NDIA 12th Annual Systems Engineering Conference

"Using Requirements Compliance Metrics to Identify Gaps Between the Technical Solution and Requirements"

October 2009

David Minchala
US Army ARDEC,
Bldg 12, RFAD
Picatinny Arsenal NJ, 07806-5000
(973) 724-
Edward.Dooley@us.army.mil

Frank Salvatore
High Performance Technologies, inc.
3159 Schrader Road
Dover NJ, 07801
(973) 442-6436 ext 249
fsalvatore@hpti.com

Outline

- Background
- State of Practice
- Improving the State of Practice
- Benefits
- Compliance Method and Tool
- Report on Progress and show data
- Summary

Background

Engineering projects that are completed on time and within budget most likely implement established "frozen" designs, e.g. roads, bridges, where there is limited opportunity to change requirements

When building new and complex systems:

- **Requirement changes are expected**
- **Requirement changes are common activities early in the lifecycle**
- **Material developers and stakeholders often "refine" the intended end-use of the system**

Background

Metrics on cost, schedule, and performance do not account for discontinuities between the defined requirements (the intent) and the delivered technical solution.

The US Army Armament Research Development Engineering Center (ARDEC) has devised a measurement and reporting method based on Requirements Engineering best practices to identify these discontinuities and facilitate fact-based management decisions.

State of Practice Metrics

Program / Project Managers (PMs) rely on various sets of metrics to:

- **Get an objective assessment of the project / program (Cost, Schedule, Performance)**
- **Formulate corrective actions**
- **Adjust budgets, schedules, and resources**

Program / Project sponsors, however, often measure program / project success or failure by met or missed:

- **Schedule**
- **Budget, and**
- **Requirements**

State of Practice Requirements Management

- **Best-practice Requirements Management (RM) requires measurement and collection of requirements metrics**
  - **Process Metrics (i.e. Change Frequency)**
  - **Requirements Metrics (i.e. # of Requirements allocated, approved, etc…)**
- **Requirements Management Reports**
  - **Traceability**
  - **Priority**
  - **Verification**
  - **Compliance**

State of Practice
Requirements Management

SYSTEM XYZ Requirements Compliance Matrix

Section	Requirement #	Requirement Text	Compliance				Rationale, comments of how the requirement was met
			Y	N	D	W	
3.2.4 Mobility	3.2.4.6. Braking	The propulsion subsystem shall enable the system to decelerate from maximum speed to full stop at a rate of 5 m/s2 with side drift not to exceed 2 m in 15 m on a dry, level, hard surface road.					
	3.2.4.7.						
	3.2.4.8.						
	3.2.4.14.						
	3.2.4.19.						

Compliance			Comments (Missing, unallocated)
Full	Partial	None	

Legend

Y - Yes, meets requirement

N - No, does not meet requirement

D - Deviation required

W - Waiver required

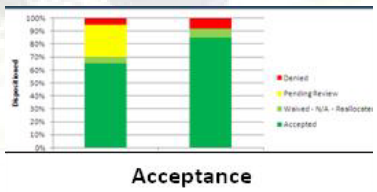
These matrices generally report the gaps between *intent* and *end-state*

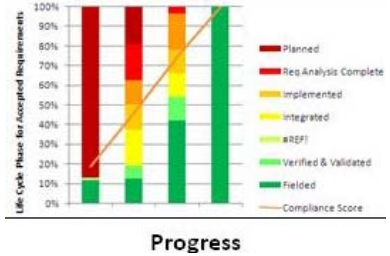
- Additionally there is no standard terminology or meaning

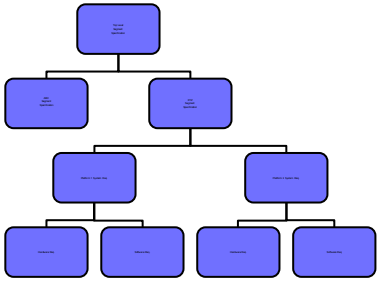
Improving the State of Practice

In our approach we take Requirements Compliance a step further by tracking progress in meeting the intent.

- **This approach provides a common language between management and developers.**







Benefits

PM visibility into implementation status

- A matrix will be maintained for each (sub)system, which will allow for metrics and reports to be generated against the system requirements. This will serve as a tool the PM can use to assess the current compliance of each (sub)system.

Facilitate communication between stakeholders

- The use of this approach will improve visibility into progress toward meeting program goals.
- Discrepancies can be discussed, clarified, resolved, documented and archived.

Help with Requirements Prioritization

- Can track incremental development with improved accuracy and identify issues with development progress sooner.

Implementation

Requirements Compliance Model defined

- The model is based on the DoD's Systems Engineering "V" approach to Systems / material development .
- The model will serve as the language that converts engineering phases to a compliance percentage.

