Property Conveyances as a Programming Language

Shrutarshi Basu
Harvard University

Nate Foster
Cornell University

James Grimmelmann
Cornell Tech
Cornell Law School

CORNELL
NYCTECH

SPLASH Onward! Oct. 24, 2019

Disclaimer

I am not a lawyer

(and I don't play one on TV*)

and this presentation (& paper) does not constitute legal advice

*though if you have entertainment industry contacts, let me know

Property Conveyances as a Programming Language

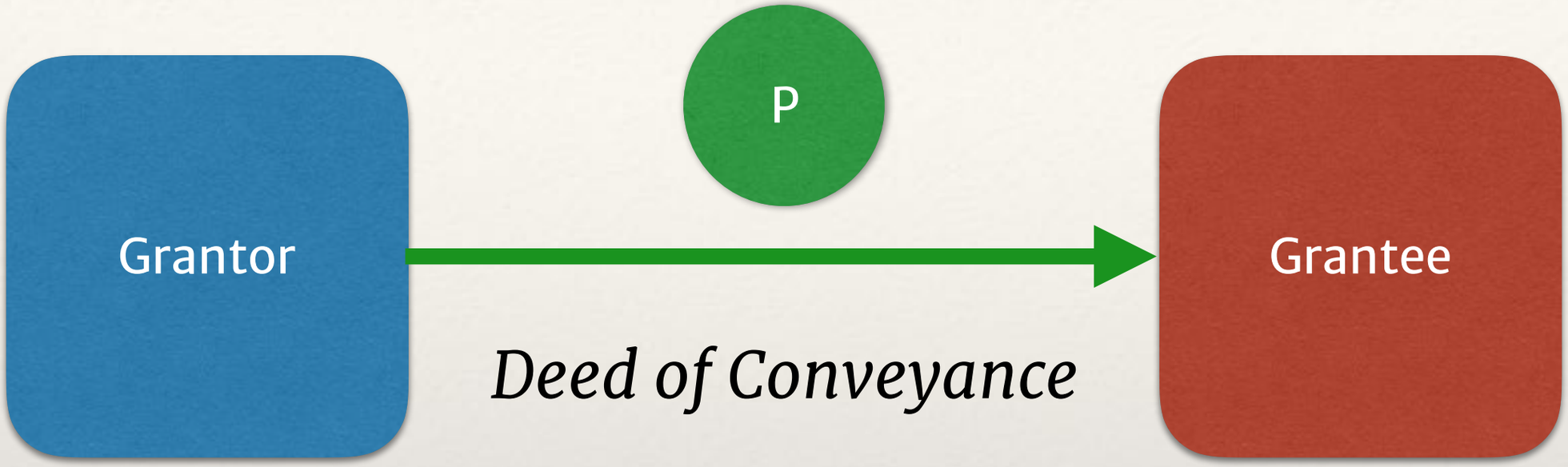
Shrutarshi Basu
Harvard University

Nate Foster
Cornell University

James Grimmelmann
Cornell Tech
Cornell Law School

CORNELL
NYCTECH

SPLASH Onward! Oct. 24, 2019



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

The right to possess the property is called an *interest.*
Possessory if owner has right to possess *now*, otherwise, *future interest.*

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.

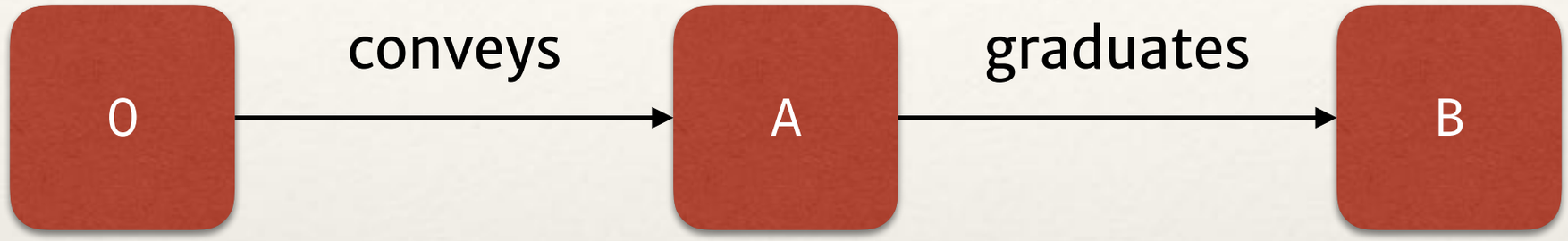
O conveys P to A for life, then to B, but if C marries to C.

Dude,
where's my
λ
?

