

FOUNDATION AND DECISION GUIDE · F12

CPU-first AI, hardware, energy, and sustainability

What the compute model changes, how
published performance and energy claims are
bounded, and how to reproduce relevance on
your estate.

DECISION

SCOPE

EVIDENCE

OWNERSHIP

PROOF

EXIT



A complete guide to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

F12

CONTENTS

From the first question to a defensible decision

This 41-page guide turns 13 owned topics into an explanation, evidence requirement, proof challenge, and closing decision record.

PRIMARY READERS

Technology, infrastructure and sustainability teams

READING DEPTH

Technical

DECISION THIS SUPPORTS

Determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

01

Read this first

03–05

Audience, reader decision, canonical boundary, terms, and guide-specific factual spotlight

02

Understand the decision and its language

06–...

The infrastructure problem being claimed · Binary and integer-oriented computation in plain terms · Core and Kera beneath the workload · Sparse cognition and exact work · Distributed and spare capacity

03

Define the mechanism, controls, and evidence

...

Hardware target matrix · Benchmark anatomy · Energy claim anatomy · Cost model · Hardware lifetime and locality

04

Prove adoption, operation, continuity, and exit

...–39

A reproducibility plan · Decision worksheet · Limits and honest comparisons

05

Sources and next action

40–39

verification records and closing decision package

06

Subject index

40–41

An alphabetical finder for concepts, controls, products, and decisions.

TARGET LENGTH

9,000–13,000 words.. Page count varies with the number and depth of topics rather than forcing every subject into the same shell.

SOURCE DISCIPLINE

Dweve-specific facts are taken directly from the English website text audited 2026-07-20; web-navigation wording is humanised for print. Evaluation guidance is editorial, and scenarios and derived visuals are illustrative. Unsupported synthesis is replaced by canonical copy or omitted.

READ THIS FIRST • READER DECISION

Who should use this guide, and what decision it supports

infrastructure, finance, sustainability, architecture, and technical strategy teams.

Determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

The worked scenario is an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. It is illustrative and deliberately bounded. Replace it with the organisation's actual workflow, systems, data, sources, policies, roles, infrastructure, commercial conditions and consequences while preserving the evidence discipline.

The guide's concrete output is a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations. Use the chapter explanation pages to align language, the evidence pages to create acceptance records, and any challenge pages to test whether the boundary survives missing assumptions and failure. End in a named decision, not an attractive summary.

COMPUTE STANCE**CPU-first**

Current guide lens; verify source and scope.

PUBLISHED CLAIM**96% energy framing**

Current guide lens; verify source and scope.

PUBLISHED FIGURES**366W per server**

Current guide lens; verify source and scope.

DECISION RULE**Reproduce workload**

Current guide lens; verify source and scope.

PRIMARY RISK

A fast binary kernel or energy-efficient component can be a small part of the full path; data movement, source/model services, preprocessing, storage, evidence, idle capacity, cooling, operations and hardware refresh can change total cost and impact.

READ THIS FIRST • DISCIPLINE

How to read claims, evidence, responsibility, and status

The website is the only factual source for this edition. The guide adds explanation, architecture and proof requirements without inventing current product, regulatory, assurance, commercial, or customer-result facts.

Six evidence layers

A published description states a current proposition. Delivered software and documentation show what is available in a selected version. Customer configuration shows how it is used. An observed run shows one result. Independent assurance has scope, method and date. A written agreement creates commercial commitments and remedies.

Responsibility stays explicit

The relevant owners include infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review. The supplier can provide product, evidence and commitments; customer owners still decide purpose, source and policy authority, professional or domain judgement, acceptable risk, infrastructure, staffing, affected-person treatment and production reliance.

Status and strong statements

Measured claims stay tied to workload, hardware and method. Research remains research. Alignment is not certification. "Open" is verified per component and licence. Pricing is re-audited at release. Roadmap and availability are not assumed. Conflicts or ambiguity become diligence items.

LAYER	EVIDENCE	CANNOT ESTABLISH ALONE
Website	website copy + captured date	Delivery, fit, contract
Delivery	Manifest + docs + artefacts	Customer result
Configuration	Data/roles/policy/posture	Operating outcome
Run	Raw output + target + packet	General performance
Assurance	Scoped test/report + findings	Universal security/compliance
Agreement	Rights + duties + remedy	Technical behaviour without test

Keep the layers separate in every chapter and decision record.

READING RULE

The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions.

READ THIS FIRST • CANONICAL SPOTLIGHT

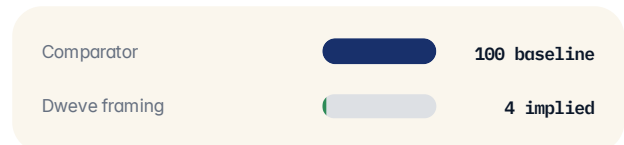
Read the energy claim as a measurement specification

The website publishes a strong comparison; responsible buyers preserve its exact boundary and reproduce the workload they actually care about.

