Requirements-driven Verification Methodology for Standards Compliance

Serrie-justine Chapman (TVS) serrie@testandverification.com
Mike Bartley (TVS) mike@testandverification.com
Darren Galpin (Infineon)

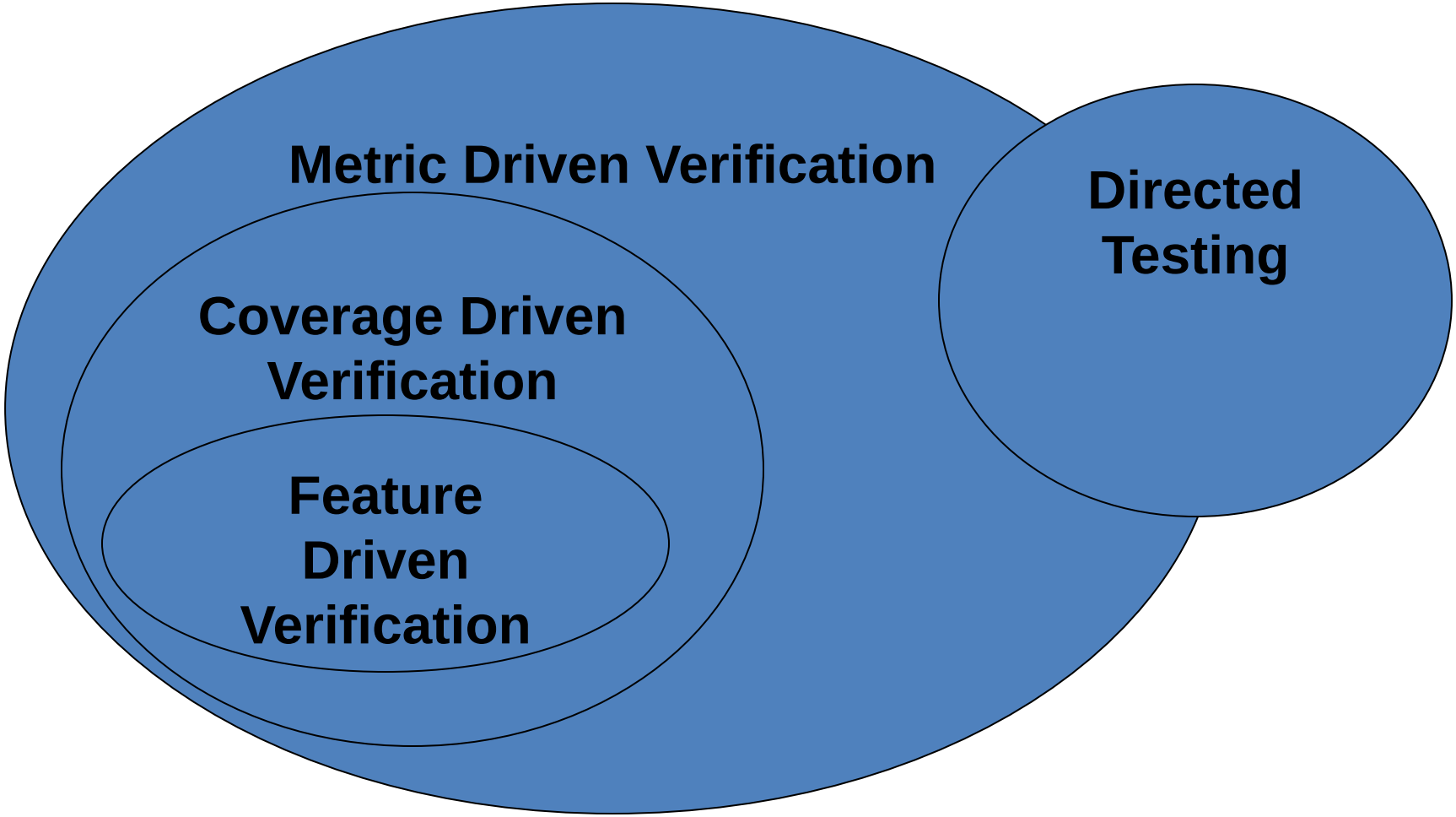
accellera SYSTEMS INITIATIVE

2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION EUROPE

Agenda

- Motivation
  - Why Requirements Driven Verification?
- Introduction to Safety
  - The Safety Standards
  - What do we need to do? And deliver?
- Supporting Requirements Driven Verification with Advanced Verification Techniques
- Tool Support
- Advantages of Requirements Driven Verification

An Overview of Verification Approaches



- **Constrained random verification**
- **Assertion-based verification.**
- **Formal property based verification.**

Why Requirements Driven Verification?

