EMPOWER YOUR WORLD · FIELD REPORT

The Governed AI Factory

How I built a 200-asset AI-governance programme on Microsoft Cowork: the architecture, the economics, the craft, and what the platform should build next.

AI didn’t replace the discipline. It raised the price of not having any.

Simon Owen · Microsoft MVP · EmPOWER Your World

v1.0 · June 2026 · empoweryour.world

Contents

About Simon Owen & EmPOWER Your World 3
Executive summary 4
1 · What I built 5
2 · The architecture that made it possible 7
3 · The operating workflows 8
4 · Quality, monitoring and the feedback loop 9
5 · Credit economics and efficiencies 11
6 · The decision frameworks worth stealing 13
7 · The capabilities the platform can’t supply 13
8 · What broke: an honest failure log 14
9 · Opportunities for Microsoft: an annex 15
10 · The reflexive insight 16
11 · The Cowork Maturity Ladder 16
Appendix A · The free Starter Kit (specification) 17
Appendix B · Method and data notes 17
Appendix C · Glossary 18

ABOUT SIMON OWEN & EMPOWER YOUR WORLD



Simon Owen is a Microsoft MVP and the creator of **EmPOWER Your World**, a complete, full-lifecycle AI-governance programme designed to stay alive after delivery: connected, living, opinionated and proven. EYW helps organisations govern AI at the altitude of the risk, from North Star intent through to sustained adoption.

This report, the model library and the Starter Kit are shared in that spirit: to help others learn to work this way, and to show what is possible when you push the platform hard. If the work is useful to you, follow along and join the conversation at **empoweryour.world**.

Adoption you push collapses; adoption people pull compounds. The same is true of quality. Build the system, and the standard sustains itself.

The Governed AI Factory

How I built a 200-asset AI-governance programme on Microsoft Cowork: the architecture, the economics, the craft, and what the platform should build next.

Simon Owen · Microsoft MVP · EmPOWER Your World · empoweryour.world

EXECUTIVE SUMMARY

Most people use AI to make things faster. I used it to build something that didn't exist before, and the hard part was never the making. It was governing the maker.

Over roughly four weeks I used Microsoft Cowork to build **EmPOWER Your World (EYW)**, a complete, connected AI-governance programme. Not a folder of slides, but a system: **34 modules across five tiers, an 11-model framework library, and a full delivery bundle for every topic: over 200 finished assets in all.** The work drew on roughly 200,000 credits of platform usage; 186,570 of those are logged against specific builds at a verified cost of about £1,493. Remarkably, it was built outside a full-time role, in mornings, evenings and weekends across a single month.

This report is the honest account of how it was done: the architecture that made it possible, the workflows that kept it consistent, the numbers that prove the quality, the craft the platform can't supply, and (candidly) everything that broke along the way. It is written for four readers at once: the **practitioner** who wants to work this way, the **leader** deciding whether AI-assisted production can be trusted, the **Microsoft partner** building delivery capability, and the **Microsoft product teams** who can build some of this into the platform itself.

EYW at a glance: the headline numbers

Scale: 200+ finished assets (225 in the score ledger, 207 audited) · 34 modules · 11 frameworks · a 16-type asset taxonomy

Quality: portfolio mean ≈ 86.1 / 100; module families consistently score 87–91 ('Excellent') against an 8-dimension rubric

Accessibility: born-accessible by default, the deck template alone scores ≈ 92 on the accessibility dimension, so every asset inherits WCAG-AA

Machinery: ≈25 custom skills authored · a single source-of-truth governance file · live quality and cost dashboards

Economics: 186,570 credits logged ≈ £1,493 · all-in ≈ £6.63 per finished asset · a typical full module family ≈ £24–£75

The thesis is simple and, I think, important. **The only way to build at this quality and scale was to run the production line itself as a governed system.** My single source-of-truth file became policy-as-code. My asset-audit skill became executable governance. My quality and cost heatmaps became runtime dashboards. EYW's own factory turned out to be a live instance of the Governance-as-Code model it teaches clients to adopt.

AI didn't replace the discipline. It raised the price of not having any.

1 · WHAT I BUILT

EYW is organised on one rule that makes everything else possible: **every asset for a topic shares the same number.** Risk Controls is family 11: the M11 deck, F11 framework, T11 catalogue, L11 learning pack and C11 card all travel together. A 16-letter type system (M, F, T, O, P, L, C, G and more) lets the library grow without ever renumbering.

