AI Agents for DV Debug

Verification Futures 2025

BRONCO

Jeffrey Pan | Co-Founder & CTO

Overview

1. The Verification Gap
2. AI Agents for DV
3. Lessons Learned Scaling AI Debug

70%

...of the design cycle is spent on verification.

It is the #1 bottleneck in chip design.

Verification is Only Getting Harder

1. **Shorter timelines:** Need to get to market faster
2. **More complex designs:** Chip design complexity is growing exponentially
3. **Never enough DV bandwidth**: With thousands of tests running every night, DV teams struggle to keep up w/ finding the needle **(chip-killing bug)** in the haystack **(hundreds of failing tests)**

The Problem: It's a massive data analysis challenge, every single day.

Why Bring AI into Verification?

Human Engineers: Excel at deep, creative analysis of a *single* complex problem.

ML Algorithms: Excel at finding subtle patterns across *thousands* of datasets simultaneously.

The Goal: Combine human expertise with AI's scale to find bugs faster.

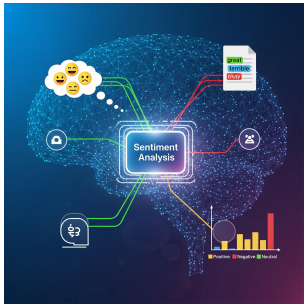
Classic ML: Limitations

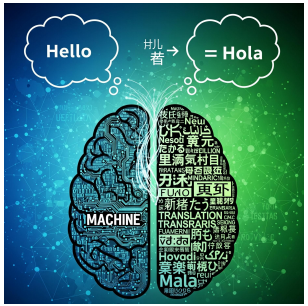
Deep learning is powerful, but...

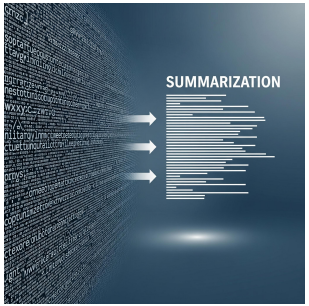
1. Doesn’t generalize well* – CPU vs. GPU vs. HBM vs. NoC...
2. Collecting data is hard – IP is expensive and sensitive
3. Labeling data is hard – requires highly skilled engineers
4. Constant distribution shift – chip 1.0 vs 1.1 vs 2.0 ...

The Next Step: Foundation Models and GenAI

*Can we formulate a task that is **easy to verify**, **easy to collect data for**, and is a **building block** for everything downstream we want to do?*



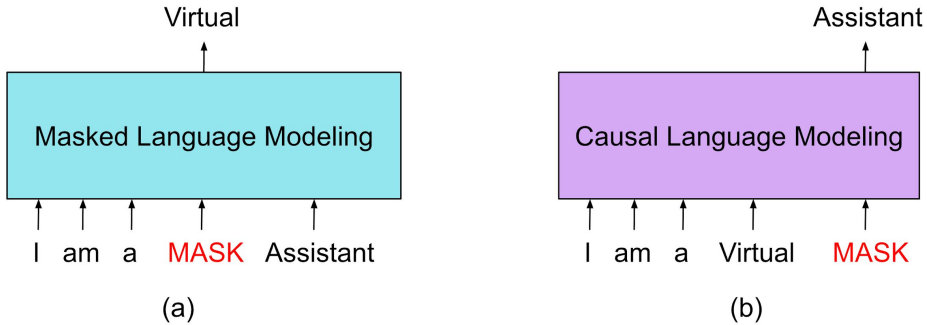




many tasks...

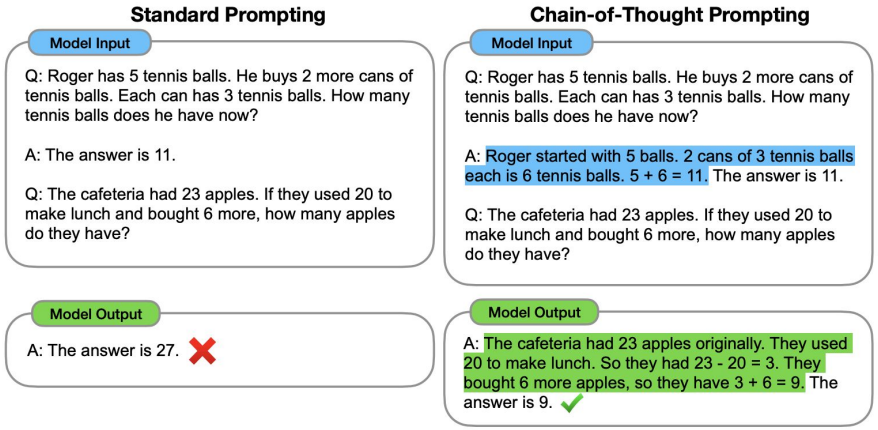
The Next Step: Foundation Models and GenAI

