Property Law as a Programming Language

Shrutarshi Basu, James Grimmelmann & Nate Foster

Cornell University & Cornell Tech
Property Law as a Programming Language

Shrutarshi Basu, James Grimmelmann & Nate Foster

Cornell University & Cornell Tech
Deed of Conveyance

Multiple types of deeds
Differing requirements by jurisdiction
Short documents with standardized language
Some Examples

O conveys $P$ to A.

O conveys $P$ to A for life.

O conveys $P$ to A until A graduates.

O conveys $P$ to A for life, then to B.

O conveys $P$ to A for life, then to B, but if C marries to C.
Dude, where’s my \( \lambda \)?
O conveys P to A until A graduates, then to B.

O conveys P to A, but if A graduates to B.
O conveys P to A until A graduates, then to B.

O conveys P to A, but if A graduates to B.
Distinguishing Limitations

- Determinable Estate (immediate transfer)
  - phrased as measure of the duration of the estate until, so long as, while, during
  - placed before the punctuation mark signaling the end of the description

- Estate Subject to a Condition Subsequent (requires claim)
  - phrased like an afterthought but if, provided that, however
  - placed after the punctuation mark signaling the end of the description
Conveyances as a DSL

- Well-defined(-ish) syntax
  - People, properties, conditions, keywords, punctuation.

- Rich language of events and conditions
  - dies, graduates, survives, marries vs. is married

- Automata model for possession
Conveyances as a DSL

- Model semantics precisely
  - How does possession change with different events?
- Verify properties of conveyances
  - Loop-freedom, "Rule Against Perpetuities"
- Synthesize (or repair) conveyances with certain properties
O conveys P to **A for life**, then to B, but if **C marries** to C.

1. From 0 to A
2. From A to B
3. From B to C

1. A dies
2. C marries
O conveys P to A for life, then to B for life, then to C if C is married.
Applications

- Querying
  - “What sequence of events allows X to gain possession?”
  - “Who gains possession after S sequence of events?”
  - Pedagogical tool for law students

- Verification
  - No person can convey more than they own
  - Rule Against Perpetuities

- Extraction to smart contracts?
  - Generate programs from conveyances
Try It Out!

http://basus.me/conveyor

basus@cs.cornell.edu