The programme spans five tiers plus an off-spine intent doorway:

Tier	Scope	Modules	Status
North Star	Intent, risk, data posture, appetite	00 doorway	Live
T0 Foundation	The prerequisite platform layer	M01–M08	Shipped
T1 Core Arc	Risk → controls → evidence	M10–M15	Shipped
T2 Operating Model	Who runs it, with what	M20–M27	Shipped
T3 ROI & Assurance	Money · markets · law	M30–M35	Shipped
T4 Change & Adoption	Transition · sustain (a loop)	M40–M45	Shipped

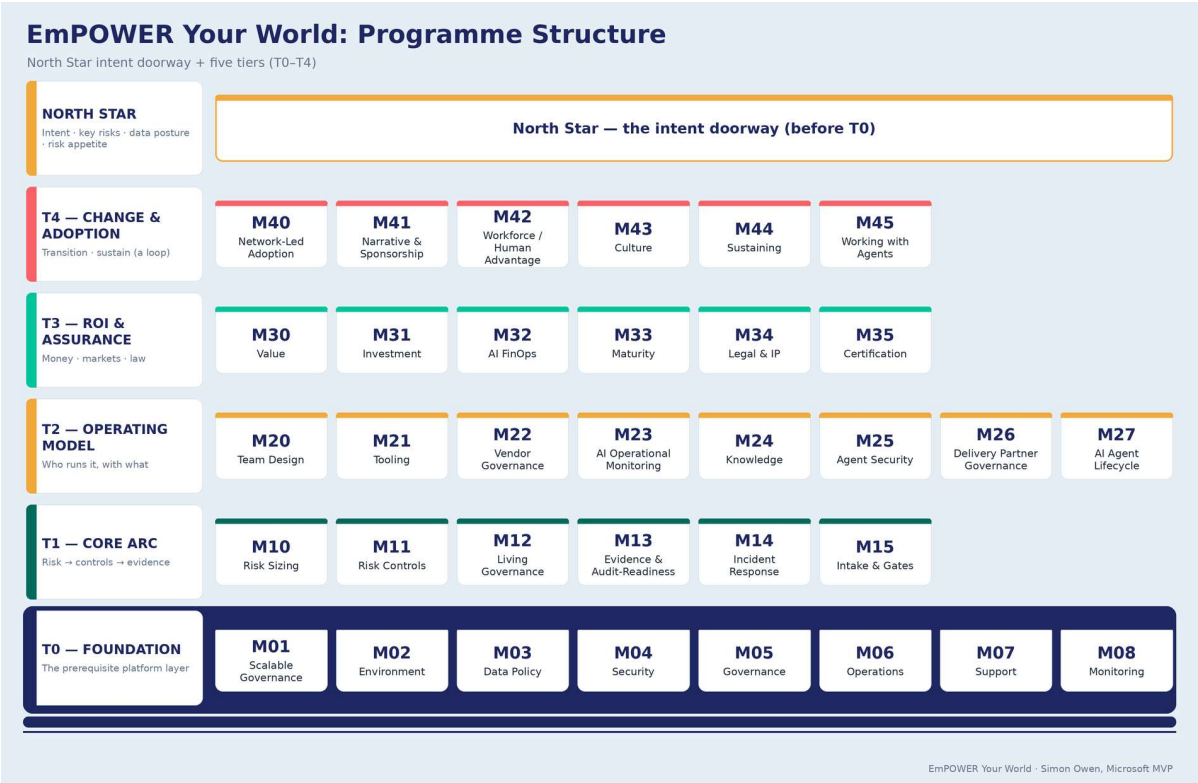


Figure 1. The EmPOWER Your World programme structure: a North Star intent doorway and five tiers, all built on the T0 foundation.

When a module deck is finished it triggers a **standard bundle**, built, versioned and cross-linked as one family:

- **M: the teaching deck · C: a one-page module card · L: a curated learning pack · O: Outcomes + Discovery capture · P: a facilitation playbook**, plus F (framework), T (catalogue) and G (governance-as-code) where the topic warrants them.

Nothing is a one-off. Every asset is connected (one shared number), living (refreshed in place, with refresh logs baked in), opinionated (each module carries a point-of-view line) and accessible from birth.

2 · THE ARCHITECTURE THAT MADE IT POSSIBLE

A prompt makes one good thing once. An architecture makes two hundred good things, on purpose, and tells you which ones slipped.

None of this was the plan. I set out to test Cowork, not to build an operating model. The operating model emerged because the alternative was to re-argue the same standards on every session and watch the quality drift. There was no clever blueprint at the start: it evolved. The efficiencies came once it was clear that a systemic problem needed a systemic fix, and the system that resulted is what made the scale possible.

Four structural pieces carry the whole system.

