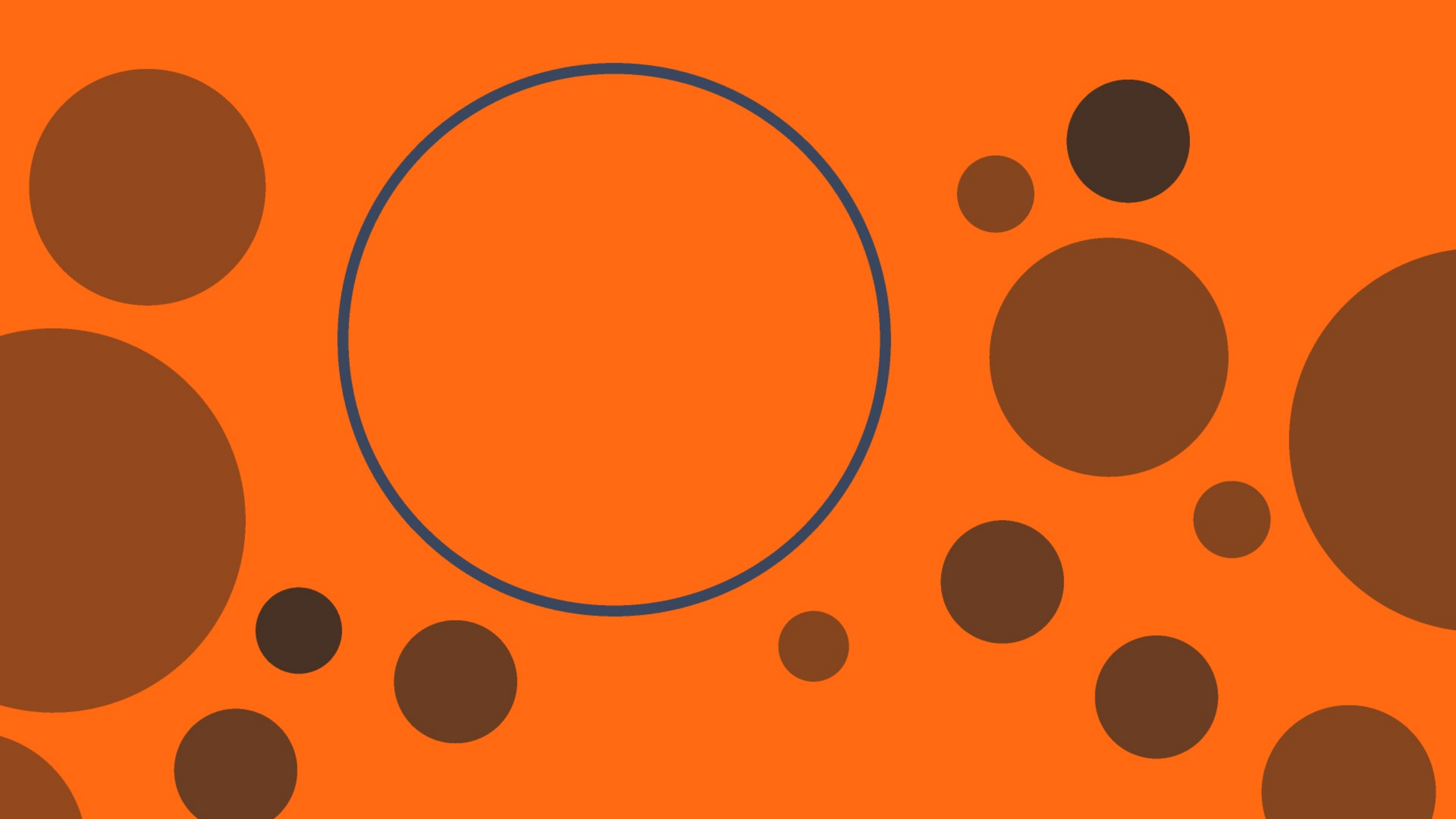


Data Parallel Proof Verification in APL

Brandon Wilson

DYALOG



What?

Formal proofs

Why?