Requirements Compliance Tool developed

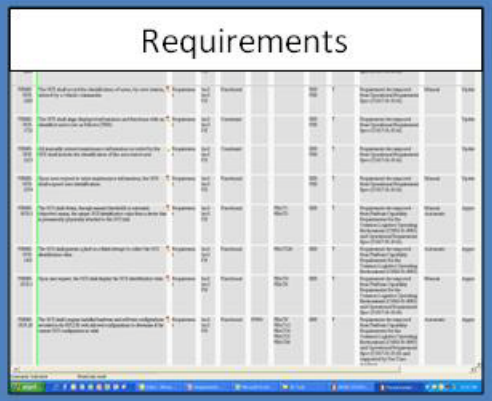
- A matrix has been constructed within DOORS which allows the following:
  - ✓ Direct linking to system/component specifications.
  - ✓ Ability to run reports to collect metrics on compliance.
  - ✓ Can export to Excel or other formats with ease.

Scripts constructed to run against DOORS Module

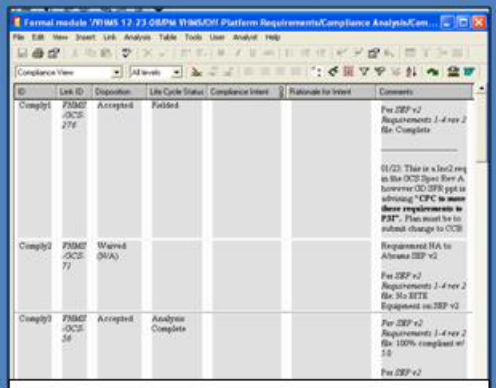
- This helps automate the process of measuring compliance.

Requirements Compliance Tool

Requirements



Assess Compliance



Compliance

Run Script to Generate Metrics

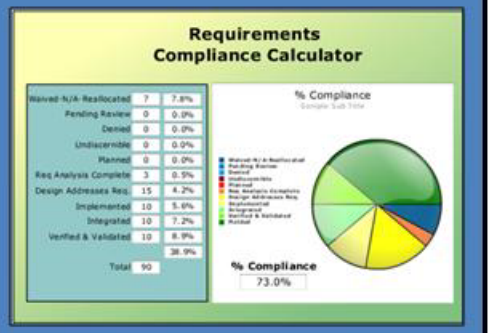
Script Automatically Generates Metrics

Criteria	Value	Score	Weight
Waived-N/A-Reallocated	7	7.8%	100%
Pending Review	0	0.0%	0%
Denied	0	0.0%	0%
Undiscernible	0	0.0%	0%
Planned	0	0.0%	5%
Req Analysis Complete	3	0.5%	15%
Design Addresses Req.	15	4.2%	25%
Implemented	10	5.6%	50%
Integrated	10	7.2%	65%
Verified & Validated	10	8.9%	80%
Fielded	35	38.9%	100%
Total	90		
% Compliance		73.0%	73.00%

	Date 1	Date 2
Fielded	10	10
Verified & Validated	0	5
Integrated	0	15
Implemented	0	10
Design Addresses Req.	0	10
Req Analysis Complete	0	15
Waived-N/A-Reallocate	5	7
Waiver Denied	5	8
Pending Review	5	0
Planned	75	15
Compliance Score	0.1875	0.4575

	Date 1	Date 2
Accepted	65	85
Waived - N/A - Real	5	7
Pending Review	25	0
Denied	5	8

