
Automating the Rules as Code Process with AI

AI does not replace legal modeling;
it reduces the cost of Rules as Code at scale.



Benoît Courty

Data Scientist
Assemblée nationale de France

Rules as Code Workshop 2026

Agenda

01 Introduction & Context

LexImpact, OpenFisca, and the parameter challenge

02 The Journey & Methodology

From 5% to 99% success with evolving AI models

03 Solutions & Workflows

UI, agents, and skills for parameter extraction

04 The Next Frontier

Full code generation from legislative text

05 Challenges, Learnings & Future

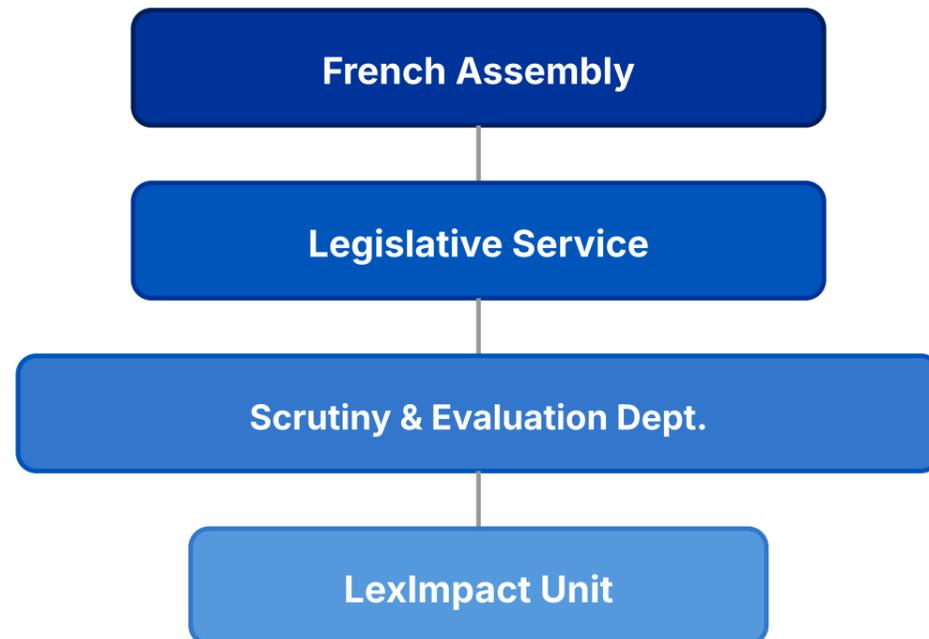
Sovereignty, cost, and human-in-the-loop



Benoît Courty

Data Scientist (Assemblée nationale de France)

LexImpact at the French Assembly



5 FTE • economists, developers, designer, data scientist

//

The Parliament shall pass statutes. It shall monitor the action of the Government. It shall assess public policies.

