

# Inherent Limitations of Digital Computers: More Reasons Why Software Fails

**David A. Fisher**  
Reasoning Technology LLC  
[WWW.pbtOmega.com](http://WWW.pbtOmega.com)

**The pbtOmega Pittsburgh Forum Meetup**  
**7:00 pm April 11, 2019**

**@ Code & Supply**  
5648 Friendship Ave. third floor  
Pittsburgh PA 15206



# Computers do Many Useful Things

Nominal purpose: Valid deductive reasoning

- Symbol manipulation, arithmetic calculations
- Simple decision making, record keeping
- Input-output device control, process control
- Search & processing of large regularly structure data
- Graphic display, communication, etc.
- Statistical decision making

# Some Computing History

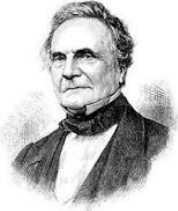
- Laws of Thought

≈ 330 BCE



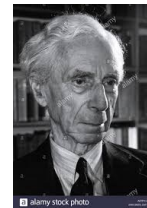
- Analytical Engine

≈1850



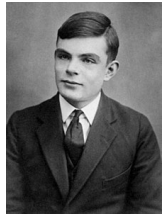
- Classical Logic

1860-1940



- Modern Digital Computers

1940-1967



- Programming Languages

1960s



# Vision for Computing

## Circa 1960s

- What we thought we saw
  - Automated devices for deductive reasoning
  - Languages and methods for reliable and error free software
  - Hardware limitations on computational speed, memory capacity, and communication bandwidth
- What we expected
  - Hardware improvements in speed, capacity, and bandwidth
  - Commensurate software advances that would enable human-like reasoning
  - Systems that would amplify human reasoning
    - the way powered machinery amplified human mechanical ability during the industrial revolution
- and ultimately
  - An intellectual revolution enabled by computer assisted human reasoning
  - Accelerated advances in science, engineering, and mathematics
  - Economic and social benefits rivaling those of the industrial revolution

# What we got 1970s and beyond

- Exponential growth in hardware speed, capacity, and bandwidth
- Exponential growth in expectations and complexity of software
- Unreliable, malfunctioning, and incomprehensible software
- High costs in software development and maintenance
- Often unrecovered costs for developments never completed
- Buggy, untrustworthy, and sometimes dangerous systems
- Broad acceptance of these conditions
- More recently: Potential for problems aggravated by widespread use of machine learning and other statistical methods

# What we did not get

- Reliable cost-effective systems
- Trustworthy software
- Sound deductive reasoning
- Guarantees of valid results
- Ability to understand & often to be understood
- Ability to address infinite and incomplete domains
- Creativity or invention

# What would it take to Solve these Problems?

- An information structure that is valid and incomplete with intrinsic meaning
- A sound mathematical foundation that is higher order, incomplete, intensional & monotonic
- An informal linguistic interface that is declarative, conversational, and interrogative, and that supports vagueness, imprecision, and incompleteness
- Automated generation of practical solutions without programmer intervention

# What gives Us Confidence that this is Possible?

- The human mind reasons with valid but incomplete information that it understands
- The logic of the human mind is sound, higher order, incomplete, intensional & monotonic
- Natural language is an informal linguistic interface to the mind
- The human mind generates solutions without programming assistance



# A Proposed Solution:

## Property Based Types

### A Theory of Human Reasoning

- An information structure that
  - is incomplete with intrinsic meaning
  - is analogous to concepts of the mind
- A sound deductive logic that
  - is higher order, incomplete, intensional & monotonic
  - guarantees valid results
- An informal language that
  - can characterize anything imaginable
  - natural language like to specify what instead of how
- An automated implementation that
  - reasons with property based types
  - performs proof in near linear time
  - exploits a persistent knowledge base
  - benefits from special purpose hardware

# The PBT Project

- The PBT logic
  - A incomplete information structure with intrinsic meaning
  - A sound incomplete intentional deductive logic
  - Uses abstraction, generalization, and analogy
- The Omega Language
  - Informal language grammar
  - Answers questions about anything imaginable
  - Calculates with incomplete types
  - Exploits persistent knowledge learned from experience
- Complete System Prototype
  - User /Developer Interface
  - Incremental Compiler
  - Runtime System
  - Persistent Knowledge Repository
  - PBT Reasoning Engine
- Currently Need Development Personnel
- Currently Need Writing & Documentation Assistance