- Metric Driven Verification
  - Allows us to define targets
  - And monitor progress
- Coverage Driven Verification
  - Most common metric driven verification approach
  - Code Coverage
  - Functional coverage
    - Might be related to features
- Feature Driven Verification
  - Features MIGHT be related to spec
    - Is that relationship captured?
  - Are features related to requirements?

Safety Standards

- IEC61508: Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems
- DO254/DO178: Hardware/Software considerations in airborne systems and equipment certification
- EN50128: Software for railway control and protection systems
- IEC60880: Software aspects for computer-based systems performing category A functions
- IEC62304: Medical device software -- Software life cycle processes
- ISO26262: Road vehicles – Functional safety

Introduction to Safety

- The life cycle processes are identified
- Objectives and outputs for each process are described
  - Objectives are mandatory
  - But vary by Integrity Level
  - For higher Integrity Levels, some Objectives require Independence

Key Elements

- Plans & Standards
- Requirements
- Design Specifications
- Reviews and Analyses
- Testing (against specifications)
  - At different levels of hierarchy
- Test Coverage Criteria
- Requirements Traceability
- Independence

Key Deliverables

- Hardware Verification Plan
- Validation and Verification Standards
- Hardware Traceability Data
- Hardware Review and Analysis Procedures
- Hardware Review and Analysis Results
- Hardware Test Procedures
- Hardware Test Results
- Hardware Acceptance Test Criteria
- Problem Reports
- Hardware Configuration Management Records
- Hardware Process Assurance Records

REQUIREMENTS ENGINEERING DEFINITIONS

Requirement:

1. A condition or capability needed by a user to solve a problem or achieve an objective
2. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification or other formally imposed documents
3. A documented representation of a condition or capability as in (1) or (2)

[IEEE Std.610.12-1990]

Stakeholder:

- A stakeholder of a system is a person or an organization that has an (direct or indirect) influence on the requirements of the system

* All Definitions taken from IREB

REQUIREMENTS ENGINEERING CORE ACTIVITIES

Requirements Engineering is a systematic and disciplined approach to the specification and management of requirements with the following goals:

- **Knowing the relevant requirements, achieving a consensus among the Stakeholders about these requirements, documenting them according to given standards, and managing them systematically**
- **Understanding and documenting the stakeholders' desires and needs, then specifying and managing requirements to minimize the risk of delivering a system that does not meet the stakeholders' desires and needs**