The sustainability website description frames a Dweve server at 366W against one NVIDIA DGX H100 under a 60% utilisation assumption and states a 96% reduction, also described as a 25× efficiency gain. It connects the figure to binary-first compute, fixed-point integer mathematics and sparse activation of a small subset of experts. These are website claims, not this guide's independent findings.

A reproduction needs equivalent work and correctness, wall-power method, idle and utilisation treatment, hardware configuration, CPU/ISA/backend, software and build, model or operation path, input/corpus, warm/cold state, batching/concurrency, duration/repetitions, latency/throughput, accuracy/recall, storage/network and cooling boundary. Report absolute values and uncertainty before extrapolating.

The organisational decision includes servers and accelerators, utilisation, electricity, cooling/PUE, rack/network/storage, staff, licences, embodied hardware and refresh, availability, data movement, site capacity and optional provider paths. Hardware reuse and locality can be valuable even when the measured reduction differs from the published comparison.



Normalised visual of the website's published 96% reduction framing, not an independent measurement. The published hardware, utilisation, workload-equivalence, and method remain attached.

GUIDE-SPECIFIC PROOF

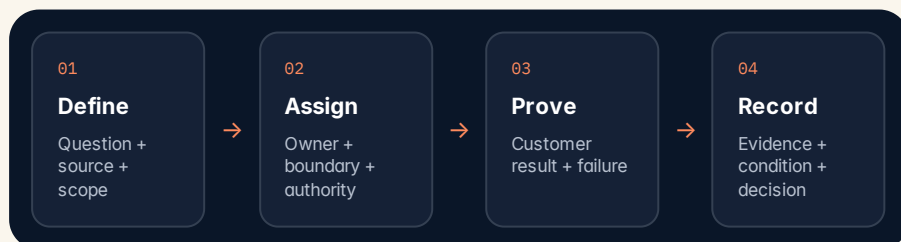
Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results.

01

CPU-FIRST AI, HARDWARE, ENERGY, AND SUSTAINABILITY

Understand the decision and its language

Define the real question, current state, responsibilities, source claims, and the artefact this guide must produce.



The same evidence discipline applies to every chapter in the section.

CHAPTER 01 • EXPLAIN

The infrastructure problem being claimed

GPU dependence, capacity queues, electricity, cooling, capital, and supplier concentration.

This chapter owns a specific part of the reader's decision: gGPU dependence, capacity queues, electricity, cooling, capital, and supplier concentration.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "The infrastructure problem being claimed", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

GPU dependence

Define the exact proposition and current authoritative source.

LENS 2

Capacity queues

Name the responsible person, system, dependency, and version.

LENS 3

Electricity

Specify the evidence and how an independent reviewer checks it.

LENS 4

Cooling

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 01 • APPLY

Turn “The infrastructure problem being claimed” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, gPU dependence, capacity queues, electricity, cooling, capital, and supplier concentration., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant source and version, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for the infrastructure problem being claimed; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 01 • CHALLENGE

Stress-test the boundary behind “The infrastructure problem being claimed”

A deep-dive topic earns an additional page for counterexamples, adversarial conditions, missing dependencies, role failure, recovery, and evidence limits.

The principal guide risk is: A fast binary kernel or energy-efficient component can be a small part of the full path; data movement, source/model services, preprocessing, storage, evidence, idle capacity, cooling, operations and hardware refresh can change total cost and impact.

For this chapter, remove or alter the strongest assumption in “GPU dependence, capacity queues, electricity, cooling, capital, and supplier concentration.”. Change the source or version, revoke a person or right, replace a delivered dependency, interrupt the selected deployment, introduce incompatible state, withhold a target acknowledgement, tamper with evidence, or make the relied-on contract condition unavailable. Observe whether the boundary refuses, degrades, escalates, reconciles, recovers, or fails silently.

Run the fault from the perspective of infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review. Each owner should see only the authorised information needed to act, understand the current state, and have an explicit safe next step. Restore normal operation, retrieve a pre-change record, and decide whether the chapter proposition still holds, now holds conditionally, requires remediation, or blocks reliance.

STRESS LANE	INJECTION	EXPECTED EVIDENCE
Authority	Role absent, revoked, conflicted or bypassed	Wait/refuse/escalate + receipt
Dependency	Source, service, package, model or key unavailable	Degraded state + manifest + recovery
Integrity	Alter, delete, reorder, substitute	Verification failure + quarantine
Operation	Load, outage, clock, storage or update fault	Alert + safe state + restore
Target	Timeout, duplicate, rejection or conflicting state	Idempotency + reconciliation
Exit	Supplier access removed or contract ends	Export + local verification + continuity result

Select applicable lanes and add topic-specific consequences before the run.