Automatic Reporting



Progress

Acceptance

Assess Compliance

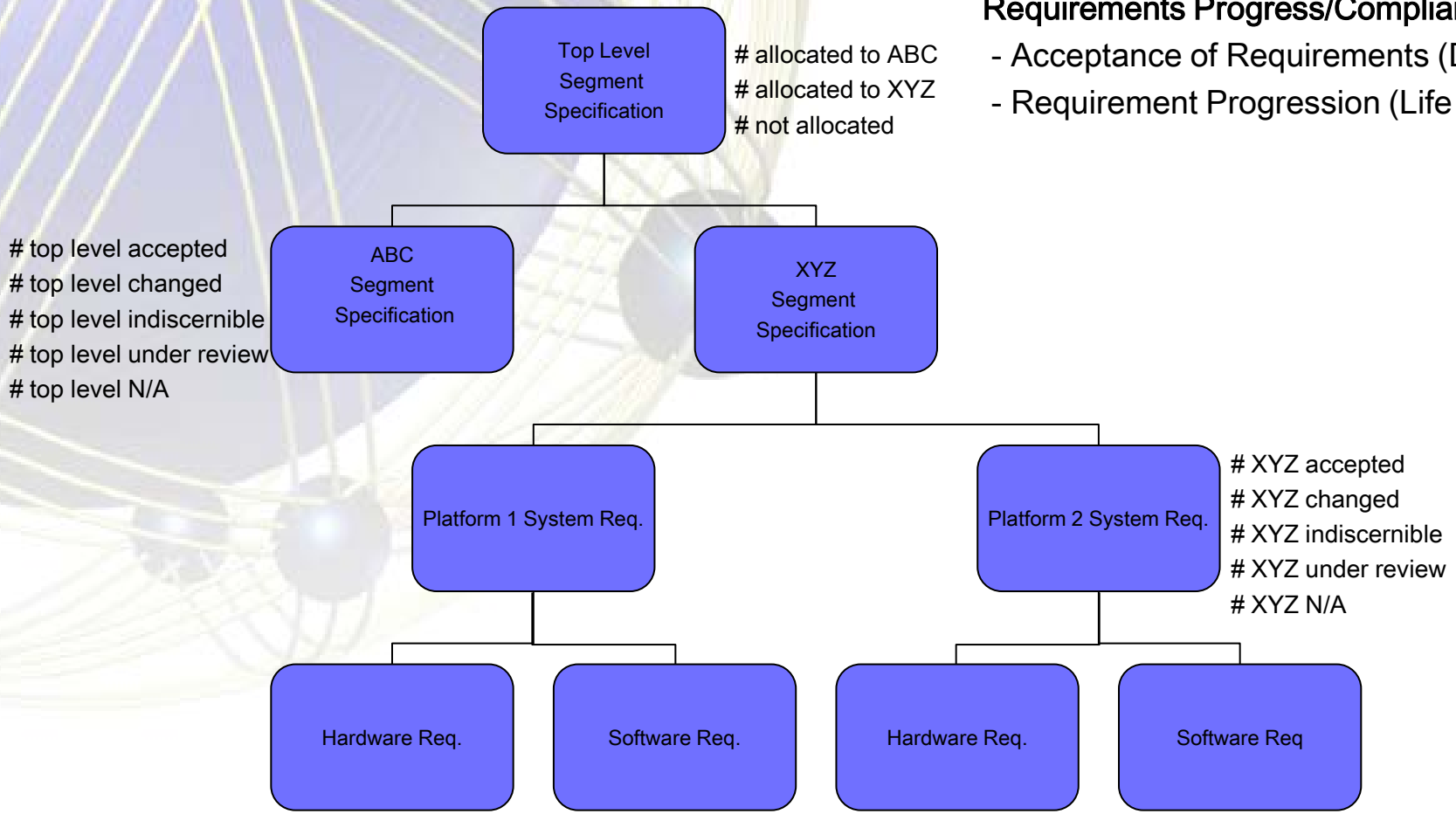
Capture Metrics

Report Results

Compliance Concept

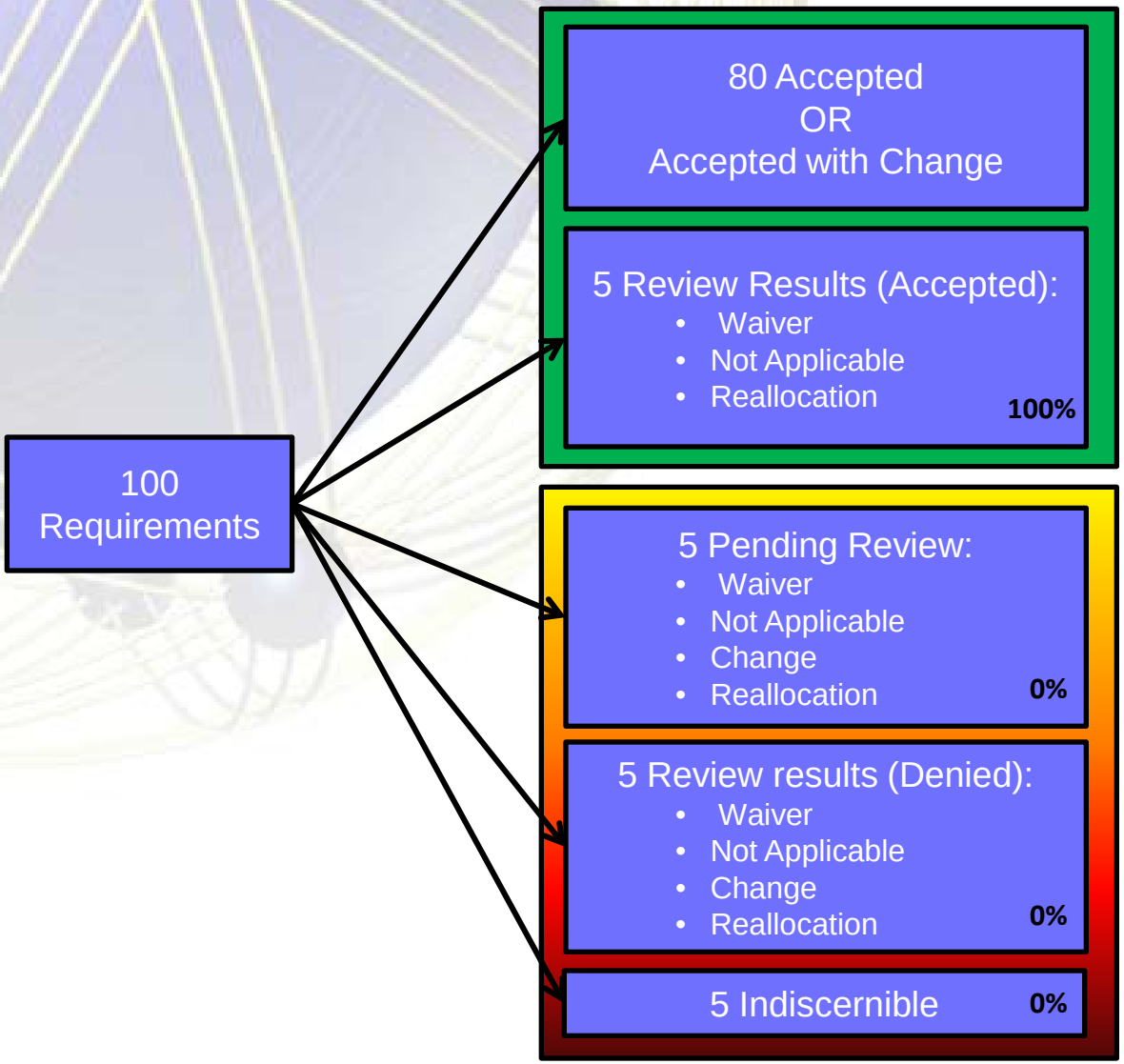
Requirements Progress/Compliance

- Acceptance of Requirements (Dispositioned)
- Requirement Progression (Life Cycle)