Article 24, French Constitution

Cost estimation of legislative amendments for parliamentarians

Built on **OpenFisca**, an open-source micro-simulation engine

LexImpact Team

Started in 2019 by 3 persons

Now 5 Full-Time Equivalent

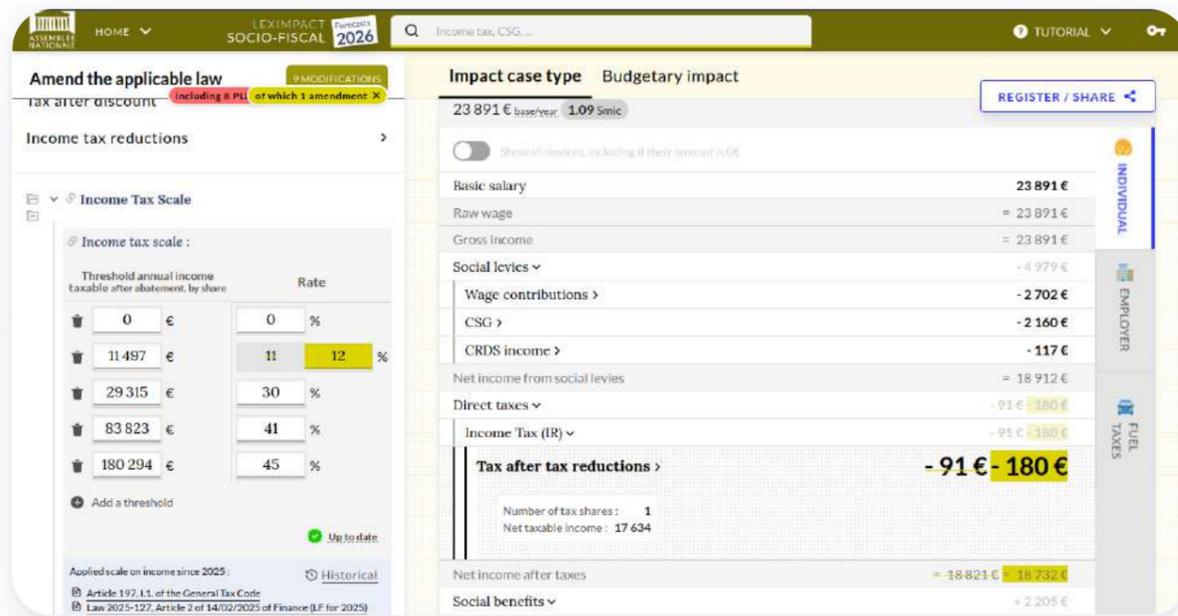
Multidisciplinary team of:

- 2 Economists
- 3 Developers (Front & Back)
- 1 Designer
- 1 Data Scientist

LexImpact's Missions: Build user-friendly online tools designed for parliamentarians, open to all citizens

LexImpact's HomePage

LexImpact's Online Tools



Income tax simulator for amendments

Applied Rules-as-Code

User-friendly online tools designed for parliamentarians to estimate the fiscal impact of their amendments.

OpenFisca-France

Open-source micro-simulation engine encoding French tax and benefit legislation into computable rules.

All Open Source

The simulator, the model, and the AI tools — everything is publicly available.

ASSEMBLÉE NATIONALE
HOME
LEXIMPACT SOCIO-FISCAL
Forecasts 2026
TUTORIAL

Amend the applicable law 9 MODIFICATIONS

tax after discount including 8 PL of which 1 amendment

Income tax reductions >

Income Tax Scale

Income tax scale :

Threshold annual income taxable after abatement, by share	Rate
0 €	0 %
11 497 €	11 % 12 %
29 315 €	30 %
83 823 €	41 %
180 294 €	45 %

+ Add a threshold

Up to date

Applied scale on income since 2025 : Historical

Article 197, I.1. of the General Tax Code
Law 2025-127, Article 2 of 14/02/2025 of Finance (LF for 2025)

Impact case type Budgetary impact

REGISTER / SHARE

23 891 € base/year 1.09 Smic

Show all devices, including if their amount is 0€

Basic salary	23 891 €
Raw wage	= 23 891 €
Gross income	= 23 891 €
Social levies >	- 4 979 €
Wage contributions >	- 2 702 €
CSG >	- 2 160 €
CRDS income >	- 117 €
Net income from social levies	= 18 912 €
Direct taxes >	- 91 € - 180 €
Income Tax (IR) >	- 91 € - 180 €
Tax after tax reductions >	- 91 € - 180 €
Number of tax shares : 1 Net taxable income : 17 634	
Net income after taxes	= -18 821 € = -18 732 €
Social benefits >	+ 2 205 €

INDIVIDUAL

EMPLOYER

FUEL TAXES

What Is a Parameter in OpenFisca?

Threshold annual income taxable after abatement, by share	Rate
0 €	0 %
11 497 €	11 12 %
29 315 €	30 %
83 823 €	41 %
180 294 €	45 %

Applied scale on income since 2025 :
Article 197, I.1. of the General Tax Code
Law 2025-127, Article 2 of 14/02/2025 of Finance (LF for 2025)



A structured value used by the model



Tax rates, brackets, ceilings, thresholds



Each needs a value, a legal reference, and a review date



Changes reflect official legislation

```
parameter.yaml
```

```
sample_rate:
  description: Percentage of the reference daily wage
  values:
    2020-10-01:
      - value: 0.50
    2025-10-01:
      - value: 0.57
  metadata:
    last_review: "2025-09-01"
    references:
      2020-10-01:
        - title: "Art. 197 of the Code..."
          href: "https://www.legifrance.gouv.fr/codes/
```

Why Automate Retrieval?