STOP RULE

If a critical assumption fails without a detectable safe state and accountable owner, stop the proof or production lane until the boundary is repaired.

CHAPTER 02 • EXPLAIN

Binary and integer-oriented computation in plain terms

bit operations, representations, numeric profiles, and what “CPU-first” does and does not mean.

This chapter owns a specific part of the reader’s decision: bit operations, representations, numeric profiles, and what “CPU-first” does and does not mean.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to “Binary and integer-oriented computation in plain terms”, the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Bit operations

Define the exact proposition and current authoritative source.

LENS 2

Representations

Name the responsible person, system, dependency, and version.

LENS 3

Numeric profiles

Specify the evidence and how an independent reviewer checks it.

LENS 4

And what “CPU-first” does

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 02 • APPLY

Turn “Binary and integer-oriented computation in plain terms” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, bit operations, representations, numeric profiles, and what “CPU-first” does and does not mean., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant delivered mechanism, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for binary and integer-oriented computation in plain terms; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 03 • EXPLAIN

Core and Kera beneath the workload

operation graph, compiler/runtime, formats, backends, and hardware mapping.

This chapter owns a specific part of the reader's decision: operation graph, compiler/runtime, formats, backends, and hardware mapping.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Core and Kera beneath the workload", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Operation graph

Define the exact proposition and current authoritative source.

LENS 2

Compiler/runtime

Name the responsible person, system, dependency, and version.

LENS 3

Formats

Specify the evidence and how an independent reviewer checks it.

LENS 4

Backends

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 03 • APPLY

Turn “Core and Kera beneath the workload” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, operation graph, compiler/runtime, formats, backends, and hardware mapping., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant customer configuration, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for core and kera beneath the workload; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 04 • EXPLAIN

Sparse cognition and exact work

activated experts, retrieval, solvers, and why not all work wakes the same compute.

This chapter owns a specific part of the reader's decision: activated experts, retrieval, solvers, and why not all work wakes the same compute.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Sparse cognition and exact work", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Activated experts

Define the exact proposition and current authoritative source.

LENS 2

Retrieval

Name the responsible person, system, dependency, and version.

LENS 3

Solvers

Specify the evidence and how an independent reviewer checks it.

LENS 4

And why not all work wakes the same compute

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 04 • APPLY

Turn “Sparse cognition and exact work” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, activated experts, retrieval, solvers, and why not all work wakes the same compute., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant observed operation, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for sparse cognition and exact work; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 04 • CHALLENGE

Stress-test the boundary behind "Sparse cognition and exact work"

A deep-dive topic earns an additional page for counterexamples, adversarial conditions, missing dependencies, role failure, recovery, and evidence limits.

The principal guide risk is: A fast binary kernel or energy-efficient component can be a small part of the full path; data movement, source/model services, preprocessing, storage, evidence, idle capacity, cooling, operations and hardware refresh can change total cost and impact.

For this chapter, remove or alter the strongest assumption in "activated experts, retrieval, solvers, and why not all work wakes the same compute.". Change the source or version, revoke a person or right, replace a delivered dependency, interrupt the selected deployment, introduce incompatible state, withhold a target acknowledgement, tamper with evidence, or make the relied-on contract condition unavailable. Observe whether the boundary refuses, degrades, escalates, reconciles, recovers, or fails silently.

Run the fault from the perspective of infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review. Each owner should see only the authorised information needed to act, understand the current state, and have an explicit safe next step. Restore normal operation, retrieve a pre-change record, and decide whether the chapter proposition still holds, now holds conditionally, requires remediation, or blocks reliance.

STRESS LANE	INJECTION	EXPECTED EVIDENCE
Authority	Role absent, revoked, conflicted or bypassed	Wait/refuse/escalate + receipt
Dependency	Source, service, package, model or key unavailable	Degraded state + manifest + recovery
Integrity	Alter, delete, reorder, substitute	Verification failure + quarantine
Operation	Load, outage, clock, storage or update fault	Alert + safe state + restore
Target	Timeout, duplicate, rejection or conflicting state	Idempotency + reconciliation
Exit	Supplier access removed or contract ends	Export + local verification + continuity result

Select applicable lanes and add topic-specific consequences before the run.

STOP RULE

If a critical assumption fails without a detectable safe state and accountable owner, stop the proof or production lane until the boundary is repaired.

CHAPTER 05 • EXPLAIN

Distributed and spare capacity

Mesh boundaries, ownership, learning/capacity claims, accounting, and security constraints.

This chapter owns a specific part of the reader's decision: mesh boundaries, ownership, learning/capacity claims, accounting, and security constraints.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Distributed and spare capacity", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Mesh boundaries

Define the exact proposition and current authoritative source.

LENS 2

Ownership

Name the responsible person, system, dependency, and version.

LENS 3

Learning/capacity claims

Specify the evidence and how an independent reviewer checks it.

