

18-642: Traceability

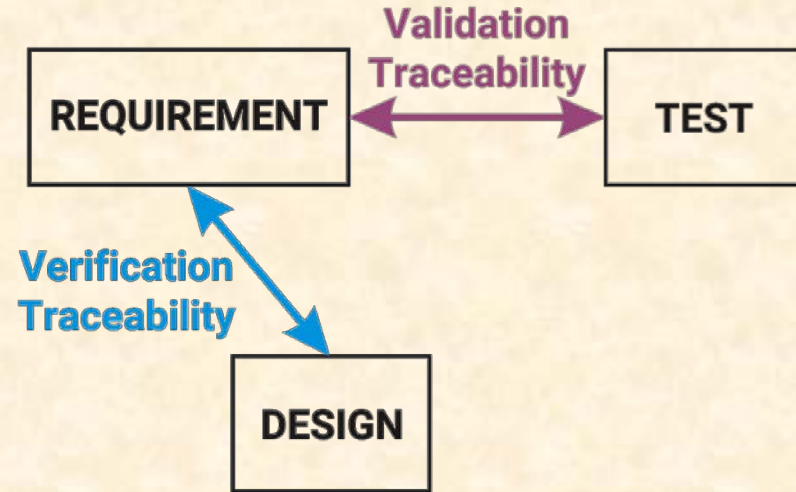
10/9/2017



Traceability

■ Anti-Patterns:

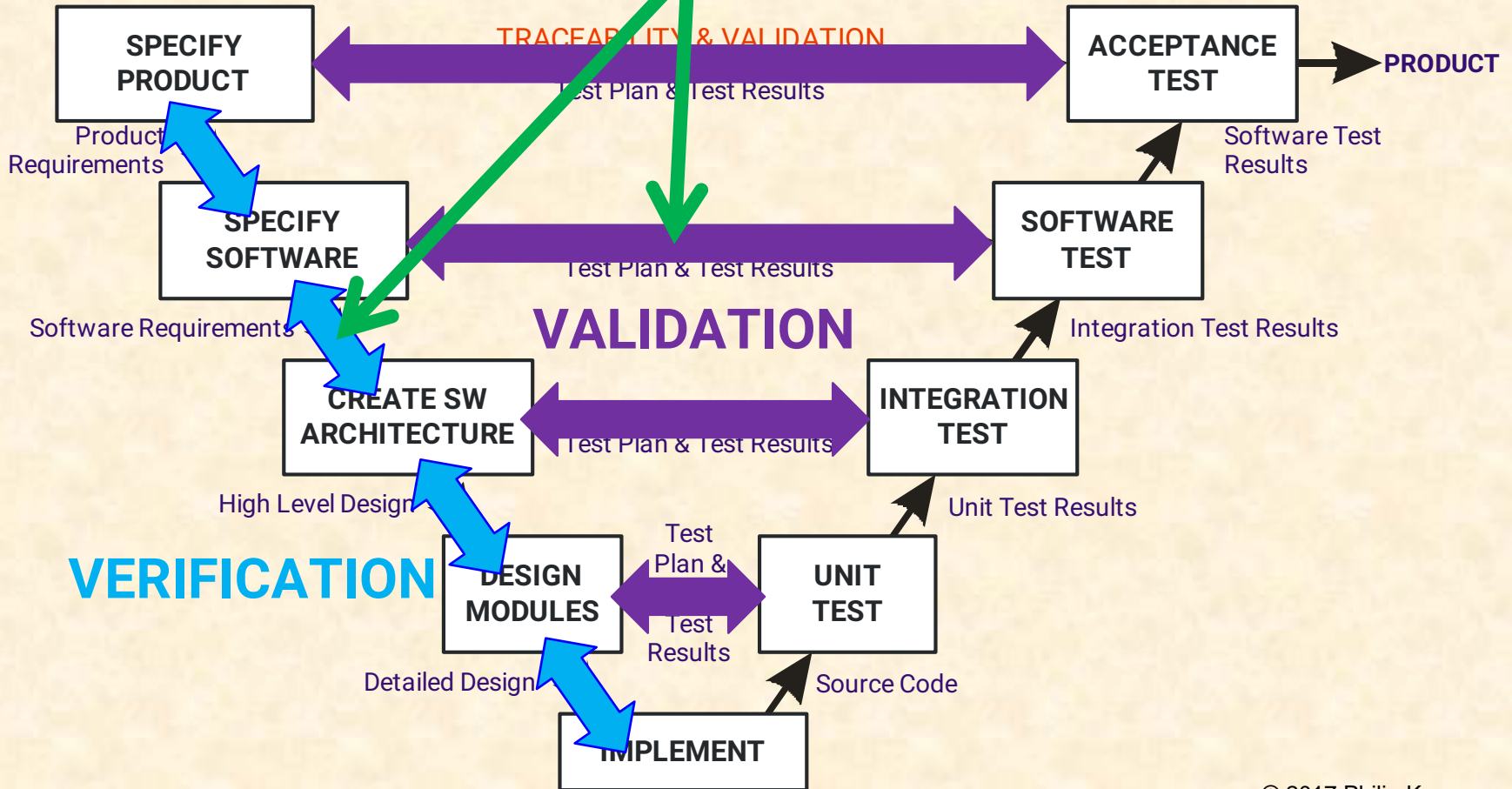
- Tests don't map to requirements
- Requirements aren't tested
- Reqts/design elements missing
- Gold plating (extra functionality)



