



Govbot

Federated, open-source legislative data for everyone

Overview

The Problem

Our Solution

What We Offer

Features

Setup + Core Functions

The Problem

Why don't we pay attention to our representatives between elections?

Legislative data is hard to parse, track, and organize. Activists, concerned citizens, and the curious may not have the time, resources, or expertise to build out duplicative tech stacks.

The Problem (cont.)

Existing solutions may be limited by the willingness of organizations and companies to continue to run and host them – such as in the case of Google's Civic Information API, which was shut down earlier this year.

What would a decentralized, open-source legislative data solution look like?

Our Solution

The Govbot team's goal is to bridge this gap – building the framework for federated, open-source, non-profit legislative data.

Built as a [Chi Hack Night Breakout Group](#), this project offers frameworks and tools built on top of OpenStates' data on state and federal legislation.

What We Offer

The main Govbot dataset currently includes legislative updates from:

- the U.S. House & Senate
- Legislatures from all 50 states
- Legislatures from U.S. territories

Data is organized as .json files using the Project Open Data catalog format, scraped and appended regularly.

Features

- A decentralized, regularly updating, legislative data catalog
- AI-powered, topic-based tagging and summaries, customized using .yml
- SQL querying via DuckDB interface
- Example applications, like custom websites (see our demo [WindyCivi](#) site), and social media bots (see our [BlueSky bot](#), made in collaboration with U.S. Representative Hoan Huynh)

Setup

You can download the setup script via one-line install, from our GitHub repository:

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/windy-civi/toolkit/main/actions/govbot/scripts/install-nightly.sh)"
```

Core Functions

Once installed, you can:

- Clone the entire dataset
- Clone specific items (state, session, or bill)
- Load metadata into a SQL-accessible DuckDB database

Project History

2022: socratic.center

The Govbot project began in 2022 at socratic.center with a vision to create a destination for simplified, summarized updates on legislative action.

The initial hypothesis: *What if citizens could easily track and understand the bills being voted on?*



Representatives for 20 N Wacker Dr, Chicago, IL
60606, USA.

Chicago City Clerk

Anna M. Valencia



Chicago City Treasurer

Melissa Conyears-Ervin

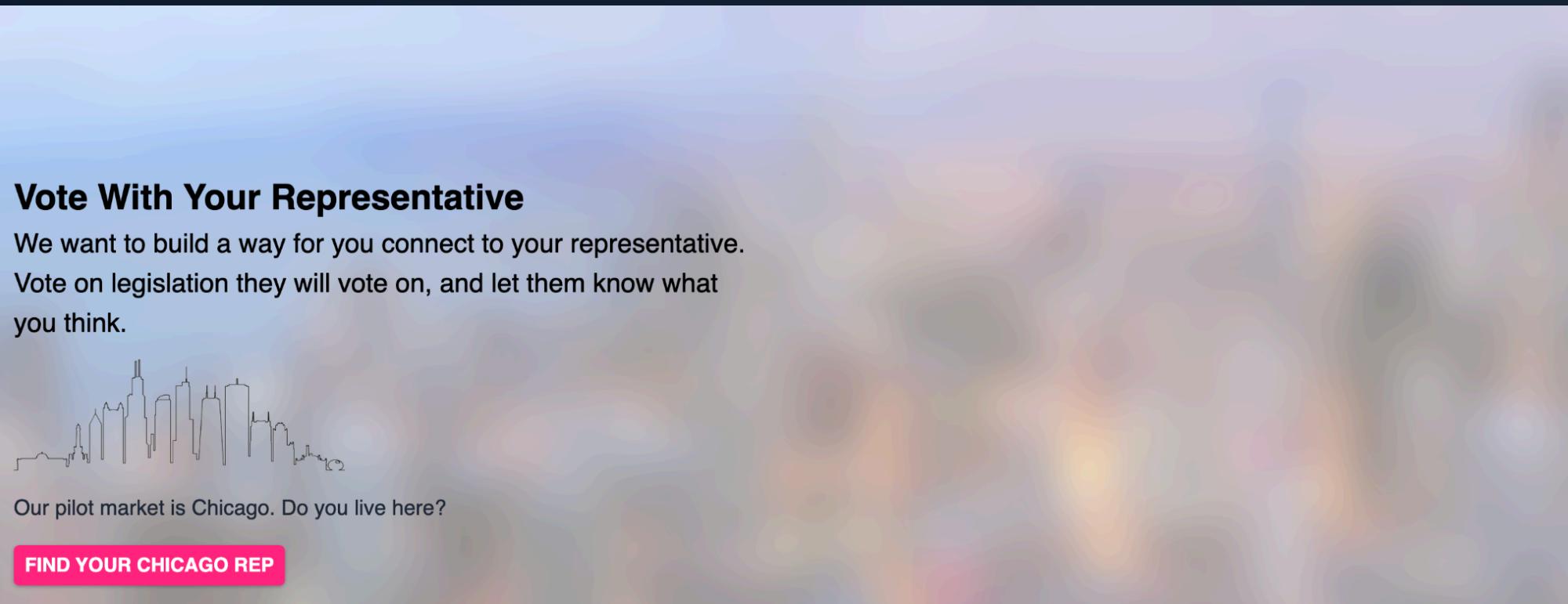


Mayor of Chicago

civi.social

We built civi.social, exploring how to make legislative information accessible and shareable on social platforms.

This experiment helped us understand how citizens wanted to engage with civic data in their existing communities.



Vote With Your Representative

We want to build a way for you connect to your representative.
Vote on legislation they will vote on, and let them know what
you think.



Our pilot market is Chicago. Do you live here?

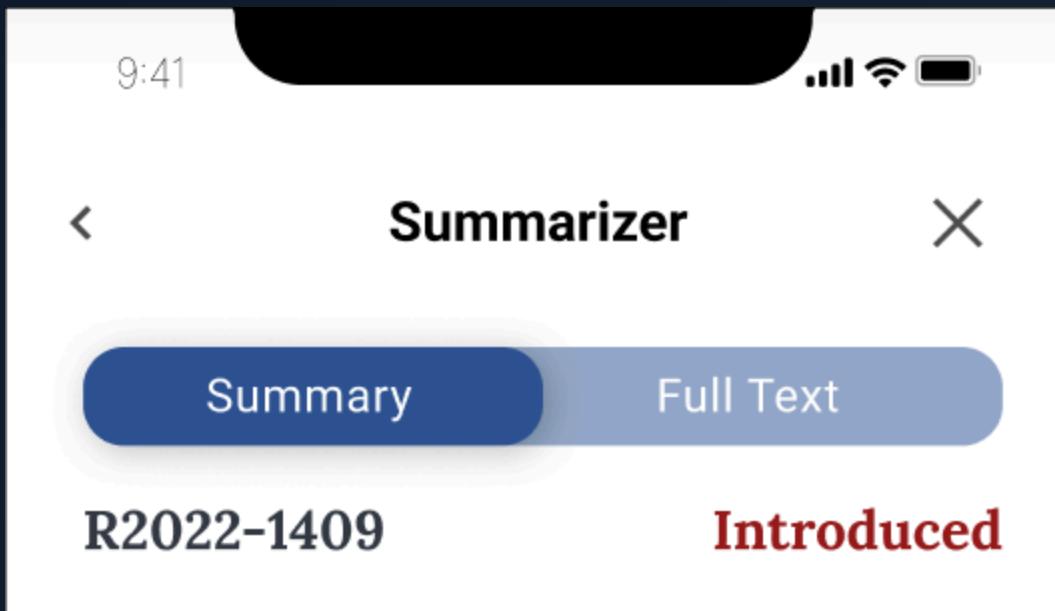
[FIND YOUR CHICAGO REP](#)

myChicago + Jarvis

We created a prototype for what integration with the myChicago platform would look like.

The goal of Jarvis, our AI-powered assistant, would have been to help users understand legislation through:

- Simplified bill summaries
- Contextual information
- Guided engagement tools

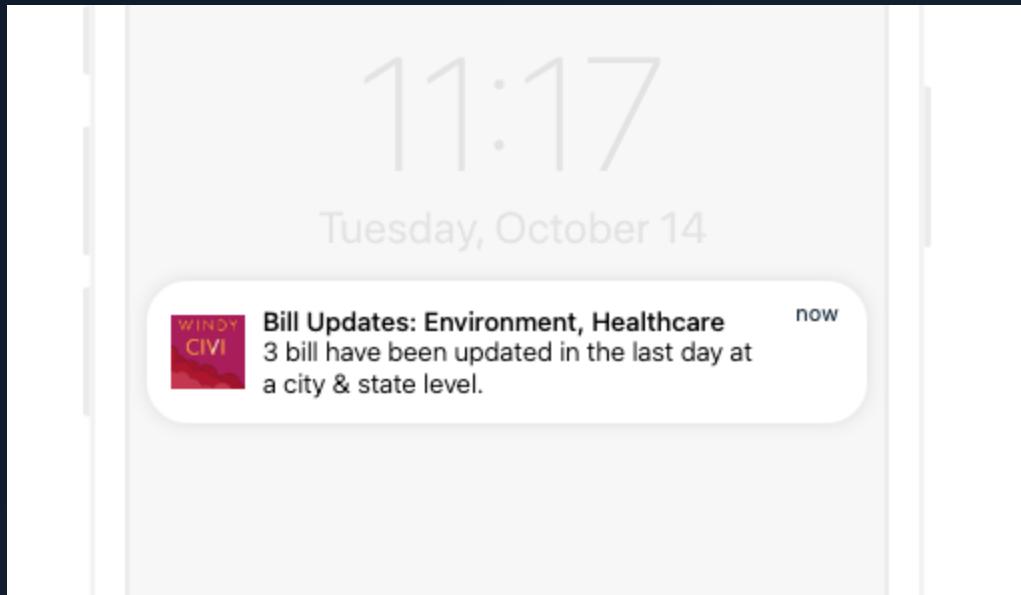


Windy Civi: Full Launch

After reflecting on previous concepts, the Windy Civi app and website launched in beta in 2024.

The goal was to enable citizens to:

- Track bills by topic
- Receive personalized updates
- Connect directly with representatives



Rethinking Our Approach

While building these solutions, we began to ask a critical question:

What are the limitations of a centrally-managed platform?

- Can it scale to serve all communities?
- What happens if we stop maintaining it?
- How can others build on this work?

Our New Vision

Our vision has now pivoted to building **the infrastructure itself**:

1. **A decentralized legislative data catalog**
2. **Reusable frameworks** for communities to build their own tools
3. **Sample applications** demonstrating use cases

Our goal: Ensure that government accountability is accessible to all.

Live Demos

Basic Setup + Commands

Querying via DuckDB

Creating Social Media Bots

Basic Setup + Commands

Install via:

