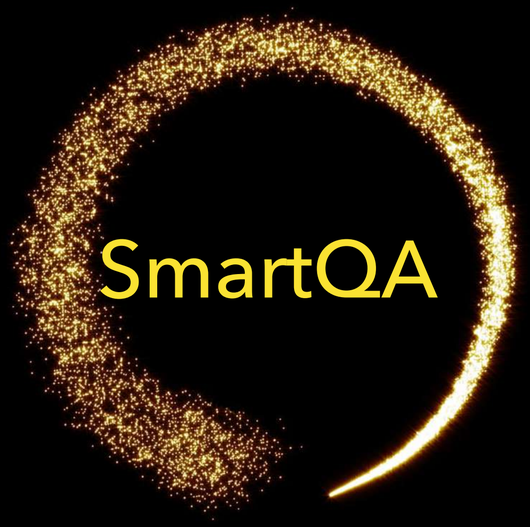


IST Masterclass
Session #4

Exploration phase in detail #1

www.stagsoftware.com

TOPICS

What is exploration?

Deep dive

Exploring specification

Design approaches

Level based test design

Test writing style

What is ‘exploration’?

RECONNAISANCE

EXPLORATION

RECOUP

Dive deep to understand entities and then evaluate them

understand what entity does/should do
what are conditions governing behaviour?
what are the acceptance criteria?
what may be potential issues probable?

RECONNAISANCE

EXPLORATION

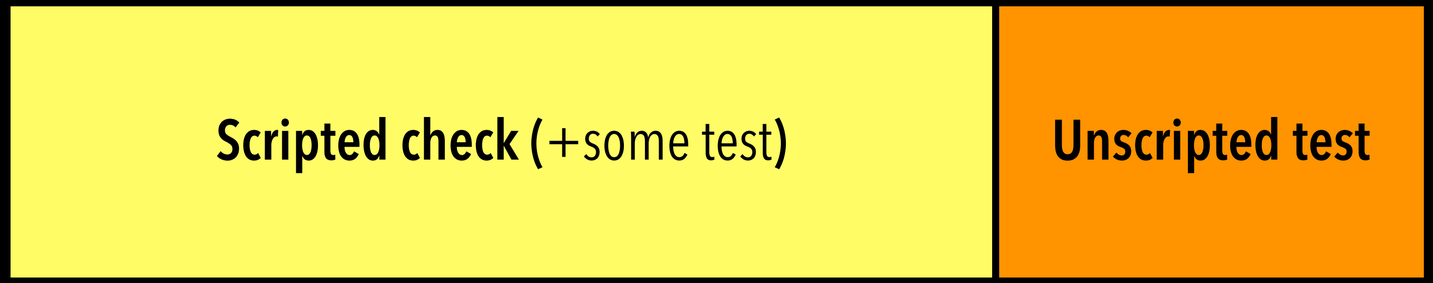
RECOUP

Dive deep to understand entities and then evaluate them

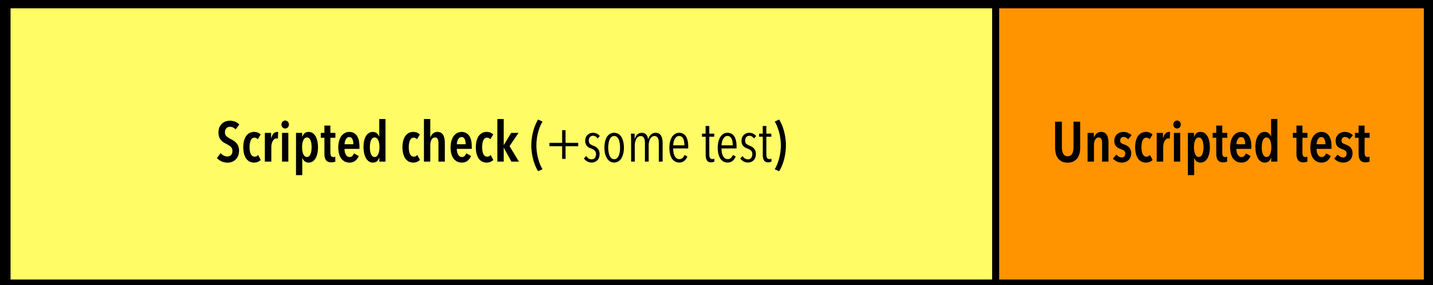
understand what entity does/should do.
what are conditions governing behaviour?
what are the acceptance criteria?
what may be potential issues probable?

come up with scenarios to try out
come up with smart checklist to check/test
create suites, review, revise
note down issues, suggestions, observations

