2022 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

MUNICH, GERMANY
DECEMBER 6 - 7, 2022

IBM

Taking Design Automation to the next level with User Experience Design

Jamie Lai, Bodo Hoppe
IBM R&D Germany GmbH

accellera SYSTEMS INITIATIVE

The Team

Jamie Lai
UX designer, IBM Z Systems
jamie.lai@ibm.com

Bodo Hoppe
Distinguished engineer, IBM Z Hardware
bohopp@de.ibm.com

accellera
SYSTEMS INITIATIVE

The IBM Telum Processor

> 5 GHz
frequency

> 22 Billion
Transistors on a module

> 19 Miles
of wires on a module

accellera
SYSTEMS INITIATIVE

Development of the IBM Telum Processor

1,500,000,000,000,000
Simulation Cycles in software Simulation

> 500K
unique discrete coverage events in the design

+ Millions of
cross-product events

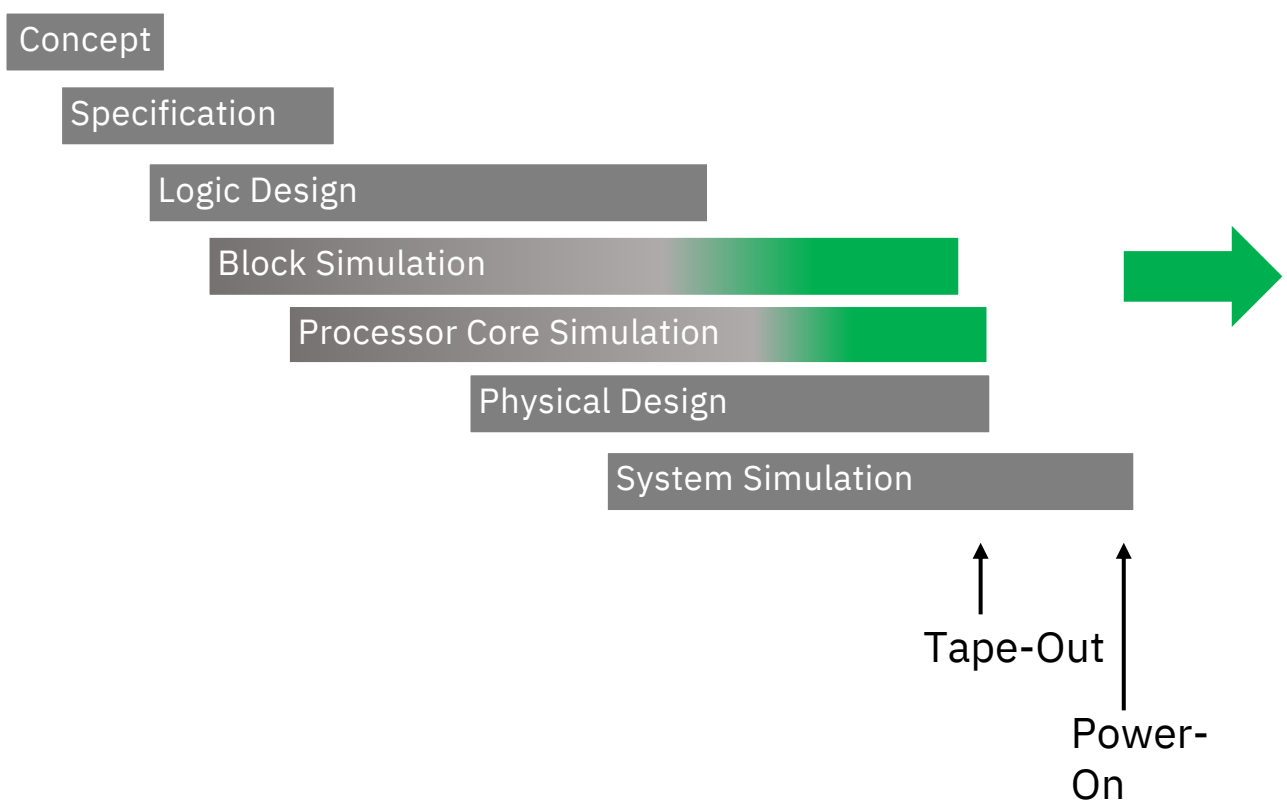
accellera
SYSTEMS INITIATIVE

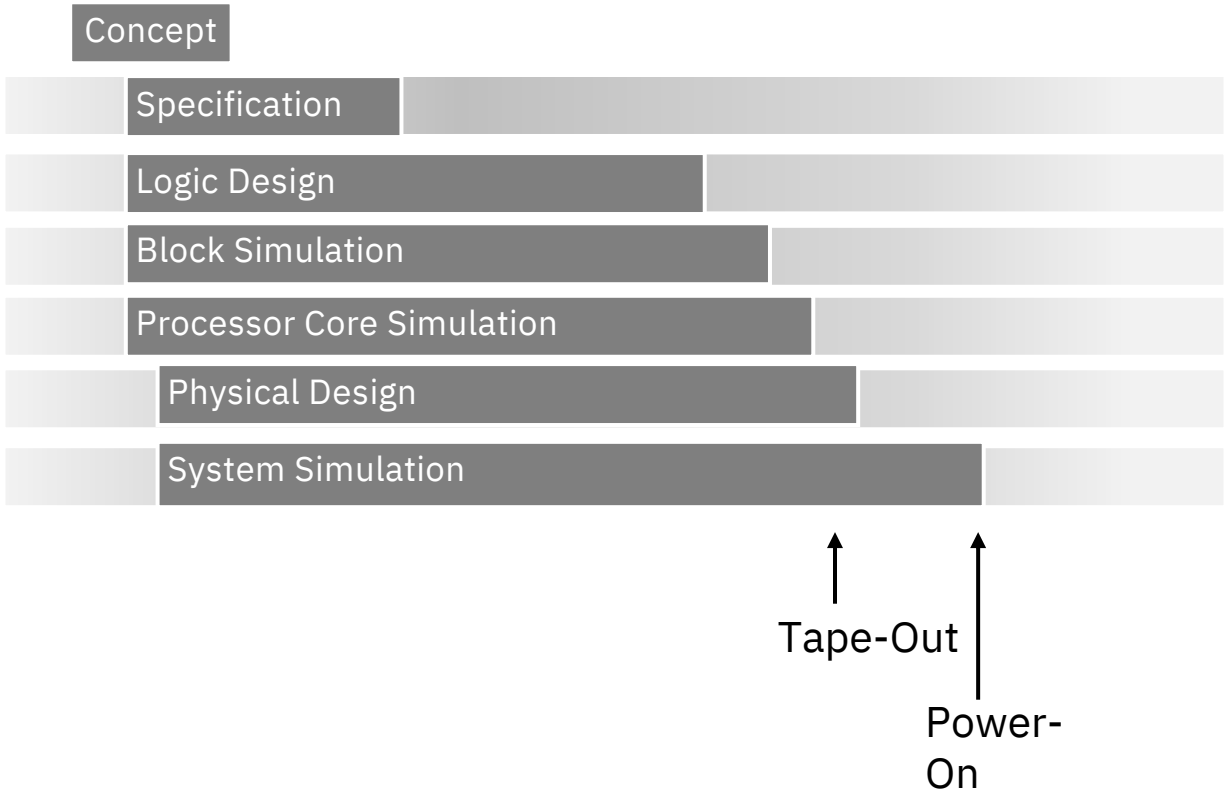
Logic Designers and Verification Engineers spend a huge amount of time on

