

Why API Governance Fails

And how to shift it left

Alastair Parker
jargon.sh

Where does your organisation spend money on APIs?

We spend millions on API programs.

Where does that money go?

API
Gateway

Runtime
Monitoring

Design ?

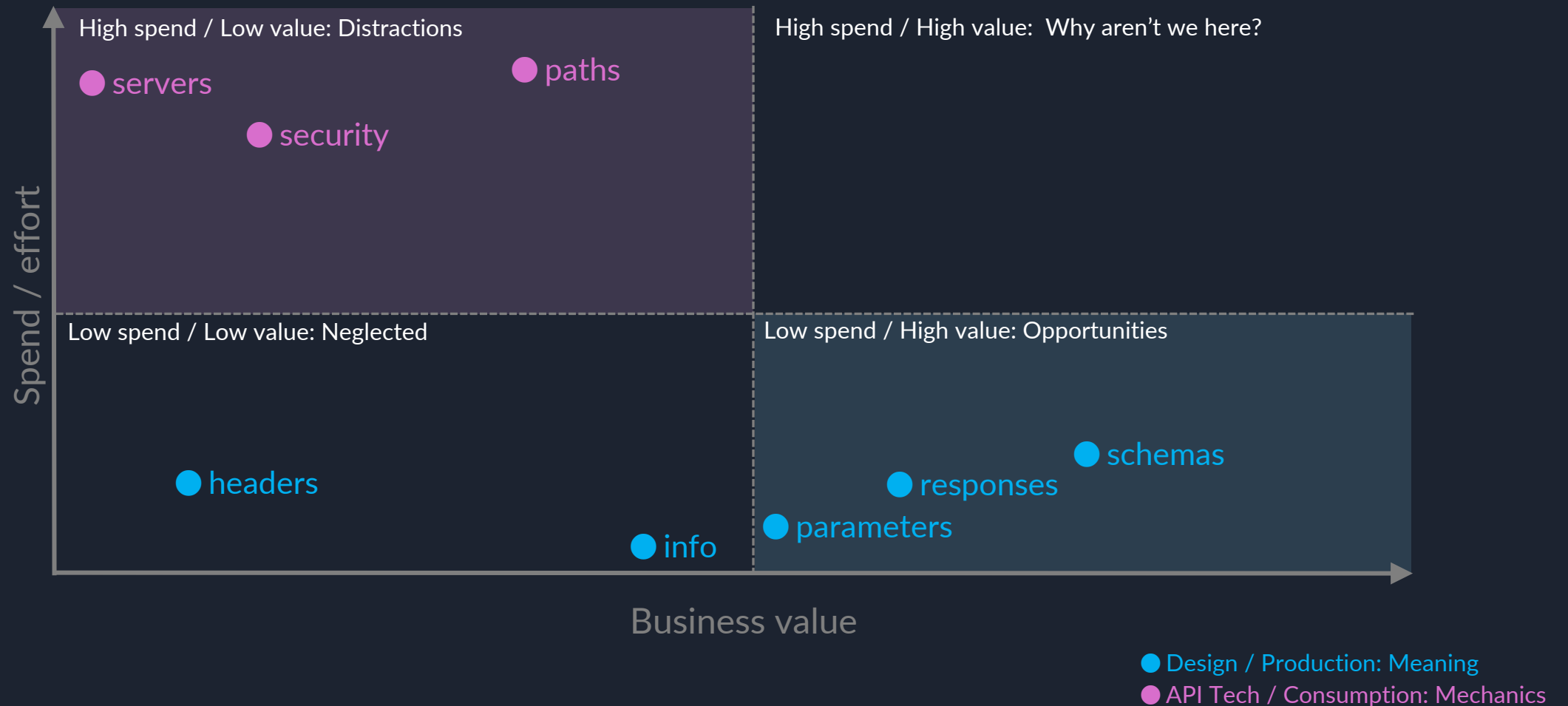


Follow the money / effort

Mapping spend against sections of the API spec

OAS 3.x.x section	Spend / effort	Where governance effort goes	Purpose
info		Linting – check not empty	Risk mitigation
servers		Gateway – deployments, env, config	Risk mitigation
security		Gateway – auth, policy management	Risk mitigation
paths		Gateway – monitoring, DX, runtime	Tech design
components.responses		Linting – check not empty, 400/500, etc	API Design
components.parameters		Linting – check not empty	Tech plumbing
components.schemas		Linting – check naming, casing, etc	API Design
components.headers		Linting – check not empty	Tech plumbing

We spend big on runtime & consumption – but starve production & design.