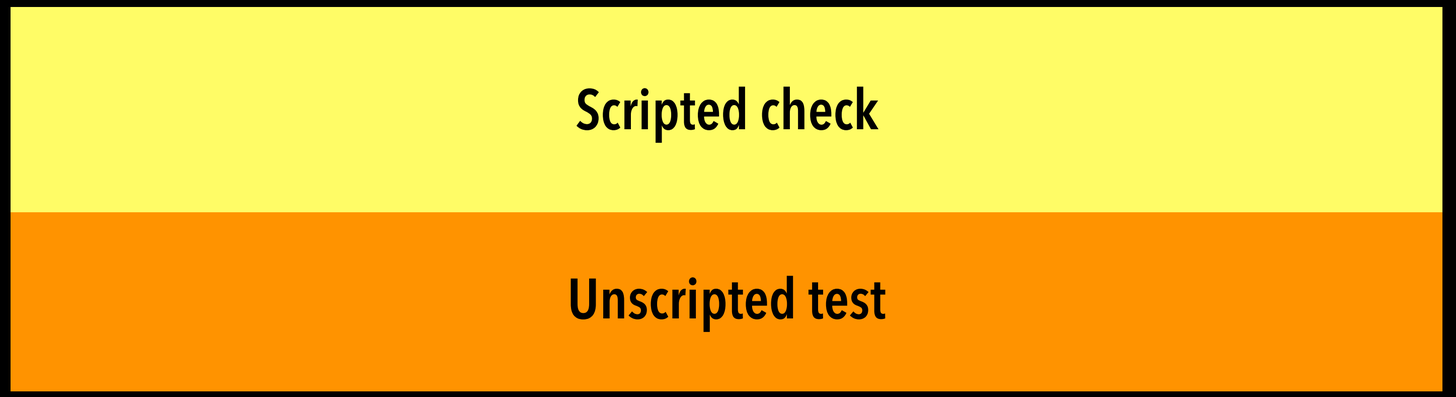
Design & evaluation in rapid tandem



Design & evaluation in rapid tandem

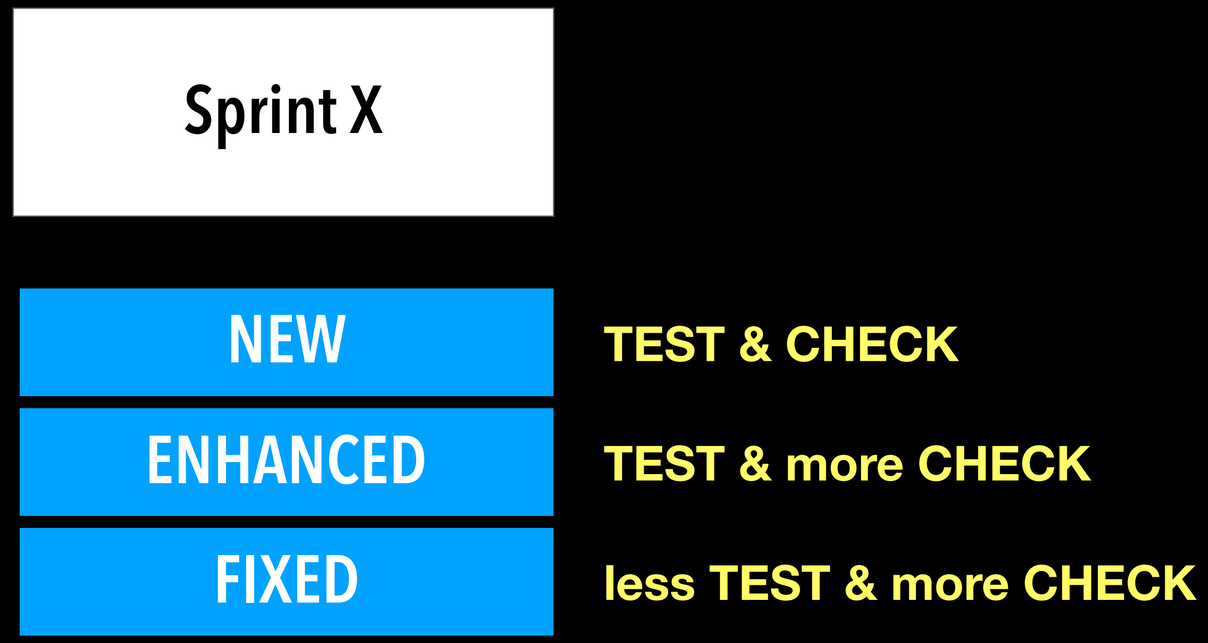


to



KEY is lightweight writing, notetaking

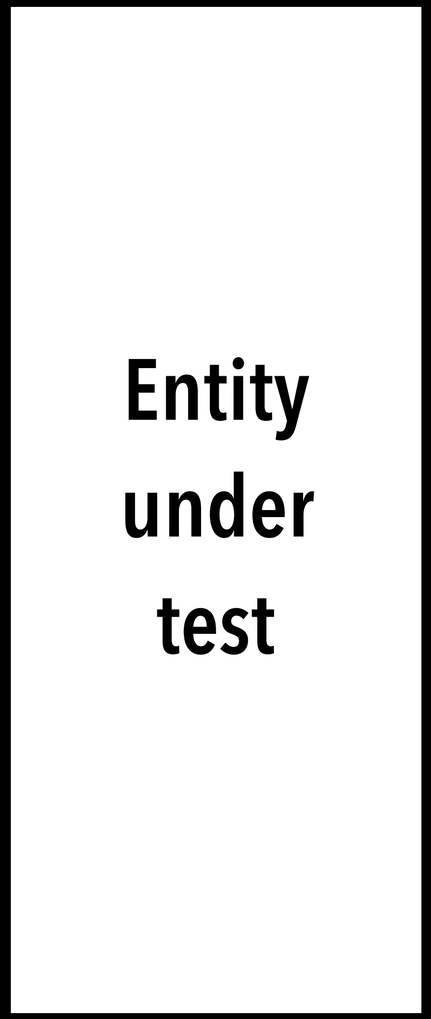
What is Test/Check in a Sprint?



How much Test & Check depends on

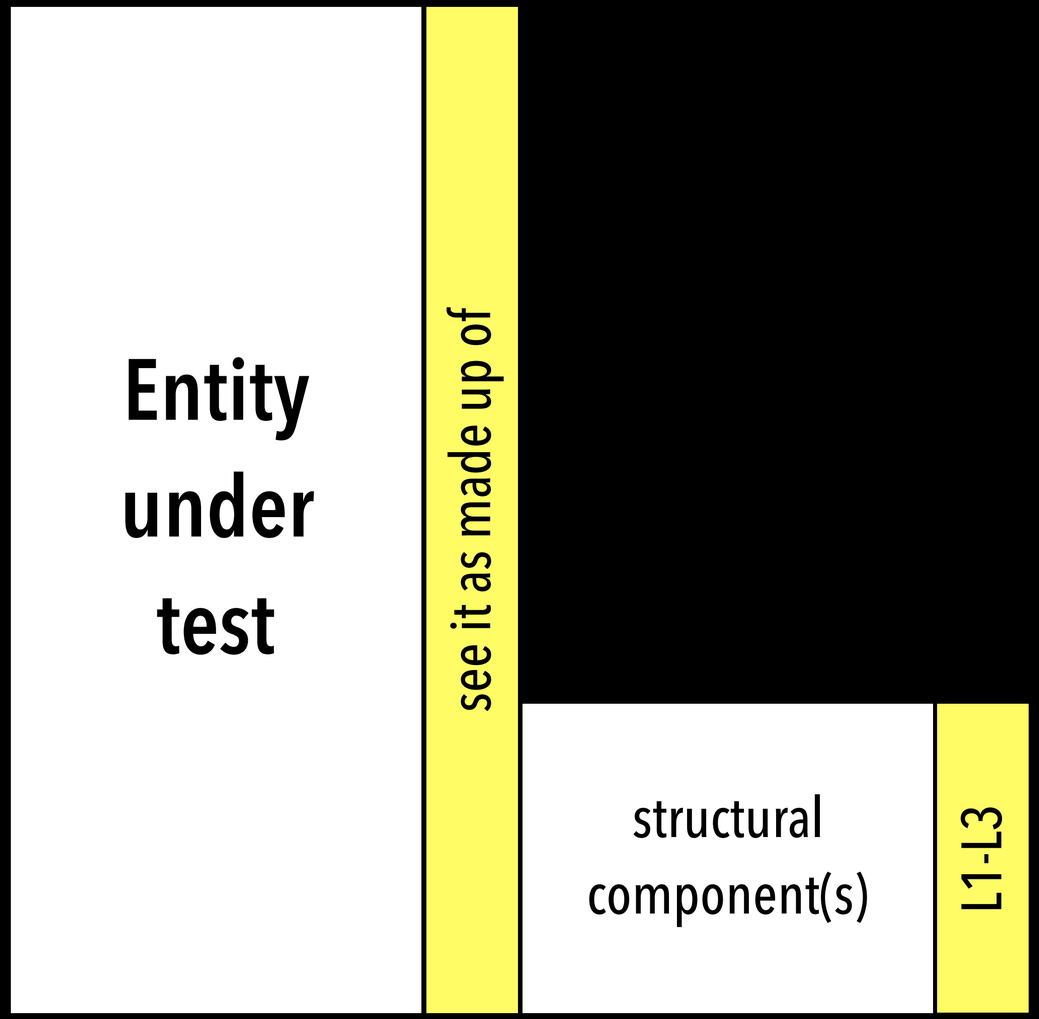
1. State of entity - New | Enhanced | Fixed
2. Spec detail for entity : Less => more Test, More => more Check