```
sh -c "$(curl -fsSL https://raw.githubusercontent.com/windy-civi/toolkit/main/actions/govbot/scripts/install-nightly.sh)"
```

Once installed, you can download and set up the data using the following commands

```
govbot # to see help  
govbot clone # to show available datasets  
govbot clone {{locale}} {{locale}} # download specific items  
govbot delete {{locale}} # delete specific items  
govbot delete all # delete everything  
govbot load # load bill metadata into DuckDB
```

Querying with DuckDB

First, set up DuckDB, which creates a simulated database from the .json log files:

```
govbot load #Load all data into a database  
govbot load -database my-bills.duckdb #Specify a custom database file  
govbot load -memory-limit 32GB -threads 8 #With memory limit and thread settings  
  
duckdb -ui govbot.duckdb #Open in DuckDB UI (opens in browser)
```

Once the DuckDB database is created, you can query as normal

```
- Load JSON extension  
INSTALL json;  
LOAD json;  
- Query all bill metadata  
SELECT *  
FROM read_json_auto('~/govbot/repos/**/bills/*/metadata.json')  
LIMIT 10;
```

Creating Social Media Bots

Technical Details

An Open Data Proposal
A Dive Into Local AI Tagging

An Open Data Proposal

A Dive Into Local AI Tagging

Thank You!

Building government accountability tools
accessible to all



Appendix

Contributing & Testing
FAQs

Contributing & Testing

Prerequisites

Knowledge of Rust and the `just` task runner required.

1. **Rust & Cargo**: Install the [Rust Toolchain](#)
2. **Just**: Install the task runner: `cargo install just`

Development Workflow

Use `just govbot ...` as your CLI "dev" environment.

Useful Commands:

- `just` – See all available tasks
- `just test` – Run all tests
- `just review` – Review snapshot test changes
- `just mocks [LOCALES...]` – Update mock data for testing

Dataset Status Key

-  The locale's data received updates since your last cloning
-  Your data is up-to-date with the most current version
-  The data is currently being updated
-  The data is not currently accessible

FAQs: Repositories

Can I See The Repo?

- Main repo: [windy-civi/windy-civi](#)
- Toolkit repo: [windy-civi/toolkit](#)

FAQs: Data Structure

How Is The Data Structured?

Find the file format structure and .json schema in the readme.md:

[DATA_STRUCTURES.md](#)

FAQs: Cloning Data

How Do I Clone This Data?

Each locale is scraped using a GitHub Actions template explained here:

[README TEMPLATE.md](#)

To manage multiple pipelines or locales, see our [pipeline manager documentation](#)

Stay Connected

How Can I Stay Updated, Or Get In Touch?

- Follow our work at [Chi Hack Night](#)
- Check commits and updates on [GitHub](#)
- Visit our [Docs page](#)
- Join the [Chi Hack Night Slack](#)