

AISE502: AI in Software Engineering II

Lecture 7: The Fit, Formally – Three Cases, the Procedure, the Matrix

Script: Part IV, Sections 30–32, 34; Section 37 (introduction)

Agenda

1. Recap: the match, seen once at small scale – and run by you last week
2. Three matches, three stages: **C6** (gate), **C1** (veto), **C2** (holistic reading and its alarm)
3. The procedure in general – and what a weighted sum would have destroyed
4. The matching matrix: cell semantics and the 7×10 grid
5. Reading the matrix as a whole: columns, rows, five support points
6. The measurement contract, introduced: fitness functions and the DORA metrics
7. This week's exercise: **the design-review gate and Deliverable A2**

Recap: where we are

demand → supply → **match** → record → **test**

Done (weeks 1–6): Part I – the framework and the C10 mini-match (L –, MM ++, MS ○); Part II – seven capability profiles $C(p)$, the consolidated table, Maxims 3 and 4, ordinal reading only; Part III opened – the C10 profile and the C1/C2 mirror pair (*weights, not dimensions, define a class*).

Last week's exercise: you ran the three stages on your own platform (knock-out → veto → ordinal reading) and began the ADR – today you will recognise every step.

Today:

- ▶ three cases from Part IV, **one per stage**
- ▶ the general statement and the **seventy-cell grid**
- ▶ the **measurement contract**, introduced

Important Note

A2 is due this week – the architecture dossier (ADR + C4 + measurement contract); the design-review gate closes the design phase; production code only after the gate.

Part IV opens: three matches, three stages

What does the three-stage procedure actually do when it runs? – usually far less than students expect: **most of the work happens before anything is scored.**

- ▶ Every Part III class ends with a **verdict and a promise** – a one-sentence primary and secondary recommendation, and the assurance that Part IV computes it through the three-stage procedure. You have seen one (C10) and the C1/C2 sketch; the other rows follow in weeks 8–9
- ▶ You own both operands: $R(a) = (w_1(a), \dots, w_{12}(a); S(a); K(a))$, $w_i \in \{H, M, L\}$ – from A1 and Part III; $C(p) = (c_1(p), \dots, c_{12}(p); S(p))$, $c_i \in \{++, +, o, -, --\}$ – the seven columns of Part II. You have watched $\text{fit}(a, p)$ run once at small scale (the C10 mini-match); not yet seen: **the machine at full load**
- ▶ Cases first, generalisation after (as Parts II and III worked) – three matches computed end to end, each exposing one stage: **C6** – knock-out screening and shape gate (Stage 1) · **C1** – veto rule with documented mitigations (Stage 2) · **C2** – holistic ordinal reading with its built-in sensitivity alarm (Stage 3). Each lands on exactly the verdict its Part III section states

today

§30 cases

§31 procedure

§32 matrix

§34 reading

§37 contract (intro)

Lecture 10

§33 rationales

§35 hybrids, evolution

§36 eight steps

Lecture 11

§37 in depth

§38 organisation

§39 limits

Case 1 – C6 against all seven: the shape gate

The nightly risk run must finish by 06:00, reproduce to the bit, and cost as little as possible: which of the seven patterns can even apply for the job?

- ▶ The C6 profile (given here as the operand; Part III, Lecture 9): High on **D2** (makespan reading), **D9** (reproducibility reading), **D10**; shape **scheduled batch**; deterministic seeds as a hard constraint
- ▶ Before comparing a single rating, hold $S(\text{C6}) = \text{scheduled batch}$ against the native-shape row $S(p)$ of the capability table (deck 6): **only PF proceeds to Stages 2–3**; the verdict column is identical to the C6 row of the matching matrix

Pattern	Native shape $S(p)$	Stage-1 outcome	Verdict
L	interactive	gate caps at $\circ$; harm clause – no answer to makespan or check-pointing, — against High D2	—
MM	interactive	gate caps at $\circ$: orchestration codebase around monolithic kernels	$\circ$
HX	(host's)	gate caps at $\circ$: ports touch no binding dimension	$\circ$
MS	interactive	gate caps; harm clause – communication cost multiplied against High D10	—
EDA	stream / async	gate caps at $\circ$: job-status glue beside scheduler and DAG	$\circ$
PF	scheduled batch	gate passed – proceeds to Stages 2–3; no veto on $\{D2, D9, D10\}$	++
SL	event-trig., short-lived	gate caps at $\circ$: burst fan-out for communication-light sections only	$\circ$

