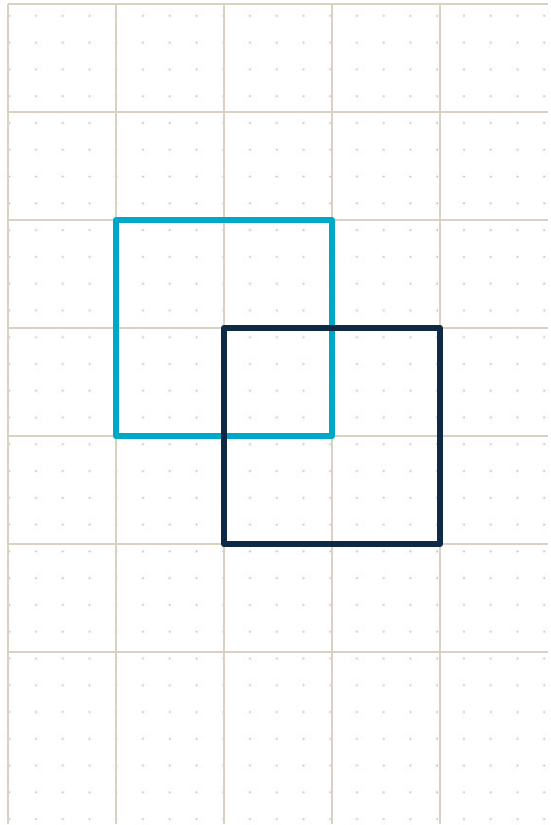
NIST OSCAL MONTHLY WORKSHOP SERIES

Assessing Critical Infrastructure with OSCAL



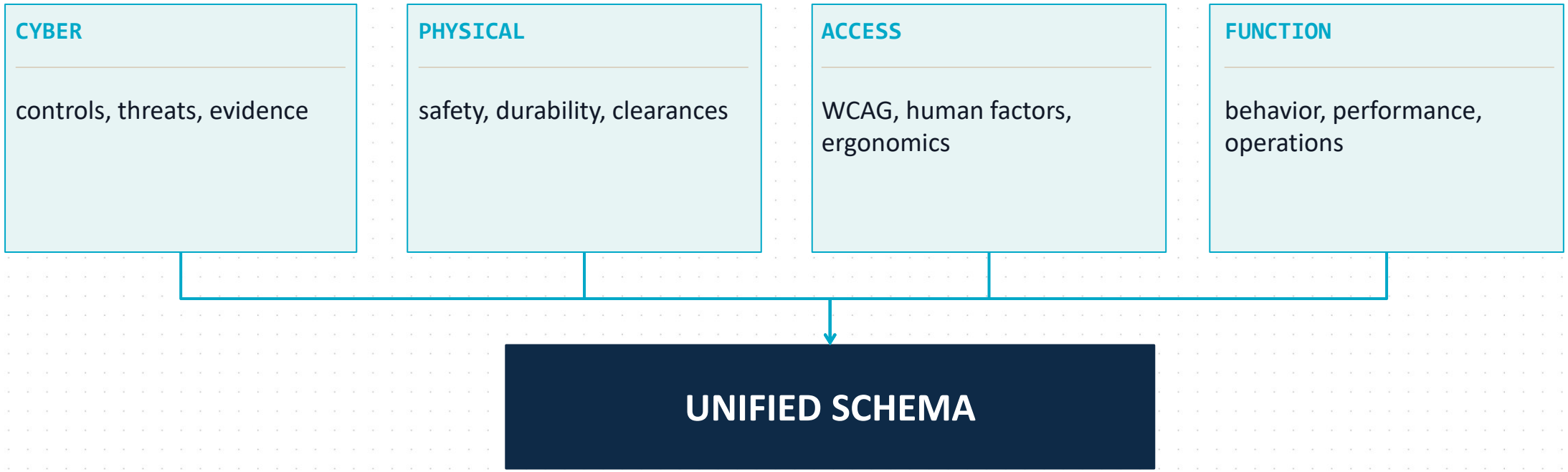
OSCAL beyond security requirements
Elections case study