- Defining coverage
- Implementing coverage
- **Debugging coverage**
- **Analyzing coverage holes**
- **Hitting important coverage events**

accellera SYSTEMS INITIATIVE

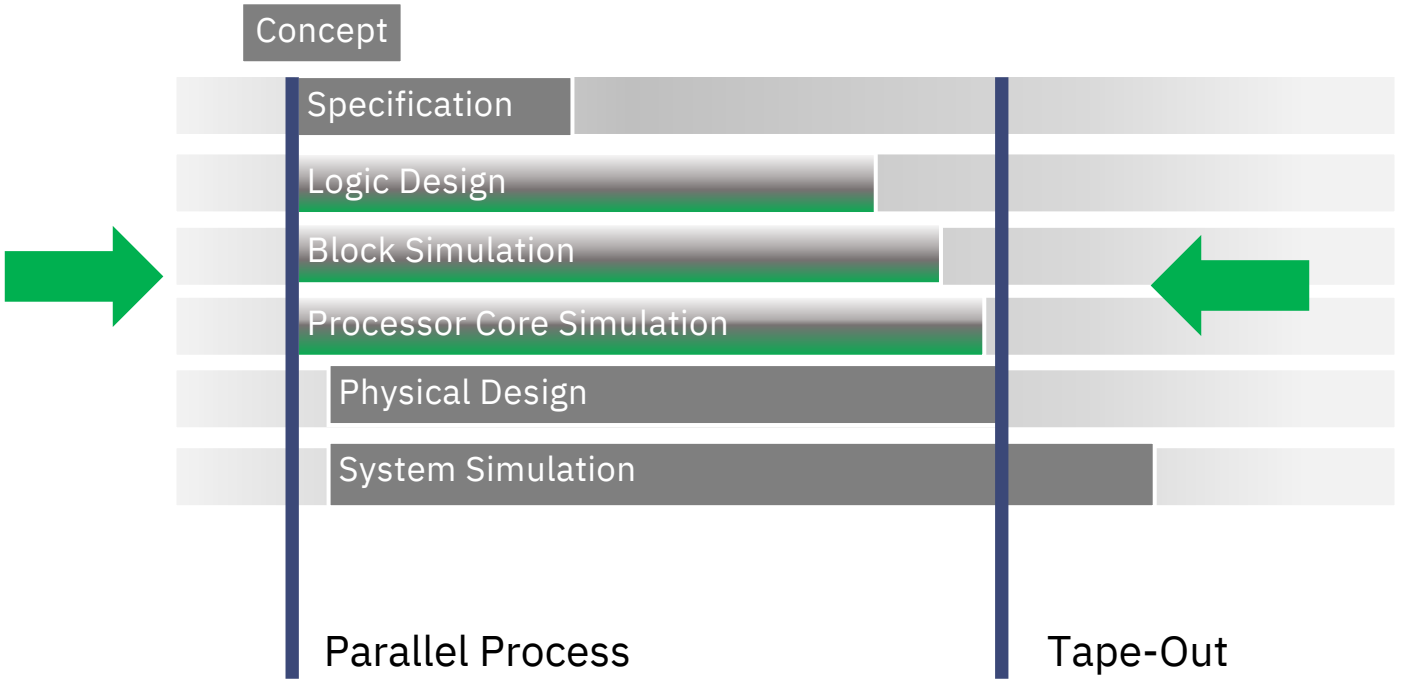
Fast time to market





Potential of Functional Coverage

- Faster time-to-market
- Higher Quality
- Less (Compute) Cost



We use various functional coverage types

- ✓ Interface coverage
- ✓ (Micro-)Architectural cross product coverage
- ✓ Configuration coverage
- ✓ Discrete design events
- ✓ Testbench coverage

accellera
SYSTEMS INITIATIVE

We have (some) collaboration

- ✓ Mark as waived/deferred/impossible
- ✓ Analyze
- ✓ Prioritize
- ✓ Comment

accellera SYSTEMS INITIATIVE

We have the technology

- ✓ Template Aware Coverage
- ✓ Coverage Driven Generation
- ✓ Finding unhittable events
- ✓ Coverage hole analysis
- ✓ Grading events & test templates
- ✓ Identifying aged out events

https://research.ibm.com/publications/template-aware-coverage-taking-coverage-analysis-to-the-next-level
https://research.ibm.com/publications/automatic-scalable-system-for-the-coverage-directed-generation-cdg-problem

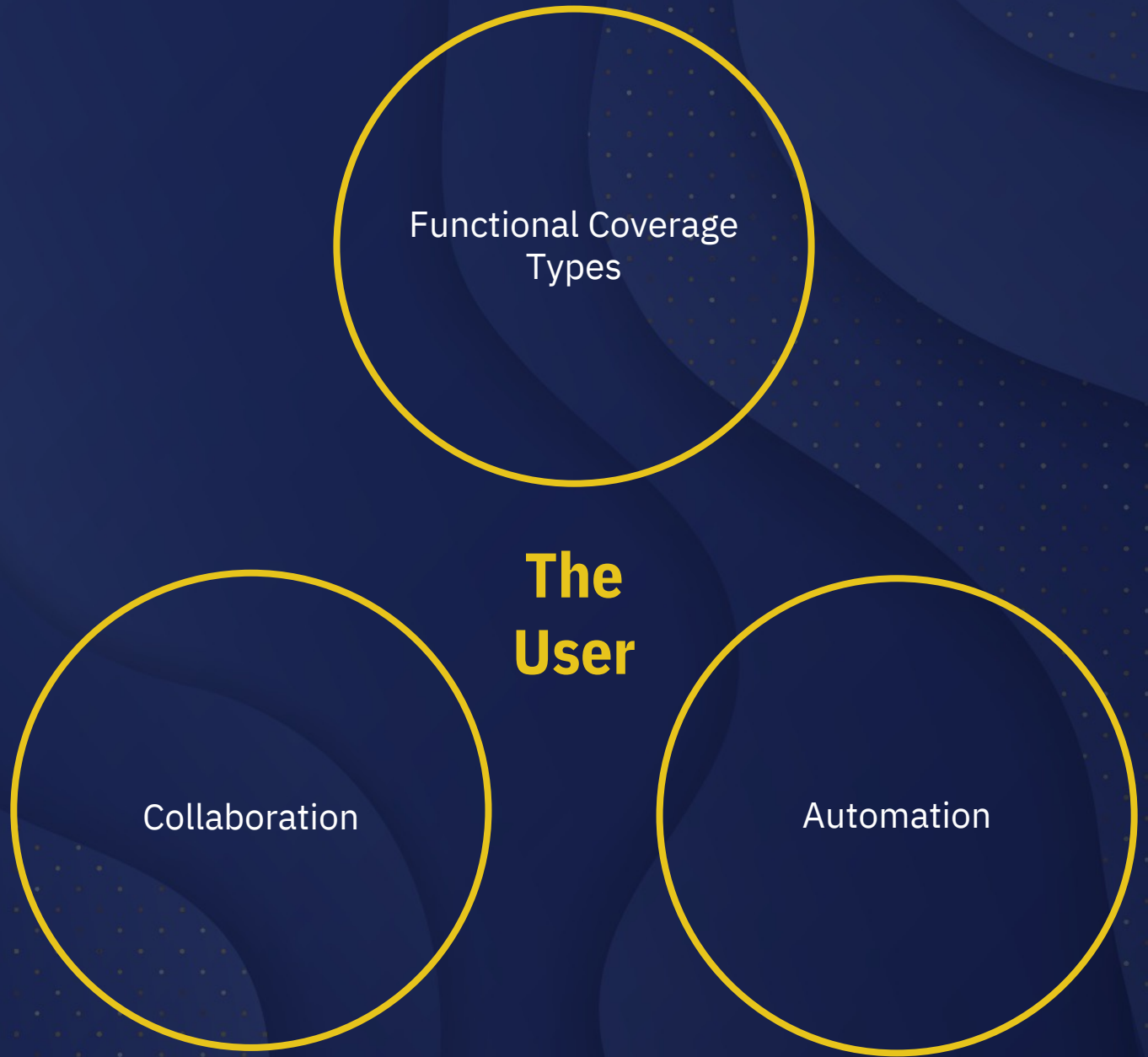
accellera SYSTEMS INITIATIVE

But...