1 · A single source of truth. One governance file (I call it R00) holds the canonical taxonomy, every module's status and version, the design standards and the working practices. The overview deck and the assistant's own instructions both derive from it. When the structure changes, that file changes first. Everything else follows. It is, quite literally, policy-as-code for the programme.

2 · Skills, references and templates: three different tools. The biggest practical lesson was learning which kind of knowledge to encode where. Getting this wrong wastes credits and produces drift:

Encode as a…	When the knowledge is…	Example in EYW
Skill	A repeatable action with judgement: 'do this, this way, every time'	Voice pass, asset audit, module-pack build
Reference	Facts and standards that must stay consistent but aren't an action	R00, the rubric, the audit profiles
Template	A fixed visual or structural form to build from, not re-derive	The Blueprint deck template + its build script

3 · A locked visual template. Every deck is built from one template and one build script: palette, fonts, geometry and accessible layouts are inherited, never re-invented per deck. This is the difference between 'on brand because someone checked' and 'on brand by construction'.

4 · Live dashboards. A quality ledger scores every asset; a cost ledger logs every build. Two heatmaps render both. I never have to guess whether the library is healthy or where the credits went. I can see it.

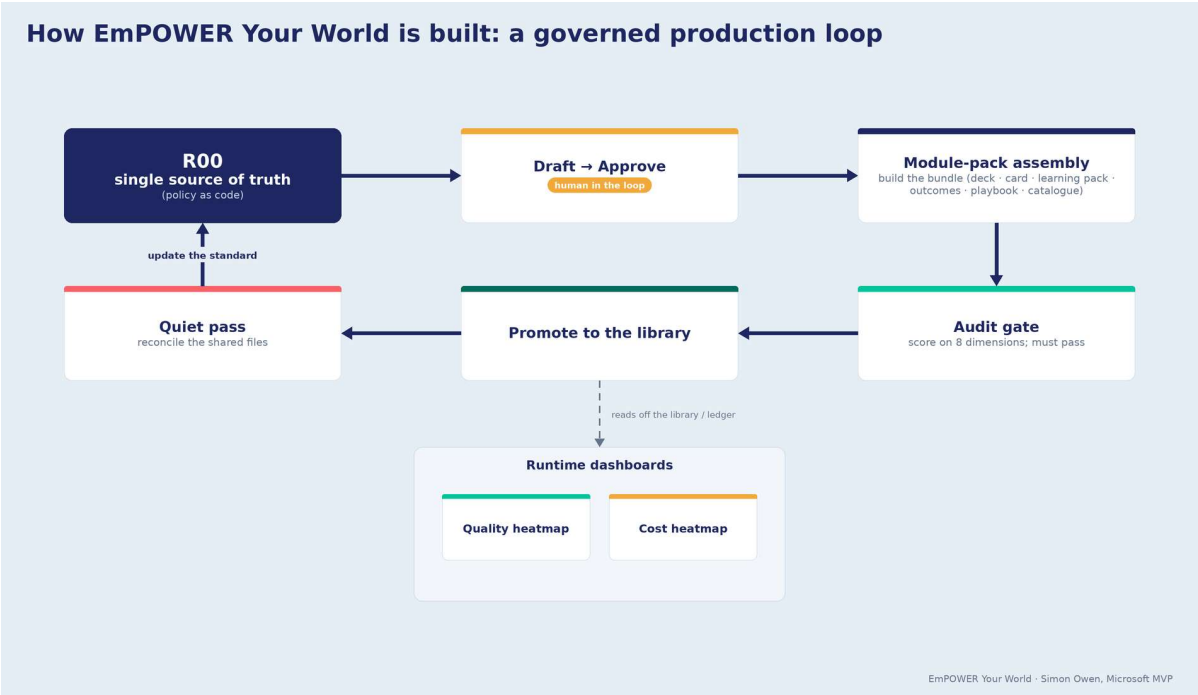


Figure 2. How EYW is built: the governed production loop, from the single source of truth through to the runtime dashboards.

3 · THE OPERATING WORKFLOWS

Architecture is potential; workflows turn it into output. Five recur:

The draft → approve loop. Anything that is genuinely my intellectual property (the model on a deck, the point-of-view line) is drafted and shown to me for approval, never auto-shipped. Mechanical assembly (companions, catalogues, accessibility) runs automatically. Knowing exactly where that line sits is the single most important decision in the whole system.

The module-pack assembly line. One command reads the current standard from the source-of-truth file and builds (or assesses) the full family, assess-first, so an existing asset is reviewed and recommended on, never blindly overwritten.