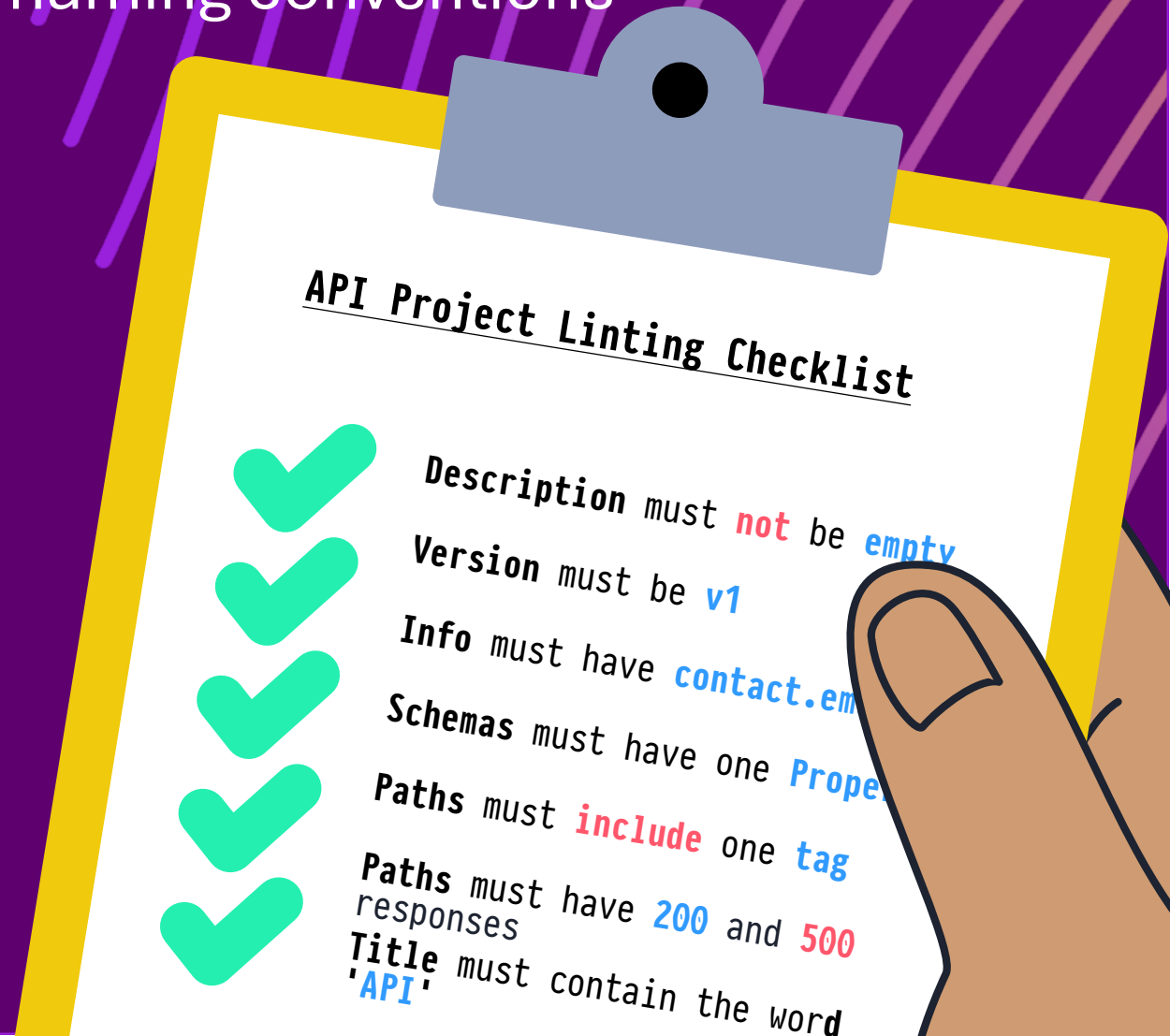
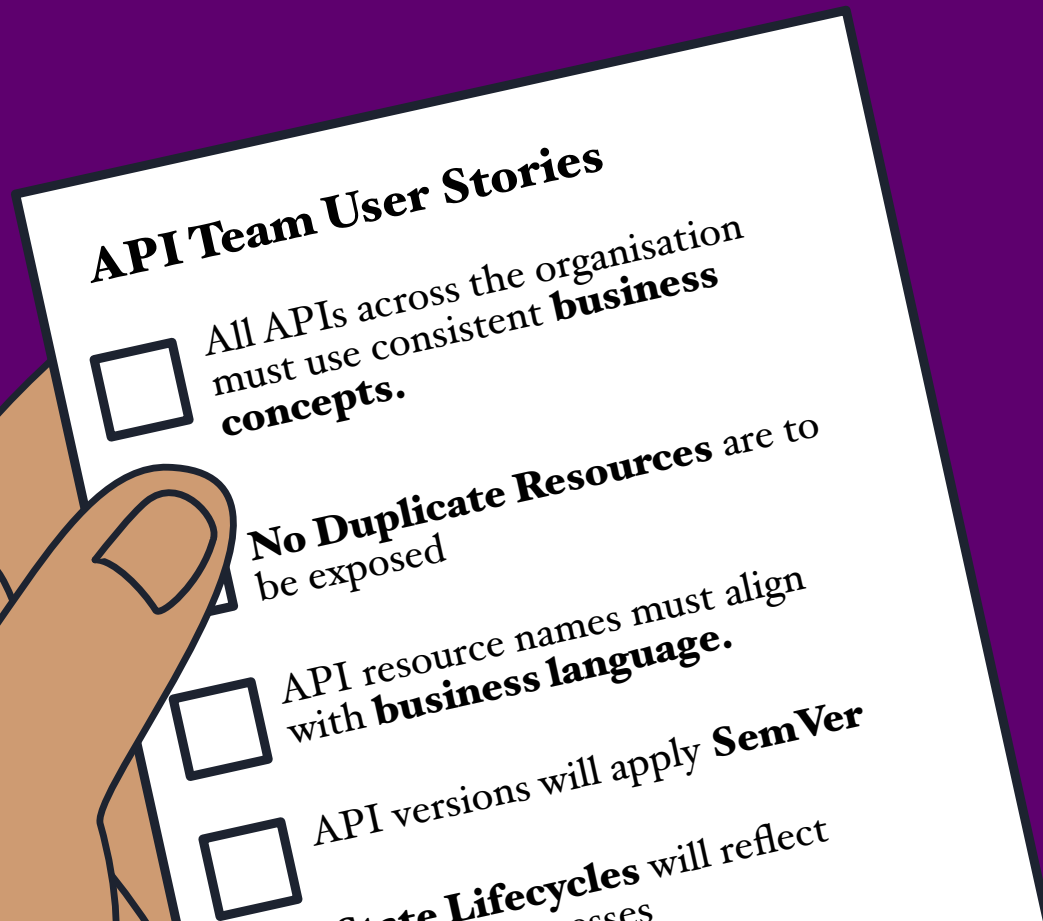
Big tech investments only govern at the end – not during API design