Case 1 – one survivor, and the didactic point

- ▶ **Stages 2 and 3 – one survivor.** Only PF reaches the veto stage, and no veto fires: against the High set $\{D2, D9, D10\}$ it rates $+$, $+$, $++$ (PF column, deck 6) – throughput from data-parallel frameworks, reproducibility by construction, utilisation-driven cost
- ▶ Its $--$ on D3 sits on a Low weight and is *inert*; the holistic reading ranks a field of one
- ▶ **Result:** $\text{fit}(\mathbf{C6}, \mathbf{PF}) = ++$, **every other pattern at $\circ$ or below** – the C6 row of the matrix, computed almost entirely at Stage 1
- ▶ The didactic point generalises: **run the cheapest test first** – six of seven candidates died before a single rating was weighed
- ▶ The knock-outs of $K(a)$ belong to the same stage and work the same way: $K(\mathbf{C1})$ eliminates any structure that cannot guarantee an ACID booking core, an immutable audit journal, and ten-plus-year retention – *before* scoring, however well it scales
- ▶ The verdict the class's Part III section states (Lecture 9): **pipes-and-filters on HPC/batch infrastructure as primary**, serverless fan-out for bursty, communication-light parallel sections as secondary – with the honest subsystem roles (the $\circ$ cells) stated, not hidden

Case 2 – C1: the gate passes both, the veto rule decides

Two candidates pass the gate, both natively interactive – and one of them cannot commit a transaction across its own internal boundaries: how does the procedure decide core banking?

- ▶ From the deck-6 mirror pair: C1 High on **D4, D5, D6, D7, D9**; $K(C1)$: ACID booking core, BCBS 239 / FINMA auditability; shape interactive with batch edges. Contested pair: **MM against MS** – the two columns side by side
- ▶ **Stage 1.** Both natively interactive → the gate passes both; no hard constraint in $K(C1)$ eliminates either – *a constraint names an obligation, not a pattern*
- ▶ **Stage 2 for MS.** $c_4(MS) = \text{---}$: no ACID transactions across service boundaries; sagas trade atomicity for choreography complexity. D4 is High in C1 → **the veto fires and caps the cell** –
- ▶ The fact is *structural*: a mitigation can only show that living without the property is *survivable*
- ▶ Monzo (deck 5): roughly 2,800 microservices in production banking – under **organisational scale plus extreme technological homogeneity** (one language, one monorepo, central migration automation): a condition most organisations do not meet; the premium is paid in platform staffing with no gain for typical team sizes
- ▶ Cap lifted **only to** $\circ$, the condition recorded in the cell rationale; MS's --- on D8 and D10 sit on Low weights – no further veto fires. **Result:** $\text{fit}(C1, MS) = \circ$

Case 2 – stage 2 for MM, and the stage-3 reading

- ▶ **Stage 2 – veto rule for MM.** $c_4(\text{MM}) = ++$ (cross-module ACID transactions) – no veto
- ▶ But $c_5(\text{MM}) = -$ on High-weight D5 $\rightarrow$ caps the cell at $\circ$ – *unless a documented mitigation exists*
- ▶ It does: **hot-standby replication of whole monolith instances** – the classical banking high-availability tactic, in production at Fineract-class core-banking systems. The cap is lifted.
- ▶ **Stage 3 – holistic reading.** MM now stands at $++$ on D4, $+$ on D6, $+$ on D7, $+$ on D9 – support on every High-weight dimension of the class, with the one structural weakness mitigated
- ▶ **Result:** $\text{fit}(\text{C1}, \text{MM}) = ++$
- ▶ The ranking $\text{MM} \succ \text{MS}$ for the C1 core is **stable under plausible weight variation**: it would flip only if D11 (team scaling) rose to High *and* the Monzo homogeneity condition held – exactly what the C1 cell rationale (Section 33, Lecture 10) records as the escalation condition