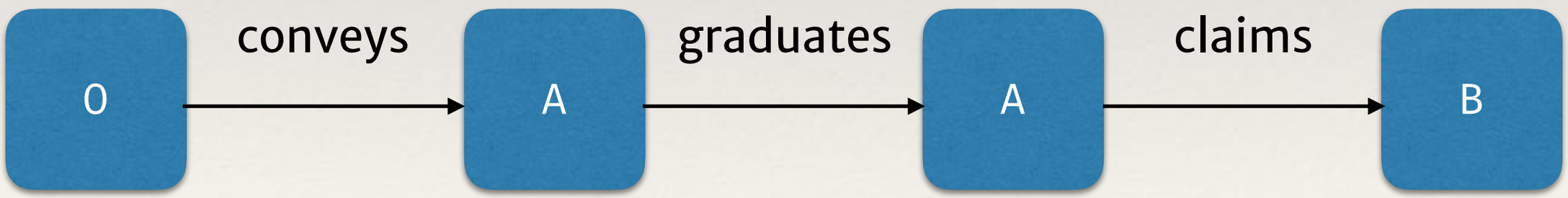
O conveys P to A until A graduates, then to B.

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.



Example: 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 the punctuation mark signaling the end of the description

Ideal for a DSL

- Well-defined(-ish) syntax
  - People, properties, conditions, keywords, punctuation.
- Rich language of events and conditions
  - dies, graduates, survives, marries vs. is married
- Deterministic, agreed-upon rules

What we did

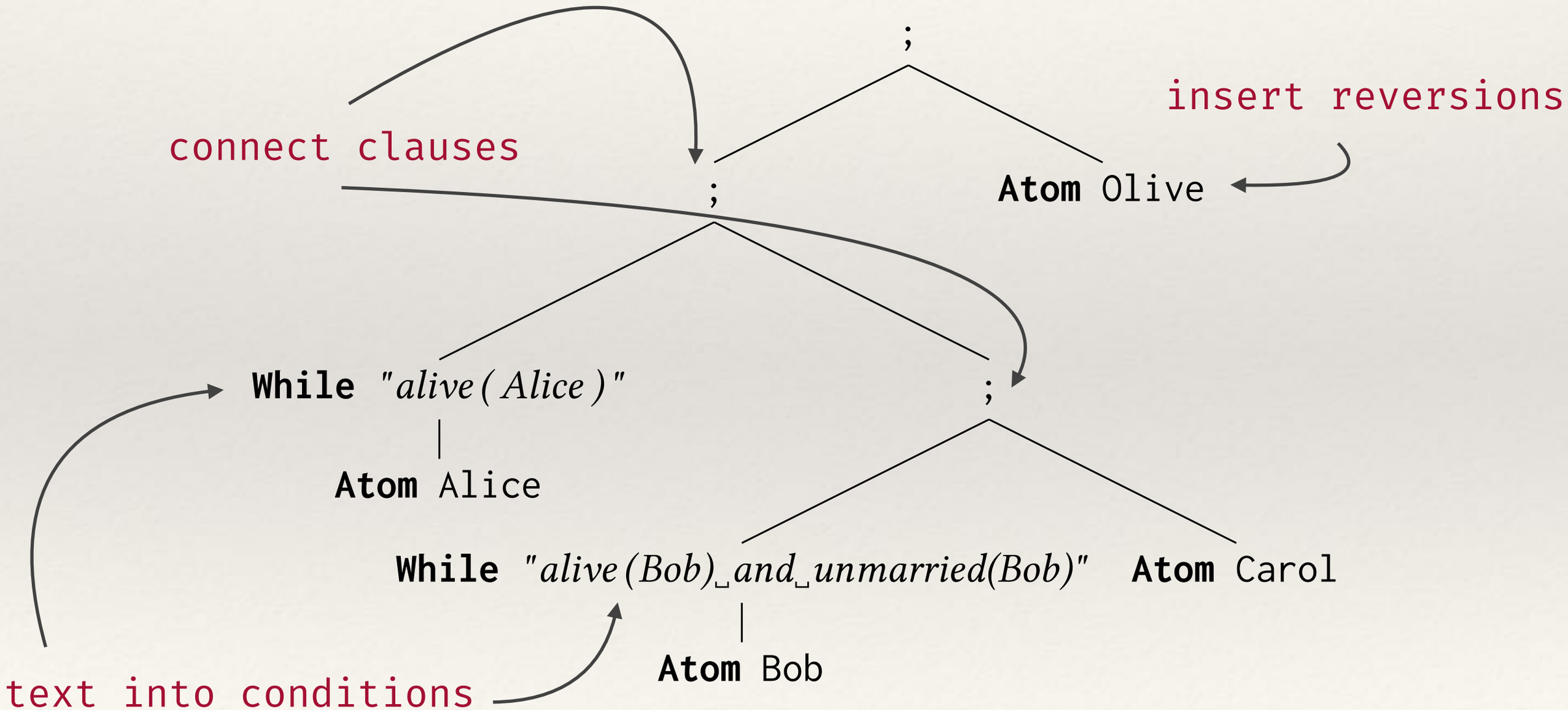
- Develop a core calculus for conveyances
- Translate surface syntax into a core calculus
- Operational and denotational semantics
- Prove equivalence between the semantics
- Prove that semantics obey property law principles
- Implementation and evaluations

```
Olive owns.
Olive conveys to Alice for life, then to Bob
    for life until Bob marries, then to Carol.
Alice dies.
Bob conveys to Dave for life.
Bob marries.
```