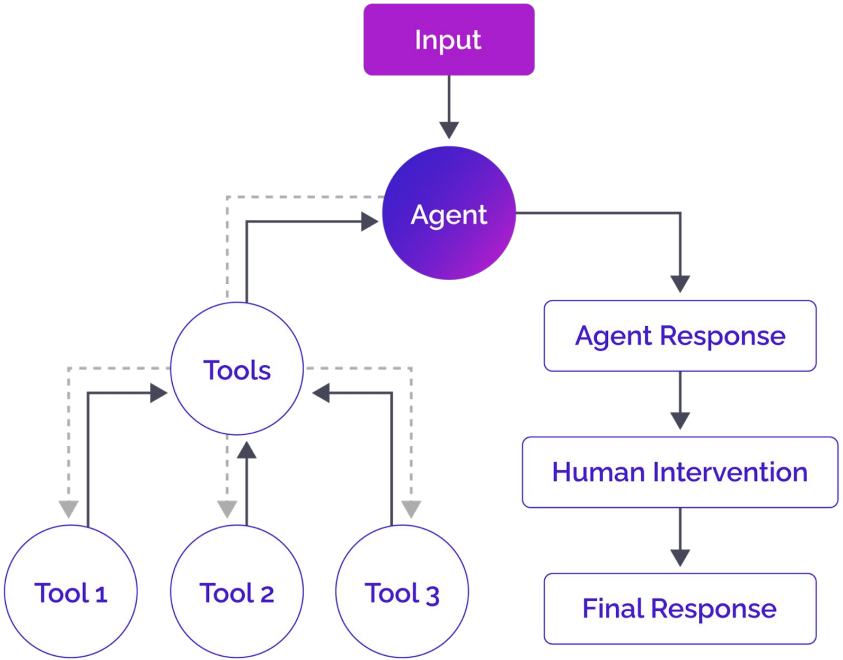
*Can we formulate a task that is **easy to verify**, **easy to collect data**, and is a **building block** for everything downstream we want to do?*



...to one!

Reasoning Models & Agents





Agents Generalize Where Models Can’t

Deep learning is powerful, but…

agents take debug to the next level!

1. ~~Doesn’t generalize well* – CPU vs. GPU vs. HBM vs. NoC…~~
2. ~~Collecting data is hard – IP is expensive and sensitive~~
3. ~~Labeling data is hard – requires highly skilled engineers~~
4. ~~Constant distribution shift – chip v1 ≠ chip vN~~

Agents Generalize Where Models Can’t

Agents work better because…

1. Strong ability to generalize – trained on internet-scale code + problem solving
2. Steerable by natural language – easy for users to steer
3. Learn metadata, not data – don’t overfit
4. Can handle and forget context – addresses IP/security issues

Bronco Platform: Accelerate DV Flows End-to-End

Verification Planning

Man-weeks of verification planning done in minutes.

Testbench Bring-Up

10x faster bring-up of new UVM TBs with Bronco's specialized DV agents.

Simulation Debug

Failure-to-Fix in minutes on **100GB+ waveforms** with Bronco Agentic Debug.

Secure Your IP

Deploy with complete security with **on-prem** and **bring-your-own-AI** options.

Ready on Day 1

Easy integration with customer EDA flows, and without custom AI training.

Self-Improving

AI-powered learning that transfers across tasks, bugs, and projects.