John Dziurłaj

Senior Solutions Architect · The Turnout
https://rabetv.org

FRAME THE GAP

Critical infrastructure needs one requirements-to-assessment thread

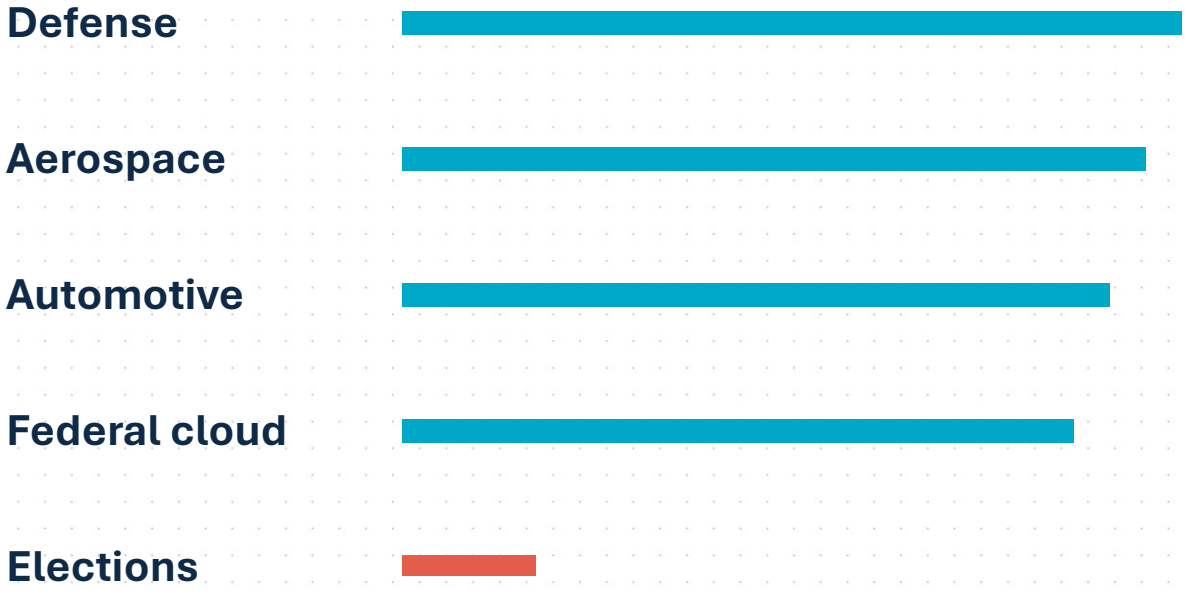


Orchestrate requirements, implementation claims, test methods, evidence, and findings in one machine-readable lifecycle.

FRAME THE GAP

Election infrastructure requirements sit in static prose

AUTOMATED, MACHINE-READABLE SPECIFICATIONS



THE ELECTION GAP

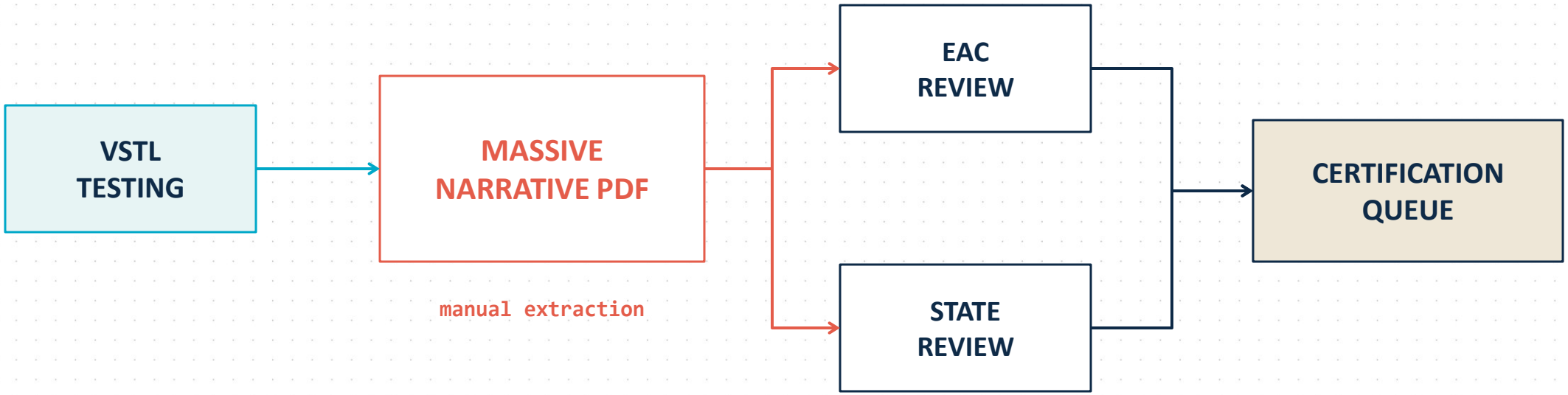
- Word templates
- Email passes
- Spreadsheet tracking
- Static PDFs

Manual requirements management compounds compliance bottlenecks, vendor lock-in, and audit fatigue.

Election systems combine high assurance with 50-state variability



Narrative test reports turn every review into manual data extraction



Minor software updates trigger monolithic re-testing—creating multi-month evaluation queues.