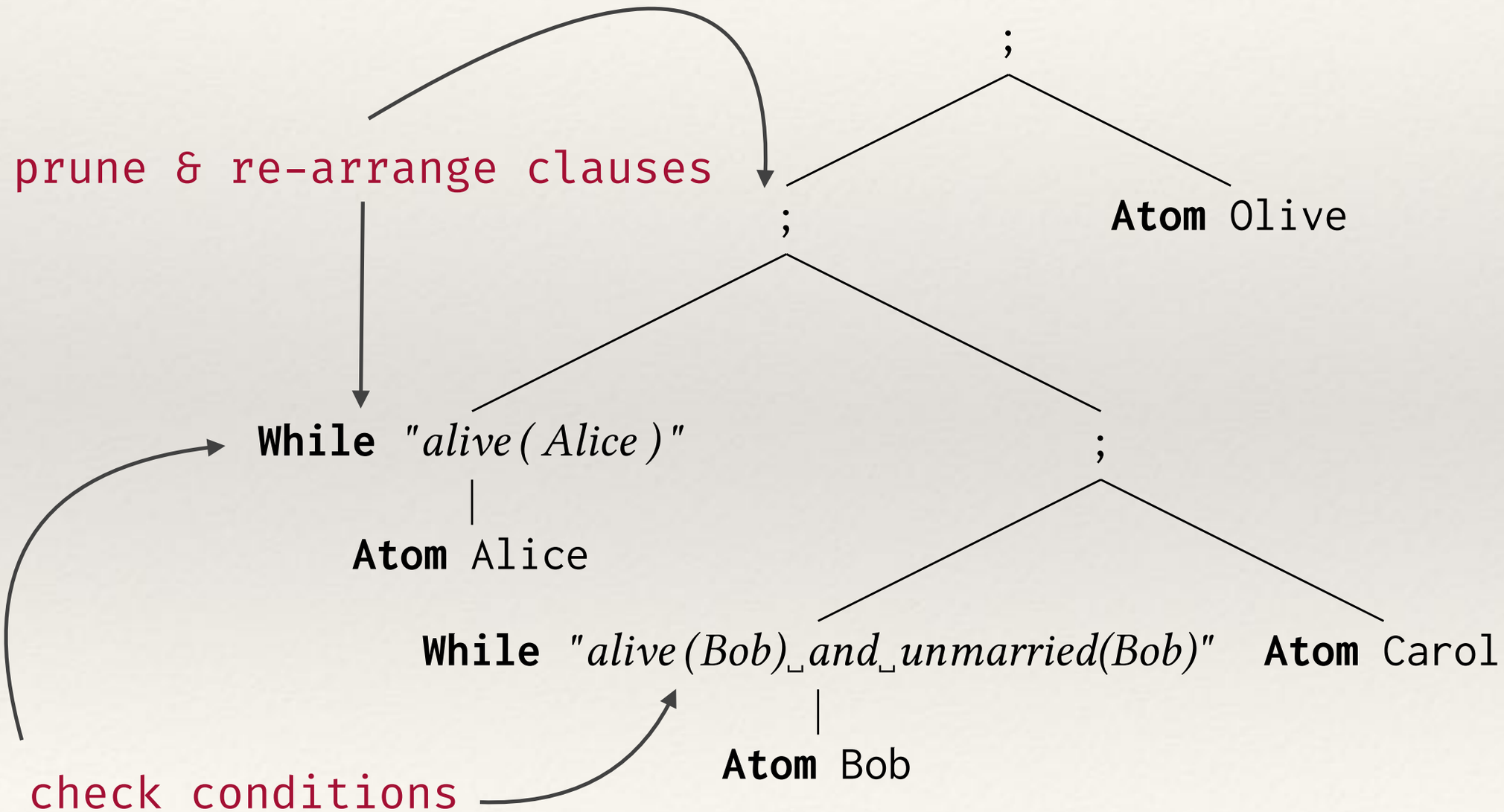
1. Olive owns.