Testing an entity



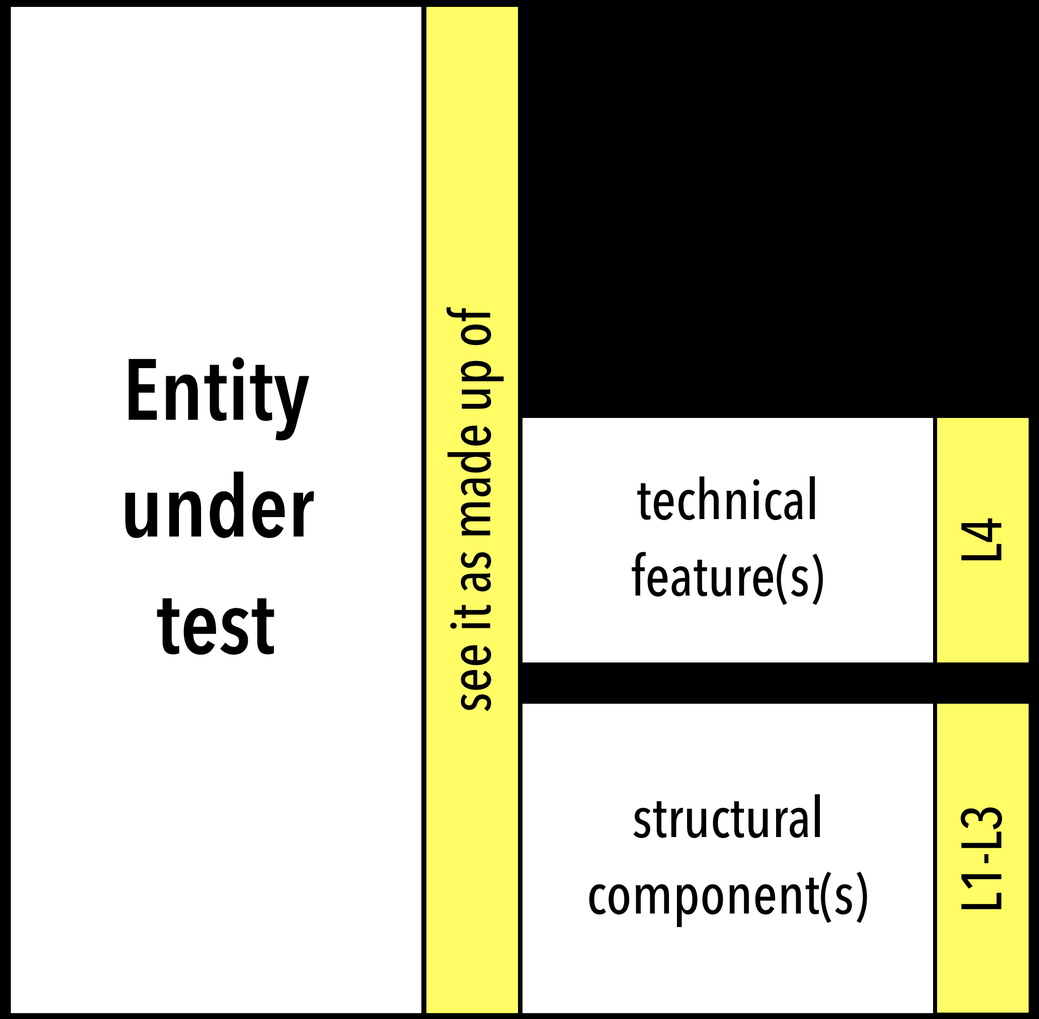
When you explore, see the EUT as a composite and mentally decompose it