Metrics can be used to compare progress & compliance to planned activities and can be sorted by increment, build, priority, capability etc.

Disposition Progress



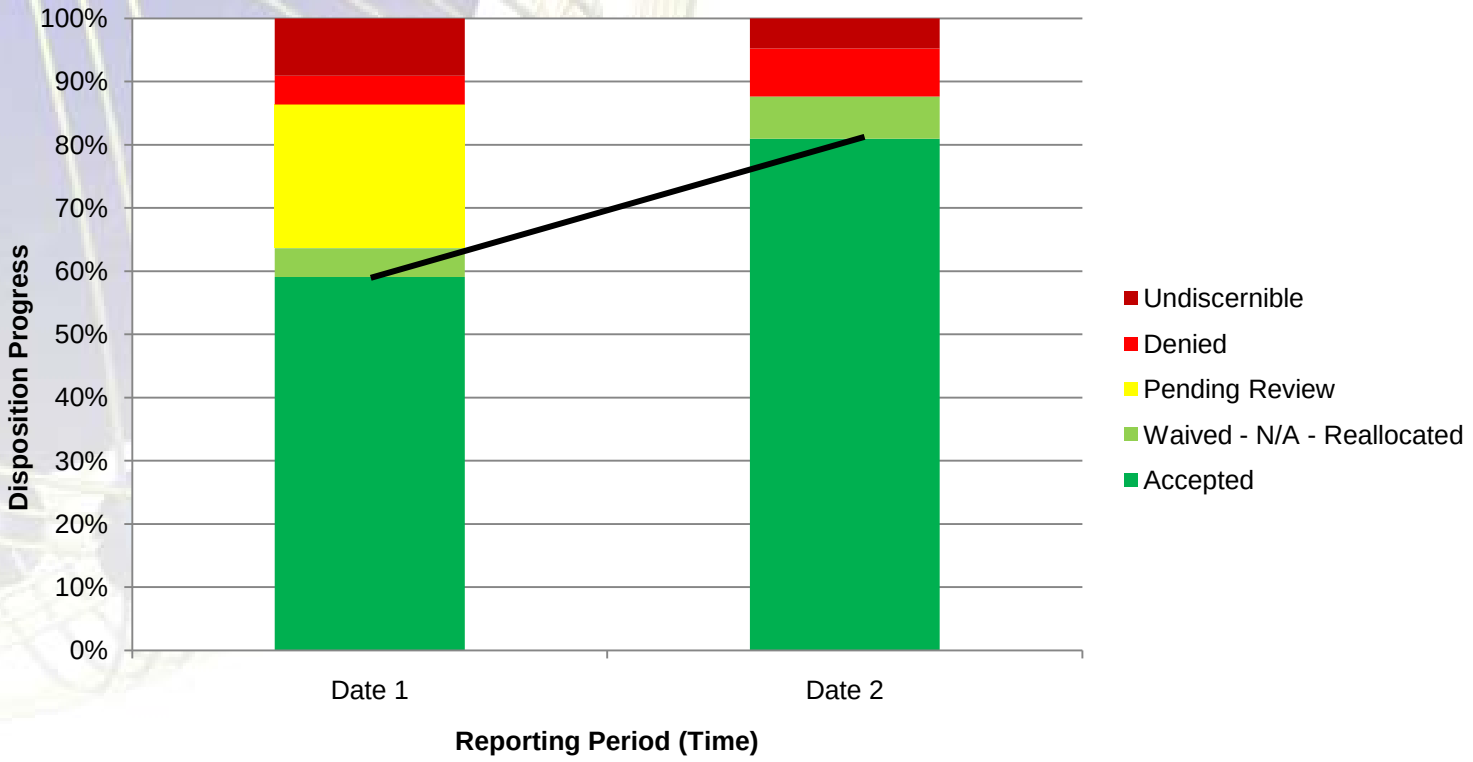
Disposition Progress

Percentage not Assigned to Accepted Requirements

This is deferred to life cycle compliance scoring.

Establishes basis for measuring the progress made towards accepting the requirements.

