and close with the two disciplines that govern them all: changing the backbone without breaking the business, and re-founding it for the era in which the systems of record themselves begin to reason. The org chart shows who decides. This book is about the machine that decides what happened.

End of free sample. The full book is available at putrabooks.github.io

ABOUT THE AUTHOR

Aiman JoeHanz writes the Building Infrastructure of Tomorrow series, taking the machinery of modern life apart page by page until its moving parts sit on the table, labeled. In this volume he opens the back office: the records, registers, payment rails, and trust architecture that every business runs on, whatever its sign says.