LENS 4

Accounting

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 05 • APPLY

Turn “Distributed and spare capacity” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, mesh boundaries, ownership, learning/capacity claims, accounting, and security constraints., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant contract and assurance, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for distributed and spare capacity; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

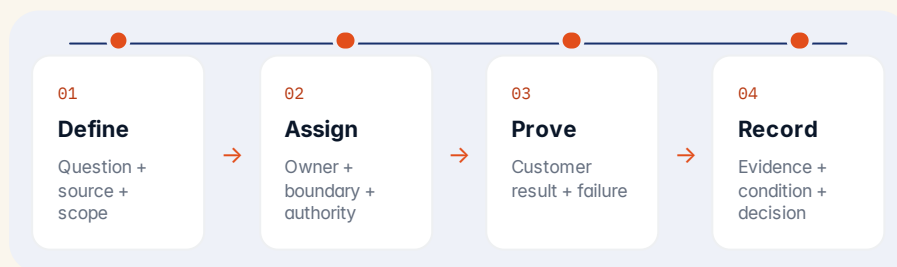
Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

02

CPU-FIRST AI, HARDWARE, ENERGY, AND SUSTAINABILITY

Define the mechanism, controls, and evidence

Turn architecture and policy language into versions, owners, boundaries, tests, records, failures, and explicit limitations.



The same evidence discipline applies to every chapter in the section.

CHAPTER 06 • EXPLAIN

Hardware target matrix

website-supported CPU, SIMD, WASM, embedded, FPGA, and other backend wording with current status.

This chapter owns a specific part of the reader's decision: website-supported CPU, SIMD, WASM, embedded, FPGA, and other backend wording with current status.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Hardware target matrix", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Website-supported CPU

Define the exact proposition and current authoritative source.

LENS 2

SIMD

Name the responsible person, system, dependency, and version.

LENS 3

WASM

Specify the evidence and how an independent reviewer checks it.

LENS 4

Embedded

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 06 • APPLY

Turn “Hardware target matrix” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, website-supported CPU, SIMD, WASM, embedded, FPGA, and other backend wording with current status., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant source and version, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for hardware target matrix; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 07 • EXPLAIN

Benchmark anatomy

hardware, corpus, operation, warm/cold path, throughput, latency, recall/accuracy, build, and non-SLA caveat.

This chapter owns a specific part of the reader's decision: hardware, corpus, operation, warm/cold path, throughput, latency, recall/accuracy, build, and non-SLA caveat.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Benchmark anatomy", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Hardware

Define the exact proposition and current authoritative source.

LENS 2

Corpus

Name the responsible person, system, dependency, and version.

LENS 3

Operation

Specify the evidence and how an independent reviewer checks it.

LENS 4

Warm/cold path

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 07 • APPLY

Turn “Benchmark anatomy” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, hardware, corpus, operation, warm/cold path, throughput, latency, recall/accuracy, build, and non-SLA caveat., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant delivered mechanism, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for benchmark anatomy; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 07 • CHALLENGE

Stress-test the boundary behind "Benchmark anatomy"

A deep-dive topic earns an additional page for counterexamples, adversarial conditions, missing dependencies, role failure, recovery, and evidence limits.

The principal guide risk is: A fast binary kernel or energy-efficient component can be a small part of the full path; data movement, source/model services, preprocessing, storage, evidence, idle capacity, cooling, operations and hardware refresh can change total cost and impact.

For this chapter, remove or alter the strongest assumption in "hardware, corpus, operation, warm/cold path, throughput, latency, recall/accuracy, build, and non-SLA caveat.". Change the source or version, revoke a person or right, replace a delivered dependency, interrupt the selected deployment, introduce incompatible state, withhold a target acknowledgement, tamper with evidence, or make the relied-on contract condition unavailable. Observe whether the boundary refuses, degrades, escalates, reconciles, recovers, or fails silently.

Run the fault from the perspective of infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review. Each owner should see only the authorised information needed to act, understand the current state, and have an explicit safe next step. Restore normal operation, retrieve a pre-change record, and decide whether the chapter proposition still holds, now holds conditionally, requires remediation, or blocks reliance.

STRESS LANE	INJECTION	EXPECTED EVIDENCE
Authority	Role absent, revoked, conflicted or bypassed	Wait/refuse/escalate + receipt
Dependency	Source, service, package, model or key unavailable	Degraded state + manifest + recovery
Integrity	Alter, delete, reorder, substitute	Verification failure + quarantine
Operation	Load, outage, clock, storage or update fault	Alert + safe state + restore
Target	Timeout, duplicate, rejection or conflicting state	Idempotency + reconciliation
Exit	Supplier access removed or contract ends	Export + local verification + continuity result

Select applicable lanes and add topic-specific consequences before the run.

STOP RULE

If a critical assumption fails without a detectable safe state and accountable owner, stop the proof or production lane until the boundary is repaired.