- Many months are still spent on coverage closure
  - In the critical path for tape-out
- We‘re still not seeing the forest because of all the trees
  - Missing potential bugs
- Still wasting compute cycles

Let’s automate more …

… and we can also …



accellera
SYSTEMS INITIATIVE

Our tech and our tools can assist, but they cannot replace

accellera
SYSTEMS INITIATIVE

How do we use the tools and tech we have in a way that will **enhance and augment** the user's abilities?

accellera
SYSTEMS INITIATIVE

2022 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

MUNICH, GERMANY
DECEMBER 6 - 7, 2022

Introducing... UX design!

accellera SYSTEMS INITIATIVE

“Design a vase”

accellera
SYSTEMS INITIATIVE

“Design a better way for someone to enjoy flowers”

accellera
SYSTEMS INITIATIVE

We look to these ways of questioning our problem spaces, to open up our creativity and scope, and to ask the Right Questions

accellera
SYSTEMS INITIATIVE

Enterprise design thinking: The loop

Observe

Reflect

Make

accellera SYSTEMS INITIATIVE

UX design: An umbrella term

Observe

Reflect

Make

UX research

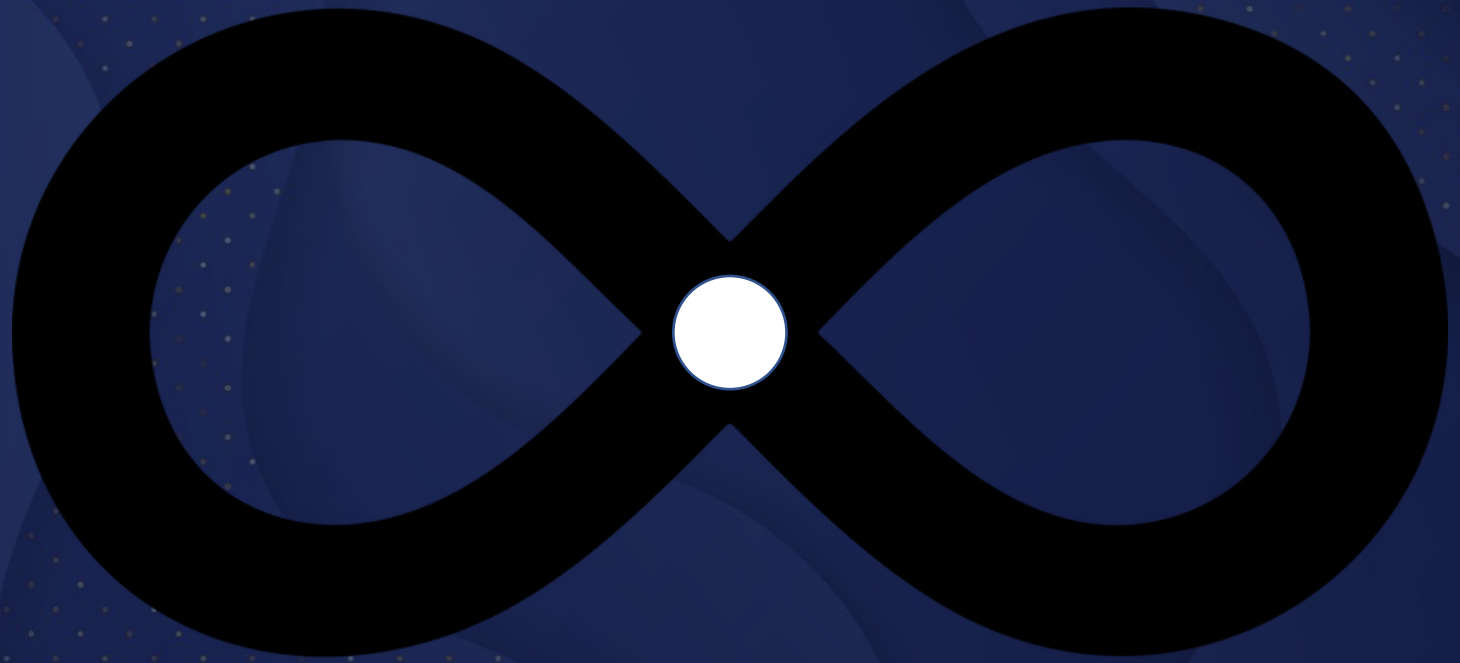
accellera
SYSTEMS INITIATIVE

UX design: An umbrella term

Observe

Reflect

Make



UX design
Systems design
Service design
Product design

accellera
SYSTEMS INITIATIVE

UX design: An umbrella term

Observe

Reflect

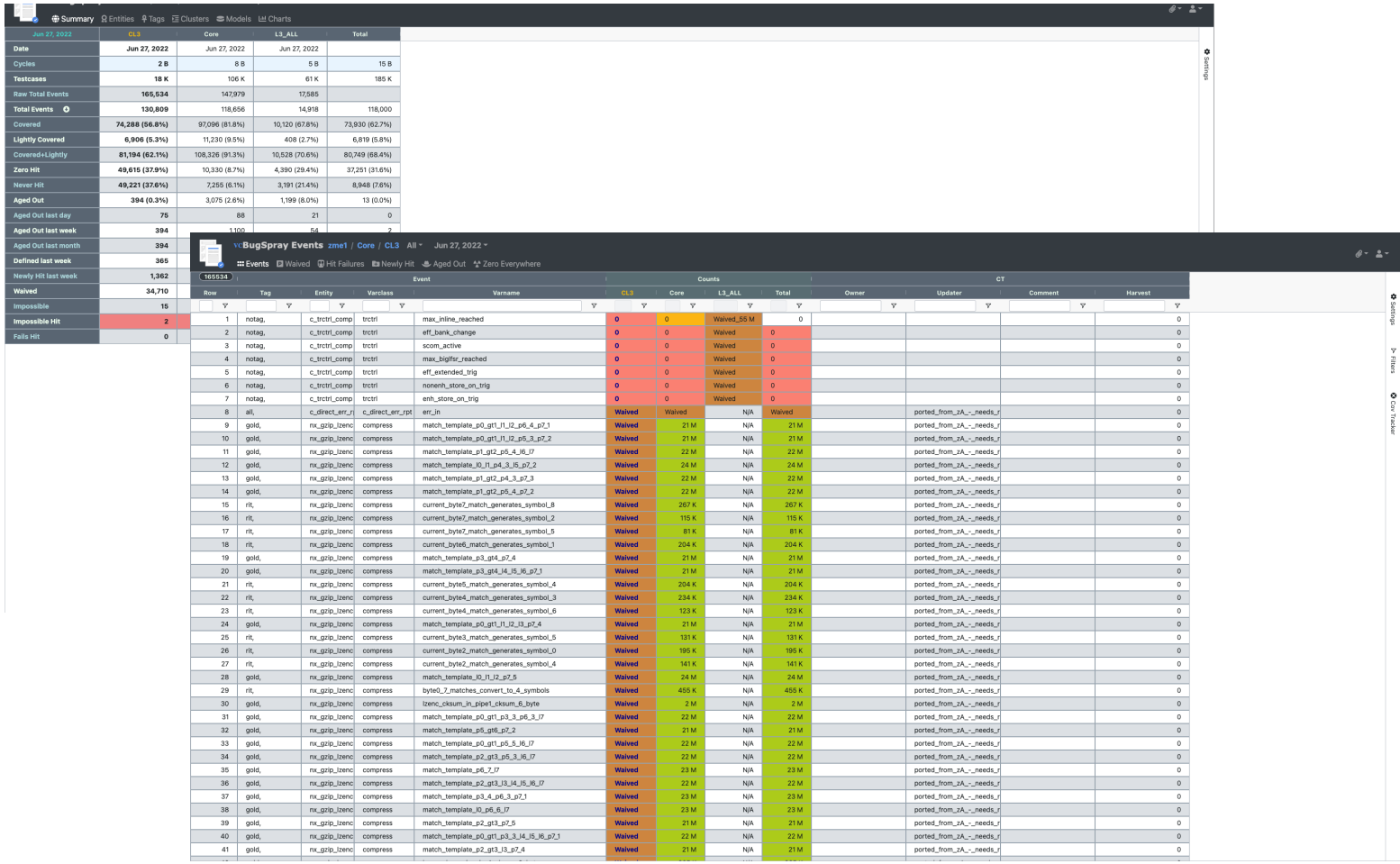