2,000+

Parameters to keep up to date

Only for French tax & social benefits for households



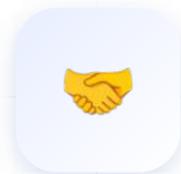
Manual updates miss changes

Leading to inaccuracies and loss of trust

The Parameters Drive Everything



Calculation Accuracy



User Trust



Legal Compliance

Project History

- 2023**
First LLM trials
5% success — models had only 4K token windows
- 2024**
Agent tooling
For parameters without legal references
- Early 2025**
99% success
With modern models like DeepSeek-R1
- 2025**
UI for updates
Better workflow with a desktop application
- 2026**
Full code generation
Generating legislative code from bill text

AI Context Window Growth



Example: [Article 199 undecies B](#) = 8,922 tokens — wouldn't fit in 2023 models!

AI Evolution

2023 models only have 4,000 input tokens size, they now handle 1 million.

- A token is roughly 4 characters
- The input tokens are filled with: law text, parameter file, prompt instructions, ...
- **2023 models could not handle some law text!**

Example: Article 199 undecies B

8,922 tokens - would require multiple API calls with older models



Rules as Code 2026 in Den Haag

Evaluation Dataset

It is mandatory to have a dataset

- We use **165 parameters** with known values and legislative references
- Used to benchmark model performance

Evaluation Workflow

- Same prompts and parameters for all LLM models
- Ensures comparability
- But not enough : models react differently to the same prompt.

Sample Evaluation

Description from OpenFisca file	Law Text	Expected Value
Percentage of the reference daily wage (SJR) for the basic unemployment insurance allowance	When the resulting amount is less than 57% of the reference daily wage,	0.57

Prompt Structure (Evaluation)

- Explain the goal of the task
- Provide the description of the parameter
- Provide the legal reference
- Ask for the value of the parameter

Evaluation Results

Model	Success Rate	Time for dataset	Price
deepseek-ai/DeepSeek-R1	90 % to 99% *	1 heure 43 minutes	N/A
GPT-4	92.9%	20 minutes	N/A
Llama-3.3-70B-Instruct-Turbo-Free	91.45%	9 minutes	N/A
deepseek-ai/DeepSeek-V3	89.74 %	15 minutes	N/A
ministral-medium-latest (08/03/2026)	86 %	4,5 minutes	0.31 €
GPT-5.4	86 %	4.7 minutes	1.85 €
ministral-large-latest (08/03/2026)	85 %	4.7 minutes	0.39 €
ministral-3b-latest (08/03/2026)	70 %	1.5 minutes	0.07 €

* Multiple runs. Seed vary so results are not exact

Results Available At

git.leximpact.dev/leximpact/exploration/fiscal-qa

Workflow

Have changed since the beginning of the project

Initially: Batch process - review all parameters monthly

Reality: More complex

- Many parameters lack references
- Need to customize AI output: fix errors, adapt legal references, manual changes
- Human reviewers focus on coherent parameter groups, not big batches or one-by-one

AI changes need to be reviewed by another human before being merged in the main branch of OpenFisca-France.

Desktop Application for Parameter Updates

Web → Desktop for better integration

Why desktop?



