METAGOV

A governance layer for the Internet
Metagov is a **software framework** for **online governance**.
How it works

Metagov is a lightweight protocol, packaged as an extensible plugin for online platforms, that helps users govern social groups and solve collective action problems. It’s built on four principles:

1. **Modularity.** Platform operators and community members should have the ability to construct governance systems by creating, importing, and arranging component parts together as a coherent whole.

2. **Expressiveness.** The governance layer should be able to implement a wide range of processes as possible, with minimal bias for one kind of process over another.

3. **Portability.** Governance tools developed for one platform should be portable to another platform.

4. **Interoperability.** Governance systems should have the ability to interact across platforms, sharing data and influencing processes.
Overview

Cross-platform toolset for governance

World-class R&D team at Harvard, Oxford, Stanford, ...

Committed industry partners & funders

Built for social networks, blockchains, online games, and messaging platforms

Toolset

Unified API gateway for platforms & governance services

Simplified installation and configuration of many platforms & services

Flexible agreement engine for groups and contracts

Powerful drivers for authoring and enforcing policies

Type system for composing services and APIs

Research Strategy

Allied experiments:
PolicyKit, CommunityRule, Govbase

Modular Politics whitepaper

University and foundation collaborations

Quick iteration through prototypes

Everything is open-source
Governance is complex, and governance systems are hard to build.
Governance complexity is a barrier to sustained growth.

Governance is necessary to growth. Ungoverned spaces like 4chan or Gab quickly become toxic. But governance at scale is complex and expensive.

To succeed and grow, online communities must use a diverse array of tools (laws, norms, markets, and code) to channel user behavior.

---

Governance is an unaddressed need

There is huge interest in decentralized governance, with experiments across Facebook, Twitter, and the blockchain. A recent example: the explosion of mutual aid groups in response to COVID-19.

However, the current solutions are ad hoc, limited, and ill-equipped for governance innovation—often featuring rigid hierarchies, static toolsets, and typologies suited more to nations than the needs of online communities.

Crowdsourced map of COVID-19 mutual aid networks, from mutualaidhub.org.
A simple idea: empower users

Platform operators are not the only ones working on governance; users and independent developers are too.

Metagov integrates these three processes, taking the load off platforms and increasing the efficiency of the entire system.
Simple building blocks

Users form *agreements*, which are interactive virtual machines that support both textual contracts and simple computations.

*Modules* provide rich functionality for governance, including enforcement. For example: voting, tokens, reputation, and chat.
The Metagov Stack

Metagov Core
A unified API gateway that connects and interoperates a wide range of platforms and governance services.

PolicyKit
A powerful authoring tool for creating and editing governance policies for digital platforms.

Agreement Engine
Software for interacting with smart contracts and other software services as both resources (to be governed) and virtual machines (that govern).
**Ecosystem**

**Engagement for platforms**
Platforms install an Agreement Engine plugin as part of their platform, or interact with Metagov's servers through an API. Implementation may vary in scope.
In doing so, they increase retention and engagement by allowing users to customize their experience within certain constraints. They also drastically reduce customer service costs when users are empowered to deal with their own problems.

**Freedom for users**
Users use the Metagov interface to manage and customize governance decisions.
Allowing users to customize aspects of the platforms they use turns them from users to co-designers.

**Support for developers**
Developers and scientists use the SDK and online development environment to design, test, and simulate complex social mechanisms before deploying to the Metagov network.
Built-in platform integrations allow developers to build fast and scale quickly to multiple platforms.
A portable, cross-platform tool

Our key innovation is the development of a truly cross-platform governance protocol, one that works across social networks, blockchains, games, and instant messaging.
Team

Metagov
Lawrence Lessig
Principal Investigator

Roy L. Furman Professor of Law and Leadership at Harvard Law School
Founder, Creative Commons
Team

Primavera de Filippi

*Harvard/CNRS*

Blockchain lawyer, firestarter, governance researcher

---

Seth Frey

*University of California, Davis*

Data scientist, social scientist, expert in online institutions

---

Nathan Schneider

*University of Colorado Boulder*

Journalist, democracy activist, platform expert
Joshua Tan  
**Oxford/Stanford**  
Computer scientist, mathematician, entrepreneur

---

Amy Zhang  
**University of Washington**  
Computer scientist, human-computer interaction

---

Federica Carugati  
**Stanford University**  
Historian, classicist, economist
Partners and funders committed to change
Seed

Seed is a massively-multiplayer simulation game being developed by Klang Games. Our core team has been working with them for over three years, helping them design the political and economic systems of the game.