Testing an entity



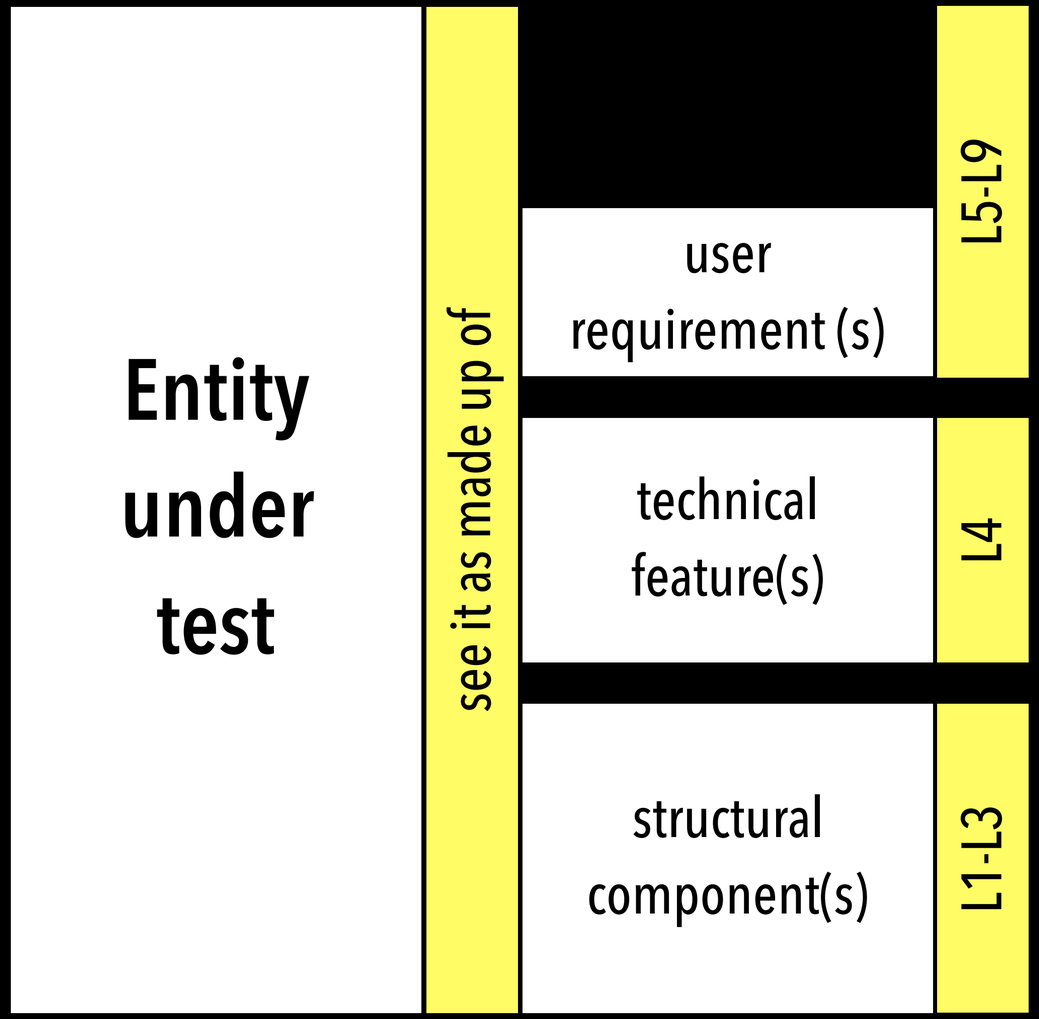
When you explore, see the EUT as a composite and mentally decompose it

Testing an entity



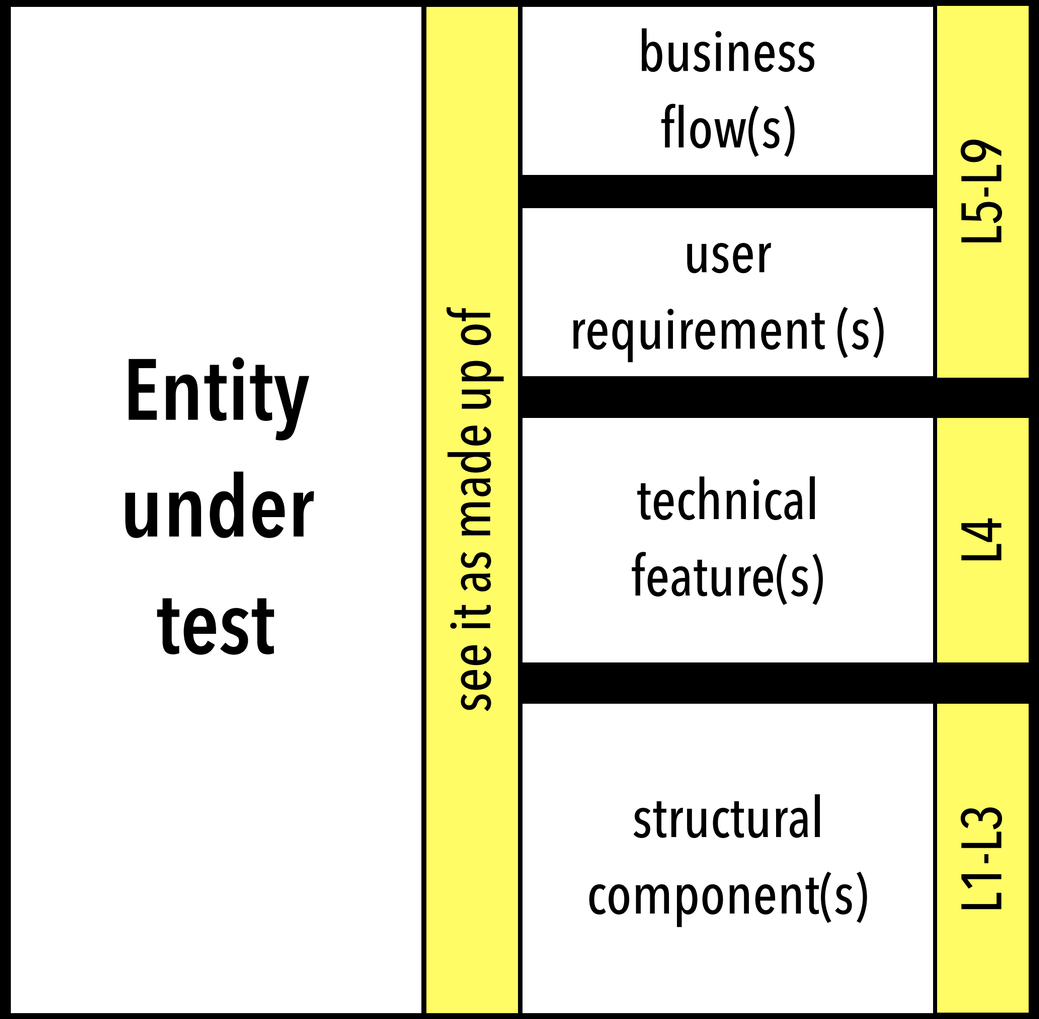
When you explore, see the EUT as a composite and mentally decompose it