The quiet pass. Derived, shared artefacts (the overview deck, the heatmaps, the source-of-truth snapshot) are not refreshed on every promotion. They are batched into one consolidated reconciliation run when no other session is active.

Parallel-session etiquette. Several chats can build different modules at once. Topic folders are safe to work in parallel; the shared files are not. They get surgical, re-read-before-edit changes only. Ordinary distributed-systems discipline, arrived at the hard way (see §8).

The audit gate. No family is 'done' until it passes the audit. The gate is part of the build skill, so assets are born voice-passed and accessible rather than patched afterwards.

4 · QUALITY, MONITORING AND THE FEEDBACK LOOP

Most AI content fails for one reason: nobody governs it. People generate, glance and ship. I refused to.

Quality here is measured, not asserted. Every asset is scored out of 100 across eight dimensions:

- Craft & Conformance · Flow & Journey · Relevance & Depth · Connectedness & Reuse
- Voice & POV · Currency & Alignment · Actionability & Grounding · Accessibility

Scores fall into grade bands: **Excellent ≥85, Good 70–84, Needs Work 50–69, Poor <50**, and every score lands in a ledger and a heatmap. The current picture:

What the numbers say

Portfolio mean ≈ 86.1 / 100 across the audited library, the top of the 'Good' band, leaning 'Excellent'.

Module families score 87–91 with strong consistency. This is not one good deck, it is a portfolio holding a line.

Accessibility is structural: the template scores ≈ 92, so WCAG-AA contrast, marked table headers, real slide titles and alt text are inherited, not retro-fitted.

Every asset clears the bar: of 225 scored assets, none falls below 'Good', and 169 (75%) are Excellent. The lowest dimension is Accessibility, which is now built into the template so new assets are born accessible.

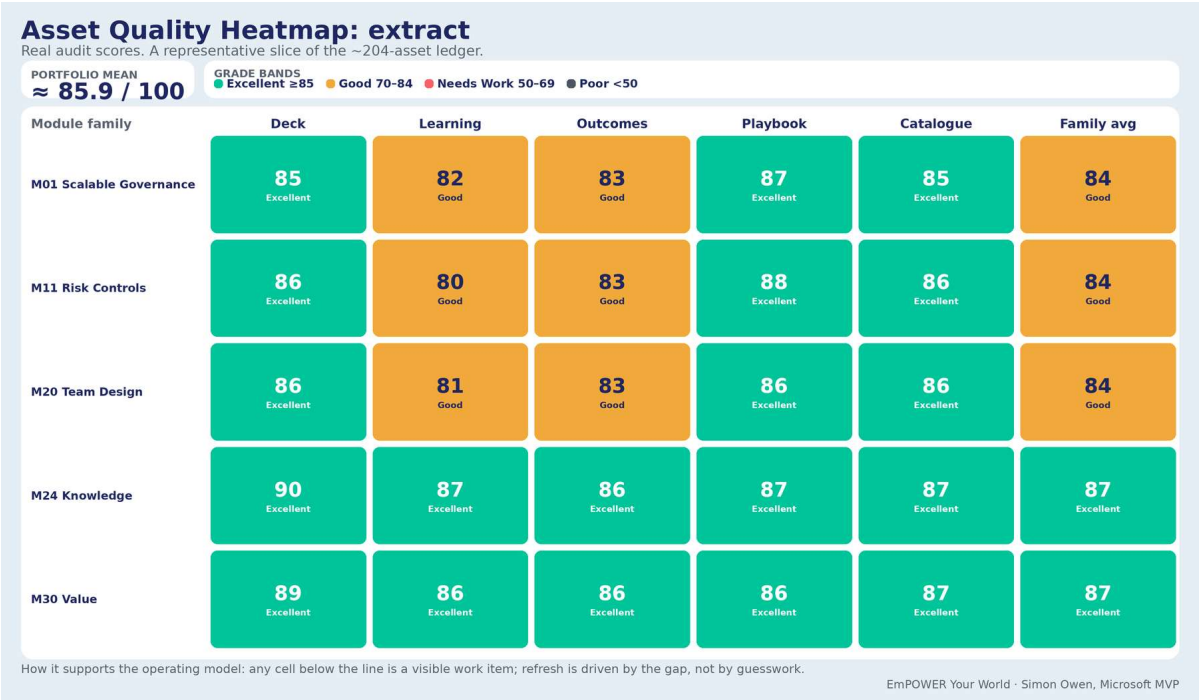


Figure 3. The Asset Quality Heatmap, by dimension: the portfolio profile across the eight-dimension rubric, with the Excellent line at 85. A snapshot of the live dashboard.