“We want players in Seed to experience a world with a huge range of different kinds of politics. We want to see constitutional democracies, medieval monarchies, corporate dictatorships, and 70s-style communes all emerging out of player-to-player interactions. We are incredibly excited to be working with Larry, Josh, and the Metagov team to make this possible.”

Mundi Vondi
CEO of Klang Games

A massive, dynamic world

Traditional sandbox MMOs like Minecraft allow players to build vast, empty cities that have no purpose except to be seen. In Seed, millions of players will come together to build vibrant cities, companies, and even nations.
The perfect testbed for governance

Players on the Seed Discord are already forming factions and imagining their own constitutions. Factions will need to motivate and control their players. Governance is not an option. But governance is also part of the fun.
DADA is a platform for art-based conversations. We are working with DADA to prototype and govern a new, “dada-ist” creative economy.

"With the creation of Dada’s Invisible Economy we are experiencing a governance process that doesn’t seem to fit any conventional model. We realized we don’t need voting mechanisms or to reach consensus, and we don’t need rules or sanctions either. Yet it’s working beautifully so far. We feel very lucky to be working with Primavera and Josh to help us make sense of our own process, and really excited that Metagov can design a new governance model that any community can use.."

Beatriz Ramos
Founder and CEO

Metagov
Vingle is an interest-based social network of 5 million users. We are working with Vingle and the CAN Foundation to build more robust self-governance tools for their interest communities.

“At Vingle, we want to explore a new paradigm of collective decision-making. We’re very excited to be working with the Metagovernance Project on a governance design to support our user communities.”

Changseong Ho
President, COO
Roadmap overview

2021
- Proof of concept deployed with one of our partners
- Modules/editor research & development (user interviews, workshops, prototypes)

2022
- Expand partnerships to incentivize protocol adoption (games, blockchains, etc)
- Sustainable funding strategies (e.g., open-source + software as service for premium features)
- Scalable module/agreement engine
1-year roadmap

Initial Prototypes
- Metagov v0.1: Govbase
- Govbase roll-out, Iterate
- Metagov v0.2: Core Plugin Architecture
- 
- Theoretical framework
- Metagov v0.3: Parallel Prototypes

Deployable Prototypes
- Metagov v0.4: Agreement Engine
- Metagov v0.5: Module Design Language
- 
- Core module development
- Mod developer community management

Deployed Proof of Concept
- Partner Integration
- Partner Workshops
- 
- Alpha Testing
- 
- Reflect on Outcomes

Metagov
PM Project Management
RT Research team
DEV Development
ERT Extended research team
ALL All
Key challenges

Translating science into engineering
Metagov is both a basic research project as well as a software development project that will be deployed on live, commercial platforms. The challenge is to balance the science and the engineering. Mapping research questions to engineering requirements is important, as is forming long-term collaborations with external researchers. The core team is experienced in running such research-backed software ventures, and we are very confident that we can address this risk.

Platform adoption
Not all platforms are interested in enabling self-governance or decentralization. We will get around this by seeking out ideological partners, building our own connectors to certain platforms, and by conducting focused advocacy that argues for the legal and economic value of self-governance.

User adoption
Governance is necessary, but it can get complicated. Once embedded in a platform, the toolset needs to be easy to use. We will take a prototype-and-iterate approach to make sure that whatever we build will appeal to actual users.
Vision

A governance layer for the Internet. A sandbox where we can test and implement not just software but also laws, norms, and markets. Simple building blocks (modules) built on a rigorous foundation (computational agreements), but leading to a huge possible range of editable governance forms and purposes, from traditional democracies to nonprofit corporations to minimalist libertarian communities. Scientific tools to study, analyze, and edit the behavior of these governance systems, allowing feedback to users and robust experiments. And finally, tools to scale up the successes and prune the failures.

“If the digital future is to be our home, then it is we who must make it so.” – Shoshana Zuboff
Vision

Too many would-be platforms regard governance as a secondary activity. Too many would-be tools think of governance in simplistic terms, in terms of proposal systems and majority voting schemes.

As communities mature, users’ wants and behaviors and expectations change, transforming the platforms that host them. But without effective governance, communities cannot mature.

Without governance, a game will always be just a game. With governance, it can become something more.
METAGOV

part of the Metagovernance Project

www.metagov.org
Research Overview
Metagovernance is the foundation

Metagov is a metaconstitutional tool. It is designed to create effective constitutional systems (a.k.a. user groups). Users then define the rules and laws of collective action within these constitutional systems.