Proofs
≡
Programs

Curry-Howard

Typed
lambda
calculus

Natural
deduction

Logic
programming



Coq
1989

Mizar
1973

Isabelle
1986

Metamath
2005

Some
existing
systems

Agda
2007

Lean
2013

Idris
2007

Strong type
systems

Pythagorean
Theorem

Central
Limit
Theorem

Formalizing 100 Theorems

Fermat's Last
Theorem

Rising popularity

①

$$2 + 2 = 2 + 2$$

$$2 = 1 + 1$$

②

$$2 + 2 = 2 + (1 + 1)$$

$$x + (y + z) = (x + y) + z$$

$$3 = 2 + 1$$

③

$$2 + 2 = (2 + 1) + 1$$

④

$$2 + 2 = 3 + 1$$

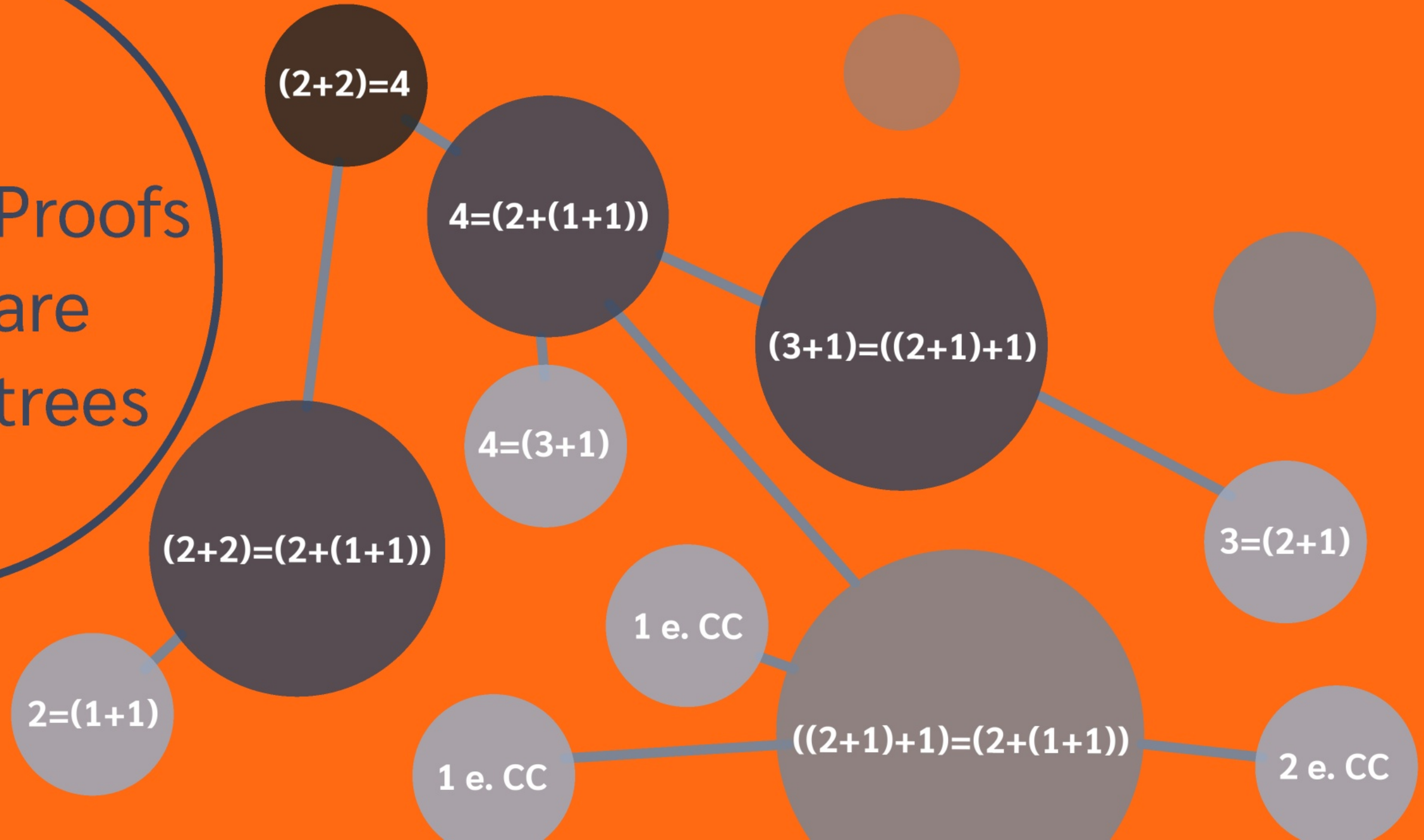
⑤

$$2 + 2 = 4$$

$$2 = 1 + 1$$

A proof
primer

Proofs
are
trees



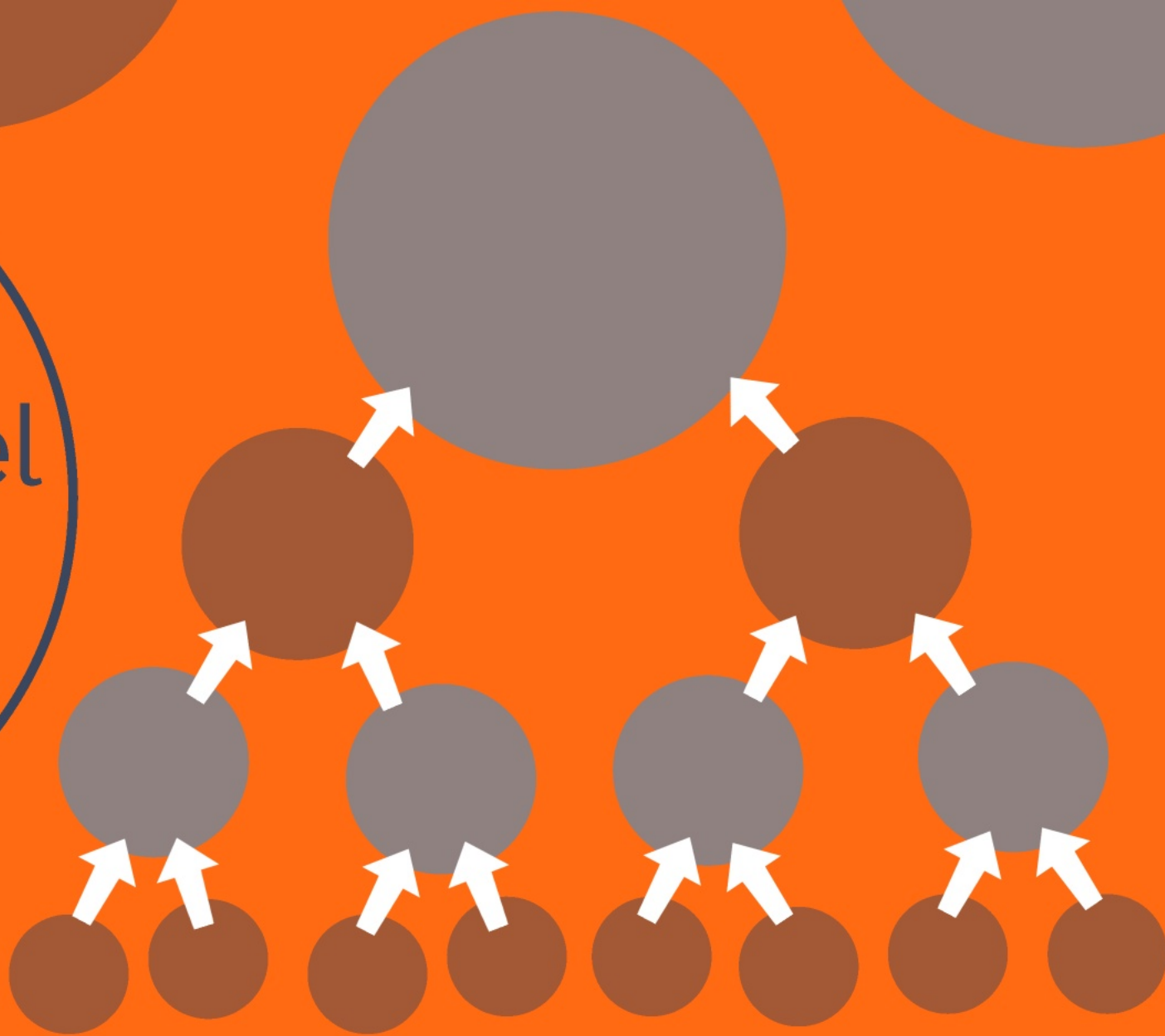
Again, in Metamath

```
17      oveq1i.1=df-2      $a |- 2 = ( 1 + 1 )
18  eqtr4i.1=oveq2i      $p |- ( 2 + 2 ) = ( 2 + ( 1 + 1 ) )
32      3eqtri.1=df-4      $a |- 4 = ( 3 + 1 )
37      oveq1i.1=df-3      $a |- 3 = ( 2 + 1 )