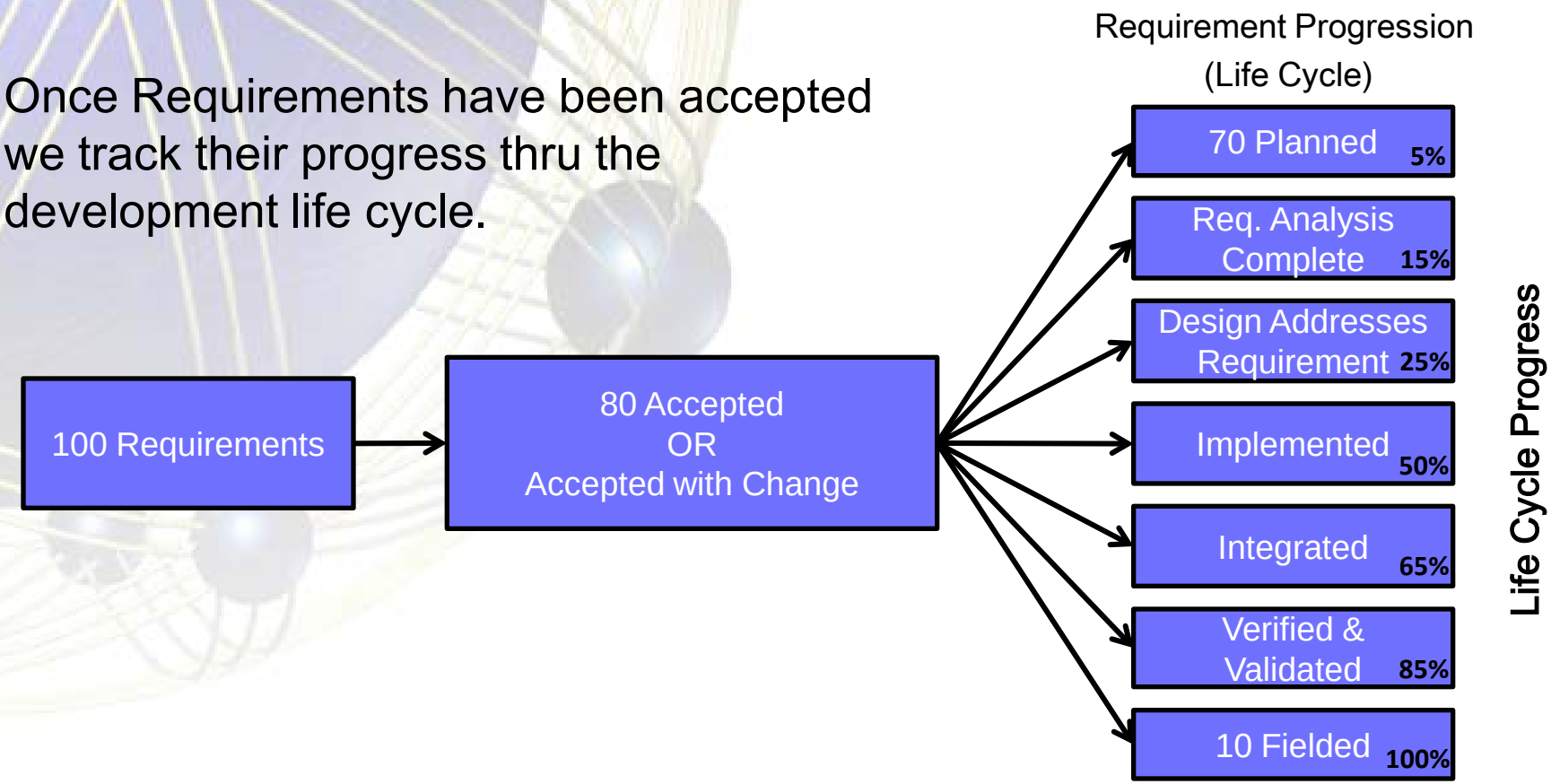
Disposition Progress
(Example Over Time)



Shows the project moving towards full acceptance/allocation of the requirements.