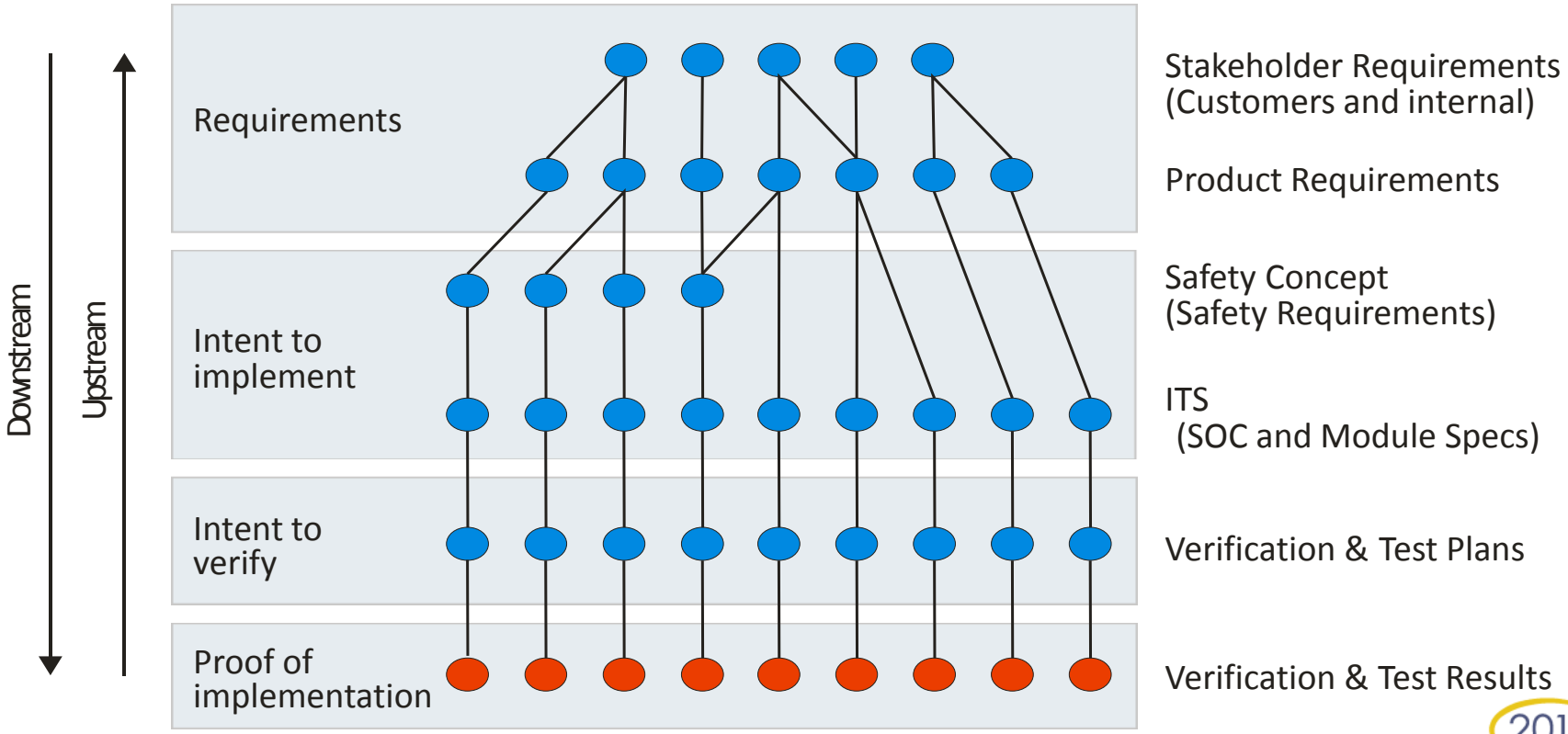
Four core activities :

- **Elicitation**
- **Documentation**
- **Validation and negotiation**
- **Management**

Requirements Traceability

- Documented:
  - all integrity levels/classes
- All requirements:
  - Tested or otherwise verified (Audit trail)