TESTING IMPERATIVE

VVSG 2.0 shifts compliance from prescribed devices to allocatable functions

VVSG 1.0 / 1.1

RULE IS ATTACHED TO A NAMED DEVICE

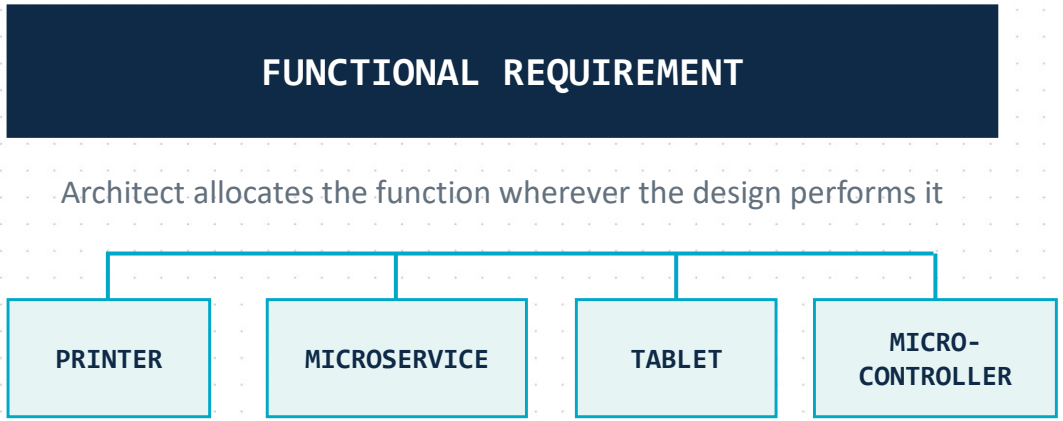
01 OPTICAL SCANNER
prescribed behavior

02 CENTRAL TABULATOR
prescribed behavior

03 BALLOT MARKING DEVICE
prescribed behavior

VVSG 2.0

REQUIREMENTS DEFINES A SYSTEM FUNCTION

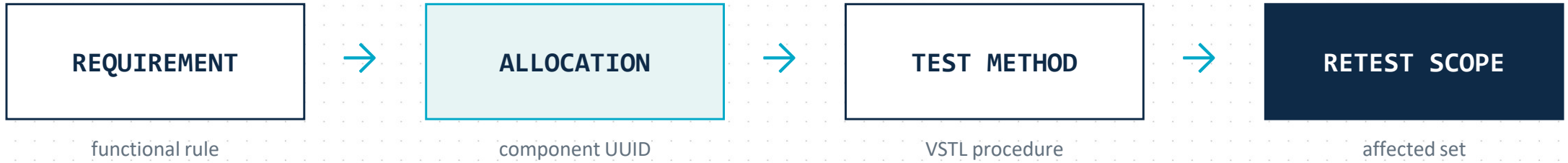


Testing moved from predictable device boundaries to architecture-dependent test boundaries.

TESTING IMPERATIVE

Traceability turns architecture into an executable test map

Each link narrows the test laboratory’s physical and logical test boundary.



PROCESS

Model relationships

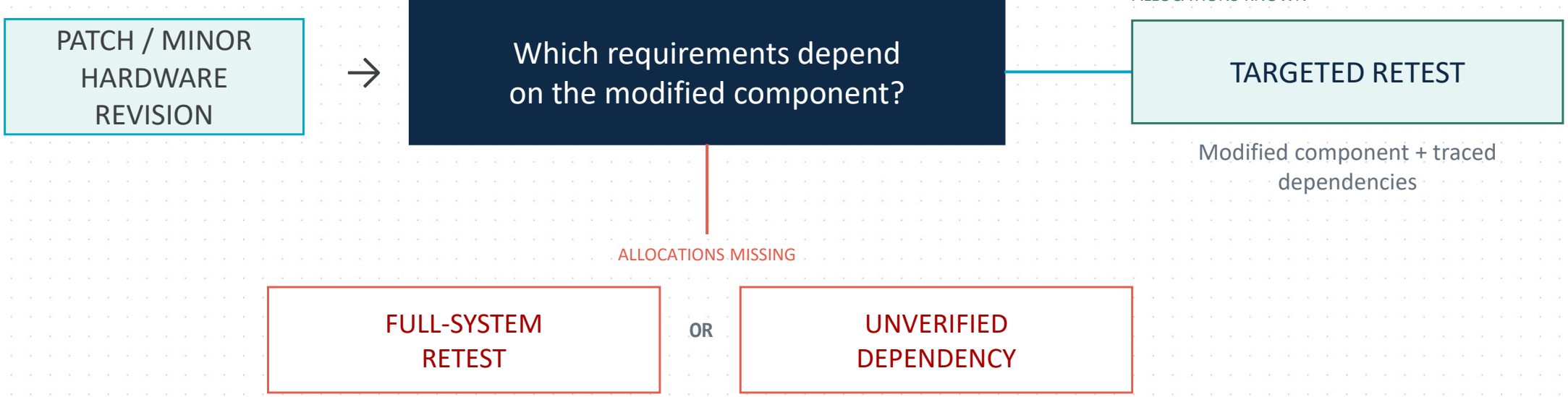
OPERATIONAL REALITY

The allocation determines what the lab must test

TESTING IMPERATIVE

Without allocation data, every change expands the lab's test surface

The same function may reside in an HSM · microservice · tablet OS · peripheral microcontroller



VARIABLE TEST SURFACE

Physical and logical boundaries change with vendor architecture.

EVALUATION BLAST RADIUS

A small revision can implicate every linked requirement.

