

The Hidden Layer: Bringing Protocol Governance into Digital Policy



Kelly Roegies, Fellow at the European Decentralisation Institute

FODSEM 2026: Legal & Policy DevRoom

How digital policy sees tech

Infrastructure - owned assets

Chips, data centres, cloud services, compute capacity, etc.

Regulation - legal obligations

Data protection, platform rules, cybersecurity frameworks, etc.

Institutions - formal authority

Agencies, funding programmes, procurement bodies, etc.



Two models of change

What policymakers understand (contract model)

- Vendor
- Contract
- Update cycle
- Accountability path

→ Change happens by agreement

What open protocols actually are (coordination model)

- Community
- Proposals
- Adoption through use
- Informal influence

→ Change happens by coordination

When digital policy ignores the protocol layer

Policy creates dependency:

- Standards references
- Interoperability mandates
- Funding and deployment choices

Structural consequences:

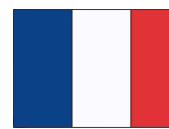
1. Interoperability becomes fragile
2. Exit options become costly
3. Change arrives as operational surprise

But does not model governance

Critical public systems depend on open protocols



civil service



TChap

300,000+ daily users



armed forces



BwMessenger

+100,000 users



healthcare

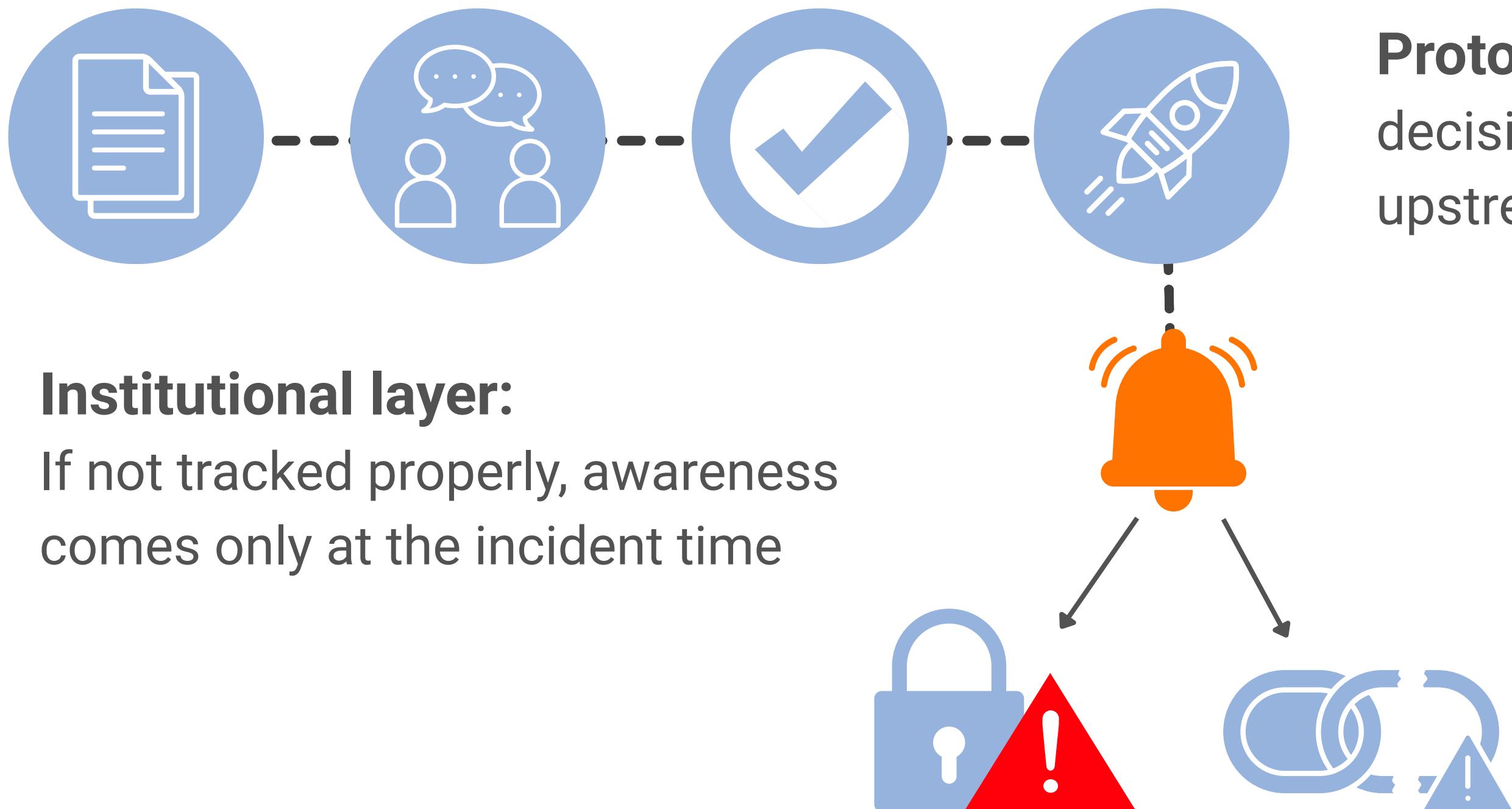


TI-Messenger

150,000 healthcare orgs

Matrix Protocol

Depending on protocols without institutional capacity



Europe's opportunity to fix a structural blind spot

DIGIBYTE | Publication 12 January 2026

Commission opens call for evidence on Open-Source Digital Ecosystems

The European Commission has launched a call for evidence on the upcoming European Open Digital Ecosystem Strategy - an initiative that will support EU ambitions to secure technological sovereignty.

Boosting European technological sovereignty is a key priority for the Commission with the open source sector considered particularly important to European ambitions. The Commission plans to set out a strategic approach to the open source sector in the EU and present a review of the 2020-2023 open source software strategy.



4 pillars for a robust Open-Source Digital Ecosystem **Strategy**

- 1. Require protocol governance assessment in public procurement**
- 2. Recognise protocol stewardship as digital infrastructure & fund it accordingly**
- 3. Build institutional capacity to track and plan for protocol evolution**
- 4. Align procurement incentives with upstream contribution**

Making protocol dependency visible & sustainable

1. Require protocol governance assessment in public procurement

- Identify protocol dependencies explicitly
- Assess how changes and upgrades are decided
- Treat protocol evolution as a procurement risk