Direct access

Access the OpenFisca working folder directly



Instant preview

See AI changes in VSCode immediately



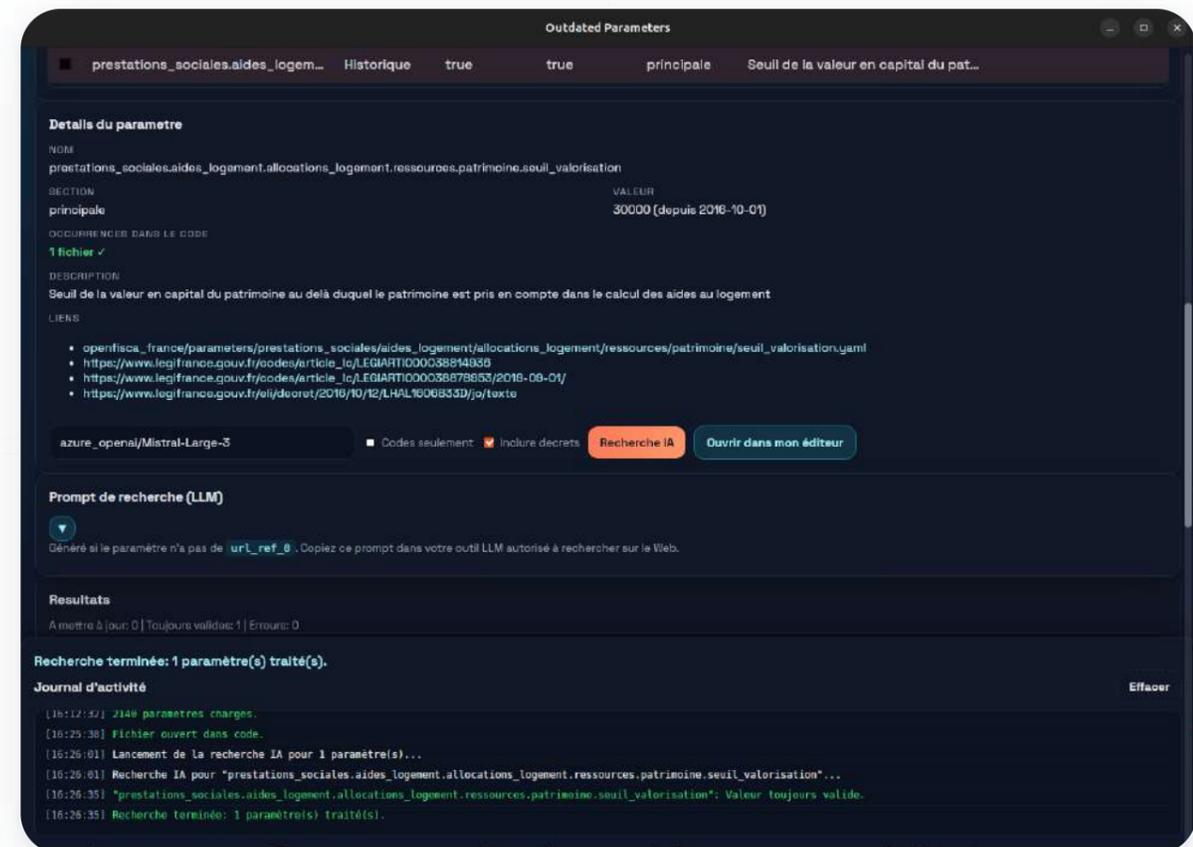
Edit before submit

Adapt changes before committing



Git integration

Submit changes through the normal code review process



Parameter update interface

Live Demo

UI for parameter updates

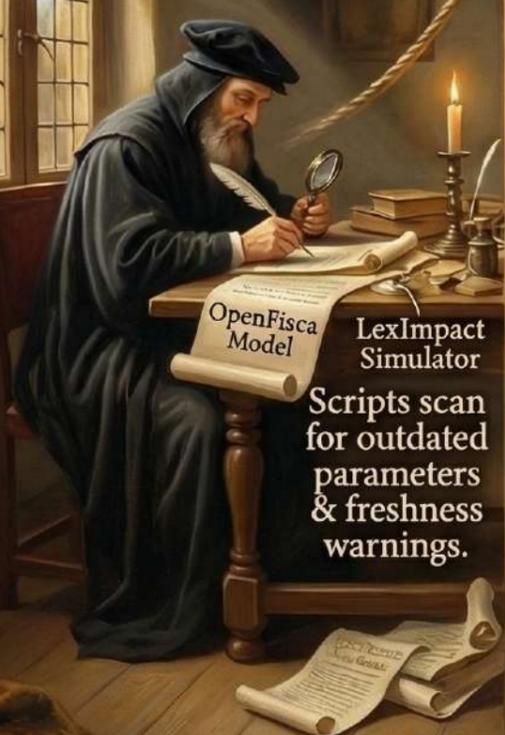


Live Demonstration

Updating a group of parameters using the desktop application with AI-assisted value extraction from Légifrance.