Case 2 – two kinds of mitigation, and the C1 verdict

Operational weakness

- ▶ MM's one-process blast radius (D5)
- ▶ *repaired outright* by a standard tactic – hot standby, pod replication
- ▶ The division of labour generalises: the veto rule does the heavy lifting, and the **“documented mitigation” clause is where engineering knowledge – not arithmetic – enters the computation**
- ▶ The rest of the row follows the same mechanics (Section 33, Lecture 10); in particular **HX – a delta discipline, not a competitor – joins MM at ++** by isolating the long-lived booking core from volatile channels and providers

Structural weakness

- ▶ atomicity surrendered at the boundary (MS, D4): no mitigation *restores* ACID across service boundaries – sagas buy coordination with compensating actions, not atomicity
- ▶ can only be *made survivable*, under a condition most organisations do not meet

The verdict the class's Part III section states (Lecture 9): a hexagonal modular monolith for the booking core (MM and HX at ++), EDA at the edges and PF for the batch runs as secondary, and microservices only when organisation size forces D11 to High – the Monzo condition – exactly the C1 row of the matrix.

Case 3 – C2: when scoring cannot separate the survivors

Two finalists carry ++ where it matters and neither dominates: what does the procedure return when scoring cannot separate the survivors?

- ▶ C2 (from the deck-6 mirror pair): the **widest High set in the catalogue** – D1, D3, D5, D7, D9, D11 – and a constraint set that knocks out almost nothing; the discrimination work is done by the *weights*, not the constraints
- ▶ **Stage 1** requires one honest observation about shape: the class core has **two constitutive paths** – the interactive read path that serves the feed, and the asynchronous fan-out path that delivers posts (the five-second delivery scenario, given here from the C2 profile of Part III, is binding for the class)
- ▶ Neither MS (natively interactive) nor EDA (natively stream/async) is shape-foreign to the path it would carry → **the gate passes both**
- ▶ **Stage 2** fires one veto against each, and documented practice lifts both: MS's – on High-weighted D3 (mitigation: edge caching, precomputed timelines); EDA's – on High-weighted D9 (mitigation: schema/contract tests, progressive delivery)
- ▶ Both reach Stage 3 intact

Case 3 – stage 3: the comparison refuses to close

High dimension (C2)	MS	EDA
D1 Read scalability	++	++
D3 Latency & predictability	(mitigated)	+
D5 Availability & isolation	++	++
D7 Evolvability	++	++
D9 Testability & deployability	+	(mitigated)
D11 Team scaling	++	+

- ▶ **Neither dominates:** MS leads where *teams* multiply (D9, D11 – independent deployments), EDA where *consumers* multiply (D3 on the asynchronous path; D7 in its attach-new-consumers reading)
- ▶ The **mandatory sensitivity analysis flips the ordering under entirely plausible variation:** weight D11 the way a several-hundred-team organisation must, and MS wins; frame the feed as what it technically is – an eventually consistent, precomputed product of an event flow – and EDA wins
- ▶ By Stage 3's own rule, that instability is **not noise**: it marks a genuine tradeoff point in the ATAM sense, to be escalated to scenario-based analysis rather than smoothed over

Case 3 – the record refuses the either/or

- ▶ Twitter's timeline architecture is **both patterns at once**: fan-out-on-write *is* publish/subscribe – an event flow whose product, the precomputed timeline, is served by independently scaled services
- ▶ The honest reading of the instability is not “the procedure failed to pick a winner” but “**the class genuinely needs both patterns, placed**” – the bridge to hybrids (Section 35, Lecture 10), where hybrids turn out to be the normal case, not the exception
- ▶ The verdict the class's Part III section states (Lecture 8): an **EDA + microservices hybrid at organisational scale** (MS and EDA at ++), a modular monolith as secondary until that scale is *measured*, not assumed – Mastodon runs the full fan-out mechanics in a Rails monolith – exactly the C2 row