Figure 1: Levels of analysis and outcomes

- Individuals’ Actions Taken that Directly Affect State Variables in the World
  - Operational Situations (Provision, Production, Distribution, Appropriation, Assignment, Consumption)

- Physical World

- Operational Rules-in-Use

- Individual’s Actions Taken that Directly Affect Rules that Affect Operational Situations
  - Collective Choice Situations (Prescribing, Invoking, Monitoring, Applying, Enforcing)

- Physical World

- Collective Rules-in-Use

- Community

- Individual’s Actions Taken that Directly Affect Rules that Affect Collective-Choice Situations
  - Constitutional Situations (Prescribing, Invoking, Monitoring, Applying, Enforcing)

- Physical World

- Constitutional Rules-in-Use

- Community

- Individual’s Actions Taken that Directly Affect Rules that Affect Constitutional Situations
  - Meta-Constitutional Situations (Prescribing, Invoking, Monitoring, Applying, Enforcing)

- Physical World

- Community

From research questions to software requirements

Questions

How do we facilitate innovation in (self-) governance?

What are the building blocks of governance?

How portable is governance?

Requirements

Modularity: users need to be able to vary governance mechanisms (including code) easily and predictably.

Interoperability: modules A, B developed for platform X work together, enabling them to function together as a coherent system.

Portability: modules (e.g. election systems) developed for platform X also work for platform Y.
The lifecycle of governance

Stage 0
- Transition Trigger
- State of Nature
  - External forums
  - External chat rooms
  - Messaging

Stage 0.5
- Deliberation Space
  - Proposal
- Proposal management (e.g., upvoting or holographic consensus)
- Document editing

Stage 1
- Agreement Engine
  - Agreement Registry
  - Law Archive
  - Constitution
- Choose (One, Few, Many) and Modify

Stage 2
- Enforcement
  - Normative
  - Police
  - Courts
  - Reputation tracking
  - Law as code
- Police, Courts, Reputation, Code

Stage 3
- Revolution Trigger
- Record of infractions and punishments

Metagov
Govbase
An open database of projects and tools in online governance. metagov.org/govbase

<table>
<thead>
<tr>
<th>Project name</th>
<th>Year founded</th>
<th>Status</th>
<th>Online / offline</th>
<th>Implements structures</th>
<th>Category</th>
<th>Subcategory</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAOcreator</td>
<td>2019</td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>software library</td>
</tr>
<tr>
<td>DAO Maker</td>
<td></td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>application/tool</td>
</tr>
<tr>
<td>Aragon Fundraising</td>
<td>2018</td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>software library</td>
</tr>
<tr>
<td>Aragon Reputation</td>
<td>2018</td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>software library</td>
</tr>
<tr>
<td>Aragon Membership</td>
<td>2018</td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>software library</td>
</tr>
<tr>
<td>Aragon Company with Bo...</td>
<td>2018</td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>software library</td>
</tr>
<tr>
<td>Aragon Company</td>
<td>2018</td>
<td>Active</td>
<td>Online communities</td>
<td>DAO</td>
<td>product</td>
<td>software library</td>
</tr>
<tr>
<td>Laosild</td>
<td>2020</td>
<td>Work in...</td>
<td>Online communities</td>
<td>Online community</td>
<td>product</td>
<td>social network</td>
</tr>
<tr>
<td>Forem</td>
<td>2017</td>
<td>Active</td>
<td>Online communities</td>
<td>Online community</td>
<td>platform</td>
<td>social network</td>
</tr>
<tr>
<td>Zendesk Gether</td>
<td></td>
<td>Active</td>
<td>Online communities</td>
<td>Online forum</td>
<td>platform</td>
<td>forum platform</td>
</tr>
<tr>
<td>Collab.land</td>
<td>2020</td>
<td>Work in...</td>
<td>Online communities</td>
<td>Online chat</td>
<td>platform</td>
<td>messaging platform</td>
</tr>
<tr>
<td>Riot</td>
<td></td>
<td>Active</td>
<td>Online communities</td>
<td>Online chat</td>
<td>platform</td>
<td>online game</td>
</tr>
<tr>
<td>Seed</td>
<td></td>
<td>Work in...</td>
<td>Online communities</td>
<td>Online chat</td>
<td>platform</td>
<td>online game</td>
</tr>
<tr>
<td>Reddit</td>
<td></td>
<td>Active</td>
<td>Online communities</td>
<td>Online forum</td>
<td>platform</td>
<td>forum platform</td>
</tr>
<tr>
<td>Substack</td>
<td></td>
<td>Active</td>
<td>Online communities</td>
<td>Online newsletter</td>
<td>product</td>
<td>application/tool</td>
</tr>
<tr>
<td>Govblocks</td>
<td>2017</td>
<td>Active</td>
<td>Online communities</td>
<td></td>
<td>standard</td>
<td>application proto</td>
</tr>
</tbody>
</table>
Modular Politics