Linting ≠ Governance

APIs don't fail governance because of naming conventions

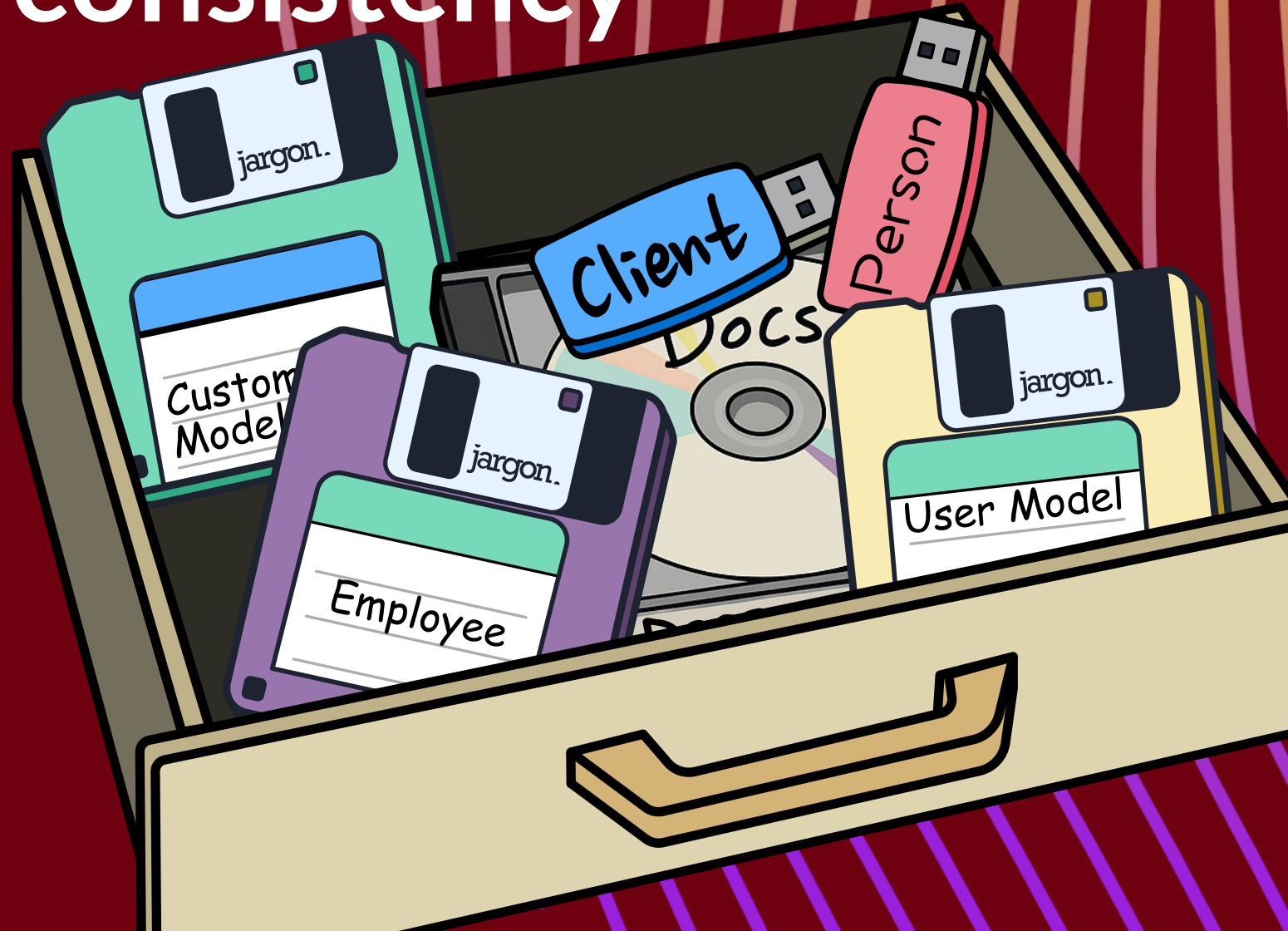
That's not governance – its hygiene



Failed Governance: Bad Outcome #1

No enterprise consistency

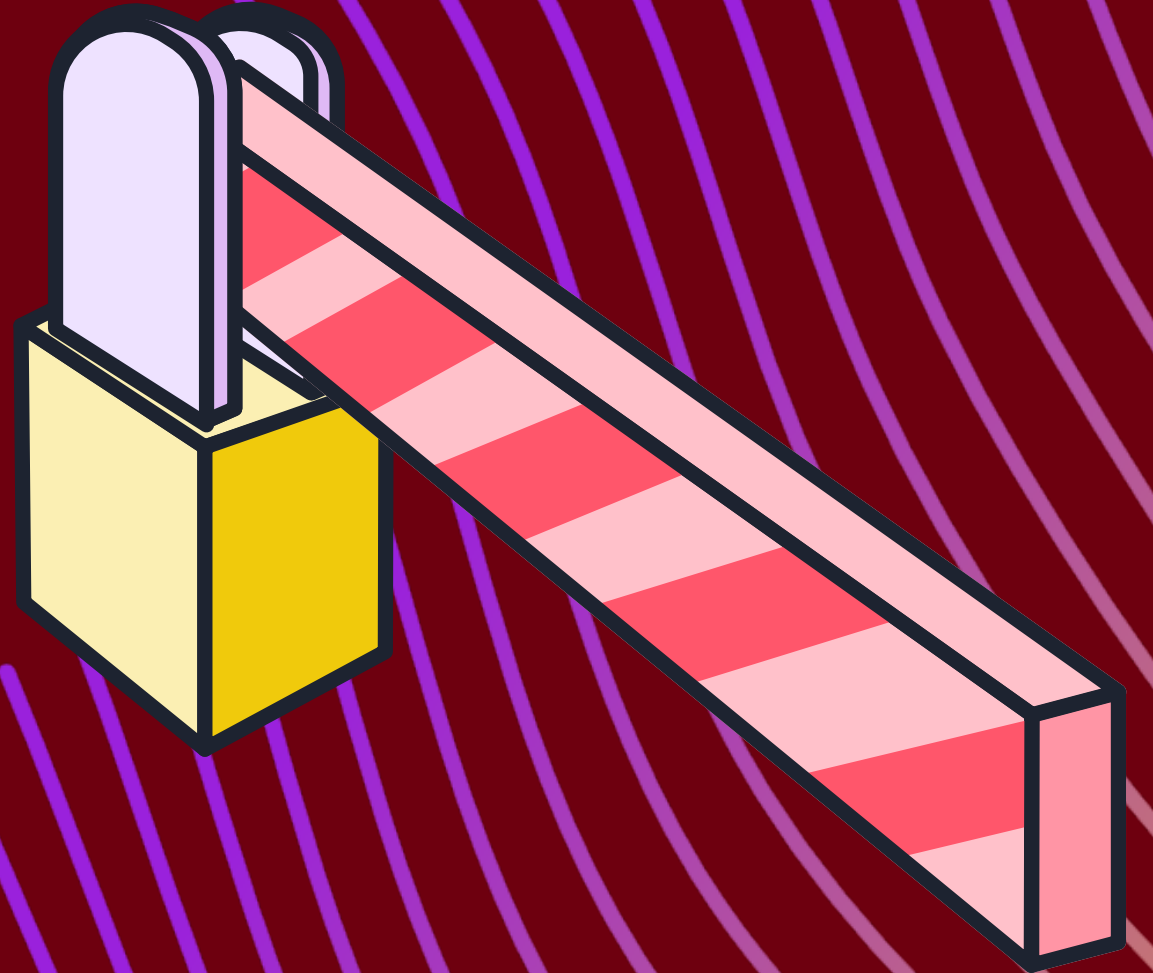