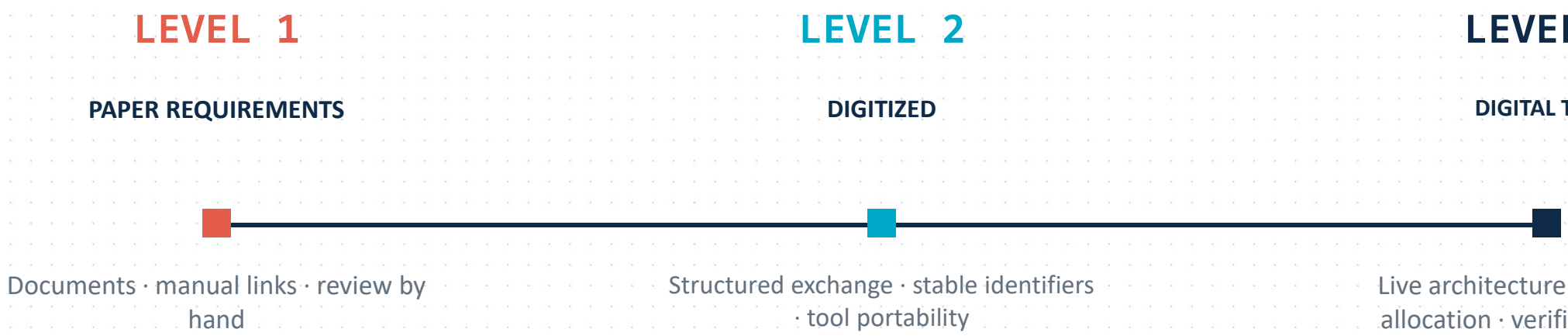
TESTING INEFFICIENCY

Technology providers absorb full-retest cost.

Existing Requirement Capture

REQUIREMENTS MATURITY

Requirements maturity rises from paper records to a traceable digital twin



Maturity is measured by retained traceability—not file format alone

Each step makes requirements more machine-readable, reusable, and directly connected to architecture and verification.

COMPARE THE FORMATS

ReqIF moves requirements cleanly—but stops before assessment execution

WHAT REQIF DOES WELL

OPEN STANDARD

OMG ReqIF 1.2

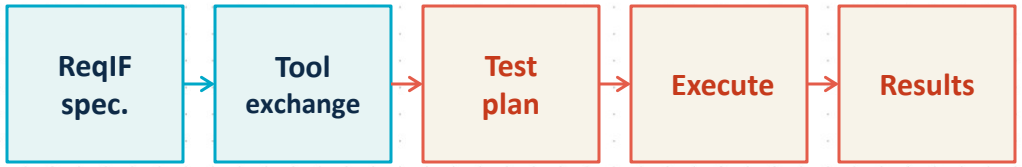
ORIGIN

German automotive · HIS / ProSTEP iViP

EXCHANGE

Requirements, attributes, hierarchies, change history

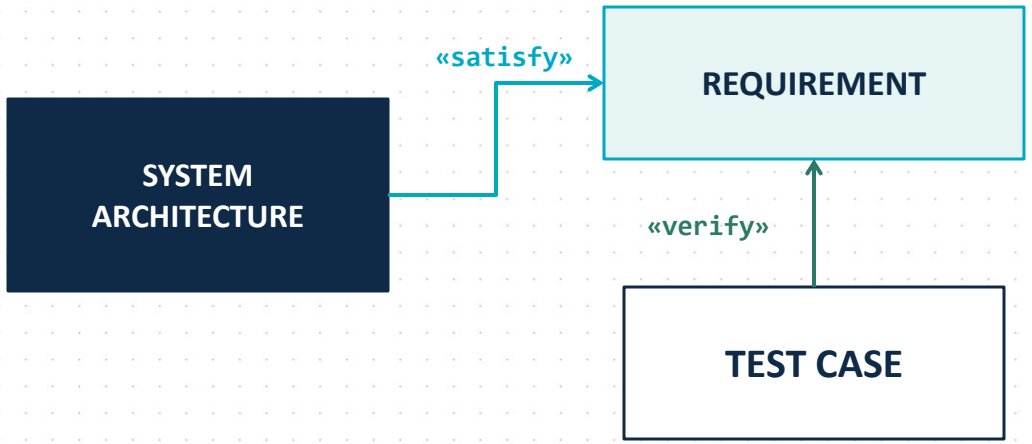
WHERE COMPLIANCE BEGINS TO BREAK



ReqIF transports specifications without vendor lock-in; it does not natively model test execution, assessment results, or continuous monitoring.

COMPARE THE FORMATS

SysML traces design intent rigorously—but not compliance results



BDD · IBD · Activity · Sequence · Requirement diagrams

MODEL-BASED SYSTEMS ENGINEERING

- **Core strength**
  Architectural decomposition and explicit trace relationships.
- **Adoption cost**
  High learning curve and expensive toolchains.
  (Changing with SysML v2)
- **Compliance gap**
  No standardized evaluation or reporting schema.