■ Traceability

- Creating something traces to a quality check on the result
 - Verification: you did something the way you said you'd do it
 - Validation: the thing you created behaves the way it should
- Ensure nothing left out; nothing added that shouldn't be there

YOU ARE HERE



Traceability Examples

■ Design traceability

- Requirement → design → implementation
- Requirement → test

■ SQA traceability

- Confirm process is being carried out
- Process step → document/artifact → quality metric

■ Safety analysis traceability

- Confirm all hazards successfully mitigated
- Hazard → requirement → mitigation → validation

■ Defect traceability

- Ensure that all bugs are fixed
- Bug report → defect identified → fix task → code check-in → regression test

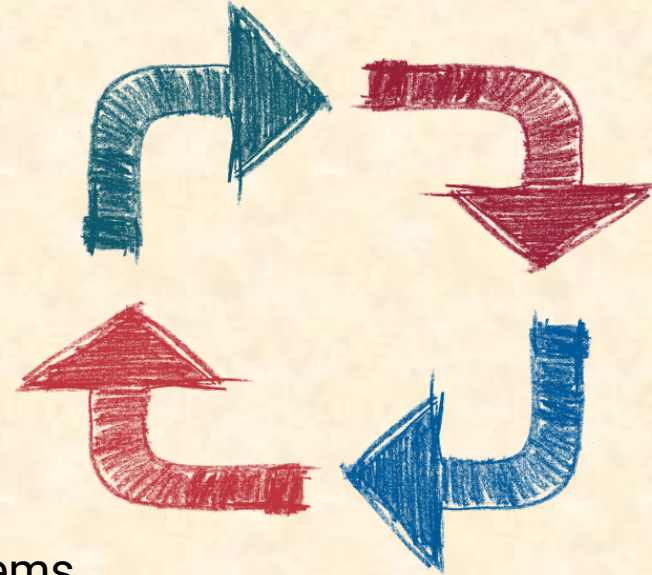
Requirements-to-Statecharts Traceability

	Requirements					
States	R2.1	R2.2	R2.3	R2.4a	R2.4b	R2.5
S2.1 IDLE	x		x			x
S2.2 EMPTY	x	x				x
S2.3 VEND	x			x	x	
S2.4 FLASH_OFF	x			x	x	
S2.5 FLASH_ON	x			x	x	
Transitions						
T2.1				x	x	x
T2.2			x			
T2.3		x				
T2.4					x	
T2.5					x	
T2.6		x				
T2.7			x			

Traceability Best Practices

■ Trace everything in design package

- Even simple traceability checks can find problems
- Everything in design has an ID tag for traceability
 - Map left and right sides of V to each other
 - Map each layer of V upward and downward
- Trace changes to see what else they affect



■ Traceability pitfalls

- Making granularity of trace IDs too big causes problems
- Re-numbering breaks auto-generated document sections used as trace IDs
- Don't use the wrong tool
 - Spreadsheets don't scale to big projects
 - Big project tools might be overkill where a spreadsheet approach will do