The loop is the point. A score below the bar is a work item; the heatmap shows exactly which dimension on which asset, and refresh is driven by the gap, not by guesswork. *It is the same closed loop (instrument, observe, act, learn) that the programme teaches clients to put around their own AI.*

5 · CREDIT ECONOMICS AND EFFICIENCIES

Nobody publishes what this actually costs. So here it is, to the penny, including the money I wasted.

Every build is logged against a module with a credit count and a cost. The verified totals:

The bottom line

186,570 credits logged ≈ £1,493, at a rate of £0.008 per credit (£200 buys 25,000).

All-in ≈ £6.63 per finished asset (829 credits): deck plus companions, audited and accessible.

A typical full module family ≈ £24–£75. The logged 186,570 sits inside a broader ≈200,000 credits of total usage. The difference is exploration, dead-ends and skill-building not logged against a deliverable. The learning cost real money; I am not hiding it.

It is worth asking what the same library would have cost built the conventional way.

What this would have cost the traditional way

Roughly 500 person-days of work to build EYW by hand: a midpoint estimate, with an honest range of about 410 to 610 days. That is two to three person-years, or a small specialist team running for several months.

The estimate is bottom-up: 34 module families at about 13 days each (deck, card, learning pack, outcomes and discovery, playbook, catalogue), plus 11 framework documents, the accessible template and build tooling, the governance architecture, and the overarching assets.

At typical day rates that is somewhere between £320,000 and £540,000 of professional effort. The logged compute cost was £1,493.

The point is not the money. It is the compression: work a team would measure in person-years, delivered in about four weeks by one person holding the whole system in their head.

A sample of real builds. Note the honesty of the data, including estimates and an expensive outlier:

Build	Tier	Credits	Cost	Note
M35 Certification (full 7-asset family)	T3	3,076	£24.61	A complete family for the price of a lunch
M41 Narrative & Sponsorship	T4	2,999	£24.00	Single-session build
M42 Workforce / Human Advantage	T4	4,743	£37.95	Deck + bundle + audit + 7 promotions
M27 AI Agent Lifecycle	T2	5,056	£40.45	Deck, bundle, governance blueprint, audit
Admin (tooling + skill authoring)	Admin	5,193	£41.55	Building the factory, not the product
M43 Culture (2 sessions)	T4	15,787	£126.30	Outlier: ~34% lost to file-sync rework (§8)
~M01 Scalable Governance	T0	12,600	£100.80	Estimated from deck size: flagged, not measured