COMPARE THE FORMATS

OSCAL spans the lifecycle that SysML designs and ReqIF transports

CAPABILITY	SysML	ReqIF	OSCAL
System architecture	STRONG	LIMITED	COMPONENT-SCOPED
Requirement interchange	PARTIAL	STRONG	CATALOG / PROFILE
Assessment planning	PARTIAL	NONE	NATIVE
Multi-assessor results	NONE	NONE	NATIVE
Evidence & monitoring	NONE	NONE	NATIVE

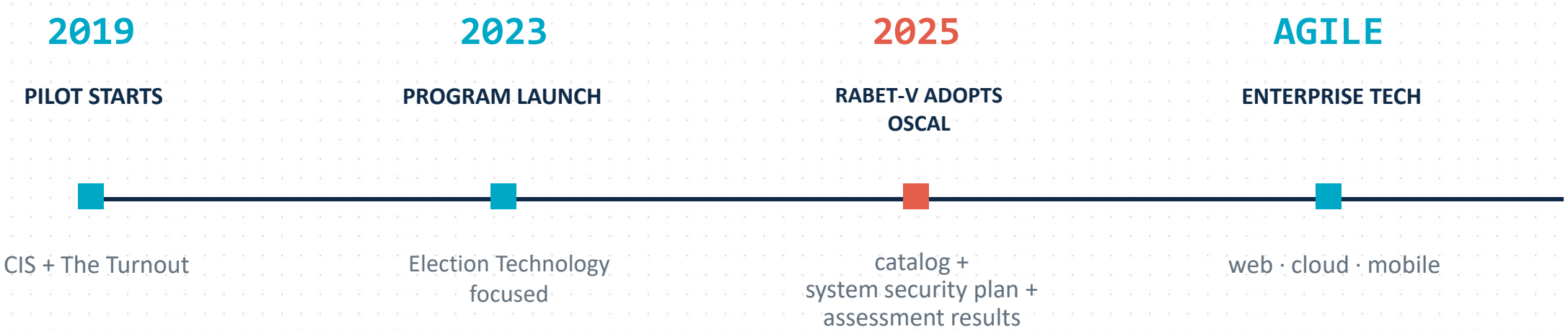
Positioning

SysML models design. ReqIF transports text. OSCAL structures implementation, assessment, evidence, and results.

OSCAL for Requirements

ELECTION CI

RABET-V made OSCAL native to agile election verification



Rapid Architecture-Based Enterprise Technology Verification

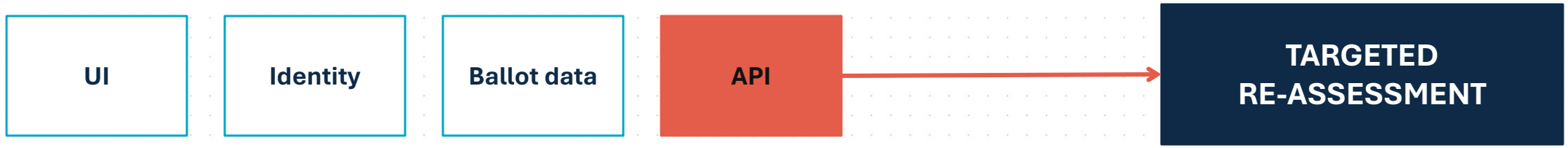
Continuous, architecture-aware verification for systems that change faster than legacy certification cycles.

https://rabetv.org

ELECTION CI

Component scoping collapses re-testing from whole systems to changed modules

SYSTEM DECOMPOSITION



Only the changed module re-enters assessment

LEGACY MODEL

Full-system retest

COMPONENT MODEL

Modified modules only

OPERATIONAL RESULT

Months → days

OSCAL FOR REQUIREMENTS

OSCAL's primitives are domain-agnostic

COMMON MISCONCEPTION

"OSCAL is only a container for SP 800-53 and FedRAMP."

SCHEMA REALITY

`catalog`

`control`

`param`

`props`

Domain-agnostic meta-models

Engineering opportunity: model functional, physical, accessibility, and interoperability constraints in standard OSCAL.

BEYOND 800-53

VVSG accessibility rules expose OSCAL's non-security potential

VVSG 2.0 · PRINCIPLE 8

Accessibility is measurable engineering data—not prose alone.

4.5:1	RATE	mm / dp	AREA
VISUAL	AUDIO	TACTILE	PHYSICAL
contrast ratios	playback speed	button dimensions	clearance zones

The schema challenge is clean expression of physical and human-computer interaction metrics in XML/JSON.

OSCAL FOR REQUIREMENTS

Standard control primitives can encode functional and human-factors rules

```
controls:
    - id: vvsg-7.1-c
      class: control
      title: Default contrast
      params:
        - id: contrast-ratio-minimum
          class: accessibility
          props:
            - name: unit
              value: luminosity-contrast-ratio
      parts:
          - id: vvsg-7.1-c-stmt
            name: statement
            prose: …
```