CHAPTER 08 • EXPLAIN

Energy claim anatomy

baseline, workload equivalence, measurement boundary, utilisation, cooling/PUE assumptions where stated, and missing information.

This chapter owns a specific part of the reader's decision: baseline, workload equivalence, measurement boundary, utilisation, cooling/PUE assumptions where stated, and missing information.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Energy claim anatomy", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Baseline

Define the exact proposition and current authoritative source.

LENS 2

Workload equivalence

Name the responsible person, system, dependency, and version.

LENS 3

Measurement boundary

Specify the evidence and how an independent reviewer checks it.

LENS 4

Utilisation

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 08 • APPLY

Turn “Energy claim anatomy” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, baseline, workload equivalence, measurement boundary, utilisation, cooling/PUE assumptions where stated, and missing information., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant customer configuration, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for energy claim anatomy; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 09 • EXPLAIN

Cost model

service metering, owned hardware, utilisation, operations, refresh, data movement, and when a flat licence changes marginal cost.

This chapter owns a specific part of the reader's decision: service metering, owned hardware, utilisation, operations, refresh, data movement, and when a flat licence changes marginal cost.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Cost model", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Service metering

Define the exact proposition and current authoritative source.

LENS 2

Owned hardware

Name the responsible person, system, dependency, and version.

LENS 3

Utilisation

Specify the evidence and how an independent reviewer checks it.

LENS 4

Operations

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 09 • APPLY

Turn “Cost model” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, service metering, owned hardware, utilisation, operations, refresh, data movement, and when a flat licence changes marginal cost., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant observed operation, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for cost model; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 10 • EXPLAIN

Hardware lifetime and locality

reuse, edge/on-prem placement, smaller infrastructure footprint, and constraints.

This chapter owns a specific part of the reader's decision: reuse, edge/on-prem placement, smaller infrastructure footprint, and constraints.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Hardware lifetime and locality", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Reuse

Define the exact proposition and current authoritative source.

LENS 2

Edge/on-prem placement

Name the responsible person, system, dependency, and version.

LENS 3

Smaller infrastructure footprint

Specify the evidence and how an independent reviewer checks it.

LENS 4

And constraints

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 10 • APPLY

Turn “Hardware lifetime and locality” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, reuse, edge/on-prem placement, smaller infrastructure footprint, and constraints., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant contract and assurance, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for hardware lifetime and locality; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 10 • CHALLENGE

Stress-test the boundary behind "Hardware lifetime and locality"

A deep-dive topic earns an additional page for counterexamples, adversarial conditions, missing dependencies, role failure, recovery, and evidence limits.

The principal guide risk is: A fast binary kernel or energy-efficient component can be a small part of the full path; data movement, source/model services, preprocessing, storage, evidence, idle capacity, cooling, operations and hardware refresh can change total cost and impact.

For this chapter, remove or alter the strongest assumption in "reuse, edge/on-prem placement, smaller infrastructure footprint, and constraints." Change the source or version, revoke a person or right, replace a delivered dependency, interrupt the selected deployment, introduce incompatible state, withhold a target acknowledgement, tamper with evidence, or make the relied-on contract condition unavailable. Observe whether the boundary refuses, degrades, escalates, reconciles, recovers, or fails silently.

Run the fault from the perspective of infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review. Each owner should see only the authorised information needed to act, understand the current state, and have an explicit safe next step. Restore normal operation, retrieve a pre-change record, and decide whether the chapter proposition still holds, now holds conditionally, requires remediation, or blocks reliance.

STRESS LANE	INJECTION	EXPECTED EVIDENCE
Authority	Role absent, revoked, conflicted or bypassed	Wait/refuse/escalate + receipt
Dependency	Source, service, package, model or key unavailable	Degraded state + manifest + recovery
Integrity	Alter, delete, reorder, substitute	Verification failure + quarantine
Operation	Load, outage, clock, storage or update fault	Alert + safe state + restore
Target	Timeout, duplicate, rejection or conflicting state	Idempotency + reconciliation
Exit	Supplier access removed or contract ends	Export + local verification + continuity result

Select applicable lanes and add topic-specific consequences before the run.

STOP RULE

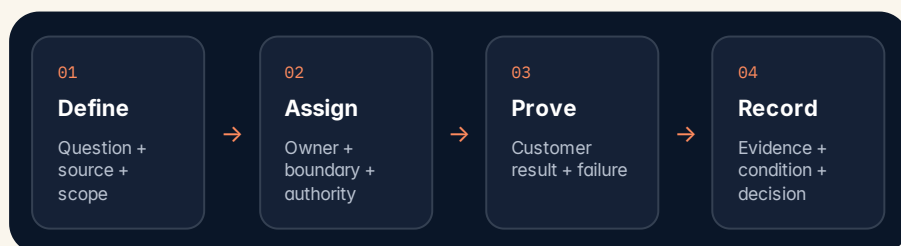
If a critical assumption fails without a detectable safe state and accountable owner, stop the proof or production lane until the boundary is repaired.