Make

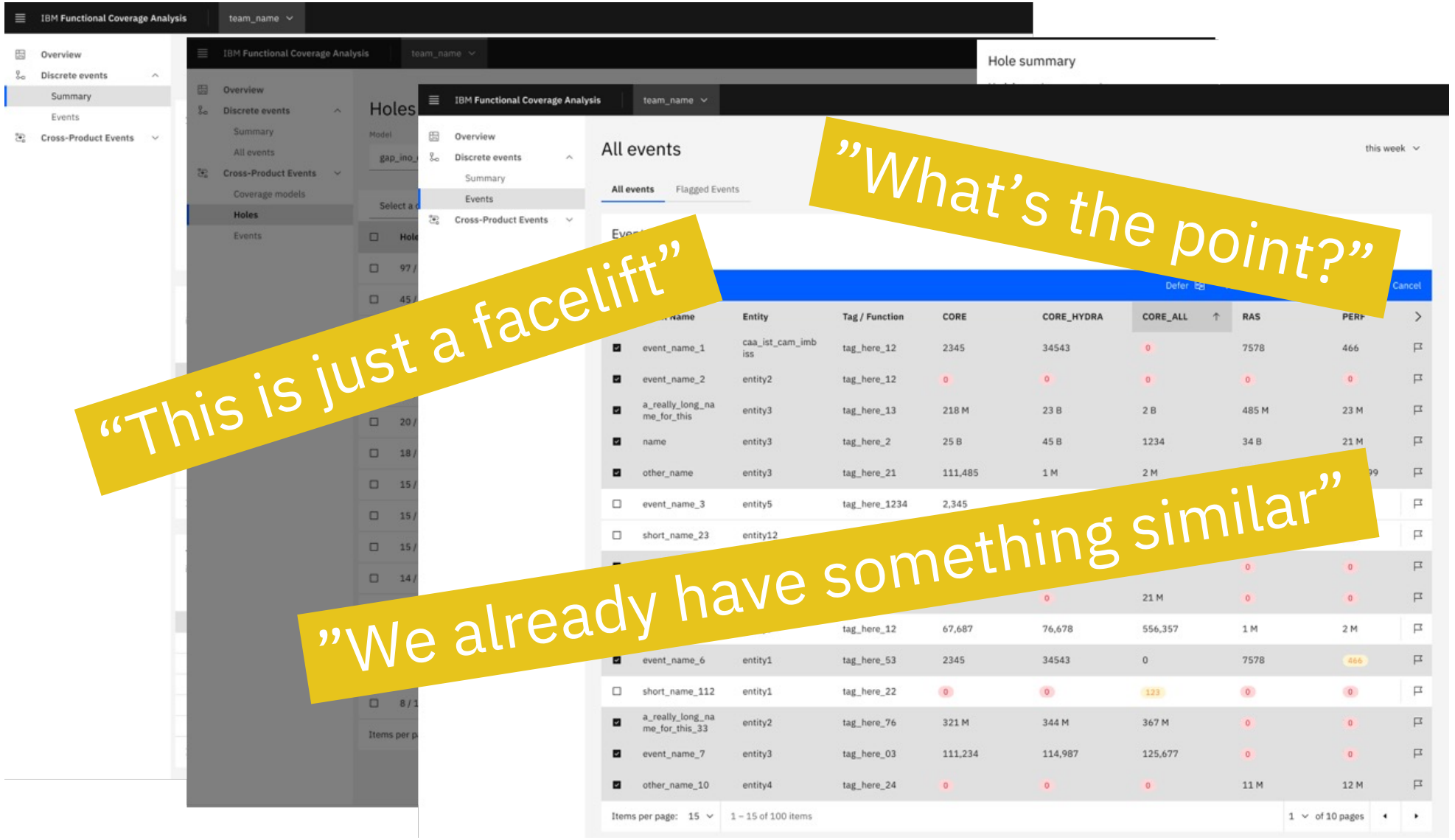
UI design
Visual design
Content design

accellera
SYSTEMS INITIATIVE

The original scope

- Improve the coverage data analysis experience
- Show user actions like waiving, deferring, marking impossible…




IBM Functional Coverage Analysis
team_name
Overview
Discrete events
Summary
Events
Cross-Product Events
Coverage models
Holes
All events
Flagged Events
this week
Hole summary
"This is just a facelift"
"What's the point?"
"We already have something similar"

What we were doing: treating the symptom of functional coverage

accellera
SYSTEMS INITIATIVE

A quick reminder…

https://scrollller.com/ui-vs-ux-3iy0tb43v4

What we should’ve been doing:
investigating the **underlying issue**...

accellera
SYSTEMS INITIATIVE

Why are there so many irrelevant and confusing events in the first place?

accellera
SYSTEMS INITIATIVE

“Design a way to improve the coverage data analysis experience”

accellera SYSTEMS INITIATIVE

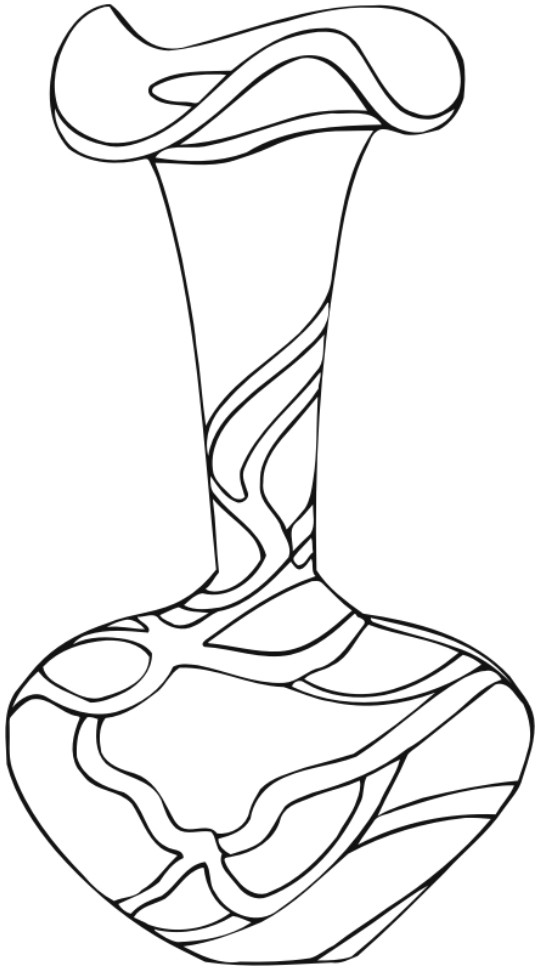
~~“Design a way to improve the coverage data analysis experience”~~

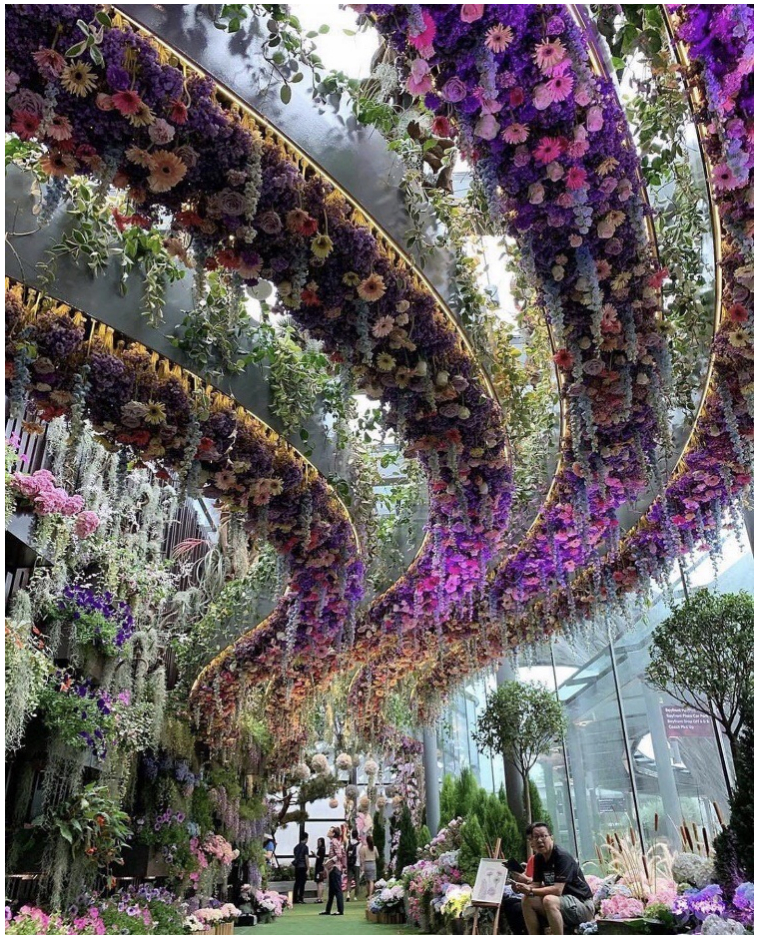
accellera
SYSTEMS INITIATIVE

“Design a way to close the design of the microprocessor faster”

accellera
SYSTEMS INITIATIVE

Feature definition	Feature implementation	Coverage analysis	Coverage closure





Floral fantasy at Gardens by the Bay, singapore
by: @unique_singapore [IG]