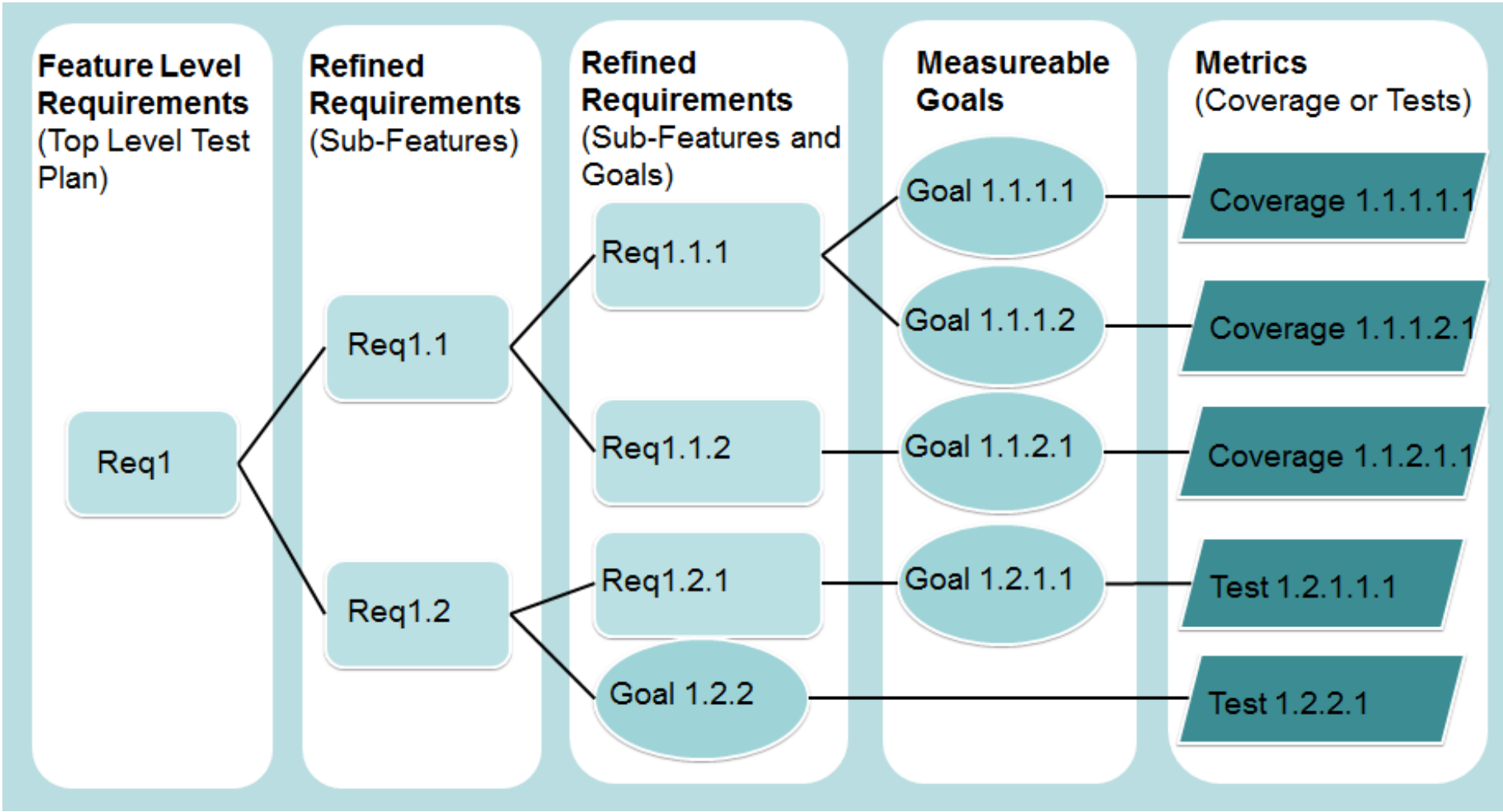
"Requirements Traceability = the ability to follow the life of a requirement, in both a backward and forward direction"