A peer-reviewed research publication (presented CSCW 2021), which describes some of the key building blocks of online self-governance. metagov.org/modpol

Fig. 1. A schematic implementation of the system described throughout this section. Here, a single server running Guilded Age also runs an Instance of Modular Politics. The Instance translates game data into assets recognizable by different Orgs and Modules. Finally, guilds in Guilded Age correspond to Orgs, and each guild may install a number of Modules and track a number of Resources. An Org may have a sub-Org, e.g. a guild may have a sub-group of officers with special permissions.
From games to institutions

A research project to construct a computational representation of social rule systems using compositional game theory.
Modeling digital platforms
Towards a functional type system to help users combine modules.
Design Appendix
Module View

The module view organizes modules according how they fit together. From left to right, modules are ordered according to their relationship to one another.

Non-technical users can use pre-made modules to create organizational forms, while technically-inclined users can go underneath the hood to edit the modules according to their own needs.
Usability vs Abstraction

Visual programming languages are quite powerful and easier to use than command line programming. However, we believe that modules should exist at a higher level of abstraction, enabling users to think creatively with modules and how to combine them to create different strcutrues.

We believe that a mix of simple to use modules (already pre-configured) with more granular and computationally driven node-based programming can exist within metagov.

We want users to be thinking about governance, not programming.
Interaction Model References

The project below (programmable blocks) strikes a good balance in making things feel simple. We're using a similar approach in our design language. We want our interface to look and play like a game, rather than something stiff and serious.
Interaction Model

Modules and their rules are summarized in a high level "constitution" format from which everything is made transparent.

Ready-made modules are easily available and ready to be configured.

Node-based language from which new modules can be assembled from.

Metagov
Modules Examples

The list below is just an example of a few modules we're considering implementing first. Each module will have editable properties and ways to customize its behavior.

Voting
- Lottery
- Approval
- Condorcet
- Quadratic
- Continuous

Curation
- Staking
- Token curated registry

Resource Management
- Automated Payments
- Tax collection

User Group
- Board Member
- User Groups
- Federations

UI
- Sliders
- Radio buttons

Proprietes
- Delegation
- Election Period
- Eligible voters
- Vote weight
**Module notification:** if new interaction is required from a user, it will show up within a specific module. A user will need to click and perform an action in order to de-activate it. Actions are: voting, delegation, and so on.

**Module Properties:** edit all the available properties for a module. (e.g. by instantiating a module, you have to set its parameters in order to instantiate it. There are no defaults for modules, users need to set them up)

**Module UI:** is where you might go as a user to interact with a parameter (e.g. a voting module comes with a voting UI that gets created after all the properties are defined)
Constitution

Preamble
Our main goal is to make players wealthy, especially newcomers. We want people associated with us to be the richest players in the game. We wish to provide advantageous contract deals for Business Owners, Investors, Employees, as well as teaching them a guide on how to achieve financial success.

Proposal Submission
All proposals will be voted on using a quadratic voting system.

```python
import members as @
import voting, quadratic_voting
members_list = (@Josh, @Nathan, @Primavera, @Louis, @Larry)
voting.mechanism = [nailer]
voting.options = quadratic
if voting.votes > 10:
  voting.vote_outcome = exception.invalid

quadratic_voting.options = (k,1,d)
quadratic_voting.period = (5,days)
quadratic_voting.scaling = 2.1
```

Roles
All roles will be filled through direct elections.

```python
import roles
roles.executive = @Bob
roles.representation = @Nathan, @Primavera, @Seth
if roles.executive = empty:
  voting.succession
```

Amendments

```python
import amendments
amendments("Proposal Submission") = (greater than .5, all_members)
amendments("Roles") = fixed
```

Quadratic Voting

Properties

Vote Eligibility

<table>
<thead>
<tr>
<th>Include Members</th>
<th>Include Members</th>
<th>Voting Period</th>
<th>Voting Period</th>
<th>Voting Period</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>3 days</td>
<td>3 days</td>
<td>3 days</td>
</tr>
</tbody>
</table>

Interface

A voting period is open: please select 3 candidates from all users from which you'd like to support. If you're not sure how to proceed, you can review all the rules set for this module, and read more about Quadratic Voting in our wiki. If you're still not sure, you can always contact the administrator for support.

Select Candidates

- [ ] Susan M
- [ ] Add Candidate

Added Candidates

- [ ] Guillem Miller
- [ ] Ostrom Mauer
- [ ] Ostrom Mauer

Confirm Changes
Data and Feedback

Users will be able to take data collected from Metagov services and plot them using built-in real-time “monitors”, i.e. governance metrics and indicators.

For power-users, Metagov will support a Jupyter notebook integration for producing embedded data visualizations.