Observe

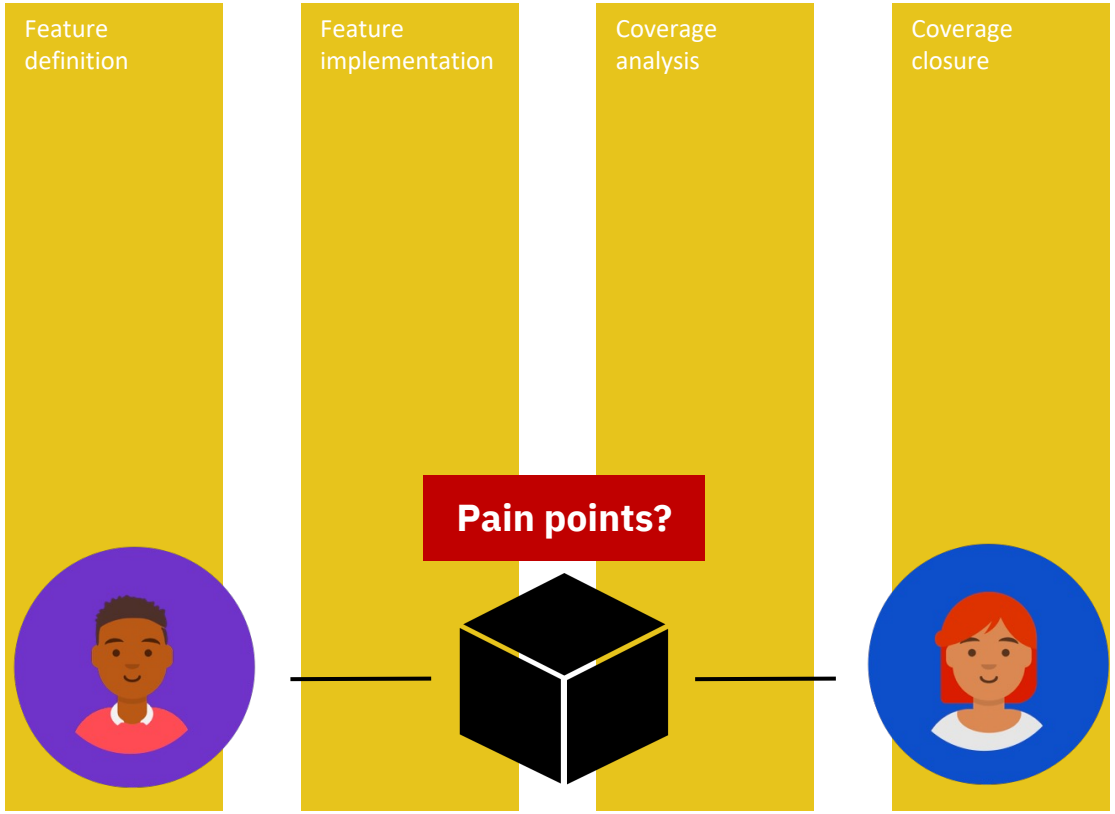
Reflect

Make

accellera
SYSTEMS INITIATIVE

User research questions

- What is the **overall process**, from feature definition to coverage closure?
- Where are the **collaboration touchpoints** between VE and LD?
- What are the **pain points** in this collaboration process?



Design toolbox: **Personas**

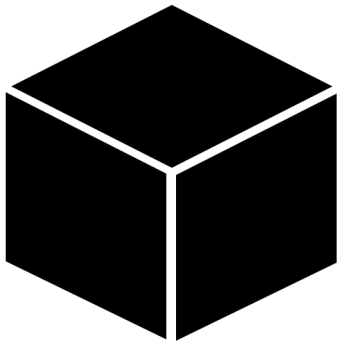
Eddie

Logic Designer

Works with architects to define the new features of the chip

Creates the logic design of the hardware

Defines spaces that need verification, using things called “events”



Jobs to be done?
Collaboration?
Pain points?

Vi

Verification Engineer

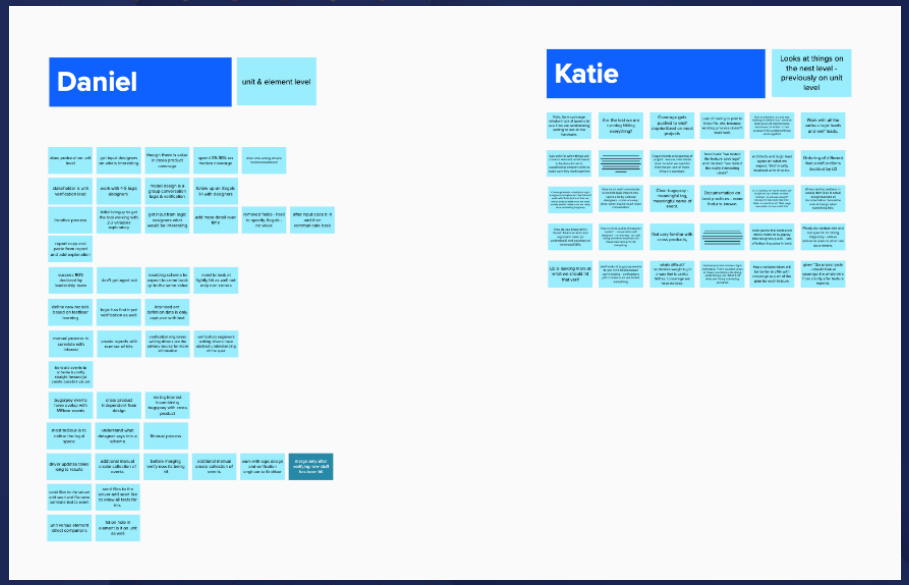
Creates test cases in C++ that stress the design of the chip

Looking to improve their test bench to hit all important coverage events to find all bugs in the HW

Sorts through all the events generated by the Verification Cockpit (VC) to look for the most interesting events

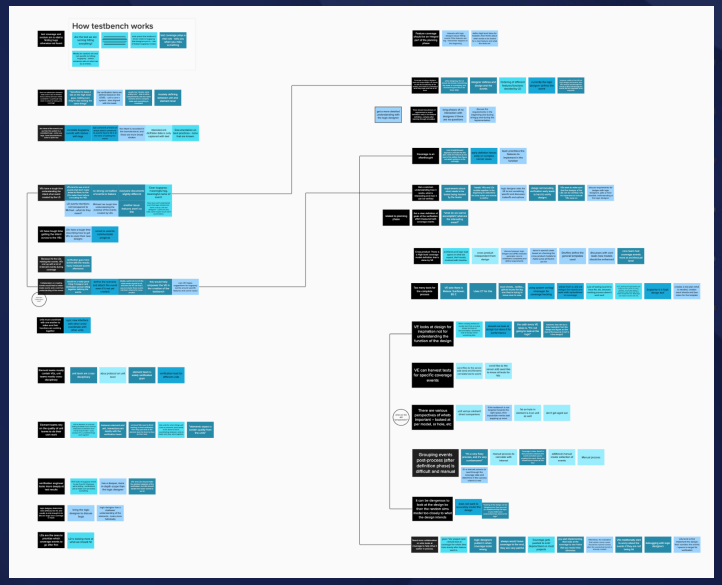
Research

Step 1: **Observe**



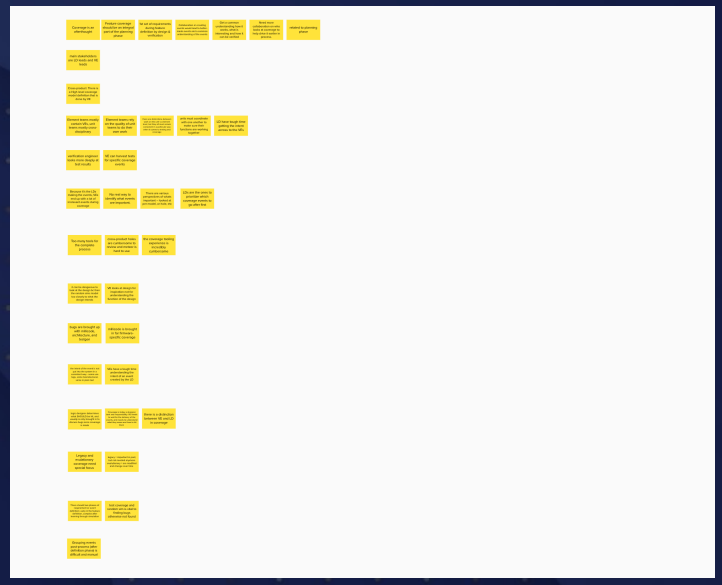
What does the user think/do?