2. Recognise protocol stewardship as digital infrastructure & fund it accordingly

- Fund maintenance, coordination, and security work
- Support specifications, not just implementations
- Ensure continuity beyond project cycles

Building capacity & aligning incentives

1. Build institutional capacity to track protocol evolution

- Monitor governance processes and security advisories
- Integrate upgrades into system lifecycle planning
- Anticipate impact before incidents occur

2. Align procurement incentives with upstream contribution

- Reward active participation in upstream governance
- Value maintenance and security contributions

What protocol communities can do

Governance Documentation

- Where/how decisions happen
- How proposals move
- Safeguards against captures
- Upgrade timelines
- Participation norms

Clear on-ramps

- Info on open calls
- Public feedback channels

Summary of the process

The MSC process consists of three basic steps:

1. Write up the proposal in a [markdown](#) document. (There's a [proposal template](#), but don't feel bound by it.)
2. Submit it as a Pull Request to this repo, marking it as a draft until it's ready for wider review.
3. Seek review from the community. Once people are generally happy with it, ask the [Spec Core Team](#) to look at it in [the Office of the SCT Matrix room](#). When the SCT are happy with the proposal, and after a successful voting process, your pull request is merged and the **MSC is now officially accepted** as part of the Matrix Spec and can be used 

For simple changes this is really all you need to know. For larger or more controversial changes, getting an MSC merged can take more time and effort, but the overall process remains the same.

Below is various guidance to try and help make the experience smoother.

Guidance on the process

1. Writing the proposal

Come up with an idea. The idea can be for anything, but the solution (MSC) needs to benefit the Matrix ecosystem rather than yourself (or your company) specifically. Sometimes this means that the solution needs to be more generic than the specific itch that you are trying to scratch.

Remember that an MSC is a formal technical document which will be used by others in the wider community to judge if the proposal should be accepted *and* to actually implement the changes in clients and servers. This means that for an MSC to be accepted it should include justifications and describe the technical changes unambiguously, including specifying what happens in any and all edge cases.

Making governance visible & easy to understand ≠ inviting control.



**Open protocols and digital
sovereignty reinforce each other
when governance is visible,
accounted for and understood.**



Thank you



/kellyroegies



kelly@furt-her.com