Atom Olive

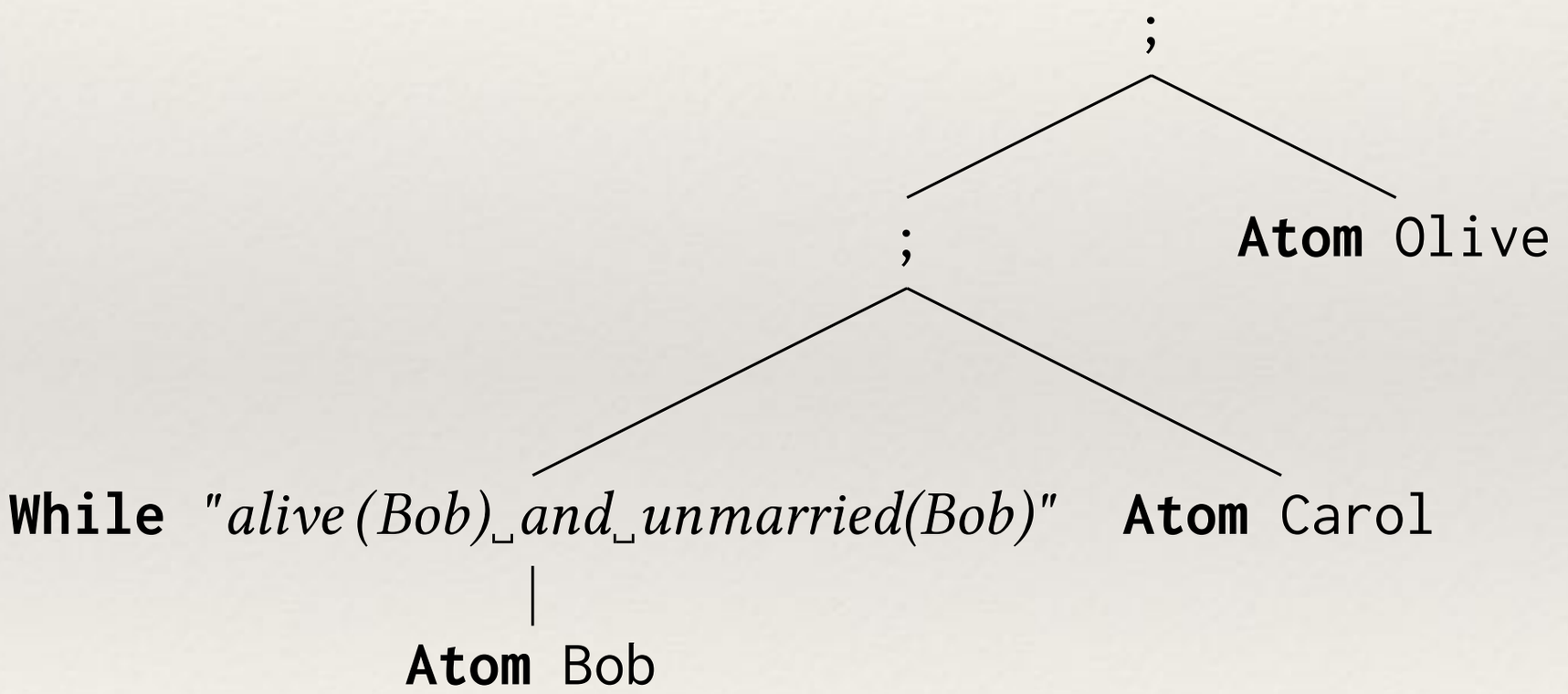
```
Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
```



Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
Alice dies.