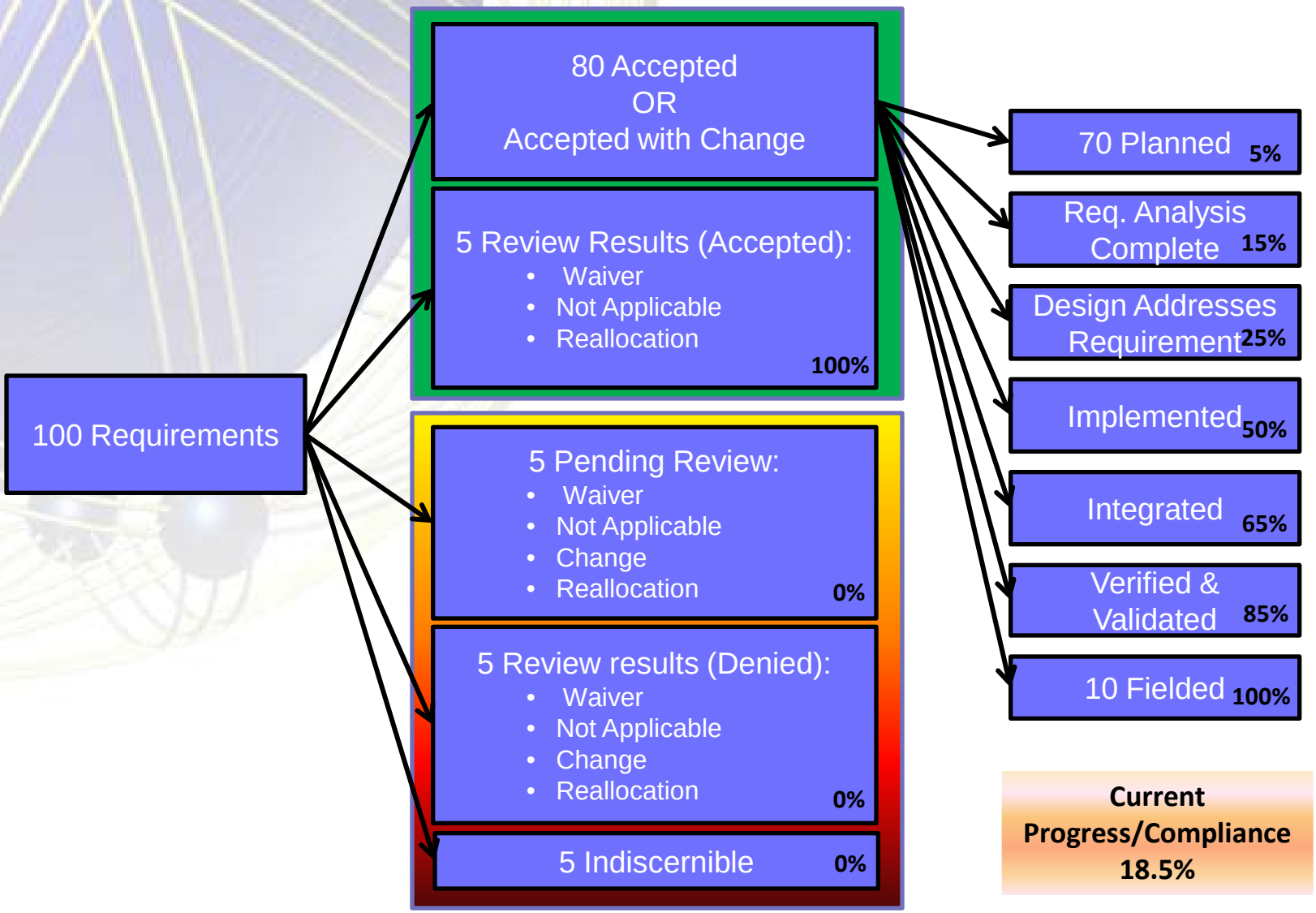
Life Cycle Progress

Once Requirements have been accepted we track their progress thru the development life cycle.