Testing an entity



When you explore, see the EUT as a composite and mentally decompose it

Testing an entity

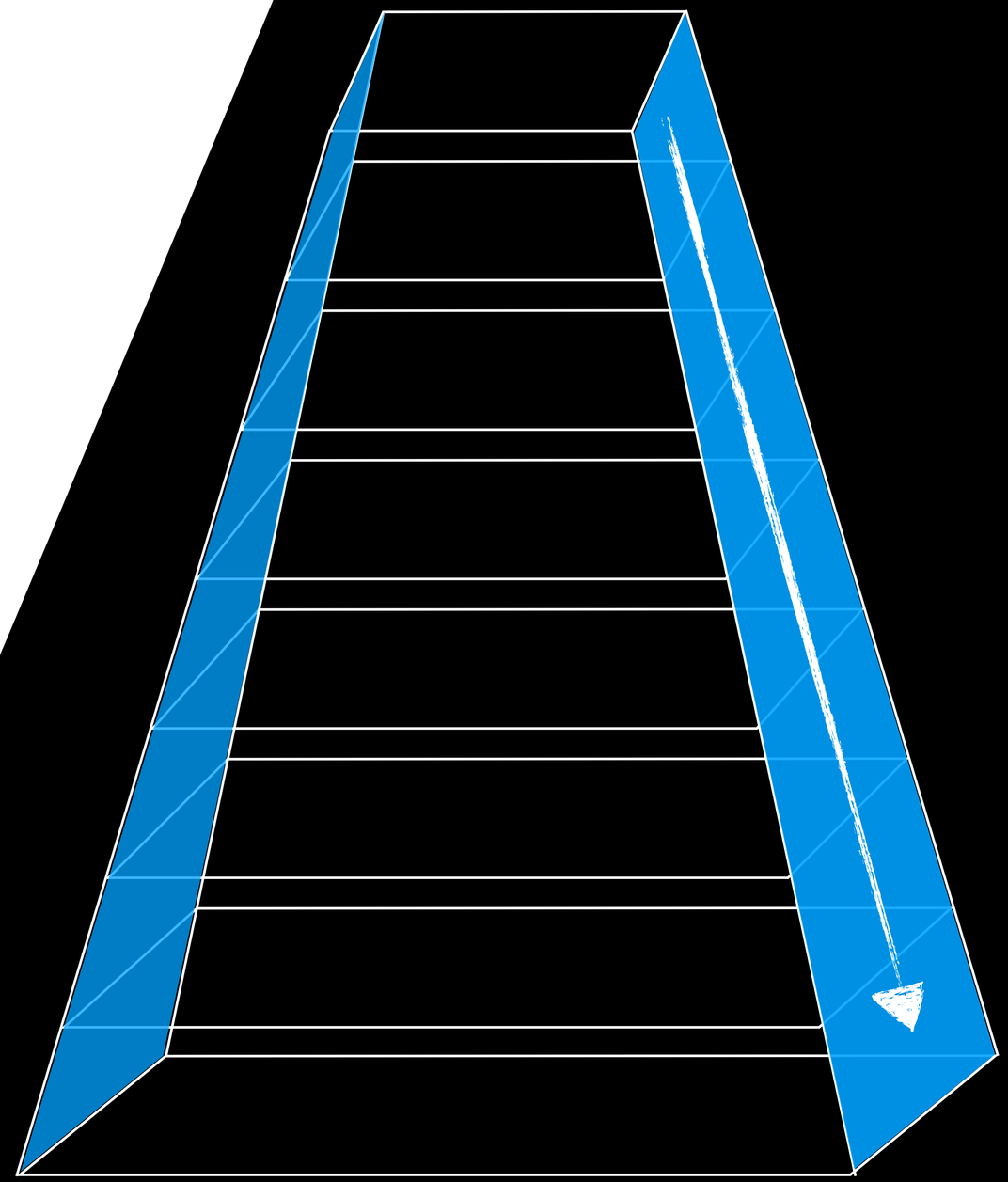


When you explore, see the EUT as a composite and mentally decompose it so that you can observe well and understand clearly at the right level to come with good scenarios/questions.

Deep dive

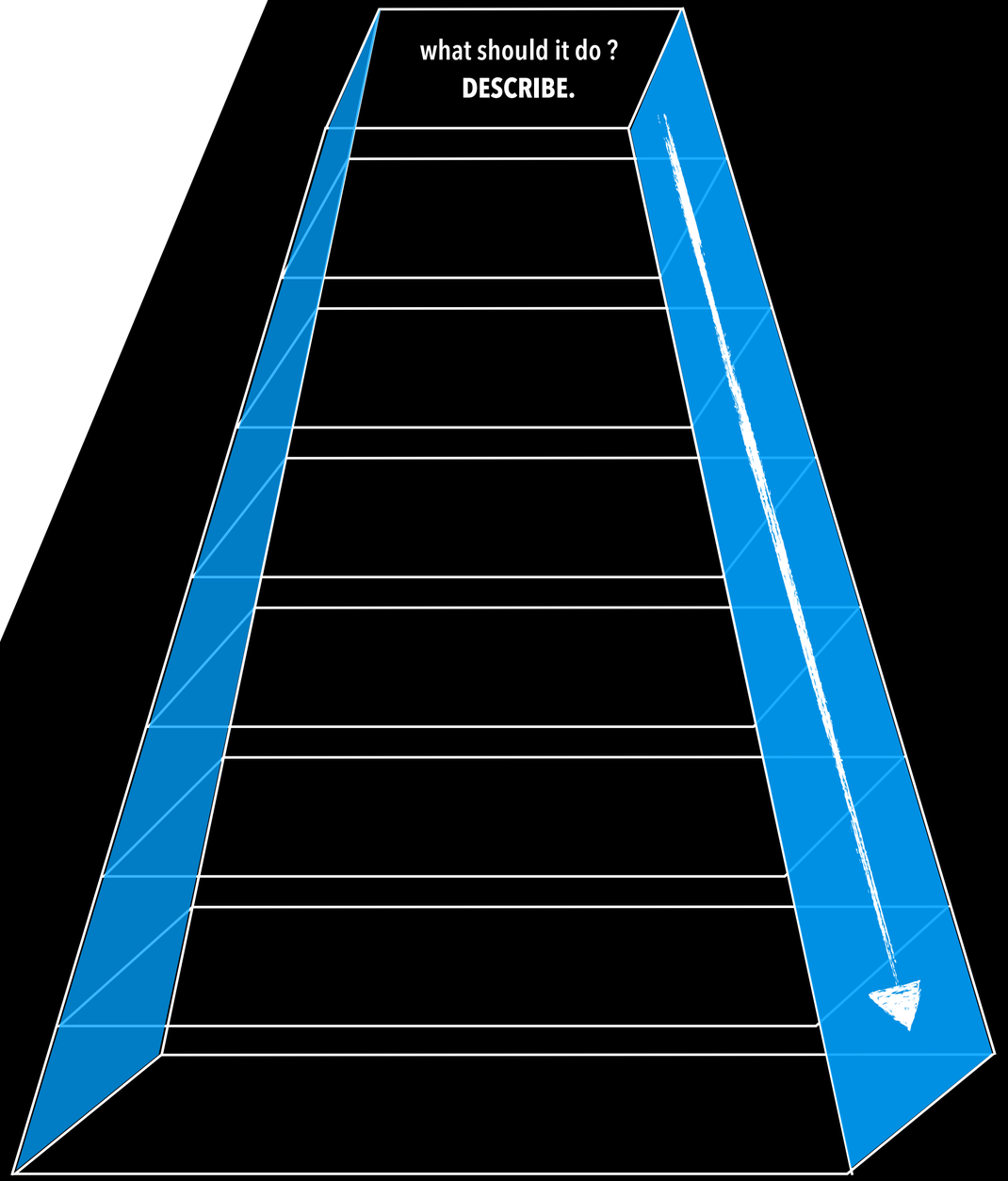
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