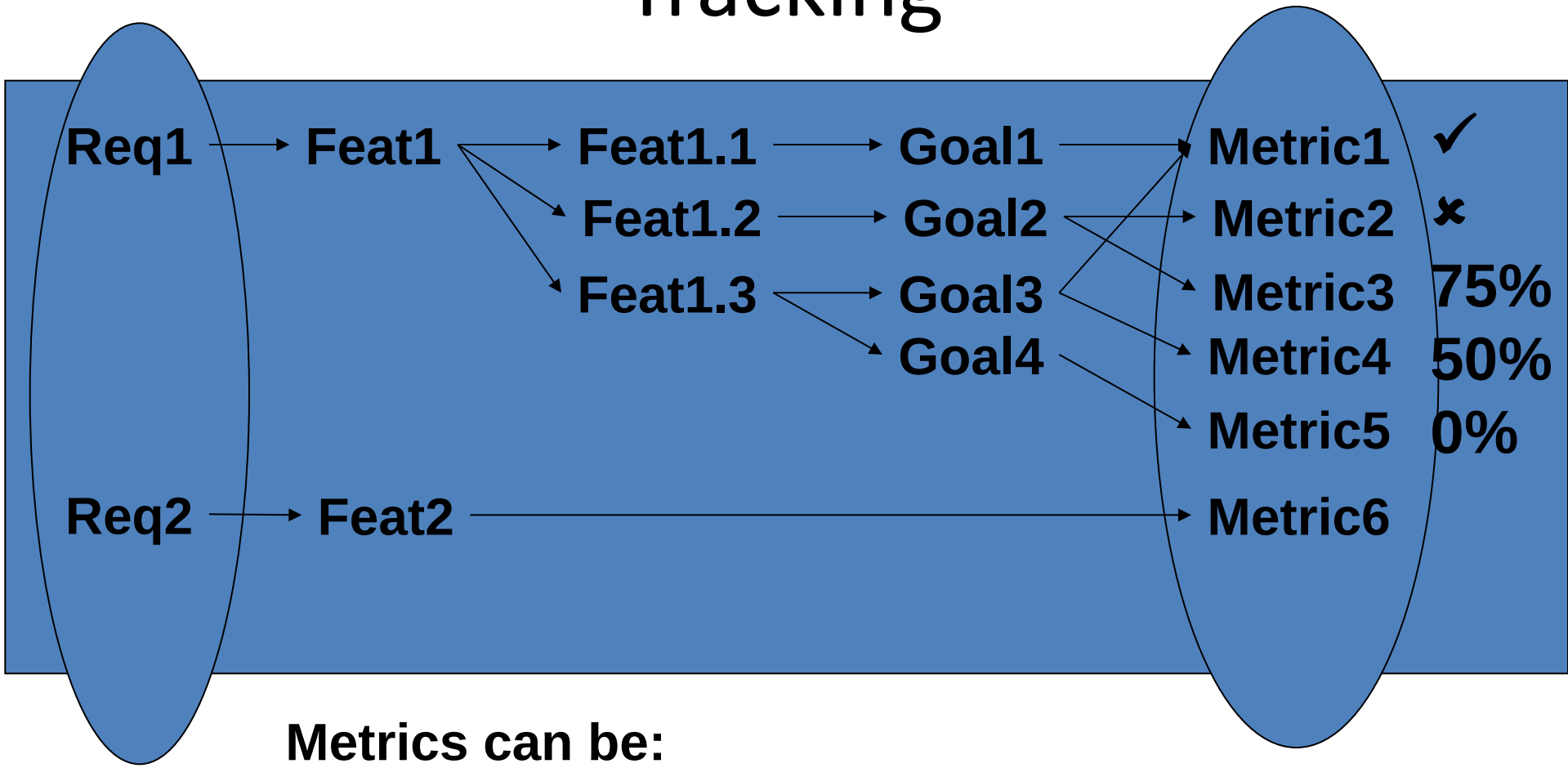
Supporting Advanced Verification

- Constrained random verification with automated checks based on models or scoreboards, etc.
- Coverage driven verification based on functional coverage models and code coverage metrics.
- Assertion-based verification.
- Formal property based verification.