- When there's no governance there's no reuse
- Integration takes longer
- No single view of 'Customer'



Failed Governance: Bad Outcome #2

Governance as a road block

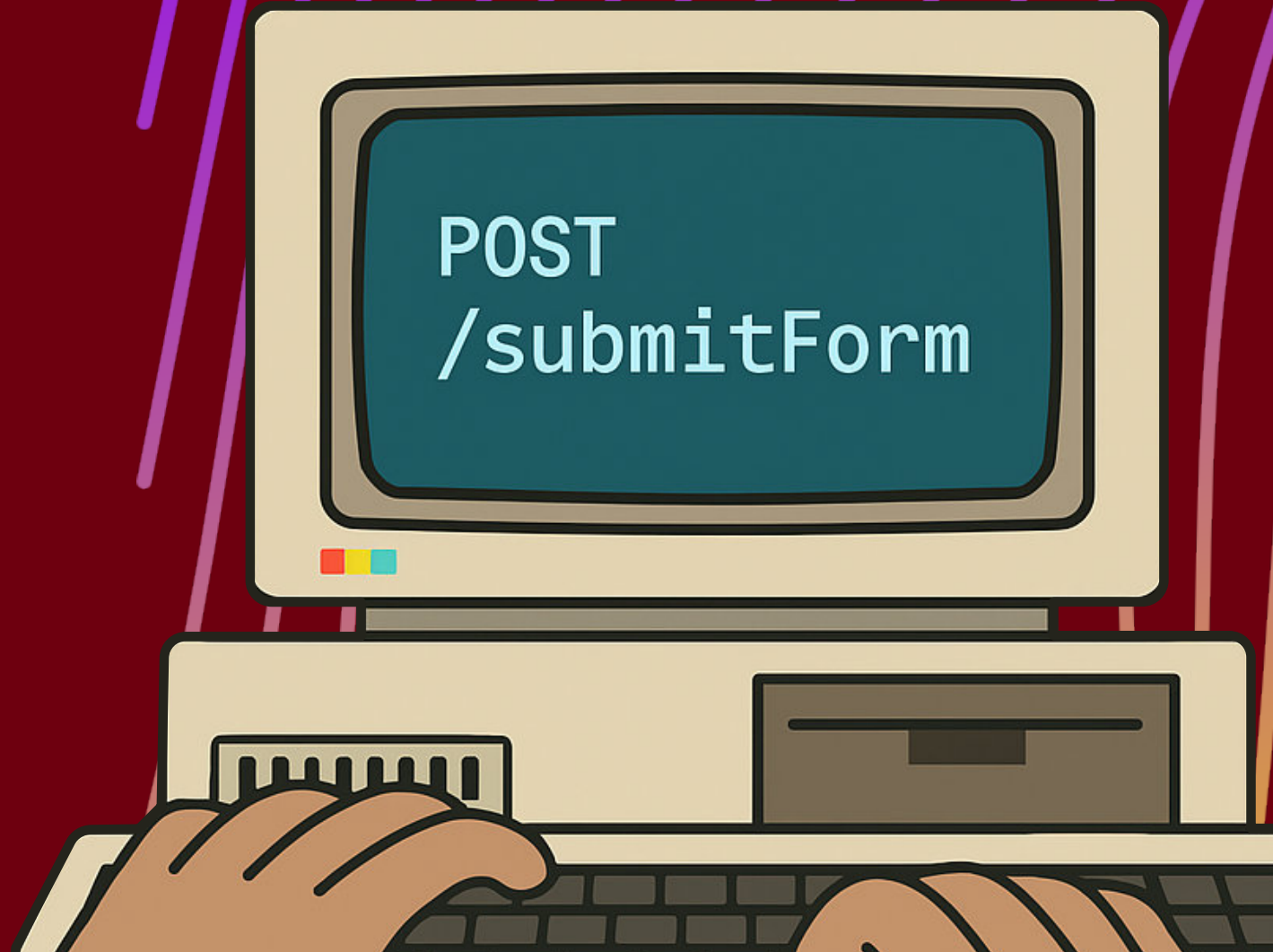
When Governance arrives last,
it can't add any value.