Step 2: **Gather insights**



What common attitudes and behaviors do our users collectively have?

Step 3: **Synthesize into big ideas**



What are the main ideas we can gather from this?

accellera SYSTEMS INITIATIVE

2022 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

MUNICH, GERMANY
DECEMBER 6 - 7, 2022

EDT toolkit: As-is scenario

accellera SYSTEMS INITIATIVE

Designer toolkit: As-is scenario overview

Phase 1
Feature definition

- Main feature and design are defined here
- Eddie draws up the specs and designs with the architects
- Vi takes a back seat

Phase 2
Feature implementation
Simulation bring-up

- Designs are created, simulation testing begins
- Eddie begins coding his designs
- Vi looks at the design specs in order to begin crafting her testbench scenarios

Phase 3
Coverage analysis

- Analysis of the testbench and discrete events begins
- Eddie and Vi collaborate to analyze the results.

Phase 4
Coverage closure

- Coverage must meet the expected quality requirements
- Vi writes special cases and continues to improve the testbench until the coverage percentage is high enough
- Eddie improves his designs

accellera
SYSTEMS INITIATIVE

Designer toolkit: As-is scenario overview

Phase 1
Feature definition

- Main feature and design are defined here
- Eddie draws up the specs and doc… with…

Eddie is often asked to deliver both good design and good coverage

Vi does not participate in design scoping

Phase 2
Feature implementation
Simulation bring-up

- Designs are created, simulation testing begins
- Eddie begins … des…
- …
- …

Because Eddie must focus on his designs, this often results in the creation of improperly documented coverage

Vi is blocked from coverage until after the implementation of the designs be Eddie

Phase 3
Coverage analysis

- Analysis of the testbench and discrete events begins
- Eddie and Vi collab…

Vi, in turn, struggles with understanding the intent of events coded by Eddie

There is no central repository for proper collaboration

Phase 4
Coverage closure

- Coverage must meet the expected quality requirements
- Vi writes … and …
- …
- E… … his de…igns

Vi has no set definition of "done"

Eddie discovers bugs super late in the implementation phase because they were not caught by the testbench earlier

accellera SYSTEMS INITIATIVE

Designer toolkit: As-is scenario overview

Phase 1
Feature definition

- Main feature and design are defined here
- Eddie draws up the specs and doc… with…

Eddie is often asked to deliver both good design and good coverage

Vi does not participate in design scoping

Phase 2
Feature implementation
Simulation bring-up

- Designs are created, simulation testing begins
- Eddie begins … design…
- …

Because Eddie must focus on his designs, this often results in the creation of improperly documented coverage

Vi is blocked from coverage until after the implementation of the designs by Eddie

Phase 3
Coverage analysis

- Analysis of the testbench and discrete events begins
- Eddie and Vi collab…

Vi, in turn, struggles with understanding the intent of events coded by Eddie

There is no central repository for proper collaboration

Phase 4
Coverage closure

- Coverage must meet the expected quality requirements
- Vi writes spec… and…
- E… designs

Vi has no set definition of "done"

Eddie discovers bugs super late in the implementation phase because they were not caught by the testbench earlier

accellera SYSTEMS INITIATIVE

Observe

Reflect

Make

accellera

SYSTEMS INITIATIVE

2022 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

MUNICH, GERMANY
DECEMBER 6 - 7, 2022

EDT toolkit: To-be scenario

accellera SYSTEMS INITIATIVE

How might we **activate** the users through **collaboration**?

Pain points

- Eddie often asked to deliver both good design and good coverage
- Vi does not participate in design scoping

To-be scenario

- Eddie defines the events without implementation, along with the scope of the stories
- Vi collaborates earlier to understand the scope and maps the events to the features created

How might we **activate** the users through **collaboration**?

Pain points

- Because Eddie must focus on his designs, this often results in the creation of improperly documented coverage
- Vi struggles to understand the coverage created by Eddie

To-be scenario

- Eddie provides the details of the coverage space and the expected results to Vi
- Vi implements the coverage defined by Eddie in her testbench in order to create thorough and well-architected coverage

How might we **activate** the users through **collaboration**?

Pain points

- There is no central repository for proper collaboration

To-be scenario

- All past tooling and features are integrated into 1 collaborative-centric platform
- The experience and the views on this platform are standardized for easier access by both personas

Design a prototype

Observe

Reflect

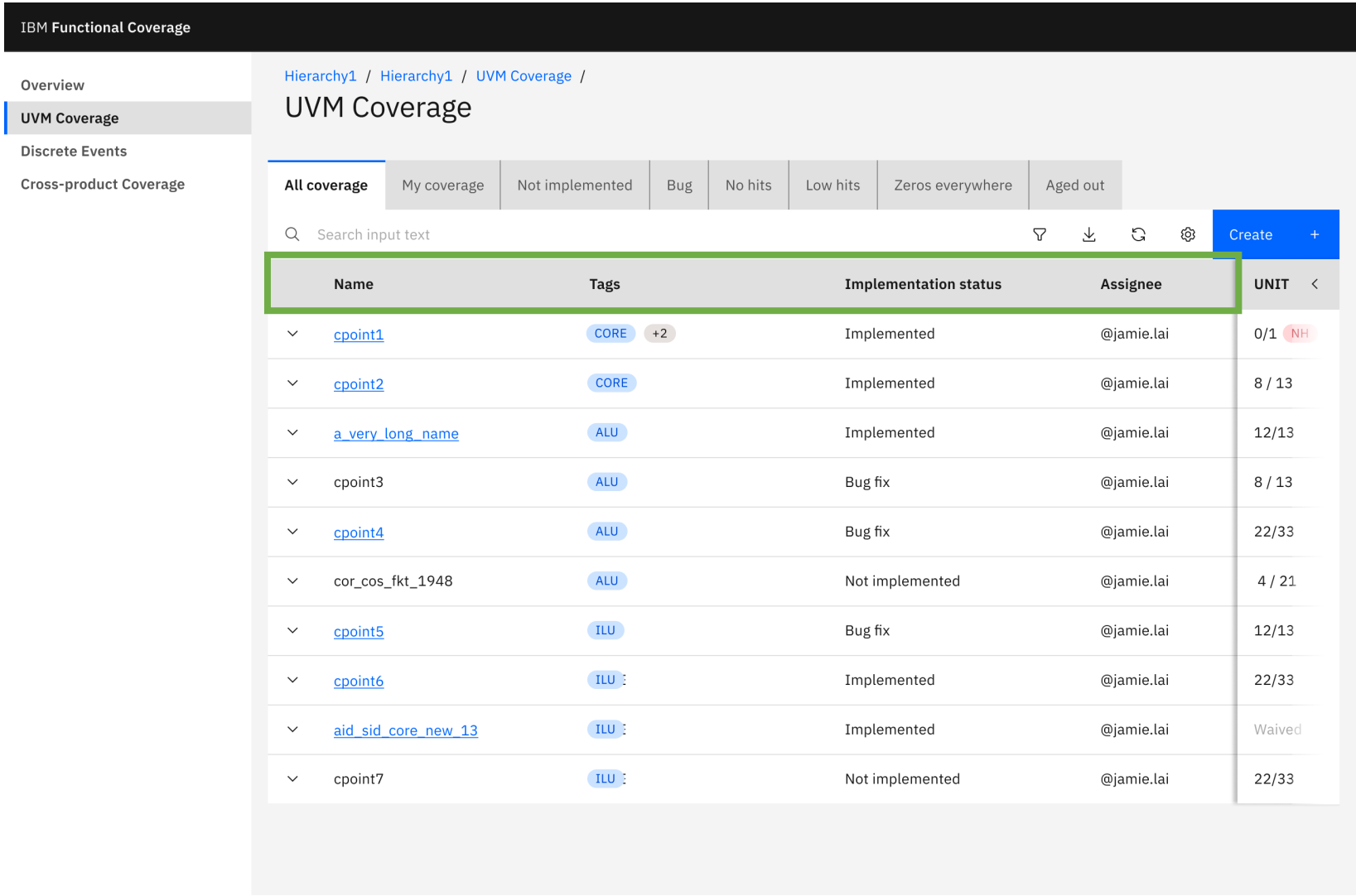
Make

accellera
SYSTEMS INITIATIVE

Overview

To-be scenario: All past tooling and features are integrated into 1 collaborative-centric platform