Supporting Advanced Verification



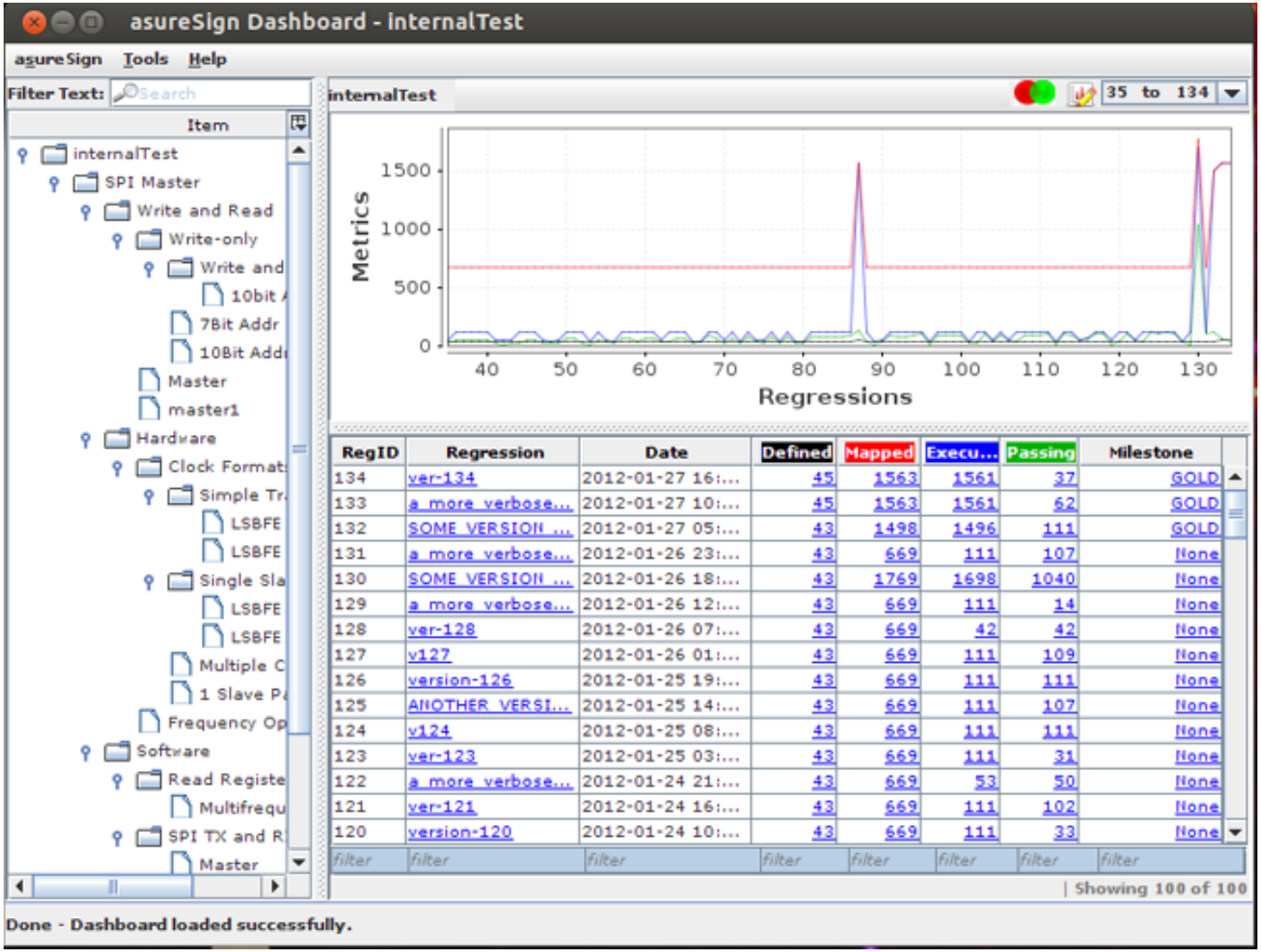
Tracking



Metrics can be:

- **From HW verification**
- **From Silicon validation**
- **From SW testing**

Track Progress on Requirements Signoff



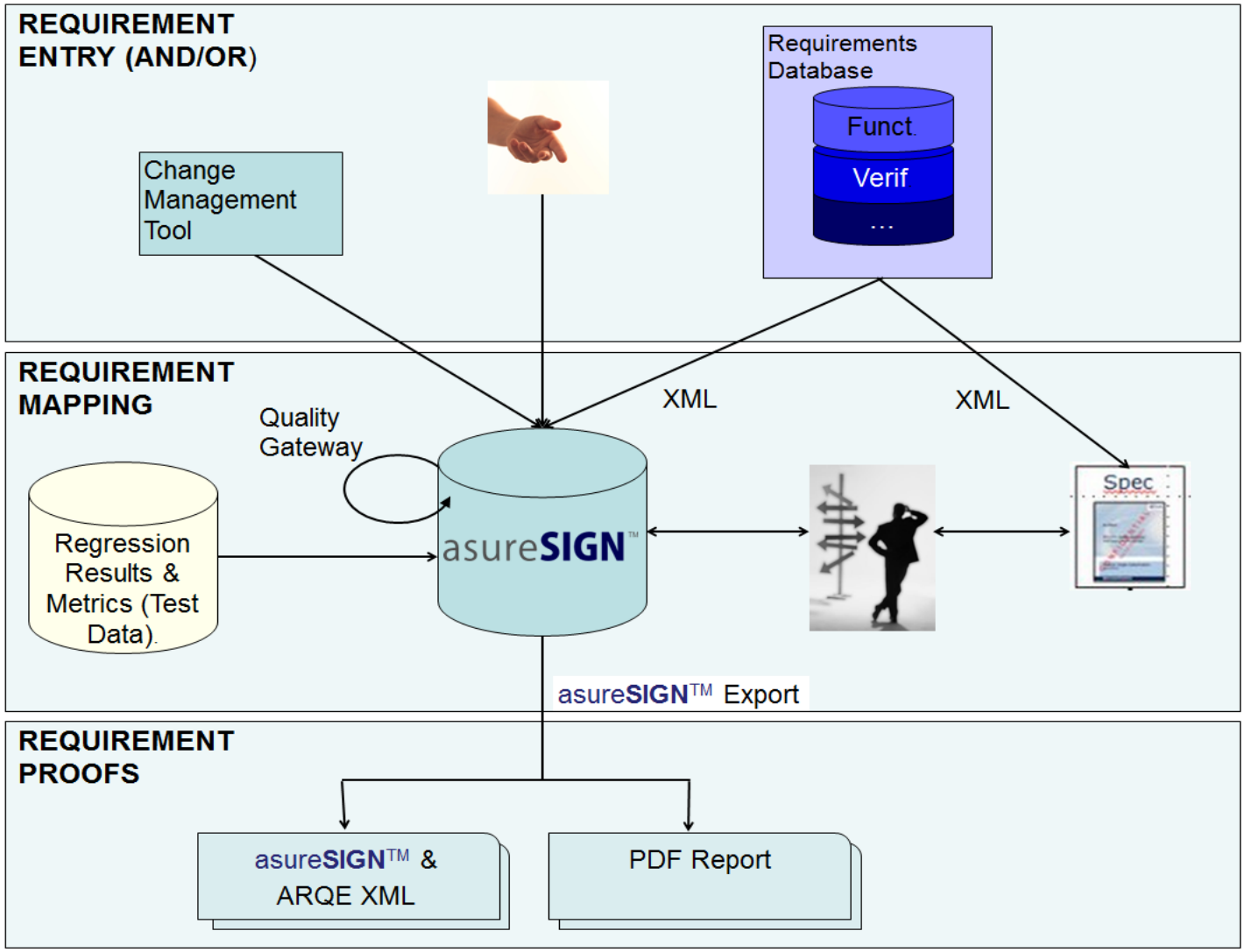
Supporting Hierarchical Verification

- A requirement might be signed off at multiple levels of hierarchy during the hardware development
  - Block
  - Subsystem
  - SoC
  - System
    - Including Software
  - Post Silicon

Tool Support Requirements

- Requirements -> test plan
- Data Integrity, hierarchy, data translation
- Change management – instant update
- Live database -> easy documentation
- Tailored Documented proof
- Allows reviews of implementation document against test plan
- Mapping
- Test management
- Compliance / Audit Management

asureSIGN Dataflow



Advantages of Requirements Driven Verif

- Requirements Management
- Verification Management
- Project Management
- Impact Analysis
- Product Line Engineering
- Variant management
- Improved Product Sign-Off

Conclusions #1

- Requirements Driven Verification
  - Compliance to various hardware (and software) safety standards
    - IEC61508: Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems
    - DO254/DO178: Hardware/Software considerations in airborne systems and equipment certification
    - EN50128: Software for railway control and protection systems
    - IEC60880: Software aspects for computer-based systems performing category A functions
    - IEC62304: Medical device software -- Software life cycle processes
    - ISO26262: Road vehicles – Functional safety
  - There are several other advantages
    - Identify test holes and test orphans
    - Track the status of the whole verification effort (planning, writing, execution)
    - Build historical perspective for more accurate predictions
    - Better reporting of requirements status
    - Support for
      - Risk-based testing
      - Prioritisation and Risk Analysis
      - Filtering Requirements based on Customers and releases
      - Impact analysis

Conclusions #2

- Advanced verification techniques can be deployed in Requirements Driven Verification
  - Requirements engineering tools to capture the verification plan & mapping
  - Verification management tools to automate collection of results
- More info
  - CRYSTAL http://www.crystal-artemis.eu/
  - White Paper http://www.testandverification.com/wp-content/uploads/tvs-white-paper-asureSIGN.pdf

Questions