03

CPU-FIRST AI, HARDWARE, ENERGY, AND SUSTAINABILITY

Prove adoption, operation, continuity, and exit

Use customer evidence, challenge, recovery, diligence, commercial terms, and a signed decision before expanding reliance.



The same evidence discipline applies to every chapter in the section.

CHAPTER 11 • EXPLAIN

A reproducibility plan

run the site's published suites on customer hardware and compare performance, energy, and output parity.

This chapter owns a specific part of the reader's decision: run the site's published suites on customer hardware and compare performance, energy, and output parity.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "A reproducibility plan", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Run the site's published suites on customer hardware

Define the exact proposition and current authoritative source.

LENS 2

Compare performance

Name the responsible person, system, dependency, and version.

LENS 3

Energy

Specify the evidence and how an independent reviewer checks it.

LENS 4

And output parity

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 11 • APPLY

Turn “A reproducibility plan” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, run the site's published suites on customer hardware and compare performance, energy, and output parity., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant source and version, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for a reproducibility plan; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 12 • EXPLAIN

Decision worksheet

workloads, latency, scale, precision, integrations, hardware, deployment, acceptance figures, and total-cost boundary.

This chapter owns a specific part of the reader's decision: workloads, latency, scale, precision, integrations, hardware, deployment, acceptance figures, and total-cost boundary.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Decision worksheet", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

Workloads

Define the exact proposition and current authoritative source.

LENS 2

Latency

Name the responsible person, system, dependency, and version.

LENS 3

Scale

Specify the evidence and how an independent reviewer checks it.

LENS 4

Precision

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 12 • APPLY

Turn “Decision worksheet” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, workloads, latency, scale, precision, integrations, hardware, deployment, acceptance figures, and total-cost boundary., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant delivered mechanism, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for decision worksheet; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 13 • EXPLAIN

Limits and honest comparisons

no universal workload equivalence, no benchmark-to-SLA leap, and no claim beyond current website evidence.

This chapter owns a specific part of the reader's decision: no universal workload equivalence, no benchmark-to-SLA leap, and no claim beyond current website evidence.

In this guide, CPU-first describes an intended compute path built around binary and integer-oriented operations, sparse expert activation, explicit compilation and supported backends. It does not mean every optional outside model disappears, every workload is equivalent, or a component benchmark predicts a complete production service. Applied to "Limits and honest comparisons", the practical task is to name the current state, the desired state, the owner of the transition, the material input and dependency, the evidence a reviewer can inspect, and the condition that would make the claim false or the design unsuitable.

The governing boundary is this: The sustainability website description publishes strong measured and modelled comparisons, including a 96% energy-reduction framing and a 366W server figure under stated conditions. Preserve hardware, utilisation, workload-equivalence, grid, cooling and arithmetic assumptions; reproduce material claims before using them in cost or ESG decisions. Keep website publication, delivered software, customer configuration, operating result, independent assurance and written contract as separate evidence layers. A strong statement at one layer must not silently become a guarantee at another.

LENS 1

No universal workload equivalence

Define the exact proposition and current authoritative source.

LENS 2

No benchmark-to-SLA leap

Name the responsible person, system, dependency, and version.

LENS 3

And no claim beyond current website evidence

Specify the evidence and how an independent reviewer checks it.

LENS 4

Scope and limits

Record scope, failure, unresolved question, and safe next action.

OWNED ARTEFACT

This chapter contributes to a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.

CHAPTER 13 • APPLY

Turn “Limits and honest comparisons” into decision evidence

Move from explanation to a record an accountable owner can use, challenge, update, and carry into proof, procurement, operation, or exit.

Begin with the chapter proposition, no universal workload equivalence, no benchmark-to-SLA leap, and no claim beyond current website evidence., and write it as one or more falsifiable questions. For each question, define scope and time, authoritative website source or customer source, delivered artefact, relevant customer configuration, named owner, comparison or acceptance rule, raw evidence, known limitation, and decision consequence.

Use the guide-wide scenario: an infrastructure team runs the published or documented component suites and a representative end-to-end workload on its CPU estate, measures output parity, latency, throughput, power and utilisation, then compares an equivalent current baseline. Walk it end to end and ask what this chapter changes at request, source, model/tool/rule, human authority, target, evidence, deployment, operation, commercial or exit state. Do not accept a screenshot or summary when the underlying manifest, result, packet, record or contractual term can be inspected.

The proof discipline for this guide is: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results. Preserve negative and unverifiable results. A gap can be remediated, contracted, accepted by the competent owner, or block the decision; it must not disappear behind roadmap language or an average score.