Failed Governance: Bad Outcome #3

Innovation stops & the Enterprise goes backwards

- Data doesn't join up
- Systems don't talk
- Silos start rebuilding
- APIs become 'point-to-point'



Shift Governance Left

Shifting left isn't just about doing things earlier.

It's about pushing quality upstream.
So it can be relied on downstream.



Proper Governance: Good Practice #1

Guardrails, not Gates

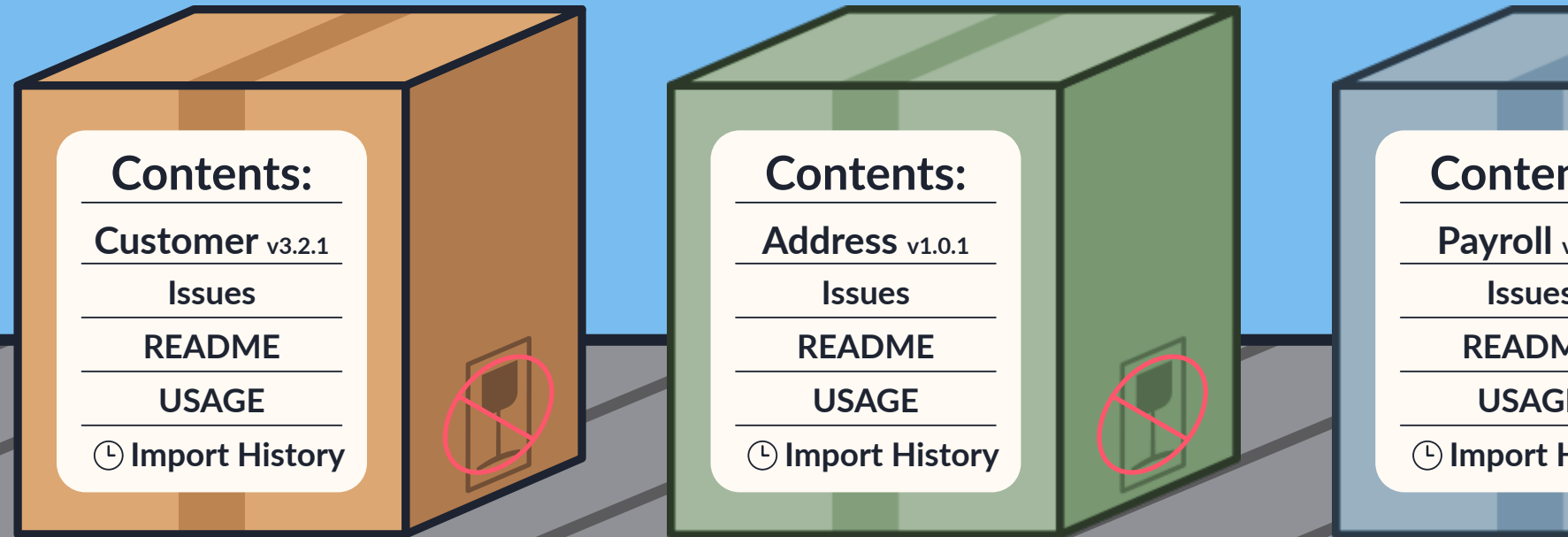
Gentle nudges to keep you safe,
not a gate at the end of the road.