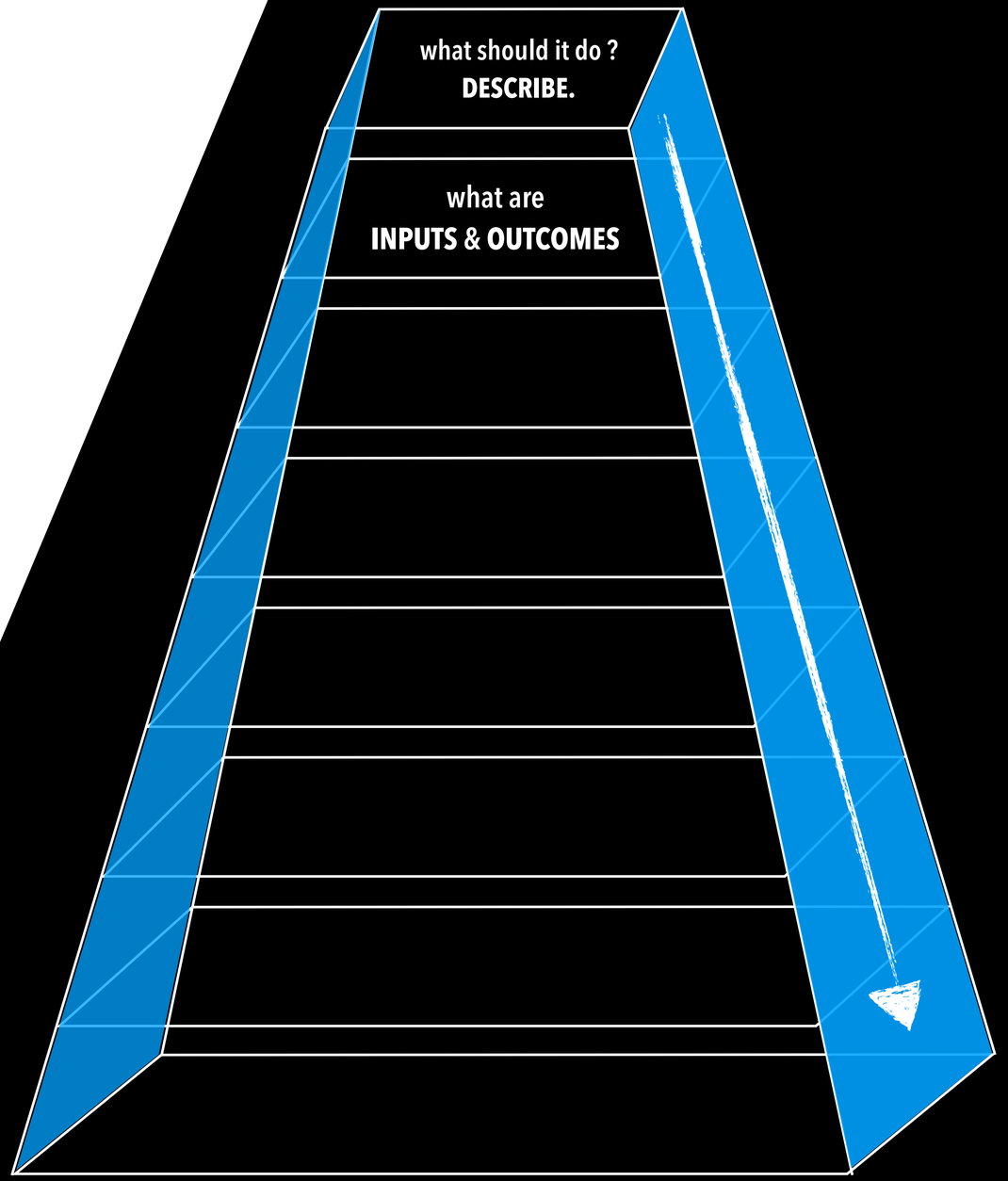
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



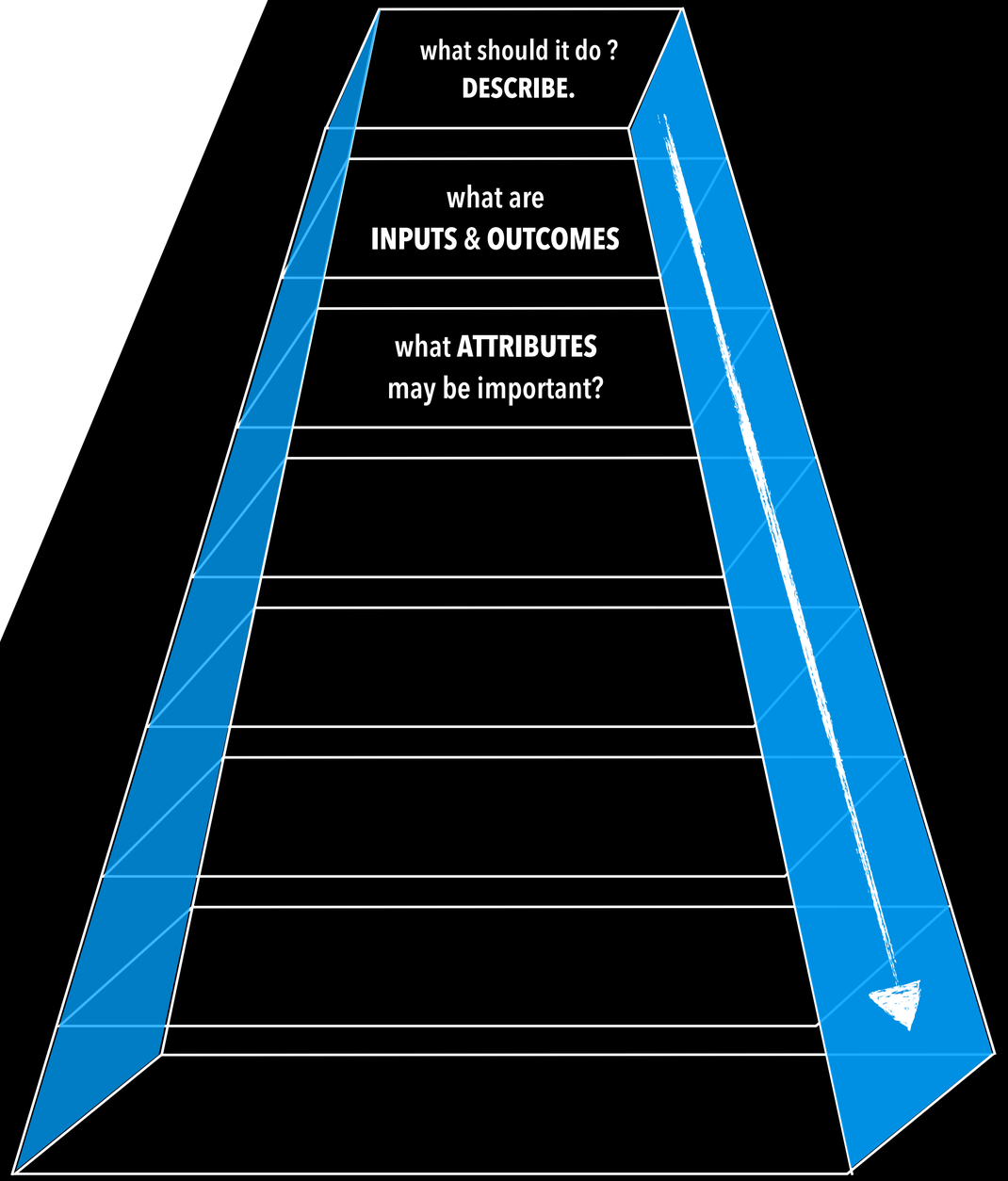
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