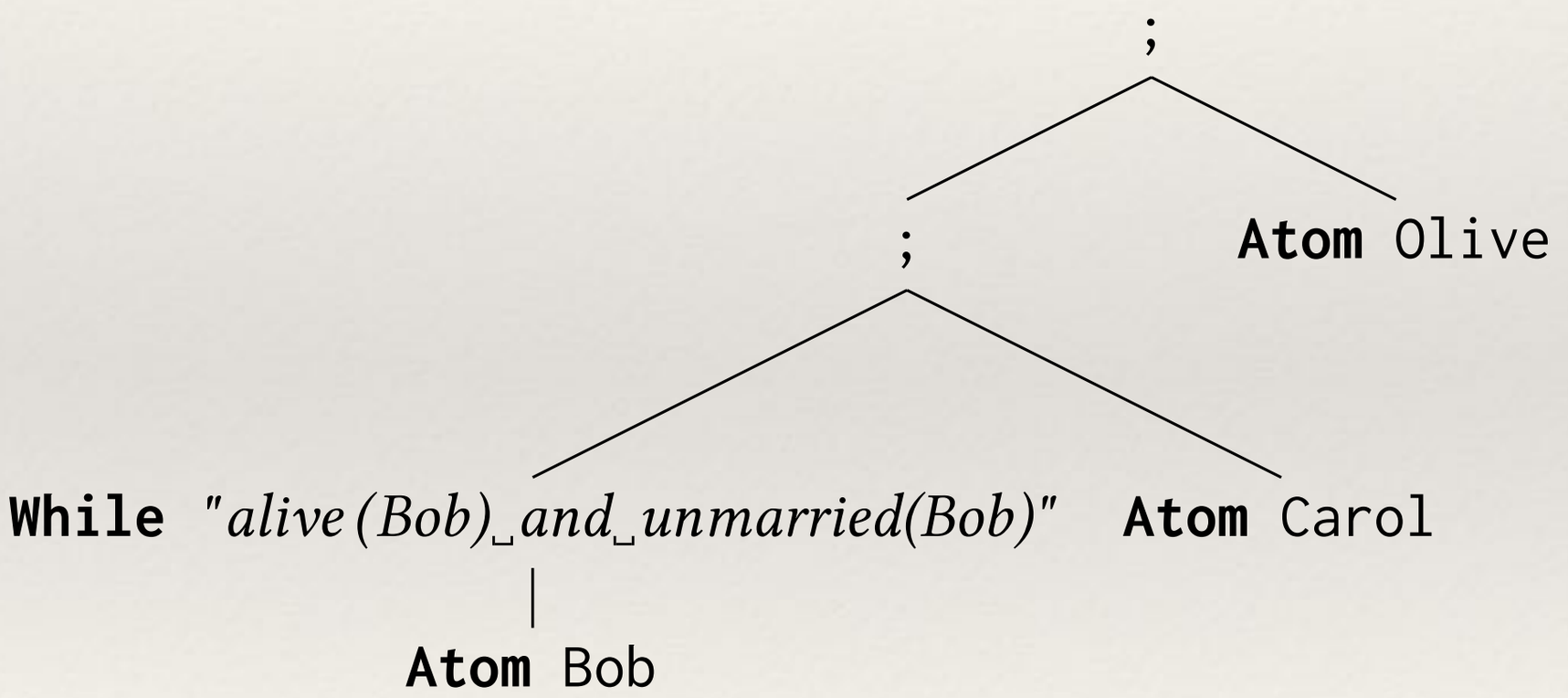


```
Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
Alice dies.
```