38      3eqtri.2=oveq1i    $p |- ( 3 + 1 ) = ( ( 2 + 1 ) + 1 )
42      axi.1=2cn          $p |- 2 e. CC
43      axi.2=ax-1cn       $a |- 1 e. CC
44      axi.3=ax-1cn       $a |- 1 e. CC
45      3eqtri.3=addassi   $p |- ( ( 2 + 1 ) + 1 ) = ( 2 + ( 1 + 1 ) )
46  eqtr4i.2=3eqtri      $p |- 4 = ( 2 + ( 1 + 1 ) )
47  2p2e4=eqtr4i        $p |- ( 2 + 2 ) = 4
```

So, how do
we treeify?

$d \leftarrow + \setminus 1$ -arity A stack depth
 $(\iota \neq d) \{ (0, \omega) [\omega \iota \alpha] \} \circ \iota'' \downarrow \circ . \geq \sim \phi d$
 $(\iota \neq d) (\iota \sim [0, \vdash) \circ \iota'' \downarrow \circ . \geq \sim \phi d$

Data-parallel
proving



Linear: 3.3 s
Parallel: 0.9 s

Some
Numbers

The image features a vibrant orange background. On the left side, there are three solid brown circles of varying sizes. On the right side, a large blue circle is outlined, containing the text "Bottom of the rabbit hole?".

