

The MACH Readiness Worksheet

The question is not whether MACH is good. It is whether it fits you, and that turns on capability and scale more than on architecture.

Score your readiness on four axes

MACH fits on capability and scale more than on architecture. Be honest rather than aspirational.

Engineering capability

We have, or will genuinely hire, developers comfortable running integrated services.

1	2	3	4	5
---	---	---	---	---

Rate of change

We change things often, and the current system is what slows us down.

1	2	3	4	5
---	---	---	---	---

Requirement specificity

No single platform fits us well. We keep hitting the walls of all-in-one products.

1	2	3	4	5
---	---	---	---	---

Scale

We operate at a scale where flexibility pays back the added complexity.

1	2	3	4	5
---	---	---	---	---

Reading it: 16 or above, a composable approach likely earns its complexity. 10 to 15, peel off one component and test before committing further. Below 10, a good monolith or SaaS platform will serve you better for far less maintenance.

Total the costs on both sides

Composable is not reliably cheaper. Fill both columns, including the lines people leave out.

Cost line	Monolith / SaaS	Assembled stack
Licences and subscriptions		
Implementation		
Integration build		
Ongoing integration upkeep		
Internal engineering time		
Upgrades		

If you proceed, start with one piece

Almost nobody should rip out the monolith in one go. Peel off one self-contained component, connect it, and learn whether your team can run it before committing further.

Which component would you peel off first, and why?

Search is the usual first candidate: self-contained, obvious pain, clear upgrade.

The real test. The first component teaches you whether your team can run an integrated service in production. That lesson is worth more than the component, and far cheaper to learn on one piece than on twelve.