Proper Governance: Good Practice #2

Versioned, Curated Domain Packages

Your Customer Domain should be a complete package, and not just a PDF

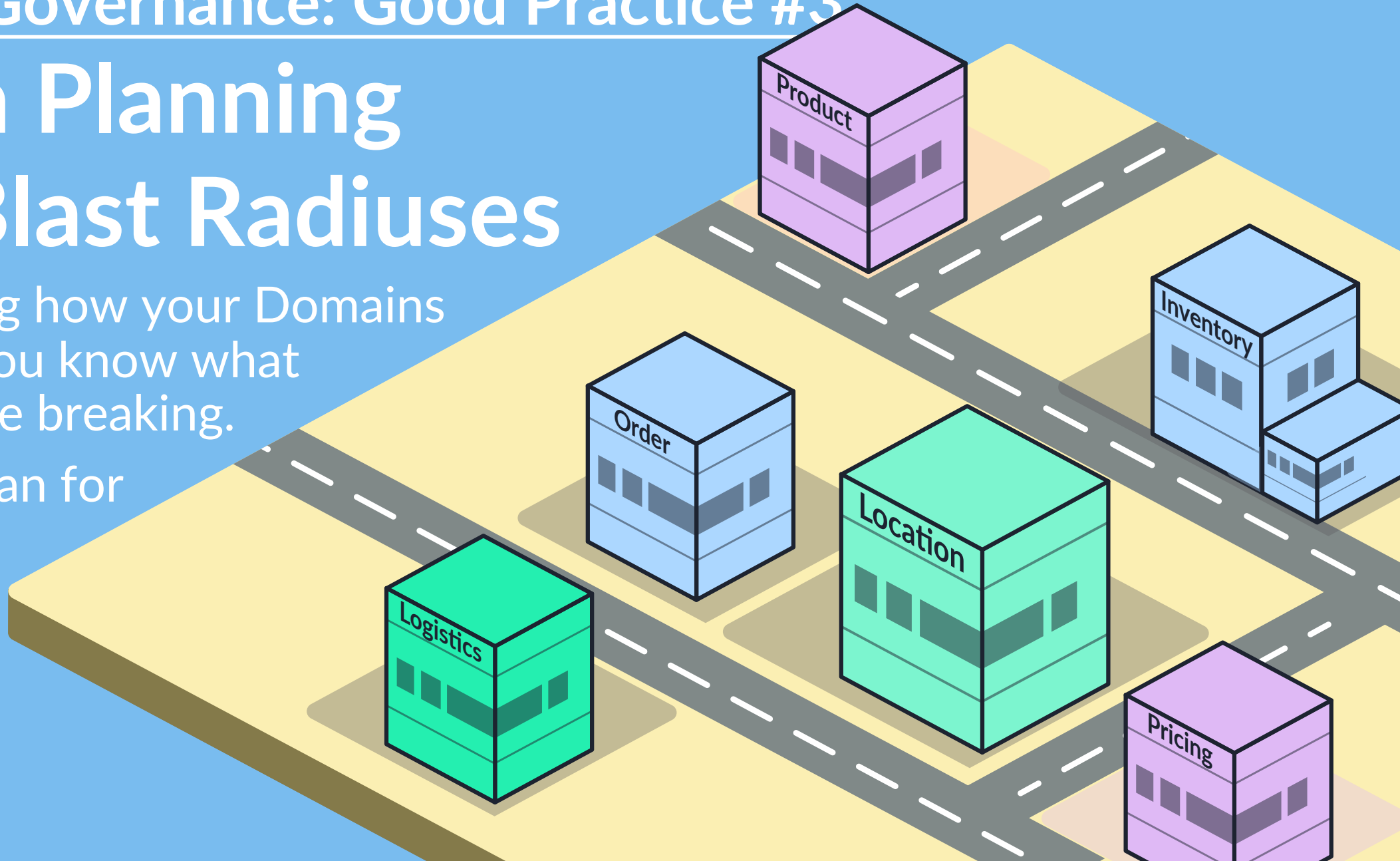


Proper Governance: Good Practice #3

Town Planning and Blast Radiuses

By knowing how your Domains connect, you know what changes are breaking.

And can plan for the fallout.