QUESTION	REQUIRED RECORD	OWNER / STOP
What is claimed?	Exact wording, scope, date, source	Guide decision owner
What exists?	Delivered manifest, interface, artefact	Technical/product owner
What is configured?	Customer data, roles, rules, posture	Customer control owner
What happened?	Raw run, target state, packet, metric	Domain/operations owner
What is assured?	Test/report scope, findings, limits	Independent challenger
What is committed?	Offer/agreement, remedy, exit term	Procurement/legal owner

Evidence ladder for limits and honest comparisons; not every question needs every row, but no row may be implied without evidence.

DECISION RECORD

Record pass, conditional pass, gap, fail, or unverifiable; link evidence, owner, remediation, date, residual risk, and effect on determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.

CHAPTER 13 • CHALLENGE

Stress-test the boundary behind “Limits and honest comparisons”

A deep-dive topic earns an additional page for counterexamples, adversarial conditions, missing dependencies, role failure, recovery, and evidence limits.

The principal guide risk is: A fast binary kernel or energy-efficient component can be a small part of the full path; data movement, source/model services, preprocessing, storage, evidence, idle capacity, cooling, operations and hardware refresh can change total cost and impact.

For this chapter, remove or alter the strongest assumption in “no universal workload equivalence, no benchmark-to-SLA leap, and no claim beyond current website evidence.”. Change the source or version, revoke a person or right, replace a delivered dependency, interrupt the selected deployment, introduce incompatible state, withhold a target acknowledgement, tamper with evidence, or make the relied-on contract condition unavailable. Observe whether the boundary refuses, degrades, escalates, reconciles, recovers, or fails silently.

Run the fault from the perspective of infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review. Each owner should see only the authorised information needed to act, understand the current state, and have an explicit safe next step. Restore normal operation, retrieve a pre-change record, and decide whether the chapter proposition still holds, now holds conditionally, requires remediation, or blocks reliance.

STRESS LANE	INJECTION	EXPECTED EVIDENCE
Authority	Role absent, revoked, conflicted or bypassed	Wait/refuse/escalate + receipt
Dependency	Source, service, package, model or key unavailable	Degraded state + manifest + recovery
Integrity	Alter, delete, reorder, substitute	Verification failure + quarantine
Operation	Load, outage, clock, storage or update fault	Alert + safe state + restore
Target	Timeout, duplicate, rejection or conflicting state	Idempotency + reconciliation
Exit	Supplier access removed or contract ends	Export + local verification + continuity result

Select applicable lanes and add topic-specific consequences before the run.

STOP RULE

If a critical assumption fails without a detectable safe state and accountable owner, stop the proof or production lane until the boundary is repaired.

FIND A SUBJECT

Subject index

Page references point to the substantive explanation, worked example, control, boundary, or decision record, not merely to a passing label.

A Acceptance criteria 3, 8, 11, 13, 15, 18, 21, 23, 26, 28, 30, 34-36, 38	A Accountability 8-9, 11, 13, 15-16, 18, 21, 23-24, 26, 28, 30-31, 34, 36, 38-39	A Adoption 32
A Appeal and challenge 3, 8-9, 11, 13, 15-16, 18, 21, 23-24, 26, 28, 30-32, 34, 36, 38-39	A Architecture 3-4, 9, 16, 19, 24, 31, 39	A Assurance 4, 7, 10, 12, 14, 17-18, 20, 22, 25, 27, 29-30, 33, 35, 37
A Authority 4, 6, 8-9, 11, 13, 15-16, 18-19, 21, 23-24, 26, 28, 30-32, 34, 36, 38-39	A Availability 4-5	B Benchmarking 3, 7, 10, 12, 14, 17, 20, 22-25, 27, 29, 33, 35, 37-39
B Boundaries 3, 5-7, 9-10, 12, 14, 16-20, 22, 24-27, 29, 31-33, 35-37, 39	C Compliance 4	C Configuration 3-5, 7, 10, 12-14, 17, 20, 22, 25-27, 29, 33, 35, 37-38
C Continuity 9, 16, 24, 31-32, 39	C Contracts 4, 7, 9-10, 12, 14, 16-18, 20, 22, 24-25, 27, 29-31, 33, 35, 37, 39	C Cost and budgeting 3-4, 7, 9-10, 12, 14, 16-17, 20, 22, 24-25, 27-29, 31, 33, 35-37, 39
C CPU-first execution 3, 6-8, 10-15, 17-23, 25-30, 32-38	D Decision records 4, 8, 11, 13, 15, 18, 21, 23, 26, 28, 30, 34, 36, 38	D Dependencies 7, 9-10, 12, 14, 16-17, 20, 22, 24-25, 27, 29, 31, 33, 35, 37, 39
D Deployment 8-9, 11, 13, 15-16, 18, 21, 23-24, 26, 28, 30-31, 34-36, 38-39	D Due diligence 4, 32	E Energy 3-39
E Evidence 3-4, 6-39	E Exit 8-9, 11, 13, 15-16, 18, 21, 23-24, 26, 28, 30-32, 34, 36, 38-39	F Failure 3, 6-7, 9-10, 12, 14, 16-17, 19-20, 22, 24-25, 27, 29, 31-33, 35, 37, 39
H Hardware 3-39	H Human authority 8, 11, 13, 15, 18, 21, 23, 26, 28, 30, 34, 36, 38	I Inputs 5, 7, 10, 12, 14, 17, 20, 22, 25, 27, 29, 33, 35, 37
K Kera 12-13	L Licensing 4, 27-28	L Limits and exclusions 8-9, 11, 13, 15-16, 18, 21, 23-24, 26, 28, 30-31, 34, 36-39
M Manifests 4, 8-9, 11, 13, 15-16, 18, 21, 23-24, 26, 28, 30-31, 34, 36, 38-39	M Mesh 17-18	M Models and solvers 3-5, 7-18, 20-31, 33-39
N Numeric profiles 5	O Operations 3, 5, 7-18, 20-39	O Outputs 3-5, 7-8, 10-15, 17-18, 20-23, 25-30, 33-38