IDENTITY

— **control.id + title**

CLASSIFICATION

— **props name / value**

THRESHOLDS

— **param blocks**

REQUIREMENT

— **statement part**

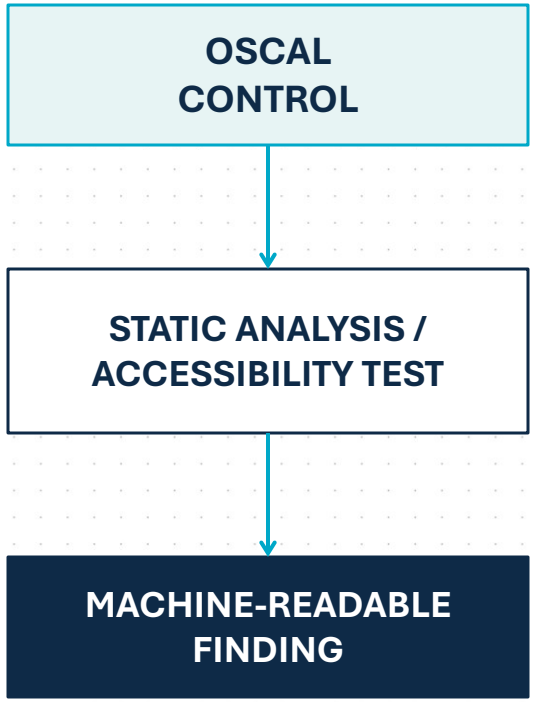
OSCAL FOR REQUIREMENTS

WCAG contrast becomes machine-testable when encoded as an OSCAL parameter

WCAG 2.1 · SUCCESS CRITERION 1.4.3

```
<implemented-requirement control-id="vvsg-7.1-c">
        <prop name="implementation-status" value="implemented"/>
        <prop name="measured-contrast-ratio" value="12.6:1"/>
        <set-parameter param-id="contrast-ratio-minimum">
                <value>10:1</value>
        </set-parameter>
</implemented-requirement>
```

INGESTION PATH



OSCAL FOR REQUIREMENTS

OSCAL parameters express physical thresholds without schema changes

HARDWARE PROFILE PARAMETERS

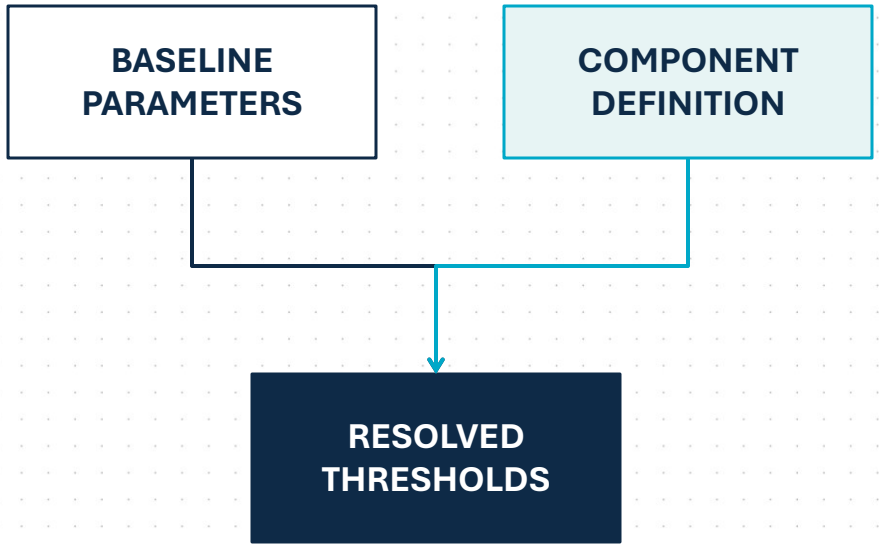
`touch-target-min`

`≥ 48 × 48 dp` interactive control

`headphone-jack-location`

`front / side` physical placement

BINDING



Schema stays stable; values change with configuration.