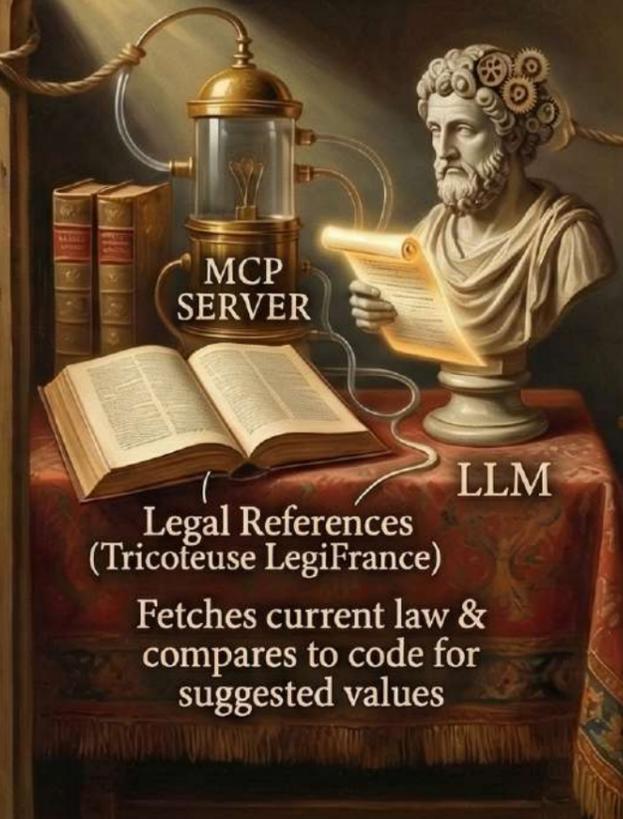
From Law to Code: The AI Pipeline

1. BATCH EXTRACTION & MONITORING



OpenFisca Model
LexImpact Simulator
Scripts scan for outdated parameters & freshness warnings.

2. AUTOMATED LEGAL VERIFICATION



MCP SERVER
LLM
Legal References (Tricoteuse LegiFrance)
Fetches current law & compares to code for suggested values

3. HUMAN-IN-THE-LOOP REVIEW



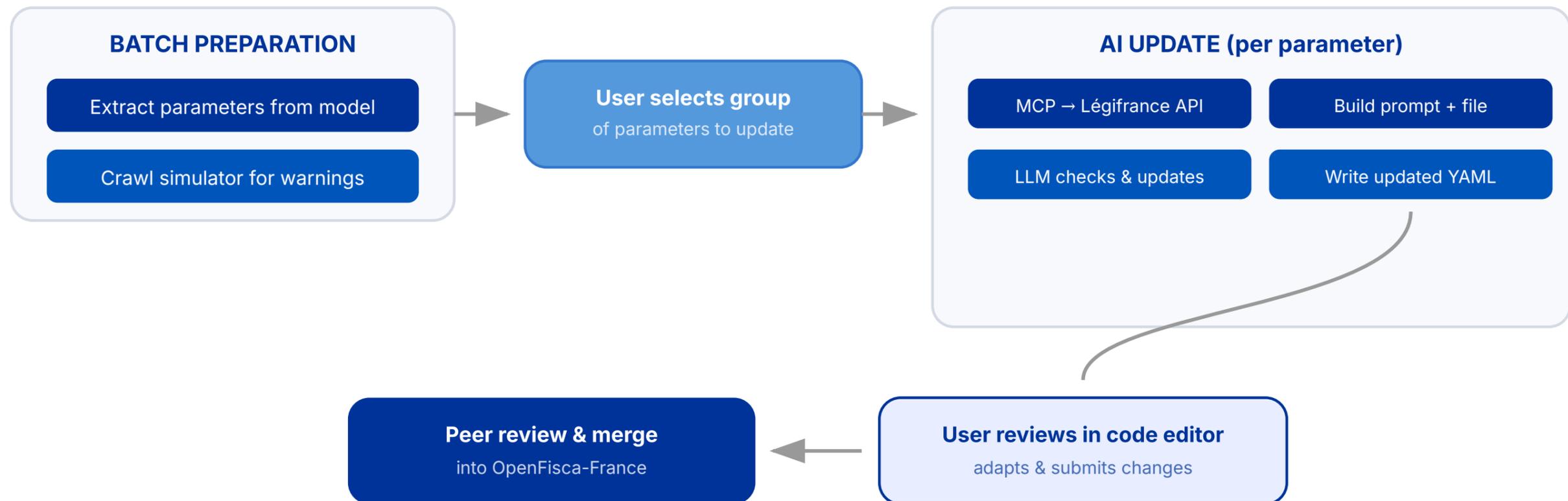
VSCode
Developers refine AI suggestions before submission.