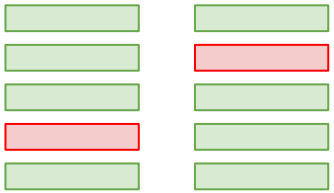
Bronco – AI Debug Agents at Scale

The moment a simulation fails, Bronco:

1. Decides how to investigate
2. Executes standard DV actions
3. Works until it finds the bug or hands off a ticket to a human

Inputs

Simulation Runs (Waves & Logs)



Project Context

- Paths to RTL, UVM, specs, docs etc.
- Input/Feedback

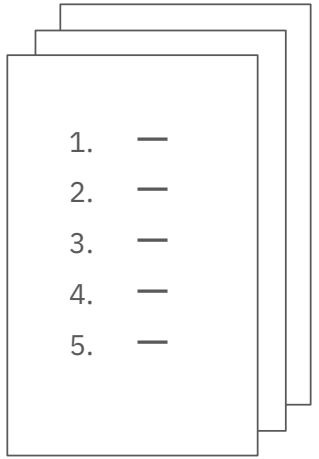
Bronco AI

Routine DV Work

1. Makes playbook for how to debug a given failure.
2. Extracts specifics from the project context.
3. Performs debug and remembers for next time.

Outputs

Root Causes & Tickets



Real Results – Next Generation ASIC

New error signature on a subsystem-level UVM test

- After 30 minutes
  - Bronco indexed/adapted to new regression
  - Agent formed understanding of the design (**200,000+ lines of RTL**)
  - Common failure patterns understood

Real Results – Next Generation ASIC (cont.)

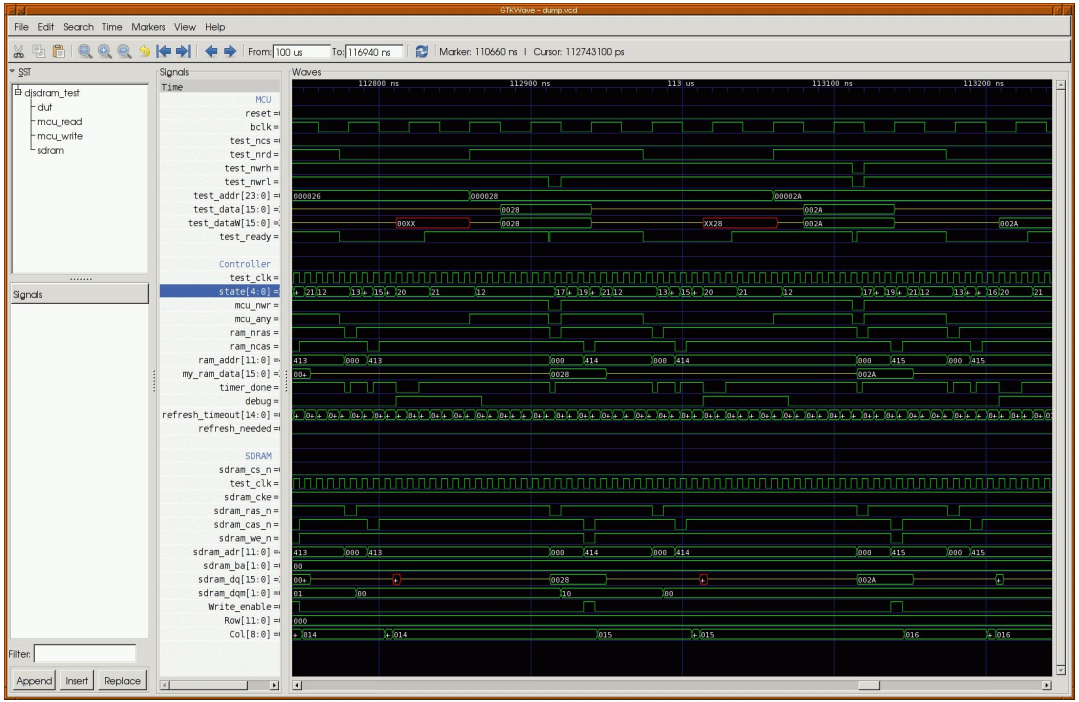
New error signature on a subsystem-level UVM test

- After 15 minutes
  - Agent analyzed run log (**100,000+ lines**), and waves (**~10GB FSDB**)
  - Agent identified root cause nested under **4 levels of hierarchy**
  - DV lead estimated they would have spent **4 hours** on debug, **new engineer may have spent 3 days**