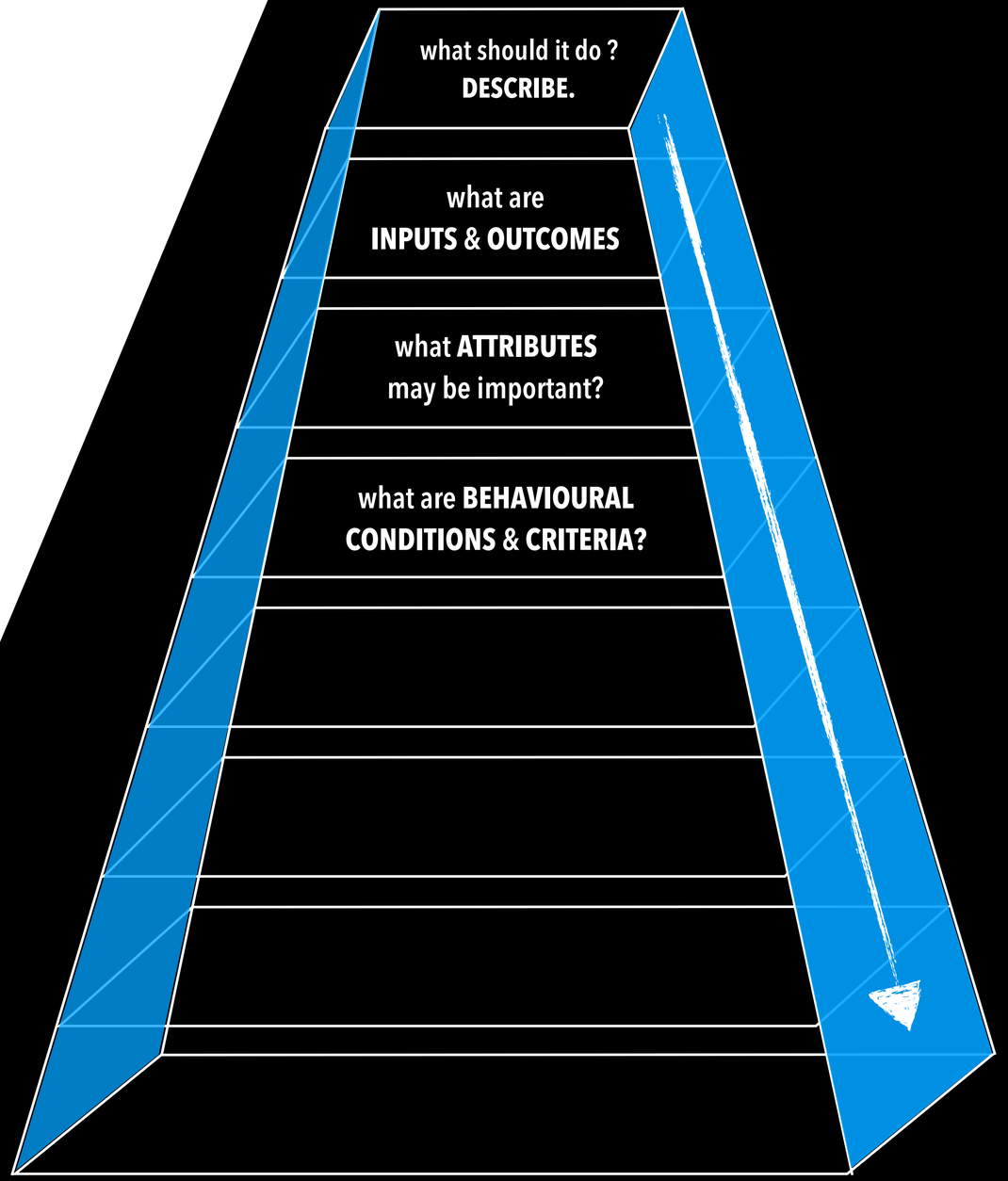
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