4. COMMUNITY INTEGRATION



OpenFisca-France Codebase
Peers review updates before merging

Workflow for Parameters with References



Prompt Structure (Extraction)

FULL PROMPT

CONTEXT

Detailed explanation of what we want (always the same)

PARAMETER

Description of the parameter and previous value

LAW

Last version of all law articles linked in the parameter's metadata

When It Fails

Understanding failure modes is key to building trust



Ambiguous law text common

Legal language can be interpreted multiple ways, leading to incorrect value extraction.



Wrong legal reference common

The reference points to the wrong article, so the AI extracts a value from the wrong source.



Outdated decree blocking

The government publishes a new decree instead of updating the old one — no automatic link.



Source outside Légifrance blocking

Some parameters come from PDFs, state agency websites, or other non-indexed sources.



Model hallucination dangerous

The LLM invents a plausible but incorrect value, sometimes with a confident explanation.



Missing reference common

No reference at all — need to find the right article before extraction can begin.

Agent Mode for Missing References

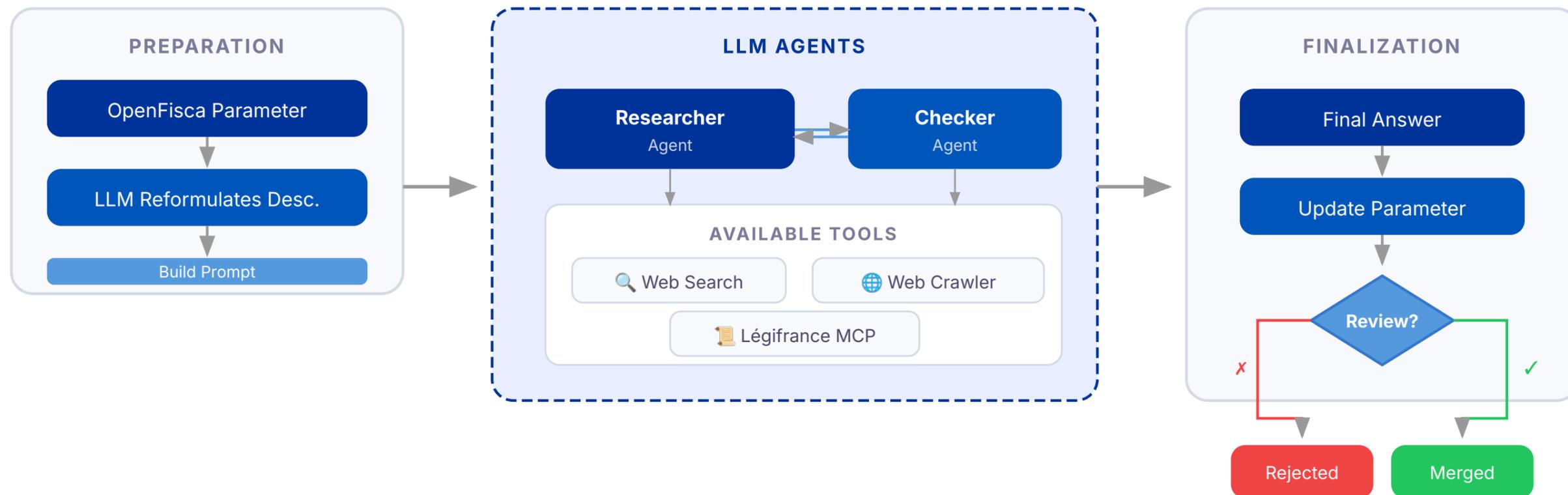
- Handles parameters without explicit legal references
- Uses model context to infer relevant updates
- Shifts focus from search to structured review