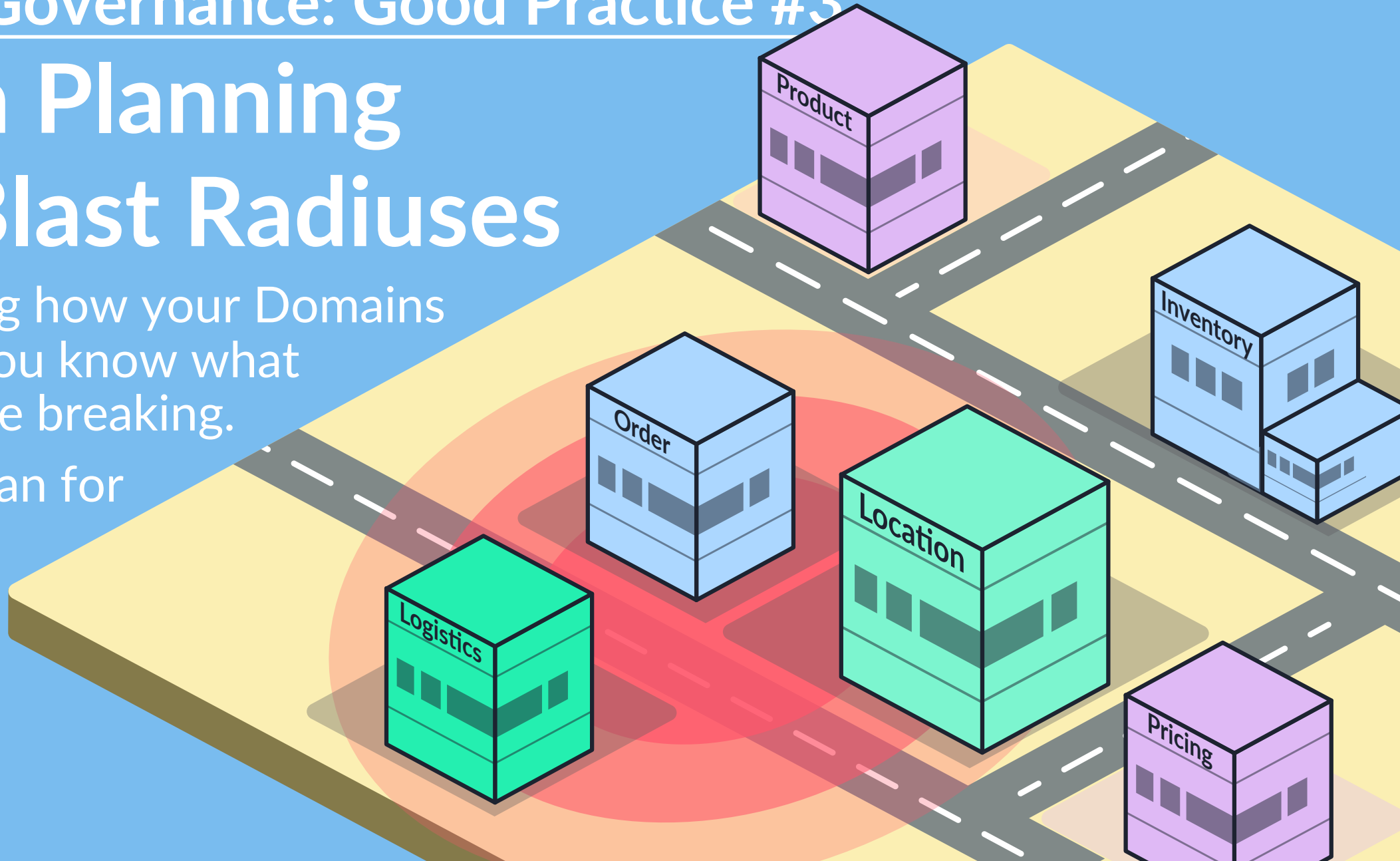


Proper Governance: Good Practice #3

Town Planning and Blast Radiuses

By knowing how your Domains connect, you know what changes are breaking.

And can plan for the fallout.

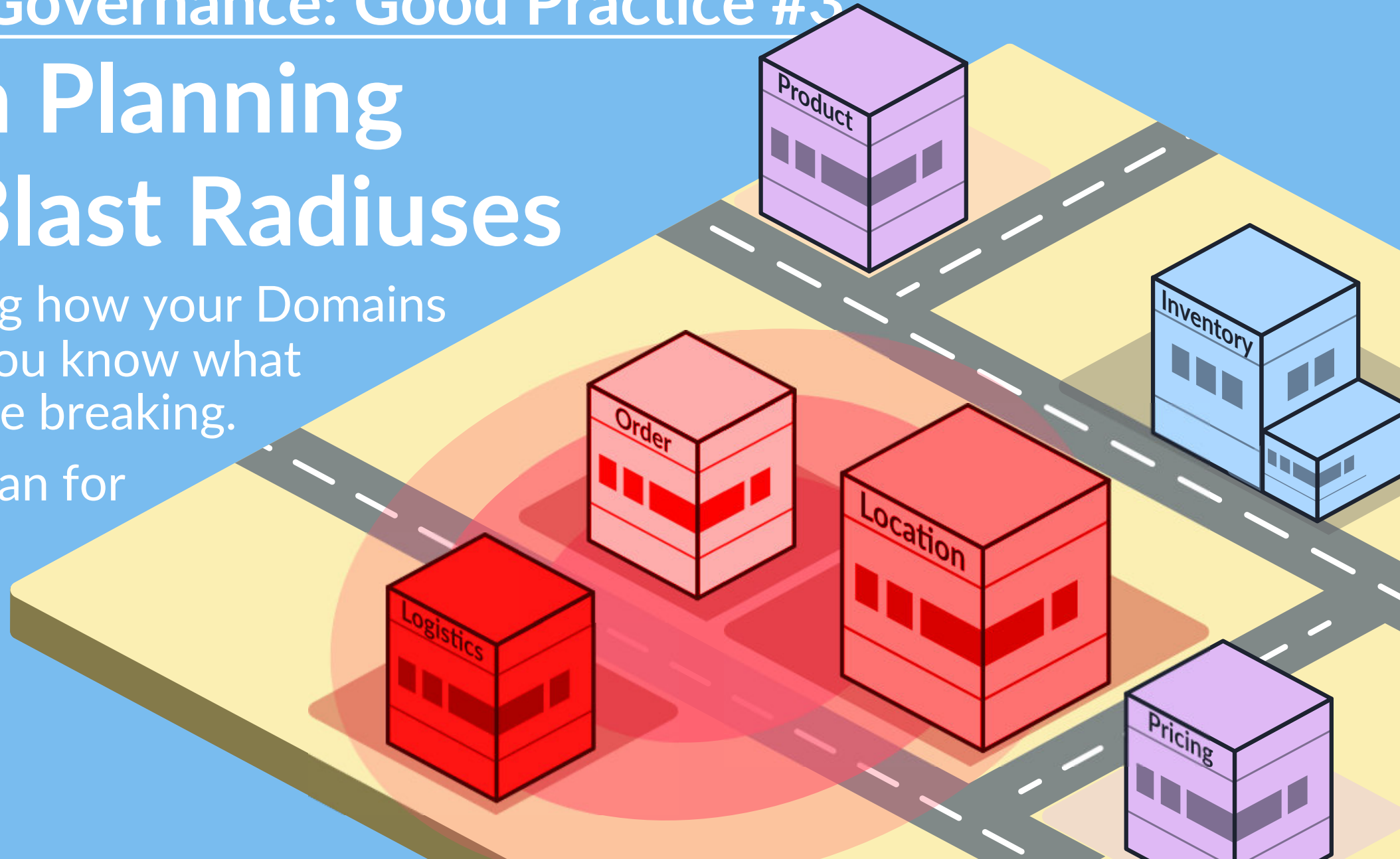


Proper Governance: Good Practice #3

Town Planning and Blast Radiuses

By knowing how your Domains connect, you know what changes are breaking.