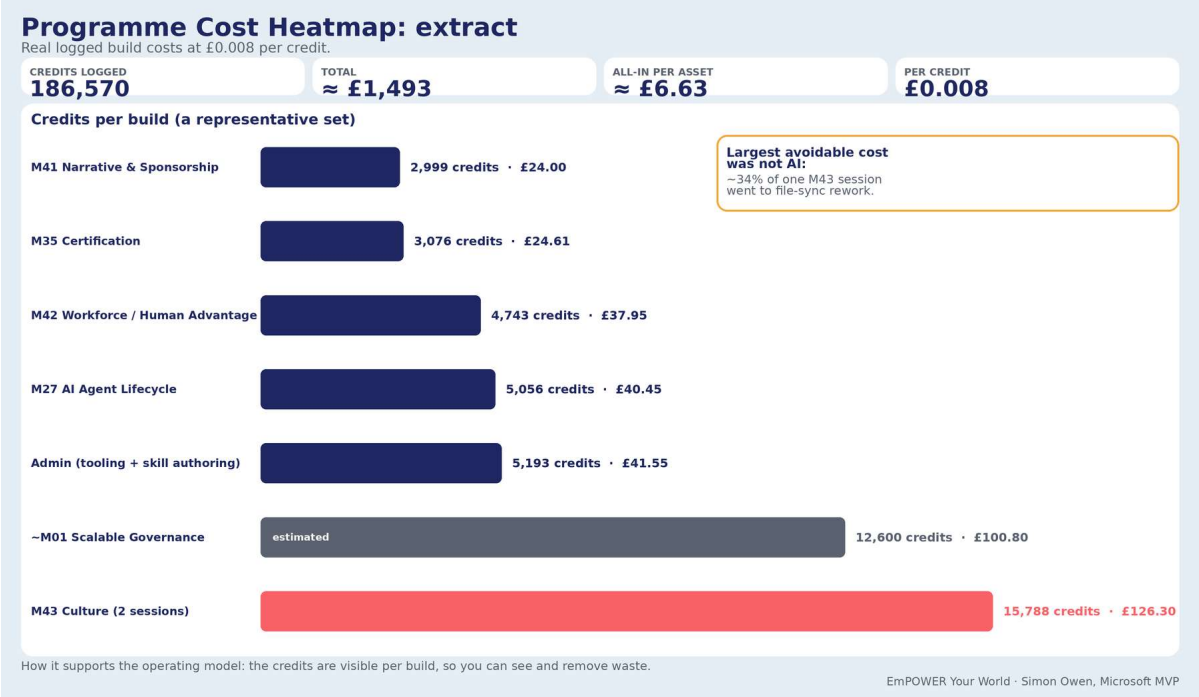


Figure 4. The Programme Cost Heatmap: the headline totals and credits per build, with the M43 Culture outlier and the estimated M01 marked. A snapshot of the live ledger.

Where the credits actually went, and what I changed:

- **Build native, not convertible.** Generating PowerPoint through one library produced files that were schema-valid but tripped PowerPoint's 'Repair' prompt. Switching vector-heavy slides to native generation fixed it for good.
- **Never round-trip through a converter to 'fix' a file.** It silently destroyed shapes. A clean preview hid a broken deck. Hard-won, expensive lesson.
- **Parallelise the mechanical.** Companion assets build concurrently via sub-agents; the human-judgement steps stay sequential.

- **Build the template first, not last.** The locked template is the cure for styling drift (Section 8). Building it on day one rather than part way through would have removed a whole class of rework.
- **The single biggest avoidable cost was not AI at all**: it was file-sync rework (§8), at one point a third of a whole session.

6 · THE DECISION FRAMEWORKS WORTH STEALING

If you take three things from this report, take these.

Skill vs reference vs template (see §2). Encode an action as a skill, a fact as a reference, a form as a template. Conflating them is the commonest and most expensive mistake.

Where to keep the human. Automate assembly, accessibility, formatting, cross-linking and scoring. Keep the human on the model, the opinion, the curation calls and the final approval. The machine earns its speed precisely because the judgement is protected, not bypassed.

How to encode a voice. A voice is a rule set: UK English, smart quotes, no em-dashes, opinionated, and external models always anchored-and-attributed, never worn as your own. Once written down, it can be applied by a skill on every draft, so the work sounds like you at scale, not like the model.

7 · THE CAPABILITIES THE PLATFORM CAN'T SUPPLY

The tool got faster every week. The thing that made it work didn't come from the tool.

The keystone capability, above all the others, is **integrative systems cognition**: the ability to hold a whole multi-dimensional model (technology, process, people, data, context, value and risk) live in your head, and to connect and re-connect it dynamically across the tiers and modules while you build. It is the capability wicked problems demand, because their parts interact and refuse to be solved in isolation.

Two moments make this concrete. Building M21 Tooling (the AI Control Tower), I felt it quietly contradict an earlier module, M20 Team Design, which had scoped *the team* too narrowly; I reached back and repaired the boundary so the two stayed consistent. And building the maturity model, M33 Maturity, I recognised that leadership intent (the North Star doorway, which sits before the foundation) belonged inside it as *dimension zero*, connecting the very front of the journey to its back. Neither was effortful. The contradiction and the connection simply surfaced.

It would be dishonest to call this an AI achievement. The platform supplied speed and reach. Five human capabilities supplied everything that made it good:

- **Orchestration**: designing the system, the bundles and the parallelism, so effort compounds instead of scattering.

- **Domain authority**: the governance expertise that makes every model correct and current, not plausible-sounding.
- **Editorial taste**: knowing what 'good' looks like, and refusing to ship the merely acceptable.
- **Curation judgement**: the calls on what to include, who to cite and where the line of integrity sits.
- **Governance instinct**: the reflex to measure, version, reconcile and prove, rather than generate and hope.

This was not a button press. The shape was not predefined. I pushed back, added perspective, changed the direction, and re-wired the interconnections as we built. EYW came from a brain: the experience, the knowledge, the vision and the curiosity. The platform made that brain faster. It did not replace it.

These are the skills worth investing in for anyone serious about producing with AI. The platform makes them more valuable, not less.

8 · WHAT BROKE: AN HONEST FAILURE LOG

A field report that only lists wins is marketing. Here is where it hurt.

Every item below cost real time or credits, and every one is a workaround I would rather not need.

