



# Rapid implementation of policy as code

By [Alex Soble](#), [Mike Gintz](#)

Published on May 12, 2020

[api](#) [10x](#) [transformation services](#) [public benefits](#) [rules as code](#)

No policy or rule stays the same forever. In response to a crisis, policy changes often come much faster, and stakes can be higher. A new rule might change the amount of benefit someone receives as part of a safety net program, or make someone eligible who was previously ineligible.

But policy changes only create impact once they are implemented. And in 2020, almost all social service programs use digital systems to manage eligibility and enrollment. Most policies written into legal code will eventually need to be programmed into these digital systems as part of implementation. These digital systems include tools used by caseworkers, front-end portals used by clients, and back-end systems that administer benefits.

If you are working on implementing new rules during this critical time, you may be wondering:

*“What can I do to speed up the implementation of these new rules? What can I do to make sure that my agency’s updated rules are interpreted and implemented accurately in our digital systems?”*

In the United States, policy staff typically write rules in English and distribute them in PDF form. Technical staff then read the complex policy rules and translate them into computer code to implement the policy in digital systems. Structuring this as a two-step process introduces lag and room for misinterpretation. Bringing technical and policy staff together to [write and publish policy rules as computer code](#) together early on improves both speed and accuracy.

Our team at 18F — the [Eligibility APIs Initiative](#), powered by GSA’s [10x](#) — has been researching and prototyping techniques for putting these concepts into practice. Back in 2018, we wrote about our early experiments [implementing rules with modern programming languages](#). Today, we want to share an example from our work that shows the potential for rapid, accurate policy implementation as code.

**Follow 18F**  
[18F on GitHub](#)  
[18F on Twitter](#)  
[18F on LinkedIn](#)  
[RSS feed](#)

## Prototyping SNAP eligibility

Eligibility for SNAP (food stamps) is complex, with many factors playing a role in determining a household’s eligibility and final benefit amount. This complicated logic has been coded and re-coded many times by different states, territories, non-profits, and legal aid organizations. SNAP eligibility is somewhat different in each location, but there is a core of shared federally set rules across all localities.

In collaboration with policy experts at the federal level, we created an experimental prototype model of SNAP eligibility that is:

- **Open and transparent:** Because we are following an open-source approach, anyone can see our [prototyping repo](#) on GitHub, check our inputs and assumptions, point out bugs, and re-use our work. We used a popular open-source coding language (Python) that many developers are familiar with, and which doesn’t require proprietary tools or licenses.
- **Shareable between systems:** Our program is meant to be used by other programs — it is built as an API, or an Application Programming Interface. For example, another system could call our API, send information about a hypothetical household, and receive information about that household’s SNAP eligibility. This could lay the groundwork for a shared service between states, territories, and even non-profits like legal aid organizations.
- **Verifiable as accurate:** Our prototype includes automated tests that check to make sure outputs stay accurate and predictable, even in a rapidly changing policy environment. We use [Gherkin](#) for our automated tests, so even policy experts unfamiliar with code can take part in confirming that the coded logic is correct.

We characterize this work as a [prototype](#) because it’s meant to be a small and lightweight proof of concept. To visualize the kind of application that our eligibility API can power, we created a pre-screener:

So far, our prototype has been focused on the financial aspects of SNAP eligibility. When SNAP’s eligibility policy changed in response to the current crisis, this provided an opportunity to test the value of our approach.

## Responding quickly to policy change

Recently, USDA [granted waivers](#) that allow states to issue emergency SNAP allotments. This change in policy affects the amount SNAP beneficiaries receive: in states with SNAP emergency allotments, households automatically receive the maximum benefit for their household size, rather than an amount calculated based on household income and deductions.

We decided to challenge ourselves and see how quickly we could add this new rule to our prototype. **We found that we could add the rule in a single morning.**

Of course, it’s much easier to change a rule in a lightweight, low-stakes prototype than in a production Integrated Eligibility System responsible for real-world SNAP determinations. This example, however, still demonstrates how rapid the implementation of new, important policy changes could be. Our approach allows for the following:

- Because our code is open-source, anyone can double-check our work, raise issues or questions with our modeling, and help us keep it up-to-date.
- Because our code uses automated tests, we can push new policies and rules with extra confidence, reduced risk, and built-in quality control.
- Because our approach is API-driven, multiple applications could all benefit at the same time from a single rules update, helping implement a new policy or rule quickly across an entire ecosystem.

If you are managing an in-production benefits system, a first step in this direction might be to bring policy and technical staff together the next time rules need to be added to or changed within the system and have them collaborate, rather than translating those rules into code after the fact.

## What’s next?

If your agency would be interested in learning more about these approaches, we want to hear from you. Would more rapid policy implementation as code help your agency implement its mission, especially in times of crisis? Might our early pre-screener prototype be of value in your state? **We are actively seeking agency partners in this work — reach out to us at [eligibility-apis-initiative@gsa.gov](mailto:eligibility-apis-initiative@gsa.gov).**

Finally: we recognize that this is a time of extreme strain and burden on all areas of the human safety net. We thank and appreciate all safety net staff and workers at this time for your critical work to serve and support others.

~

*Thank you to Elizabeth Ayer, Alex Pandel, and Abbey Kos for feedback and comments on early drafts of this post.*

[◀ Previous post](#)

[Saving time and improving data quality for the National School Lunch & Breakfast Program](#)

[Next post ▶](#)

[Design & research in critical times](#)

### Catching up with the TANF Data Portal project

Around 800,000 low-income American families receive cash assistance through Temporary Assistance for Needy Families (TANF) each month. 18F and the Administration for Children & Families’ Office of Family Assistance partnered on building a new data portal for TANF. We caught up with Office of Family Assistance leaders to see how their agency is continuing with the work.

### Saving time and improving data quality for the National School Lunch & Breakfast Program

In their 2020, pilot states reported back on their experience implementing and using a new data validation service during the fall of 2019. They agreed that it had significantly improved the data collection and submission process for their teams, citing time savings, fewer errors returned, greater ability to plan ahead, and lower stress.

### Long-term teams, not sudden handoffs

Don’t let your agency waste knowledge and opportunity. Instead of planning for a handoff to operations and maintenance, plan for a long-term team. Instead of launching your project and then keeping it running, plan for ongoing development.



Work with us to plan successful projects, choose better vendors, build custom software, or learn how to work in new ways.

[Contact us](#)

#### Pages

- [Our work](#)
- [Work with us](#)
- [About 18F](#)
- [Guides](#)
- [Blog](#)
- [Contact](#)

#### Policies

- [Linking policy](#)
- [Open source policy](#)
- [Vulnerability disclosure](#)
- [Code of conduct](#)

#### Contact

- [Get in touch](#)
- [Press](#)
- [Report a bug](#)
- [Join 18F](#)

#### Social

- [GitHub](#)
- [Twitter](#)
- [LinkedIn](#)