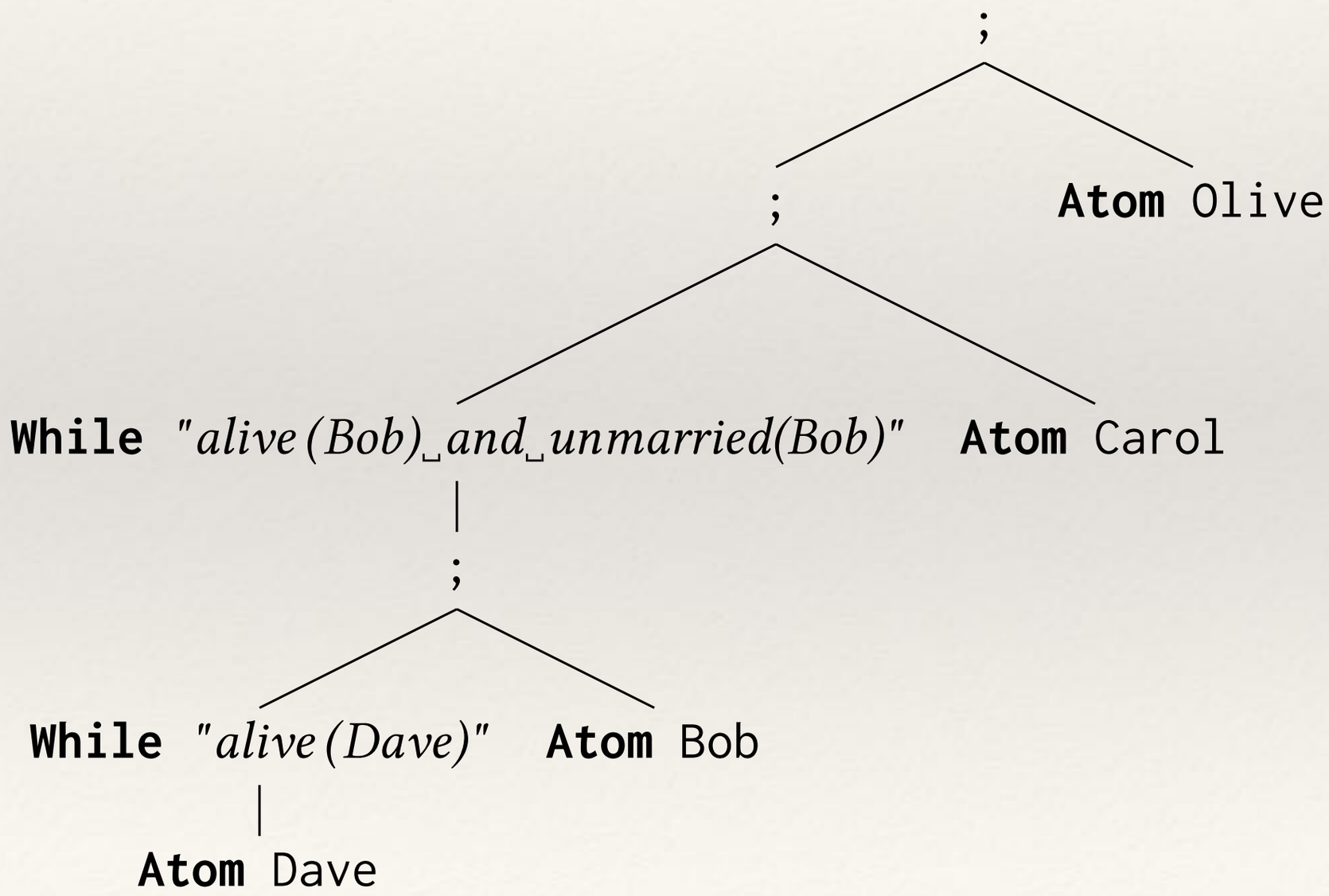


```
Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
Alice dies.
```

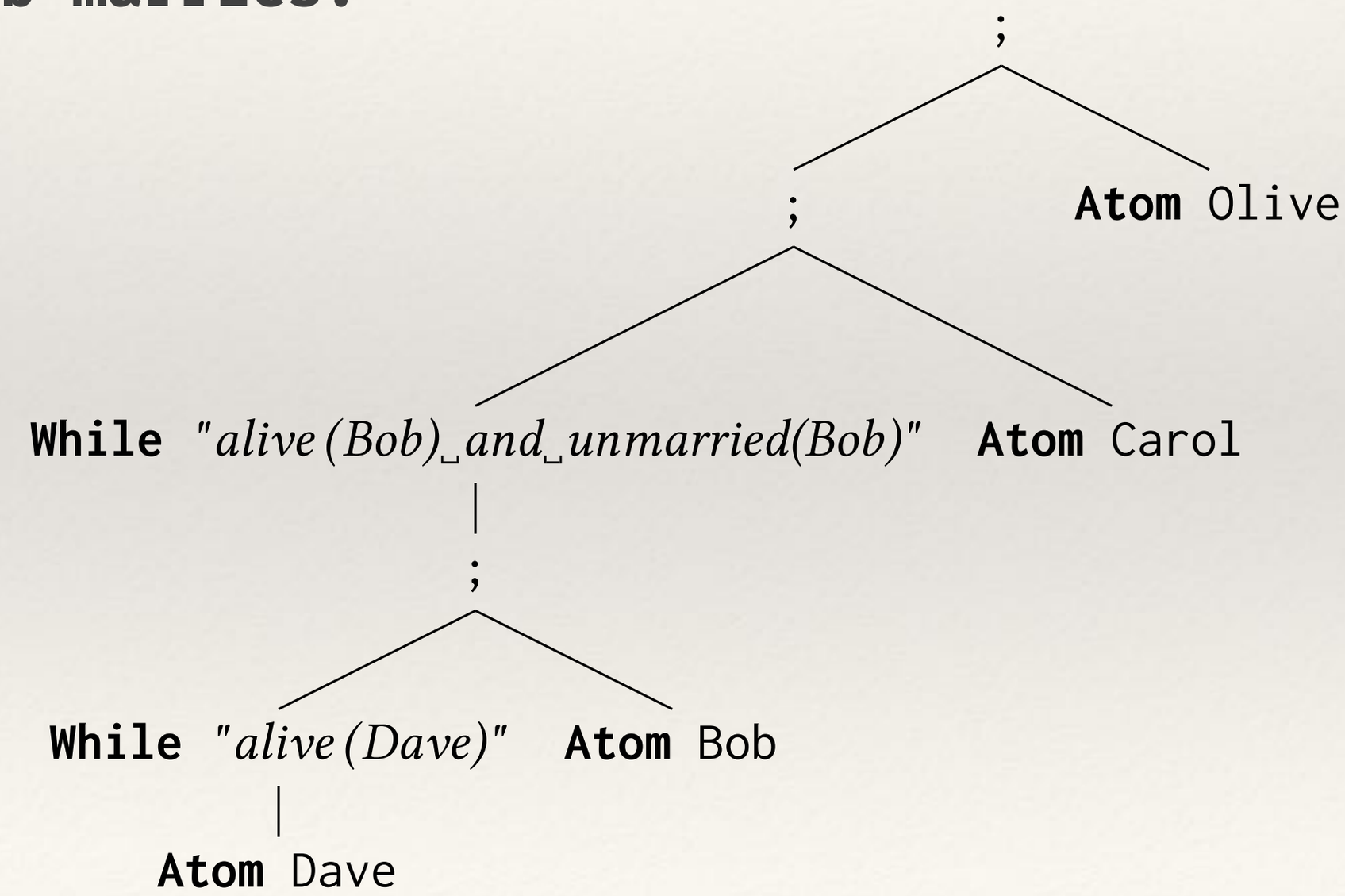
Bob conveys to Dave for life.