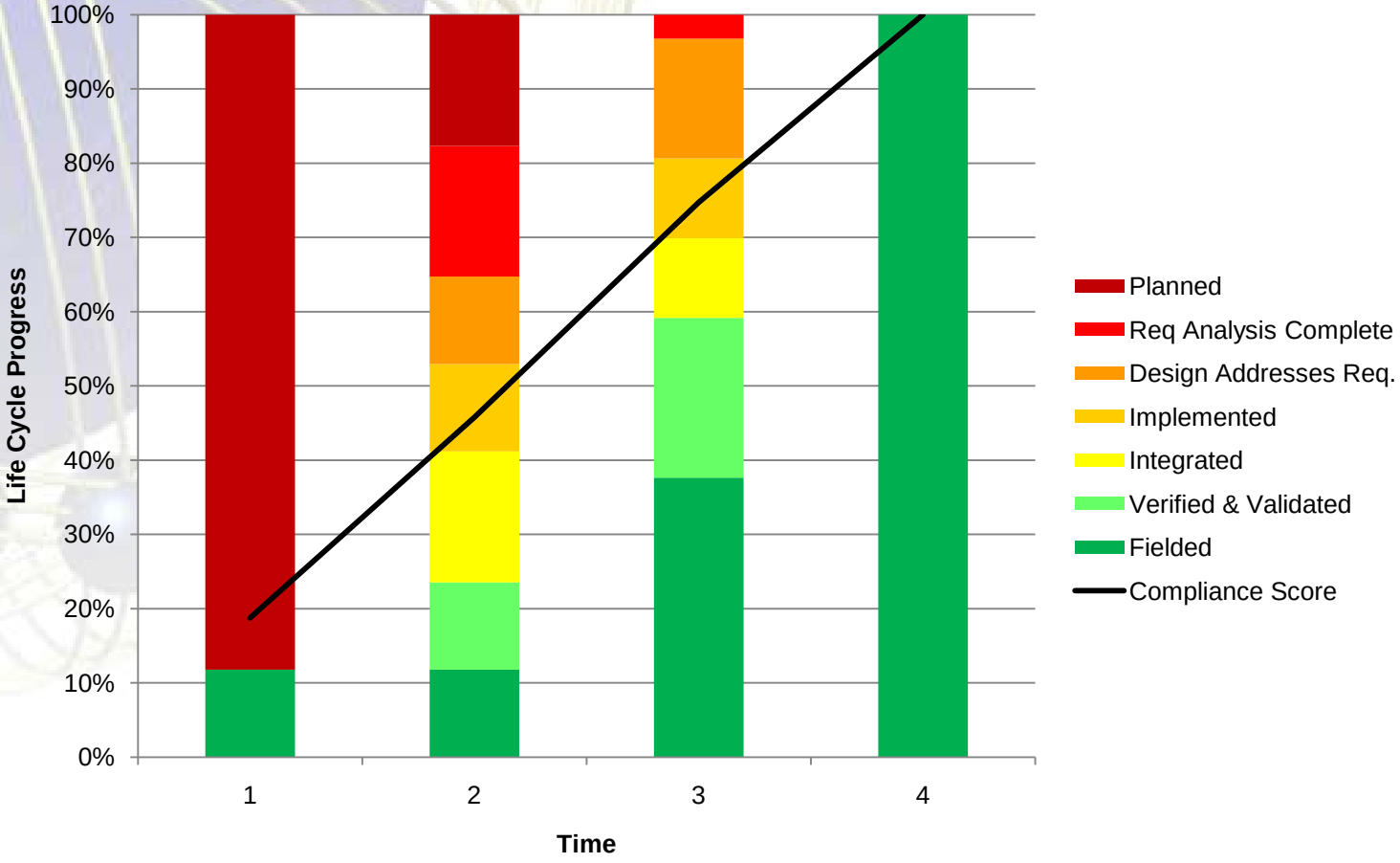


Credit is given when a requirement has finished each phase of the Life Cycle

Requirements Compliance Score

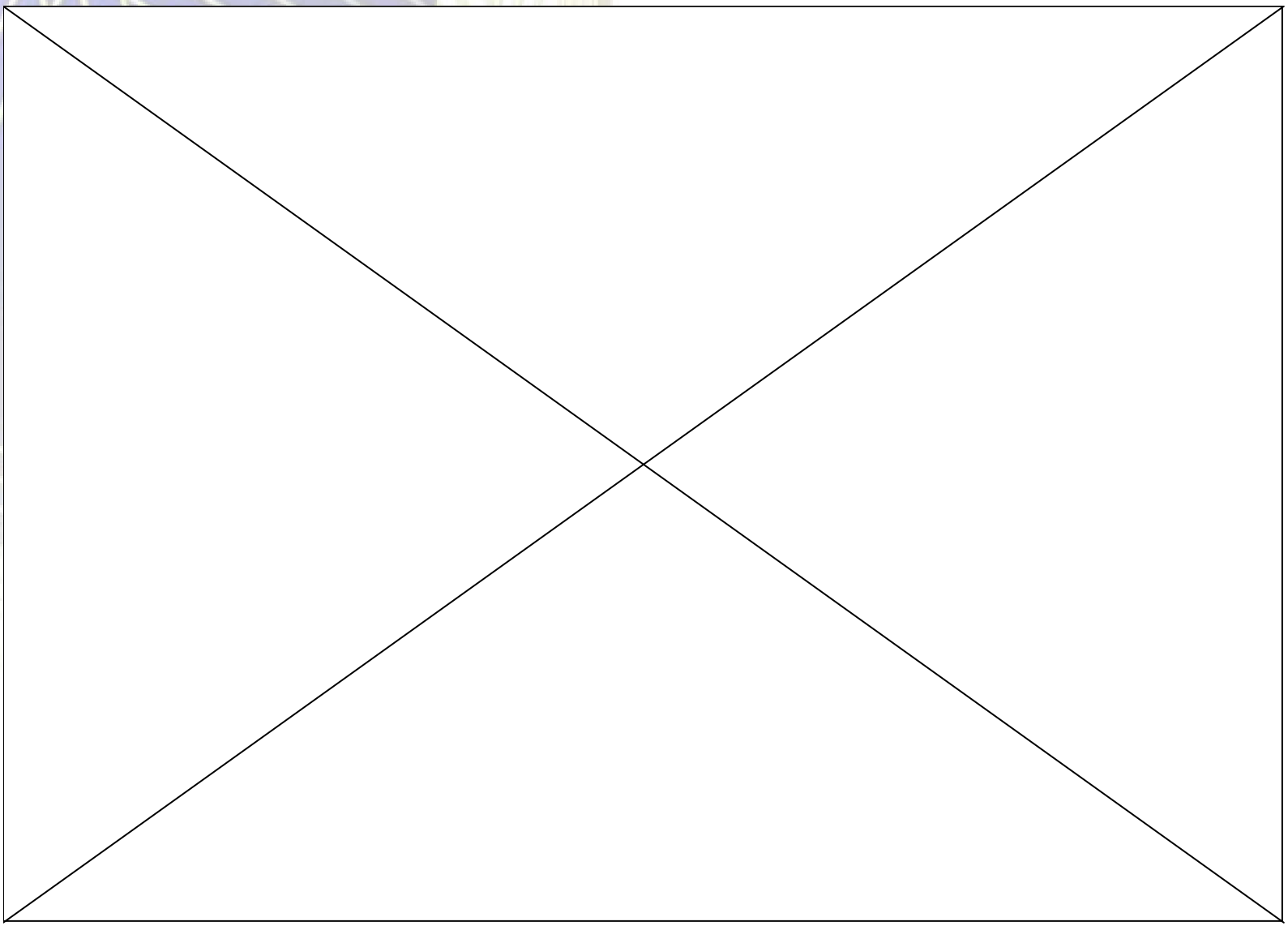


Life Cycle Progress
(Example Over Time)



Shows the project moving towards full fielding of the accepted requirements.