- **Config-area resets.** My custom skills and the score ledger were wiped from the working area more than once. The library survived in OneDrive, but tooling had to be reconstructed and the ledger rebuilt from the heatmap.
- **OneDrive sync reverting live edits.** The desktop sync engine overwrote freshly-edited files back to their originals before they could be promoted, silently undoing work.
- **Same-name conflicts and hash de-duplication.** Landing three updated files once took 10+ supersede/rename/copy cycles (roughly a third of that session) because of how identical-named files are de-duplicated server-side.
- **No delete.** Files can't be deleted through the available interface, so superseded versions are retired by renaming. The library accumulates 'superseded' copies that must be tidied by hand.
- **The PowerPoint 'Repair' trap.** Schema-valid files that one generator produced still tripped PowerPoint's repair prompt; a clean preview was never proof the file would open.
- **Destructive conversion.** 'Fixing' those files by converting them through another tool silently deleted shapes, worse than the original problem.
- **Parallel-session collisions.** Concurrent chats clobbered each other's edits to shared files until I wrote explicit etiquette rules to serialise them.
- **Styling drift across the library.** Deck to deck it felt consistent and on brand, but jumping back to an earlier tier exposed differences I had not noticed. I built the locked template too late. Setting up a proper template on day one would have saved real rework, stress and credits. It was my single biggest frustration.
- **Sessions running out of context.** Some long sessions exhausted their context window, hung, and became unresponsive, forcing me to start a fresh session that referred back to the broken one. Work in flight had to be picked up again from the library.

9 · OPPORTUNITIES FOR MICROSOFT: AN ANNEX

Read §8 again and a pattern appears: **almost everything I had to build by hand is a missing platform feature, and I have a working prototype of each.** Offered constructively, in priority order:

Opportunity	What I built to cope	Priority
File-sync stability + same-name / dedup handling	Manual supersede-rename cycles; promotion via temp names	Critical
Durable persistence for skills & data	Rebuild scripts; ledger reconstructed from the heatmap	Critical
Native asset & version ledger	A JSON score/version ledger maintained by hand	High
Built-in quality scoring / heatmap surface	An 8-dimension audit skill + two heatmaps	High
Proper delete / supersede semantics	Retire-by-rename convention	High
PowerPoint-safe export guarantee	Native-generation rule for vector-heavy slides	High
Long-session stability + graceful handoff	Splitting work across sessions; referring a new session to the broken one	High
Credit metering by asset / workflow	A cost ledger + per-build cost logging	Medium
Parallel-session locking on shared files	Hand-written ‘etiquette’ rules	Medium
Skills · references · templates as first-class concepts	A naming + folder convention to fake it	Medium

None of these is exotic. Each is the kind of capability a serious creator hits within their first few thousand assets, and building them into the platform would let the next person reach this scale without re-deriving the plumbing.

There is a larger question sitting behind all of these. They are asks at the level of one creator and one piece of work. The same needs scale up: how does an organisation, from a small business to a global enterprise, run AI-assisted production across many people and teams, with shared standards, governed quality and managed cost? That is the enterprise version of everything in this report, and it is a programme in its own right.

10 · THE REFLEXIVE INSIGHT

I governed my AI factory exactly the way I tell clients to govern theirs. That isn't a coincidence. It's the whole point.

The programme's most advanced idea is Governance-as-Code: a governance design should become executable, enforcing artefacts, not just a document. Building EYW, I realised I had already done it to myself:

The model says…	In my factory it is…
Policy, written as code	R00: the source-of-truth file every build reads
Executable governance / a CI gate	The asset-audit skill that blocks 'done' until quality passes
A runtime dashboard	The quality and cost heatmaps
A reconciliation job	The quiet pass that re-syncs the shared artefacts
Human-in-the-loop control	The draft → approve boundary on anything that is real IP

The reference implementation of the method I teach turned out to be the workshop I built it in. That is the strongest possible proof that it works, and the most honest way to sell it.

11 · THE COWORK MATURITY LADDER

To make this reusable, here is the journey as a model others can place themselves on, a new EYW framework, born from this report:

Rung	Looks like	The shift to climb
1 · Ad-hoc	One-off prompts; output copied out by hand; nothing reused	Start naming and saving what works
2 · Repeatable	Saved prompts and a few habits; quality varies by mood	Write the standard down once
3 · Standardised	Templates and a house style; consistent look	Encode the standard as skills
4 · Governed	Skills, a source of truth, scoring and a human gate	Make the dashboards drive the work
5 · Self-sustaining	The system reconciles, measures and improves itself; you steer	Keep raising the bar, not the effort

EYW runs at rung 4, reaching for 5. *The ladder is the real deliverable of this report: most people are on rung 1 and think the answer is a better prompt. The answer is a better system.*

APPENDIX A · THE FREE STARTER KIT (SPECIFICATION)

The most convincing way to share this is to give away the working machinery, not a description of it. The plan (to build as a focused follow-on) is a free, sanitised **Governed AI Production Starter Kit**, offered from empoweryour.world:

- **A generic source-of-truth template**: the R00 pattern with all EYW specifics stripped. The centrepiece.
- **The 8-dimension audit rubric** + a blank quality-heatmap template (spreadsheet and HTML dashboard).
- **A cost / credit ledger template** and the cost-heatmap pattern, so others can see their own economics.
- **A curated starter set of skill patterns**: generalised, with a 'write your own' guide rather than EYW's proprietary skills.
- **A 'run your own governed AI factory' quickstart** that ties them together and points back to this report.