SUBJECT INDEX CONTINUED

Subject index · O–W

Terms are grouped alphabetically. Consecutive page references are collapsed into ranges.

O Ownership 4, 6–39	P Performance 3–4, 7, 9–10, 12, 14, 16–17, 20, 22, 24–25, 27, 29, 31, 33–35, 37, 39	P Policies 3–4, 19
P Pricing 4	P Procurement 4, 8–9, 11, 13, 15–16, 18, 21, 23–24, 26, 28, 30–31, 34, 36, 38–39	P Proof 4–5, 8–9, 11, 13, 15–16, 18, 21, 23–24, 26, 28, 30–31, 34, 36, 38–39
R Recovery 9, 16, 24, 31–32, 39	R Reproducibility 3–5, 7–8, 10–15, 17–18, 20–23, 25–30, 33–38	R Responsibility 4, 6
R Rights 4	R Risk 3–4, 8–9, 11, 13, 15–16, 18, 21, 23–24, 26, 28, 30–31, 34, 36, 38–39	R Rules 3–4, 8–9, 11, 13, 15–16, 18, 21, 23–24, 26, 28, 30–31, 34, 36, 38–39
S Security 4, 17–18	S Service levels 22–24, 37–39	S Sources 3–4, 6–39
S Sovereignty 4–5, 7–8, 10–15, 17–18, 20–23, 25–30, 33–38	S State 5–18, 20–31, 33–39	T Testing 3–5, 8–9, 11, 13, 15–16, 18, 21, 23–24, 26, 28, 30–31, 34, 36, 38–39
V Verification 3, 9, 16, 24, 31, 39	V Versioning 4, 7–10, 12, 14, 16–17, 20–22, 24–25, 27, 29, 31, 33–35, 37, 39	W Workflows 3

CLOSING DECISION RECORD

Close with one evidence-backed, reversible decision

The guide is complete when its reader can decide whether to proceed, narrow, remediate, contract a condition, hold, reject, or exit, and another person can understand why.

Assemble a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations. State the question, scope and exclusions, scenario or workflow, current baseline, canonical source snapshot, delivered manifest, customer configuration, corpus and method, results and failures, owner decisions, risks, operational and commercial conditions, evidence links, remedies, stop/exit and review triggers.

Run the final discipline: Pin software and hardware, reproduce the published methods exactly, retain raw output, measure wall power and complete-path work under steady/burst/degraded conditions, check correctness and recall, include infrastructure/operations, test optional external-model paths, and publish both favourable and unfavourable results.

Have infrastructure and platform architecture, performance engineering, workload/product owner, finance/FinOps, sustainability/energy, operations, procurement, model/method and independent measurement review sign only the boundaries they own. Record pass, conditional pass, gap, fail or unverifiable per proposition. Preserve dissent and raw evidence. A brochure can organise the decision; it cannot make the evidence true or take responsibility from the people who rely on it.

DECISION RECORD	cpu-first-hardware-energy-decision	
QUESTION	determine whether the website's CPU-first and energy claims are relevant, reproducible, and material for the target workload and hardware estate.	BOUNDED
SOURCES	website copy + delivered material	CAPTURED
PROOF	corpus + failures + recovery	RUN
OWNERS	domain + technical + assurance	SIGNED
CONDITIONS	gaps + remedies + contract	DATED
DECISION	go / narrow / hold / stop / exit	RECORDED
The closing record remains useful after the presentation, release, team, supplier, or contract changes.		

GUIDE OUTPUT

Produce a workload and hardware benchmark dossier with exact operation path, corpus, build, target, configuration, correctness, performance, power/energy method, baseline, cost boundary, raw output and limitations.