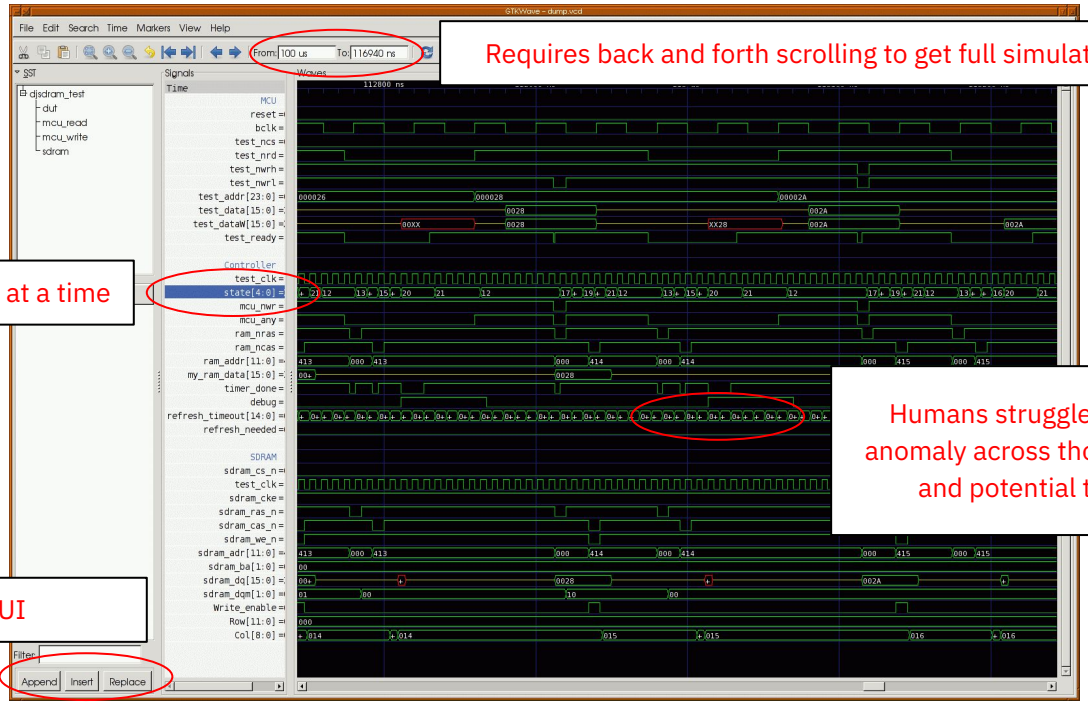
Lessons Learned

1. AI Agents require **rethinking human actions.**
2. AI Agents deliver **value through scale**.
3. Fast-time to value requires **continuous learning**.

Reimagining Waveform Debug



Reimagining Waveform Debug



Requires back and forth scrolling to get full simulation context

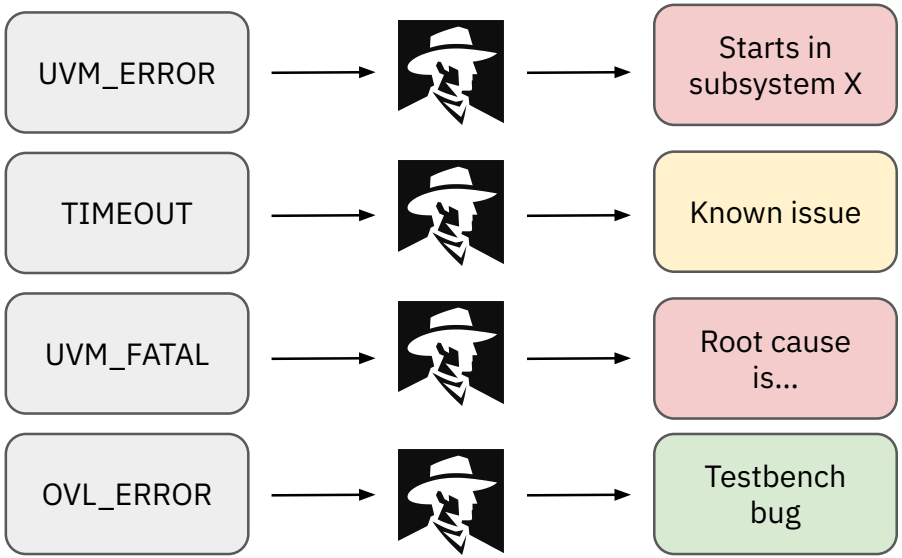
Go down one train of thought at a time

Humans struggle to spot the tiny anomaly across thousands of signals and potential time windows