TRACEABILITY

Component definitions bind controls directly to technical units



COMPONENT PRIMITIVE

Technical components

software · hardware · service

Documentary components

policy · process · procedure

Native mapping

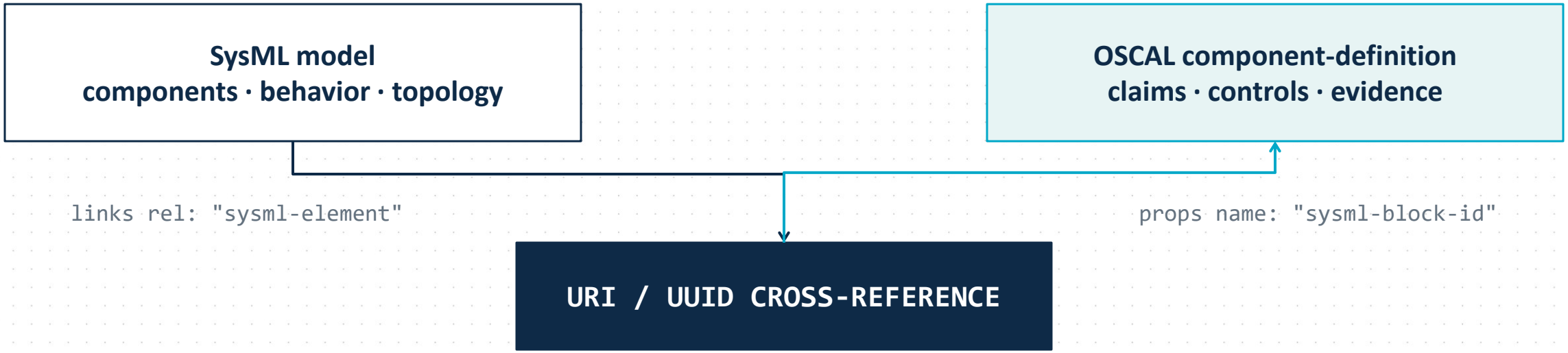
sub-components · binaries · modules

TRACEABILITY

Linked data lets SysML and OSCAL coexist without duplicating authority

ARCHITECTURE AUTHORITY

COMPLIANCE AUTHORITY



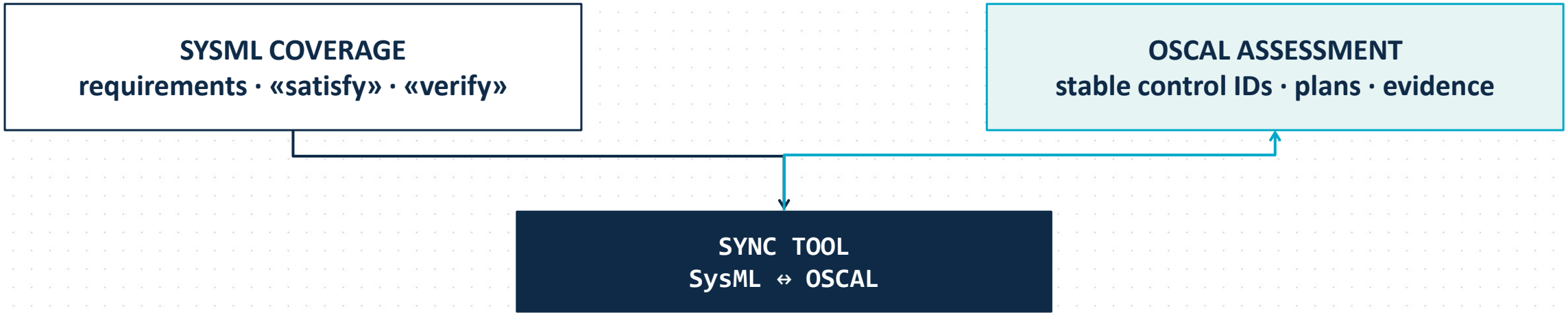
Best fit: complex hardware/software systems already using full MBSE toolchains.

TRACEABILITY

Coverage analysis dies if requirements live in only one tool

ARCHITECTURE STAYS IN MBSE Tool

ASSURANCE STAYS IN OSCAL



Cameo DataHub proves the pattern: DOORS ↔ SysML

SYNC OBJECTS

control-id

↔ requirement element

component-uuid

↔ block

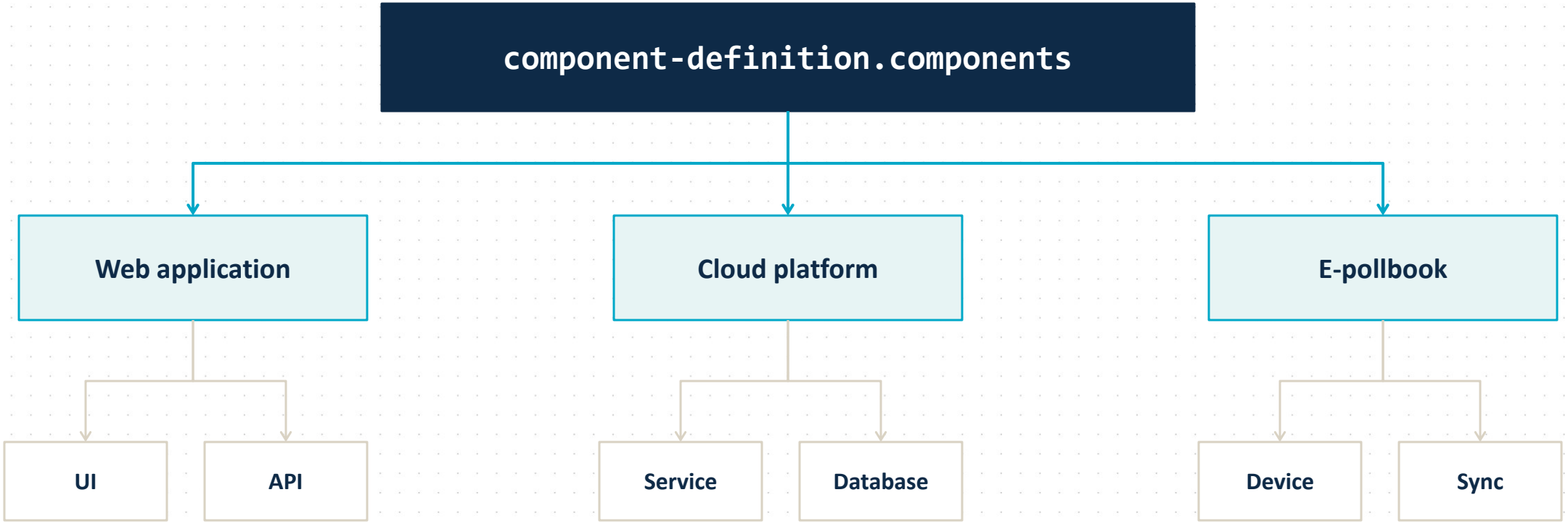
param values

↔ constraint / tag

Two copies drift. Word is not a third source of truth. IDs stay shared.

