PROOF

Future States

Open Tax Strategies: Let the machine figure it out

Challenge: Sophisticated tax strategies are out of reach for most

Most of us intuitively realize that taxes are for 'common people'—wage earners are taxed higher than investors, high-net-worth individuals have easy access to family offices, off-shoring strategies, and tax law experts. This trend (and its impact) only increases as the tax code grows in complexity and size: anything outside the bounds of consumer products like TurboTax grows less and less accessible to individuals or businesses that are complex enough to lose out, but have yet to accumulate enough wealth to justify premium services.

Opportunity: Tax-Law-as-Code

One might be tempted to think this problem is solvable with emerging AI models, but they suffer from myriad problems when the results need to be precise to an individual. And as we all know, taxes need to be *precise*. The key insight to solving this problem is the realization that *law* and *code* are basically the same thing: code is rules for software; law is code for people. The parallel carries through to very foundational levels: law has constants, variables, loops, conditionals, even recursion—just like programming languages do.

A completely new programming language, Catala, aims to turn these parallels into working software. Created by the French National Research Institute for Computer Science, Catala allows legalese to be written as executable code which can be evaluated. This code is expressive enough to model any contract or piece of legislation, including the US Tax Code.

Proof's goal is to transcribe the entirety[1] of the US Tax Code in Catala, and begin releasing automated tools to leverage it, so that anyone can not only programmatically ensure their compliance with tax laws, but automatically discover new financially advantageous tax strategies.

Design

[1] 6,781 pages as of Oct 4, 2023

The language itself is straightforward—even more readable than the Tax Code itself (not to say that's a high bar). Here's Section 121(b)(2)(a), the '$500,000 limitation for certain joint returns', per the Tax Code:

> *Paragraph (1) shall be applied by substituting "$500,000" for "$250,000" if—*
> *either spouse meets the ownership requirements of subsection (a) with respect to such property;*
> *both spouses meet the use requirements of subsection (a) with respect to such property; and*
> *neither spouse is ineligible for the benefits of subsection (a) with respect to such property by reason of paragraph (3).*

And here it is in Catala:

```
scope Section121TwoPersons:
  rule section_121_b_2_A_condition under condition
  (return_type with pattern JointReturn)
  and
  # i)
  (section121Person1.requirements_ownership_met or
     section121Person2.requirements_ownership_met)
  and
  # ii)
  (section121Person1.requirements_usage_met and
     section121Person2.requirements_usage_met)
  # iii)
  and
  (not (section121Person1.section_121_b_3_applies))
  and
  (not (section121Person2.section_121_b_3_applies))
  consequence fulfilled

  exception
  rule section121a_requirements_met under condition
    section_121_b_2_A_condition
  consequence fulfilled

  exception
  definition gain_cap under condition
    section_121_b_2_A_condition
  consequence equals $500,000
```

If nothing else, cross-references are replaced with easy-to-follow names.