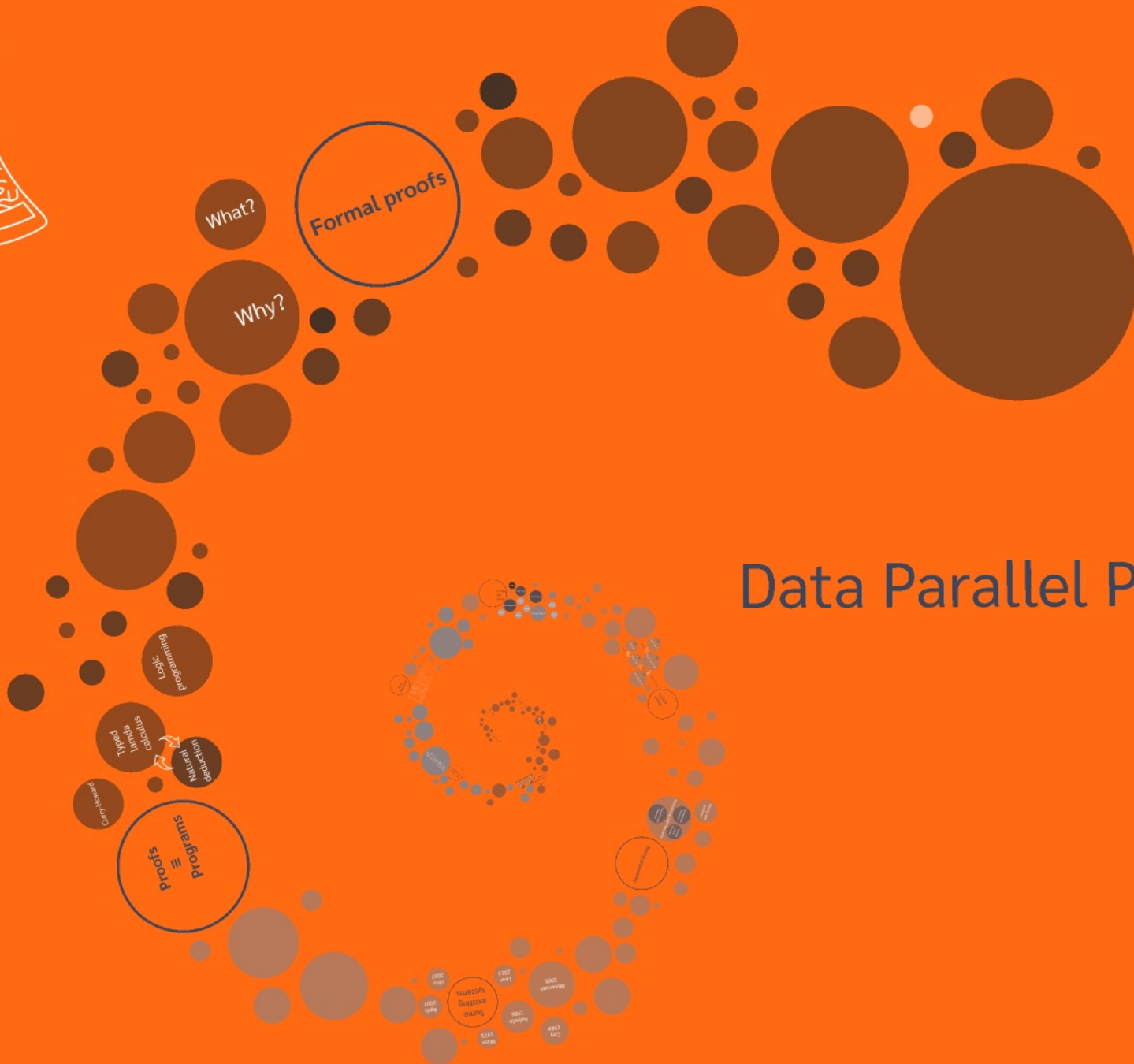
Bottom of the
rabbit hole?



What did we see?



Thank you



What?

Formal proofs

Why?

Formal program

Logical
specification
proof

Natural
deduction

Programs
≡
Proofs

Some
existing
systems

1981

1983

1985

1987

1989

1991

1993

1995

1997

1999

2001

2003

2005

2007

2009

2011

2013

2015

2017

2019

2021

2023

2025

2027

2029

2031

2033

2035

2037

2039

2041

2043

2045

2047

Data Parallel Proof Verification in APL

Brandon Wilson