Implementation

- Built our own agent workflow with LangChain
- Seems less necessary now with agent capabilities in chat interfaces
- Still useful for complex workflows with specific process requirements

Agent Mode for Missing References

A multi-agent pipeline built with LangChain



Why Skills Help

A simpler evolution of AI workflows



Browse the OpenFisca model

Skills allow AI to navigate the codebase semantically



Better semantic grounding

Improves updates of historical brackets with domain context



No code required

Domain experts write skills themselves — no developer needed

Inspiration from PolicyEngine

UK/US non-profit that built their micro-simulation model from OpenFisca. Strong use of Claude Code Skills.

> Raise the CTC to \$3k. Impact on a family of 4 earning \$55k?

CTC: \$4,400 → \$6,000

Net change: +\$1,280/yr (+\$107/mo)

90 min For a complete agentic workflow
vs. 2 weeks without AI for PolicyEngine

2026: From Text to Code

The next frontier: full code generation from the finance bill

New Capability

- ✓ Generate OpenFisca Python code directly from the finance bill
- ✓ LF 2026 example: tax update implemented in one shot
- ✓ Complex features like the accompte were handled correctly

FULL PROMPT

GOAL

I want to code a reform in OpenFisca-France

CONTEXT

Current legislative text

CHANGE

Proposed amendment



Live demo: AI coding a new law in one shot

Sovereignty & Open Source



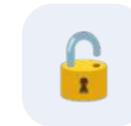
Public Institution Interest

Strong interest in using open-source models to ensure transparency and control over our tools.



Journey to Model-Agnostic

Started with the goal of open-source → developed with proprietary models while waiting → now model-agnostic.



Switch Anytime

Architecture allows switching to open models as they become competitive. No vendor lock-in.

As a parliamentary institution, **transparency and control** over AI tooling is not optional — it's a democratic imperative.

Cost of AI

On-premise deployment

Open-weight model "Mistral-Large-3-Instruct FP8" needs a 8× NVIDIA H200 server at 300,000 €

Better option: API calls

Pay-per-use through cloud providers or OpenRouter (10€ one-time for testing).

Model (2026)	\$/1M input tokens
Mistral Large 3	€0.50
Ministral 3 — 14B	€0.20
GPT-5.2 Pro	\$21.00
GPT-5.2	\$1.75
GPT-5-mini	\$0.25

~100 €

Estimated monthly cost

For 2,000 parameters with a modern model

Other options

EuroHPC

Free access to huge computers for EU public institutions. Limited to offline training or batch inference. We used it for fine-tuning.

Environmental cost

Estimate with CodeCarbon or EcoLogits. AI has an environmental footprint that must be considered.

Note: Agentic AI sends much more tokens per task than simple prompts. Factor this into cost planning.

Takeaways

Key Insights



AI is great, not magical

It requires a lot of work to build a reliable workflow



Large-scale updates are now feasible

AI made 2,000+ parameter updates practical



Tooling is mandatory

For accuracy and reproducibility at scale

Next Steps

- Optimize the review process for human reviewers
- Share common AI practices with the RaC community
- Automate non-parametric reforms

⚠️ **Keep Human in the Loop**

Human review is mandatory for accuracy and trust. It will become harder to maintain as models improve — reviewers will lower their guard when errors become rare.

Thank You

Questions & Discussion

Benoît Courty — LexImpact Team

Assemblée nationale de France

`git.leximpact.dev/leximpact/exploration/update-openfisca-with-ai`