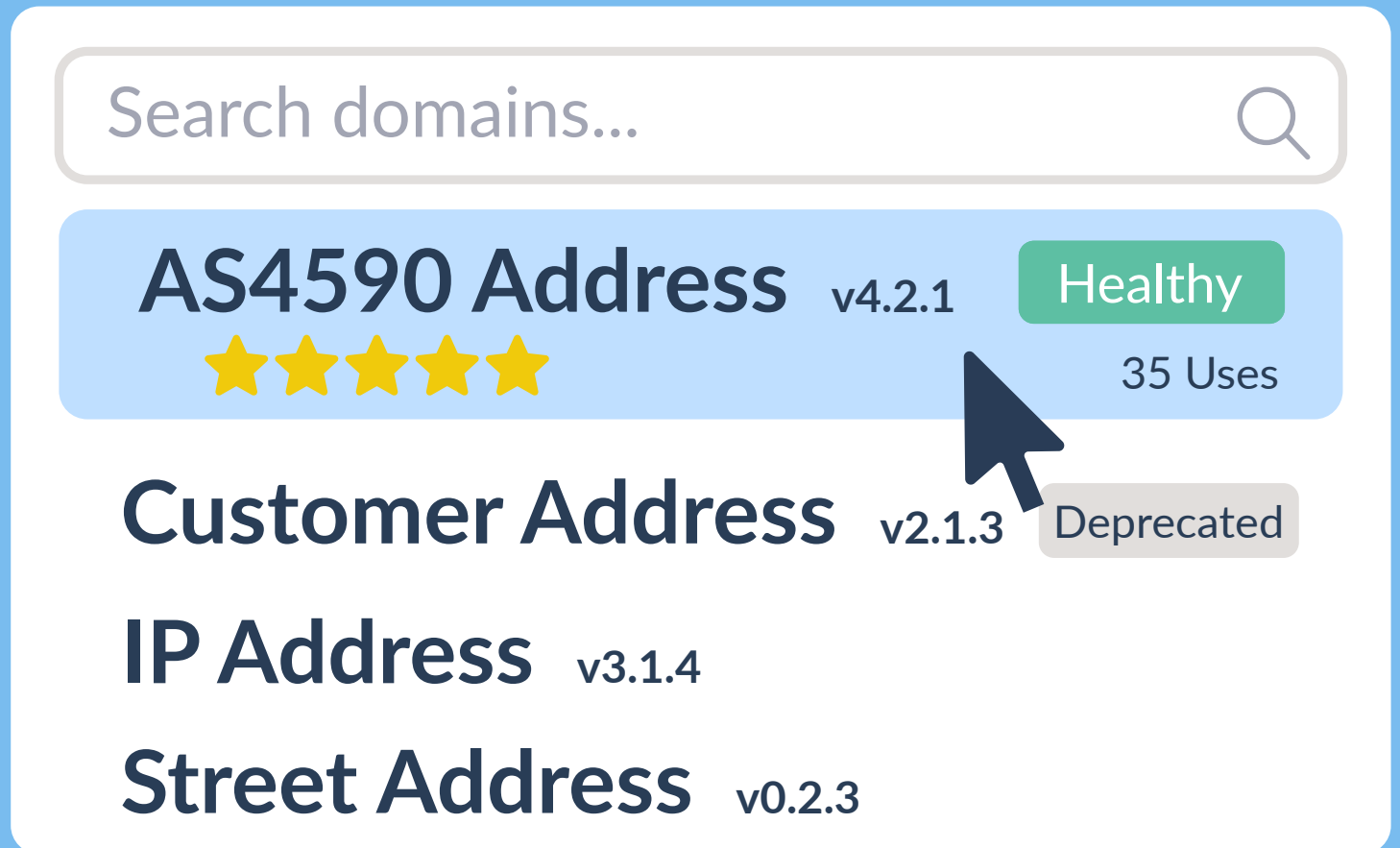
And can plan for the fallout.



Proper Governance: Good Practice #4

Observable Quality at Design Time

Finding the right reusable Domains shouldn't rely on luck.



A screenshot of a domain search interface. At the top is a search bar with the placeholder text "Search domains..." and a magnifying glass icon. Below the search bar is a list of domains. The first domain, "AS4590 Address", is highlighted with a blue background. It has a version of "v4.2.1", a "Healthy" status in a green box, and "35 Uses". Below the domain name are five yellow stars. The second domain, "Customer Address", has a version of "v2.1.3" and a "Deprecated" status in a grey box. The third domain, "IP Address", has a version of "v3.1.4". The fourth domain, "Street Address", has a version of "v0.2.3". A mouse cursor is pointing at the "Customer Address" domain.

Domain	Version	Status	Uses
AS4590 Address	v4.2.1	Healthy	35 Uses
Customer Address	v2.1.3	Deprecated	
IP Address	v3.1.4		
Street Address	v0.2.3		

Proper Governance: Good Practice #5

Executable Governance via CI/CD

If your Governance isn't automated and tied into your existing DevOps pipeline ...

Then it doesn't exist.



Governance Recap

Bad

- No enterprise consistency
- Governance as a road block
- Innovation stops & the Enterprise goes backwards

Good

- Guardrails, not Gates
- Versioned, Curated Domains
- Town Planning & Blast Radius
- Observable Quality at Design Time
- Executable Governance via CI/CD

Shift Governance Left

Invest business value at the start
and get the benefits at the end



Alastair Parker
jargon.sh