MODERN INFRASTRUCTURE & IAC · FRAMEWORK #140

Infrastructure State Gravity

Why you can no longer redesign the platform you built

01 — THE START

Infrastructure Starts Flexible

One owner. One consumer. A clean interface.

Nobody reviews a change because nobody else depends on it yet.

Month 1

1 consumer

State Accumulates

By month six: a second consumer, then a fifth.

The interface hasn't changed. What it's checked against has.

Change Costs Rise

By month fourteen: forty consumers.

The module didn't get more important. It got heavier to touch.

> *“The transition happens when changing the platform becomes more expensive than governing it.”*

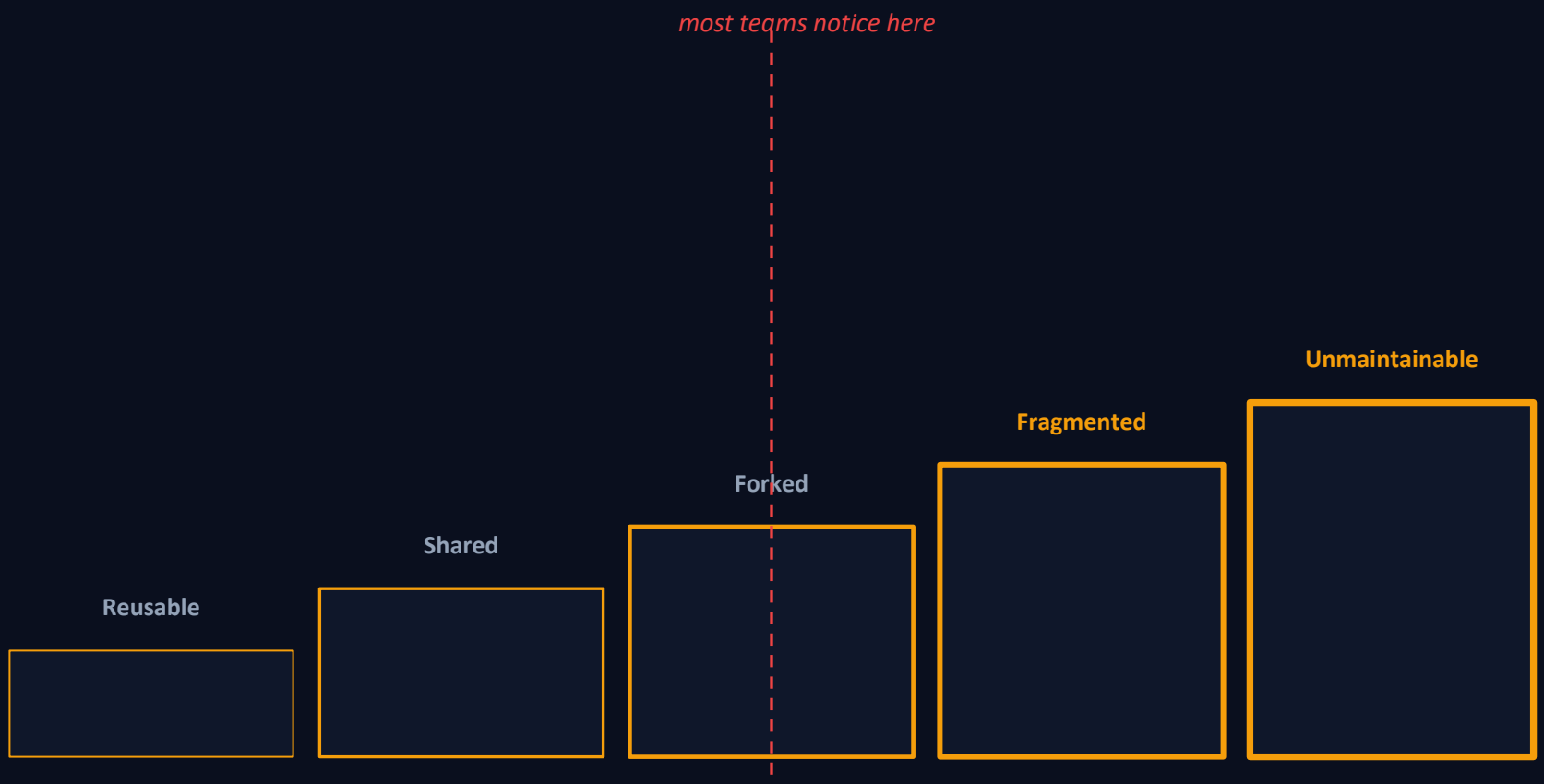
State Gravity Emerges

Not the accumulation of state itself —
the increase in change cost that accumulated state causes.

FRAMEWORK #140

Technical debt is the cost of past shortcuts.
State Gravity exists even when no shortcuts were taken at all.

The Module Lifecycle Curve



Consumers Become Constraints

State gravity increases dependency concentration.

Dependency concentration increases bus-factor risk.

Two frameworks, one accumulating mass:
change cost on one side, people-risk on the other.

07 — THE TAKEAWAY

Infrastructure Becomes a Product Before Anyone Calls It One

Nobody decided to run infrastructure as a product.
The infrastructure decided — the moment redesign got more expensive than governing it.

State Gravity Is Why.

Rack2Cloud.com | Follow The Architect for more