Compliance Calculator



Acceptance of Requirements
(Sample output)

Accepted	140
Accepted with Change	128
Pending N/A Review	1
Pending Change Review	31
Not Applicable	6
Not Allocated	7
Total Requirements (inc 1-3)	313

- Discovered a program that claimed almost full acceptance, but was actually changing over 50% of the requirements
- Discovered 32 requirements whose disposition had not yet been fully reviewed
- Discovered 7 High Level Requirements that were not allocated

Accepted	196
Accepted with Change	53
No Compliance Data	43
Indiscernible	16
Not Allocated	5
Total Requirements (inc1-3)	313

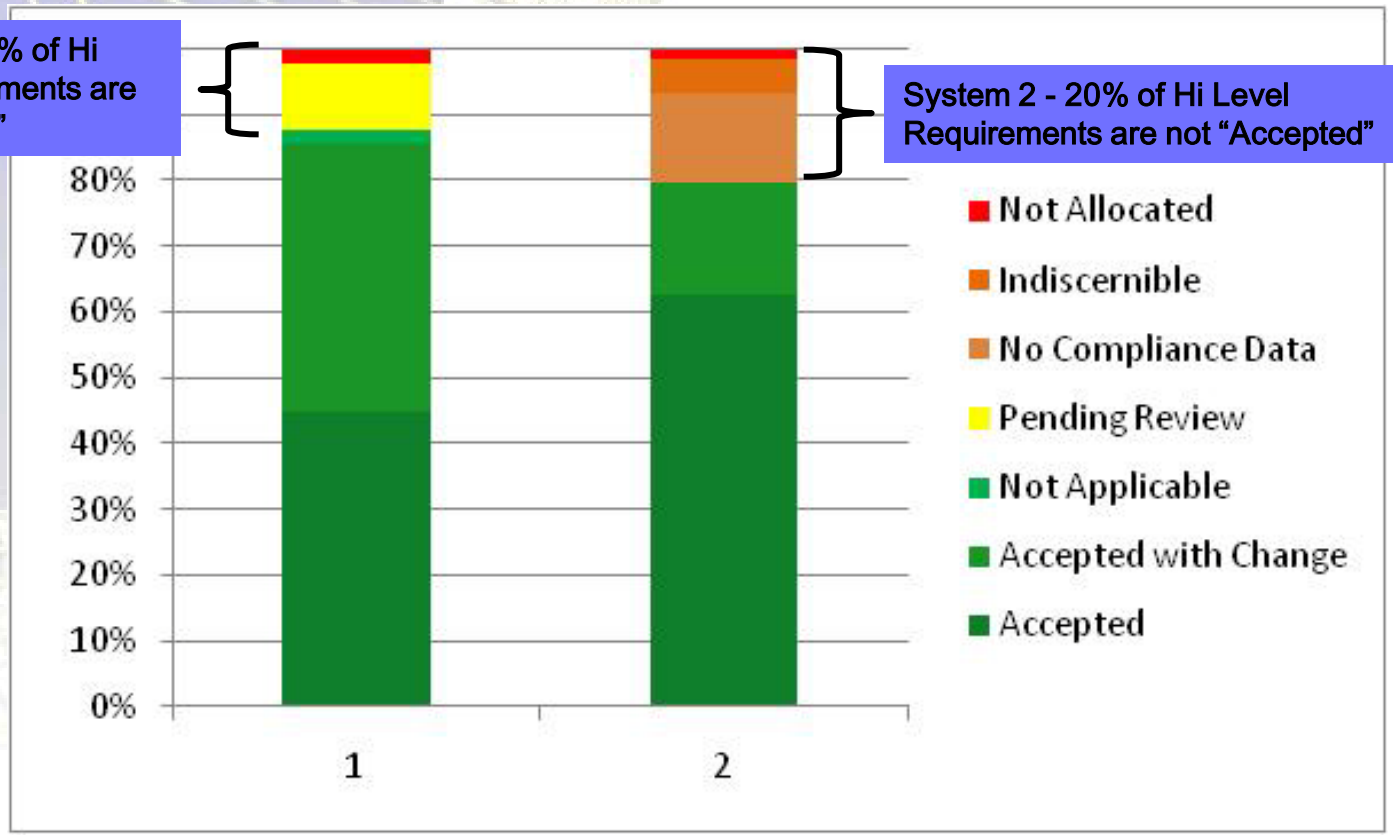
- Documented 43 requirements that still had not been dispositioned although they were allocated
- Discovered 16 problem requirements that developers were having trouble understanding
- Discovered 5 High Level Requirements that were not allocated

High Level Requirement Results Based on Actual Data

Acceptance of Requirements

System 1 - 14% of Hi Level Requirements are not “Accepted”

System 2 - 20% of Hi Level Requirements are not “Accepted”



Summary

Benefits observed are positive proof that there needs to be a well understood approach to reporting requirements

Gaps already found and reported to Customer

Just starting to roll out Life Cycle progression.