Laggy 20 year old GUI

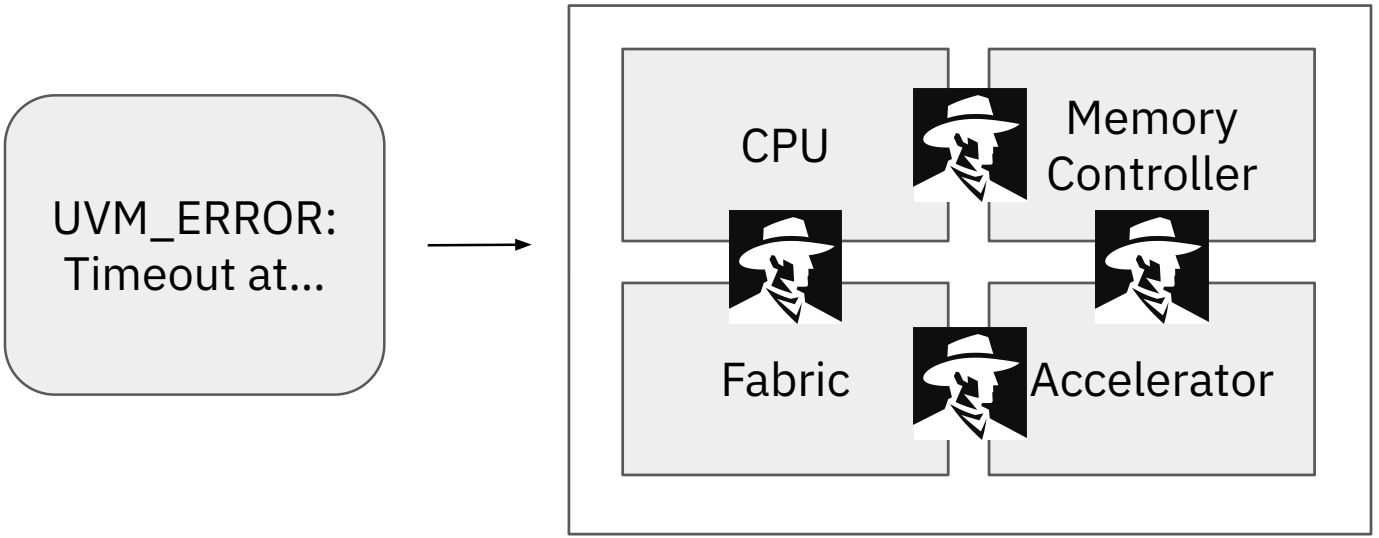
AI Agents Deliver Value Through Scale

Process many failures at once on each regression

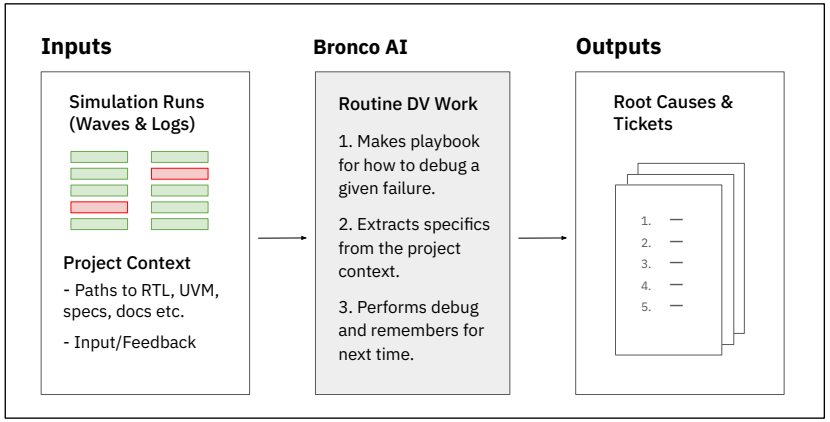


AI Agents Deliver Value Through Scale

Check many things at once on each failure.



Building Systems that Learn



Efficient shortcuts: “it’s faster to find X in the wave, but Y is in the logs”

Generalized intuition: “these types of issues show up on NoCs but not CPUs”

Human preferences: “Company X prefers tickets in this format”

Questions?

bronco.ai

bronco.ai/jobs