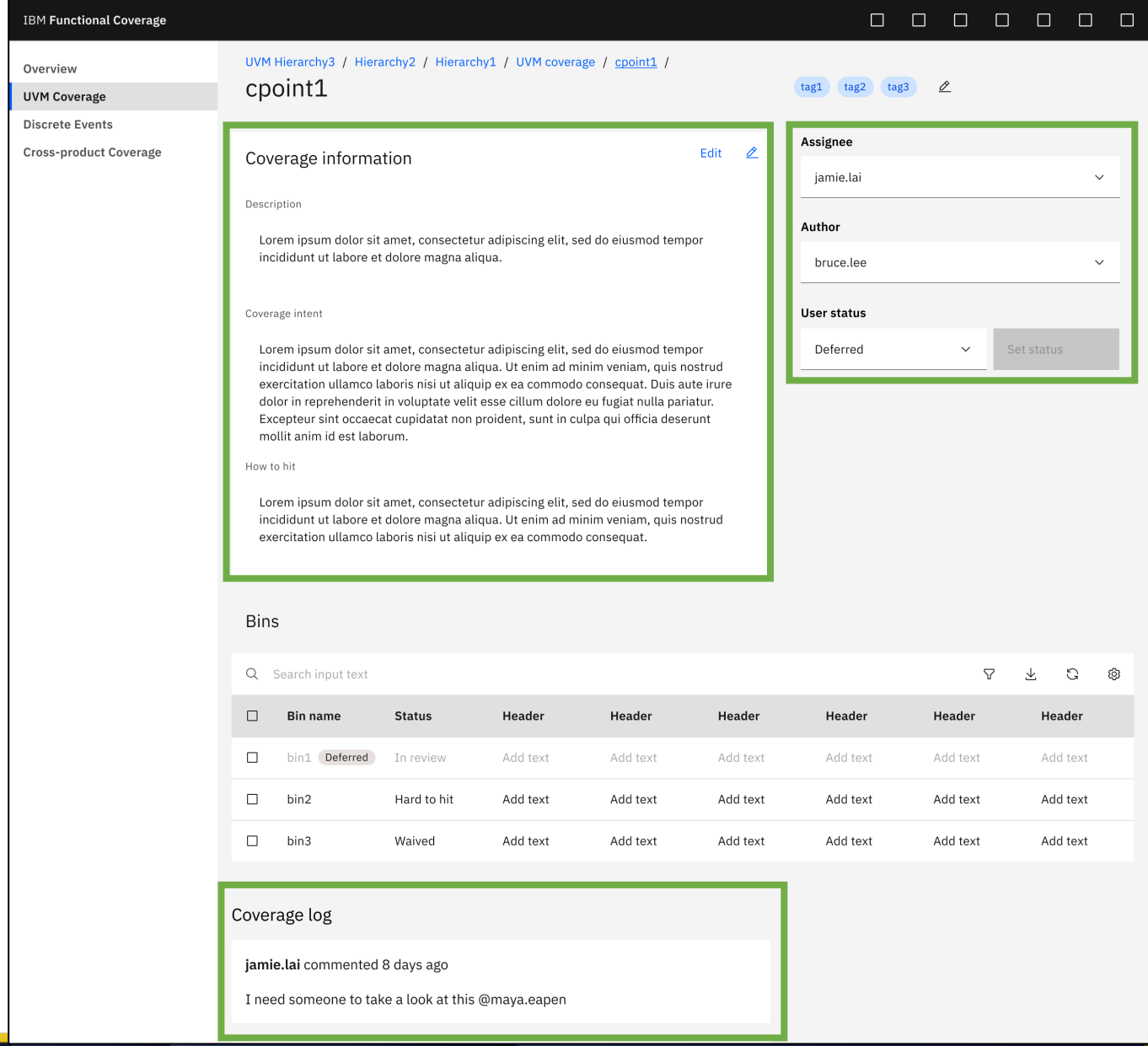
- Implemented with a **standardized design system**
- The user can see important metadata in order to provide the necessary context of the event
- Advanced filtering



Coverage details

To-be scenario: Eddie provides the details of the coverage space and the expected results to Vi

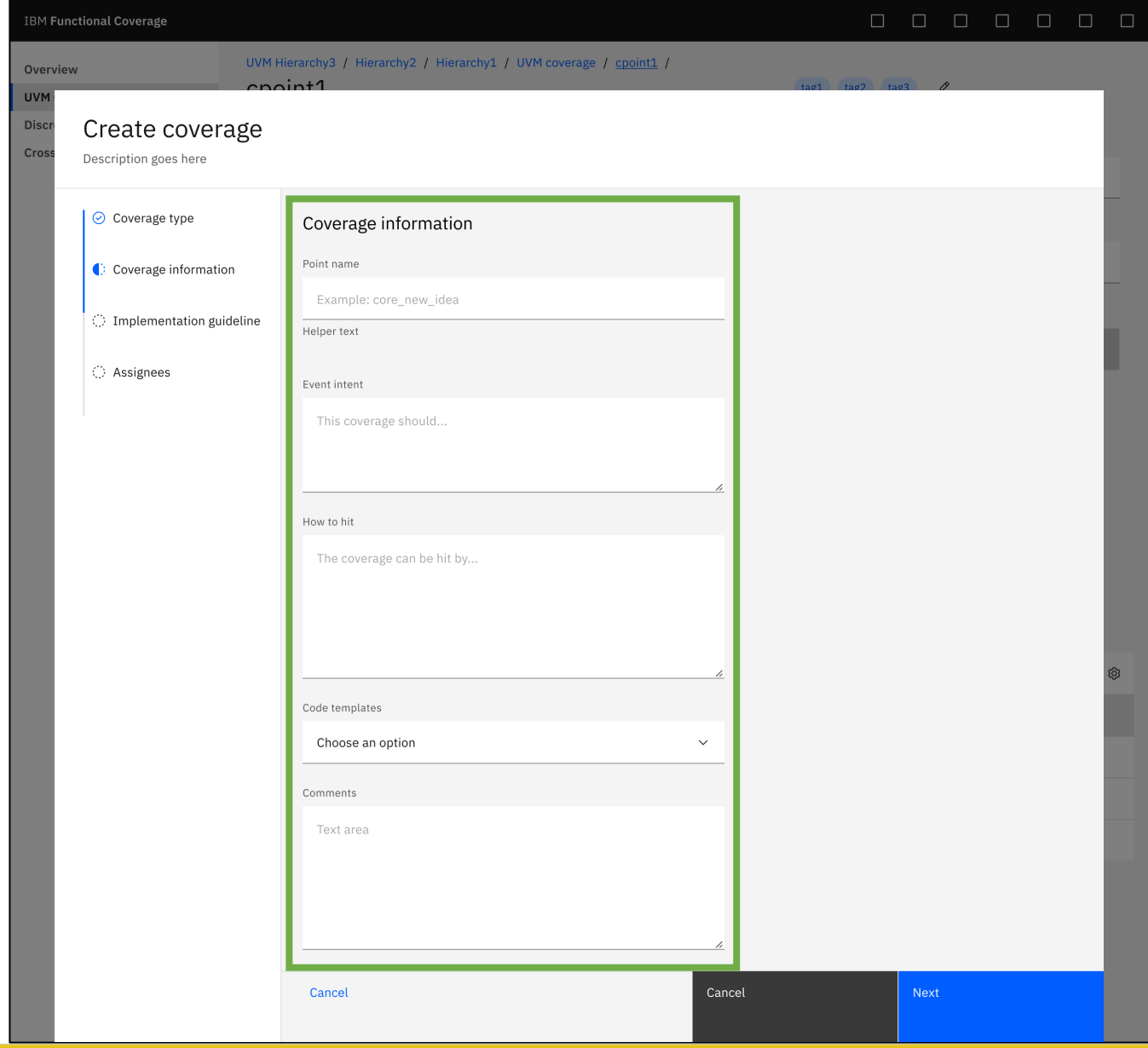
- A new “more info” page gives more detailed info about the coverage point
- This is where Vi would find information about coverage, fleshed out by Eddie
- History/log of events and comments can be found here