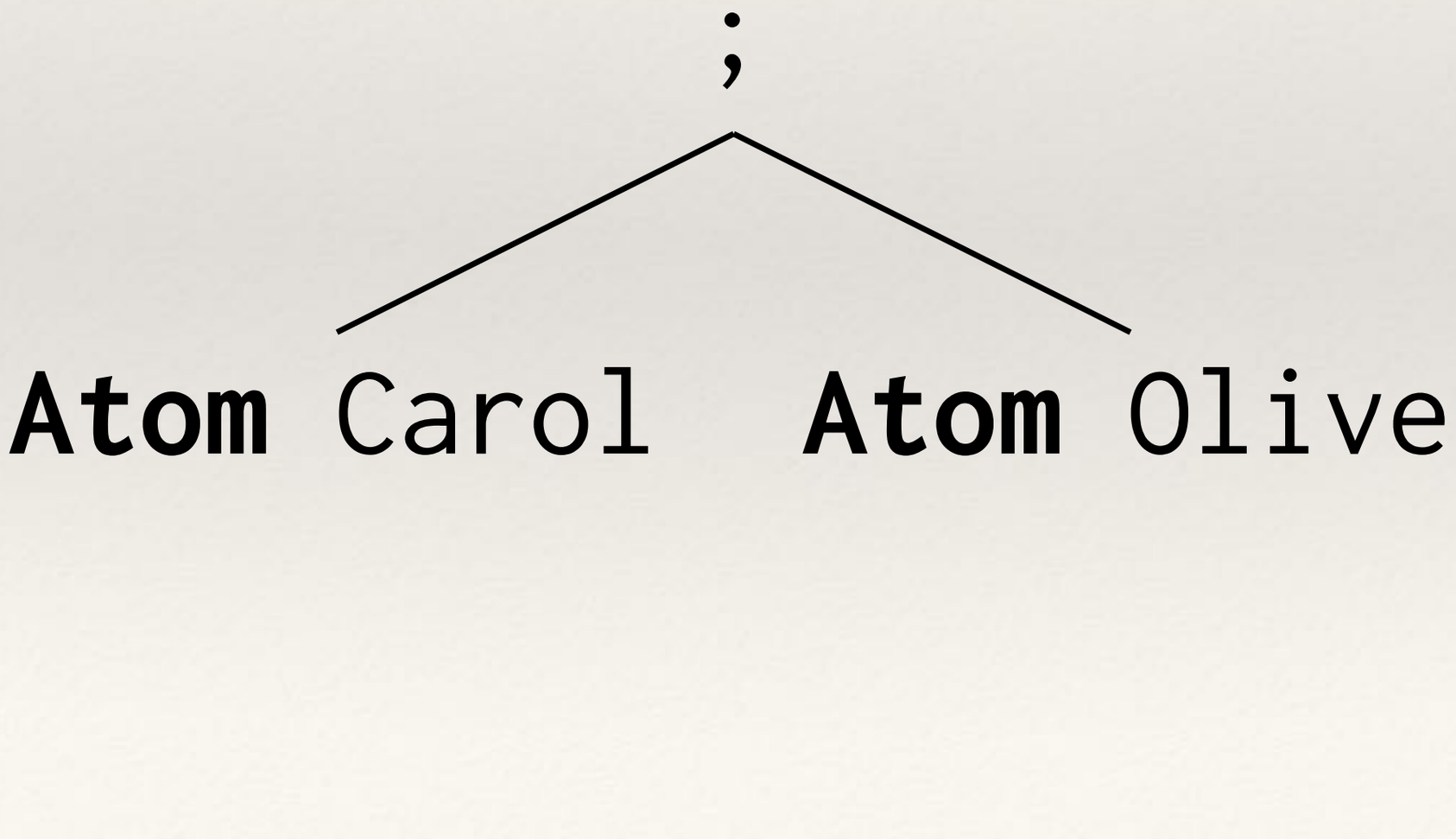
```
Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
Alice dies.
Bob conveys to Dave for life.
```



```
Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
Alice dies.
Bob conveys to Dave for life.
Bob marries.
```



```
Olive owns.
Olive conveys to Alice for life,
 then to Bob for life until Bob marries,
 then to Carol.
Alice dies.
Bob conveys to Dave for life.
Bob marries.
```