How it works as a lead magnet

Read the story → grab the kit → come to empoweryour.world.

The kit gives away the method while EYW's actual library, voice and domain authority stay the moat. It is genuinely useful on its own, which is exactly why it earns the visit.

APPENDIX B · METHOD AND DATA NOTES

- **Sources.** Figures are drawn from the programme's own records: the canonical structure file (current to 20 June 2026), the cost heatmap (refreshed 19 June 2026) and the quality score ledger.
- **Logged vs total credits.** 186,570 credits are logged against specific builds (≈£1,493). The wider ≈200,000-credit figure is total platform usage including exploration and tooling not tied to a deliverable.
- **Why not every session is logged.** Per-session cost logging only began part way through the work, and it has to be captured live, as a snapshot inside each session, so it cannot be applied in bulk after the fact. A number of modules were scored retrospectively; some earlier sessions were never logged. The logged total is therefore a floor, not a ceiling.
- **Estimated rows.** Some early modules' costs are estimated from deck size (marked '~') rather than measured, and are likely to overstate; they are shown as estimates, not facts.
- **The 'top 1%' claim.** My belief that this is further than most have taken Cowork is stated as a belief, not an audited fact.

- **Calculations.** Derived figures (per-asset cost, credits per asset) were computed from the totals above, not estimated by hand.
- **Person-days estimate.** The traditional-build figure is a transparent bottom-up estimate (families × per-asset days, plus frameworks, template, governance and overarching effort); day-rate costs are illustrative and vary widely.
- **A snapshot, not a fixed figure.** These numbers are a snapshot as of 20 June 2026, taken from the live score and cost ledger. The library is living: as modules are re-aligned to the template, some quality scores and costs will change. Treat the figures as current, not final.

APPENDIX C · GLOSSARY

A short reference for the shorthand and named parts of the programme used across this report.

Source of truth (R00) The single canonical document holding the programme's taxonomy, structure, standards and working practices. Everything else derives from it; in this report it functions as policy-as-code.

Tiers (T0 to T4) The asset roadmap. T0 Foundation is the prerequisite layer; T1 to T4 are the AI-governance overlay built on top.

North Star The intent doorway before T0: ambition, key risks, data posture and risk appetite, set with the leadership team.

Asset types (M, F, T, O, P, L, C, G) The letter codes for each kind of deliverable: M Module deck, F Framework, T Template or catalogue, O Outcomes and Discovery capture, P Playbook, L Learning pack, C Module Card, G Governance-as-Code blueprint.

The bundle The standard family of assets that ship together for one topic, all sharing the same number.

Governance-as-Code Turning a governance design into executable, enforcing artefacts rather than a static document.

Draft-then-approve loop The human-in-the-loop control: the AI drafts, and the author approves anything that is genuine intellectual property before it ships.

Quiet pass A batched reconciliation run that refreshes shared, derived artefacts (the overview, the dashboards, the snapshot) when no other work is in flight.

The eight dimensions The quality rubric every asset is scored against: Craft & Conformance, Flow & Journey, Relevance & Depth, Connectedness & Reuse, Voice & POV, Currency & Alignment, Actionability & Grounding, Accessibility.

Grade bands Excellent 85 and above, Good 70 to 84, Needs Work 50 to 69, Poor below 50.

Born-accessible Built to meet accessibility standards (WCAG-AA contrast, alt text, marked table headers, meaning carried in text not colour alone) at creation, not retro-fitted.

On brand by construction Brand consistency that is guaranteed because every asset is built from one locked template and build system, not because someone checks each asset by hand. The opposite of "on brand because someone checked".

Credit The unit of Microsoft Cowork platform usage; costed here at about £0.008 per credit.

Integrated Systems Cognition The capability of holding a whole multi-dimensional model live in the head and connecting it dynamically across the tiers and modules.

NIST AI RMF The US National Institute of Standards and Technology AI Risk Management Framework.

ISO/IEC 42001 The international management-system standard for artificial intelligence.

EU AI Act The European Union's regulation governing AI systems by risk.