Request Coverage

To-be scenario: Eddie defines the events without implementation, along with the scope of the stories

- Eddie is able to request coverage and give a detailed blueprint of how it should be implemented
- Vi will take this blueprint to implement coverage into her testbench



Enabling Collaboration using Python

- Python is a common language
- Open Source Development
- System Verilog Constructs well proven over years

```
@vsc.covergroup
class my_weak_pht_cnt_cg(object):
   def __init__(self):
      self.with_sample(
        a = uint8_t(5)
        b = uint8_t(5)
      )
      self.cp1 = vsc.coverpoint(self.a, bins=dict(
          A = vsc.bin_array([],1,2,3,4)),
          illegal_bins=dict(
              illegal_Val = vsc.bin(5)
          )
      self.cp2 = vsc.coverpoint(self.b, bins=dict(
          A = vsc.bin_array([],[5,31]))
      )
      self.cp1x2 = vsc.cross([self.cp1, self.cp2])
```

Credits to Matt Balance' PyVSC library:
https://pyvsc.readthedocs.io/en/latest/introduction.html

accellera
SYSTEMS INITIATIVE

Implement a e2e prototype

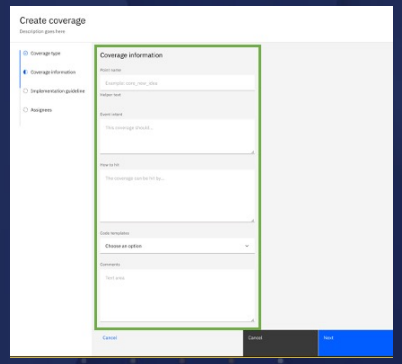
Observe

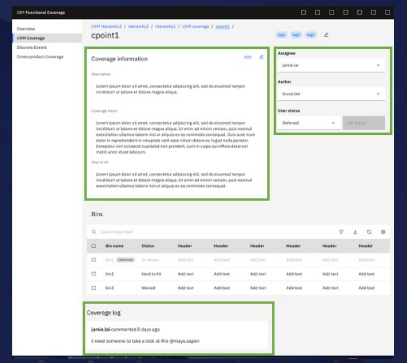
Reflect

Make

accellera
SYSTEMS INITIATIVE

Develop an end-2-end prototype in 2 weeks

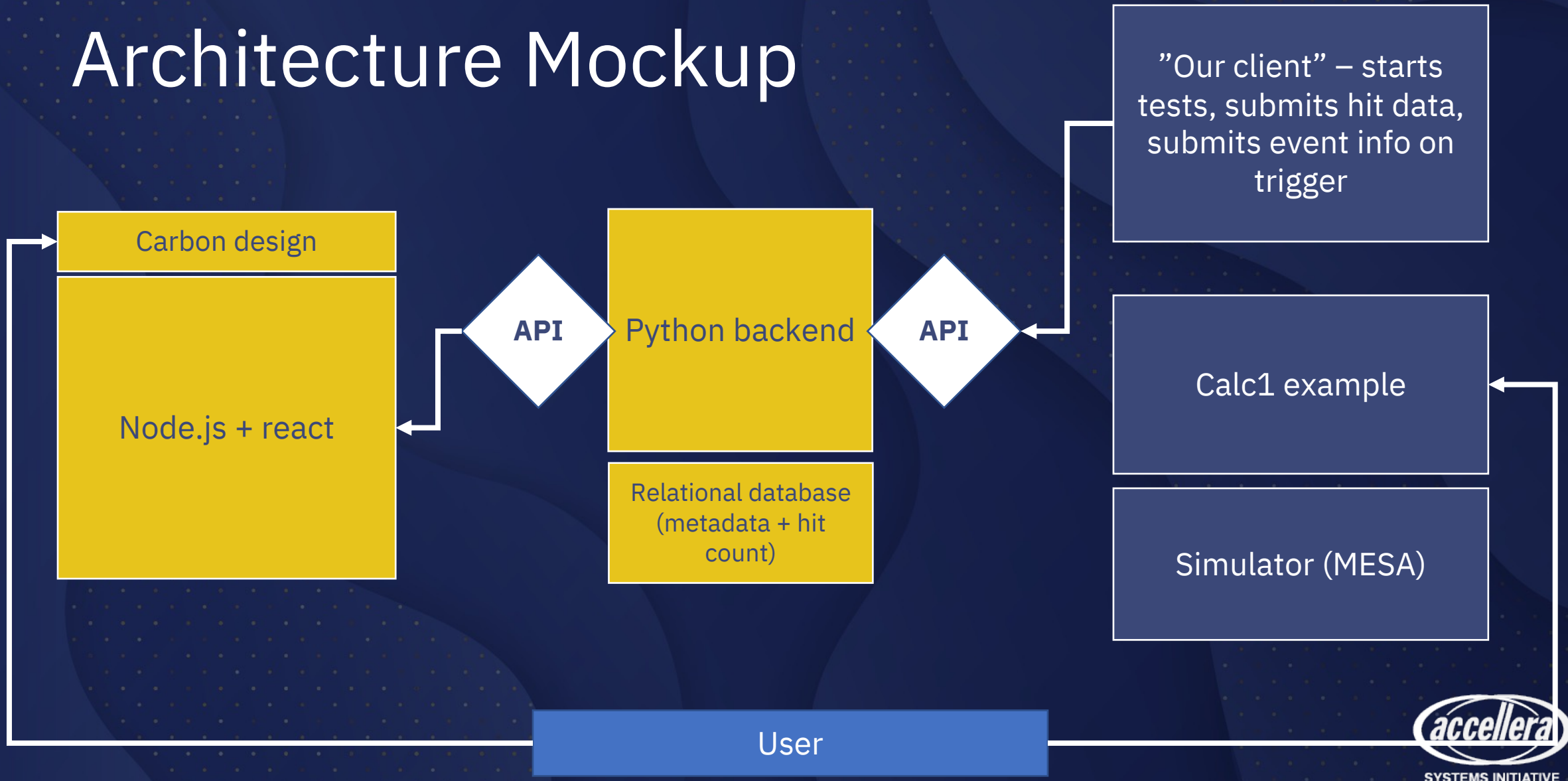




Define Implement Collect Analyze

accellera SYSTEMS INITIATIVE

Architecture Mockup



accellera SYSTEMS INITIATIVE

Summary

Enterprise Design Thinking is key

- ✓ **With the user in the loop**

UX design is key to

- ✓ Deeply understand the developers‘ needs
- ✓ taking a holistic view at the process
- ✓ fast prototyping
- ✓ Drive usability testing

accellera
SYSTEMS INITIATIVE

Conclusions

- ✓UX design and Prototyping allow fast iterations
- ✓Can change the way how we work!
  - better
  - better together
- ✓ Allows the developer to take full advantage of the available technology
- ✓ Can significantly improve the overall develpment effort and schedule

accellera SYSTEMS INITIATIVE

Questions?

Jamie Lai
UX designer, IBM Z Systems
jamie.lai@ibm.com

Bodo Hoppe
Distinguished engineer, IBM Z Hardware
bohopp@de.ibm.com

accellera
SYSTEMS INITIATIVE