Core Calculus

$$
\begin{array}{rrll}
\text{Interest } \Gamma \ni \gamma & ::= & \text{To } (n \in \mathbb{N}, g, o \in \mathbb{P}) & \text{Transfer} \\
\\
\text{Terms } t \ni T & ::= & \text{Atom } \gamma & \text{One Interest} \\
& | & \text{Seq } (t_1, t_2) & \text{Sequencing} \\
& | & \text{While } (c, t) & \text{Termination} \\
& | & \text{If } (c, t) & \text{Precondition} \\
& | & \text{Bottom} & \text{Reversion} \\
\\
\text{Statement } S \ni s & ::= & \text{Conveys } (\gamma,\ t) \mid e & \\
\text{Program } \pi & ::= & [\ \text{Owns } p;\ s_0 \ldots s_n\ ] &
\end{array}
$$

p, but if c then $q \rightarrow$ `while(not` c, (p ; `Atom` g)); q

Operational Semantics

- Conditions as black boxes
  - $c : E* \rightarrow \{true, false\}$; stepping function
- Interleave *stepping* and *simplification*
  - *Stepping* moves conditions ahead & expands terms
  - *Simplification* removes expired terms
- *Observe* possessory interest from leftmost term
- Based on derivatives of regular expressions

Evaluation

Theorems

Implementation

Theorems

- A fee simple is perpetual and unconditional
- Ownership is always unambiguous
- *Nemo dat quod non habet*
- First in time, first in right
- Conservation of estates
- Proved using the denotational semantics

Implementation

Computational Conveyances

Choose an example or write your own

```
O owns Blackacre.
O conveys Blackacre to B until B dies.
B dies.
```

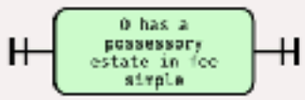
Interpret! Show trace

Examples for Onward! 2019

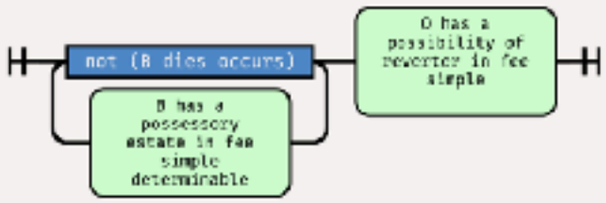
- O conveys Blackacre to A.
- O conveys Blackacre to A for life.
  A dies.
- O conveys Blackacre to A for life, then to B.
  A dies.
- O conveys Blackacre to A for the life of X.
- O conveys Blackacre to A for life, then to B for life, then to C.
- O conveys Blackacre to A until B marries.
- O conveys Blackacre to A until B marries, then to C.
- O conveys Blackacre to A, but if B marries then to C.
- O conveys Blackacre to A for life until B marries.
- O conveys Blackacre to A for life, but if B marries then to C.

See the results

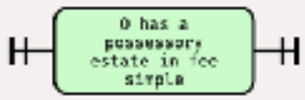
O owns the property



Conveyance by O



B dies



Implementation

- 119 examples from *Estates in Land and Future Interests: A Step-by- Step Guide* by Linda Edwards
- 104 passed without issue (26 required minor edits)
- 13 had complicated conditions; 2 required extra syntax

Why?

- It’s an interesting experiment!
- An exploration of (dis-)similarities between legal & computational reasoning
- Potential pedagogical tool for law students
- Proof assistants & explorers for legal reasoning

https://conveyanc.es