Key Concept

Three cases, three stages, one division of labour. The knock-out screening and shape gate kill most candidates before any scoring (C6: the cheapest test runs first); the veto rule disciplines the High set and prices every mitigation as documented engineering rather than optimism (C1); the holistic ordinal reading ranks the survivors while flagging its own instability as a finding, not an error (C2). *Every one of the seventy cells was produced by exactly this division of labour.*

The formal statement – the deck-3 box, plus one clause

What rule were the three matches following? Stated briefly – every element has already done visible work: $\text{fit}(a, p)$ is the ordinal aggregate of the dimension-wise comparison of $R(a)$ and $C(p)$, same five-step scale, **non-compensatory, three stages**.

Definition: Architecture–application fit $\text{fit}(a, p)$

1. **Knock-out screening and workload-shape gate.** $K(a)$ eliminates before any scoring; $S(p) \neq S(a)$ caps at $\circ$ (subsystem role), $+$ only for a *constitutive* subsystem of a shape-hybrid class, $-/-$ where it would harm the binding scenarios. **Not shown in deck 3 – the gate is read per constitutive path:** a pattern is not shape-foreign to a class one of whose binding scenarios constitutes a path of its native shape (Case 3: C2's fan-out delivery scenario).
2. **Veto rule on High-weight dimensions.** $c_i(p) = --$ on a High dimension caps at $-$; $c_i(p) = -$ caps at $\circ$ – unless a documented mitigation exists (a tactic or hybrid composition with production evidence): the cell says so, the cap is lifted.
3. **Holistic ordinal reading with mandatory sensitivity analysis.** Survivors ranked by support of the High set; clustered Medium conflicts downgrade one step; a *ranking with exclusions*, never “12 % better”; a flip under plausible weight variation marks a tradeoff point (ATAM) $\rightarrow$ scenario-based analysis.

Three stages, decreasing hardness – each with a worked face

The three stages are ordered by **decreasing hardness**, and each now has a worked face:

Stage	What it encodes	Worked face
1 Knock-out and shape gate	facts no merit elsewhere can compensate – a violated BCBS 239 obligation, an interactive pattern asked to carry a scheduled-batch core	Case 1 (C6): this stage running the show, emptying six of the row's seven cells on shape alone
2 Veto rule	Assumption A4: the High weights come from the (H, H) leaves of a utility tree (deck 2), so a structural failure on such a dimension fails precisely the scenarios that define the class – unless engineering practice has produced a documented way around it	Case 2 (C1): both halves of the rule – a mitigation that <i>repairs</i> (MM's hot standby) and one that merely makes <i>survivable under condition</i> (MS's Monzo condition)
3 Holistic ordinal reading	deliberately the softest: produces an ordering, and carries a built-in alarm for its own instability	Case 3 (C2): the alarm fired and returned a hybrid rather than a false winner

Why the fit is not a weighted sum

- ▶ Deck 3 (Part I): $V(p) = \sum_i w_i \cdot v_i(p)$ presupposes cardinal scales, preferential independence, and weights as trade-off rates – all three violated by ordinal profiles (A2); AHP inherits rank reversal
- ▶ **What the three cases add – a demonstration of what the formula would have destroyed:**
 - ▶ **C6:** it would have averaged the shape gate away under good scores elsewhere
 - ▶ **C1:** it would have let MS's missing cross-service ACID be compensated by team scaling
 - ▶ **C2:** it would have manufactured a decimal-point winner exactly where the honest output is a flagged tradeoff point
- ▶ Kept from multi-criteria decision analysis: the *explication discipline* (criteria, weights, assumptions forced into the open); dropped: its arithmetic pretensions – the matrix is an **explication and communication instrument**, not a computation that determines decisions; behind every contested cell stands **ATAM**, and, where money decides, **CBAM** (utility-response curves, return on investment)

Key Concept

The fit computation is **non-compensatory by design**: constraints knock out before anything is scored, structural failures on High-weight dimensions veto unless a documented mitigation exists, and only then does a holistic ordinal ranking follow – with mandatory sensitivity analysis. *A weighted sum over ordinal profiles would be formally illegitimate and would average away exactly the failures that matter most.*