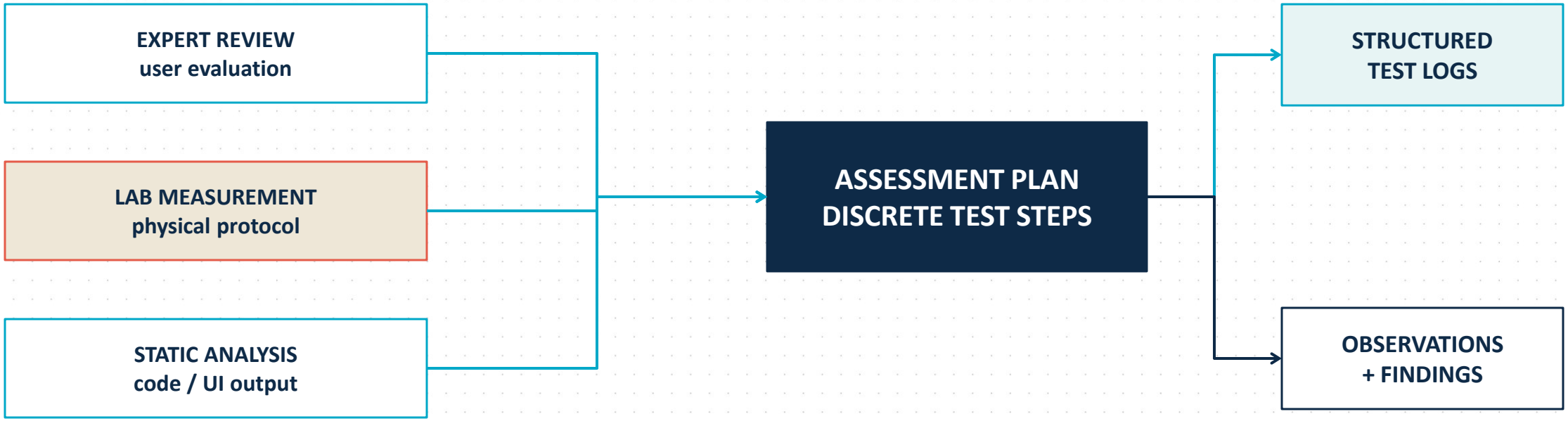
TRACEABILITY

Compliance-focused systems can model architecture directly in OSCAL



Direct XML/JSON authoring removes heavyweight MBSE tooling when architecture exists primarily to scope compliance.

Assessment plans can orchestrate lab, expert, and automated methods



One execution schema harmonizes physical measurements and automated test outputs.

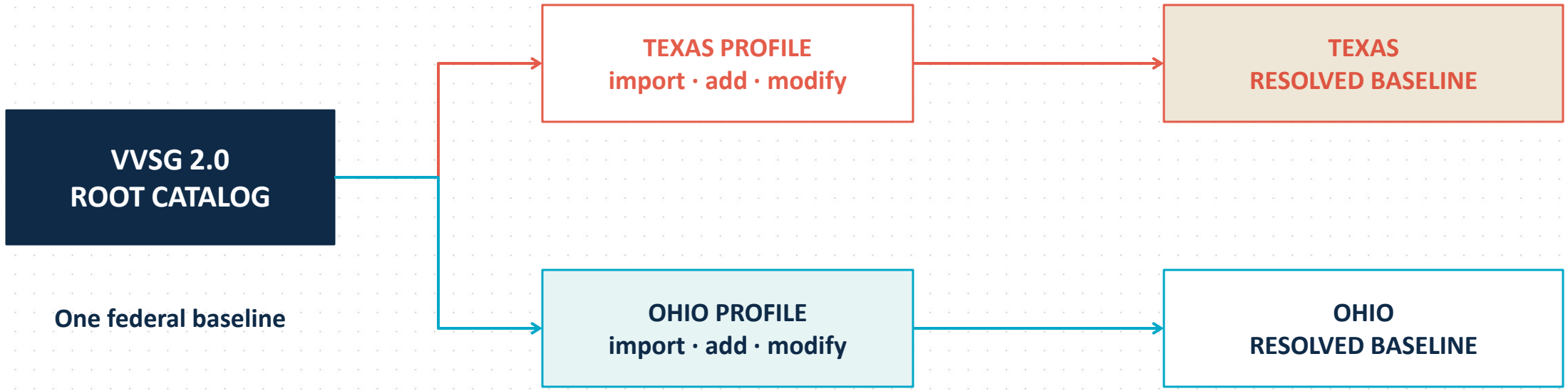
BEYOND 800-53

Separate disciplines can emit one machine-readable assessment result



Diverse technical findings stay separately attributable—but aggregate into one executable record.

Profiles resolve federal baselines into state-specific overlays



Dynamic resolution avoids split code bases while preserving jurisdiction-specific rules.

BEYOND 800-53

Machine-readable links make traceability and impact analysis continuous



← backward traceability

forward traceability →

CHANGE IMPACT

A code or component change can flag affected test plans and assessment findings immediately.

VALIDATE

complete links

TECHNICAL DISCUSSION

OSCAL bridges requirements engineering and executable assurance.

01

Connects SysML/ReqIF intent to compliance execution.

02

Binds requirements to components, plans, evidence, and findings.

03

RABET-V shows machine-readable models can accelerate verification.

Q&A

John Dziurłaj · Senior Solutions Architect · The Turnout
https://rabetv.org/