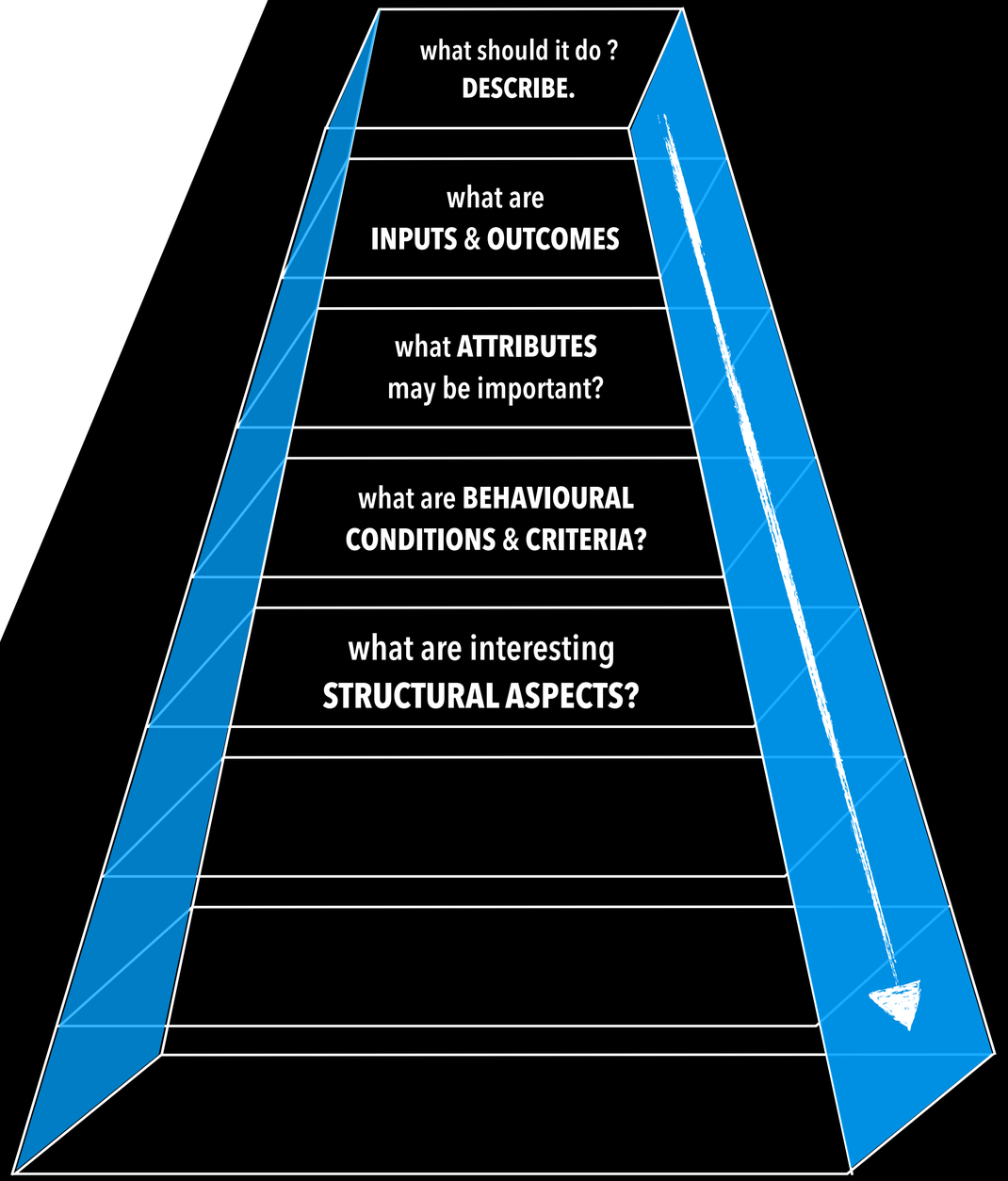
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



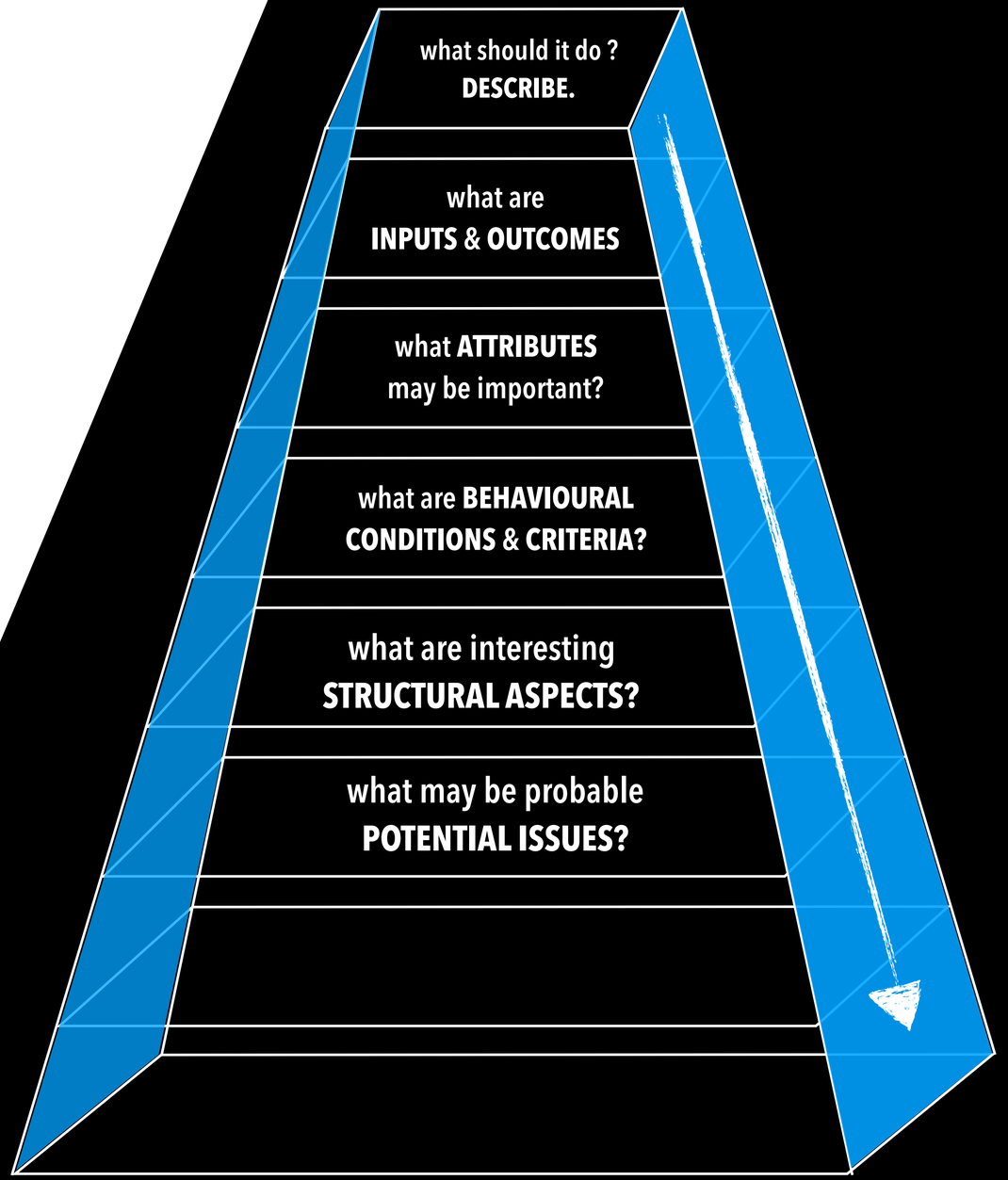
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



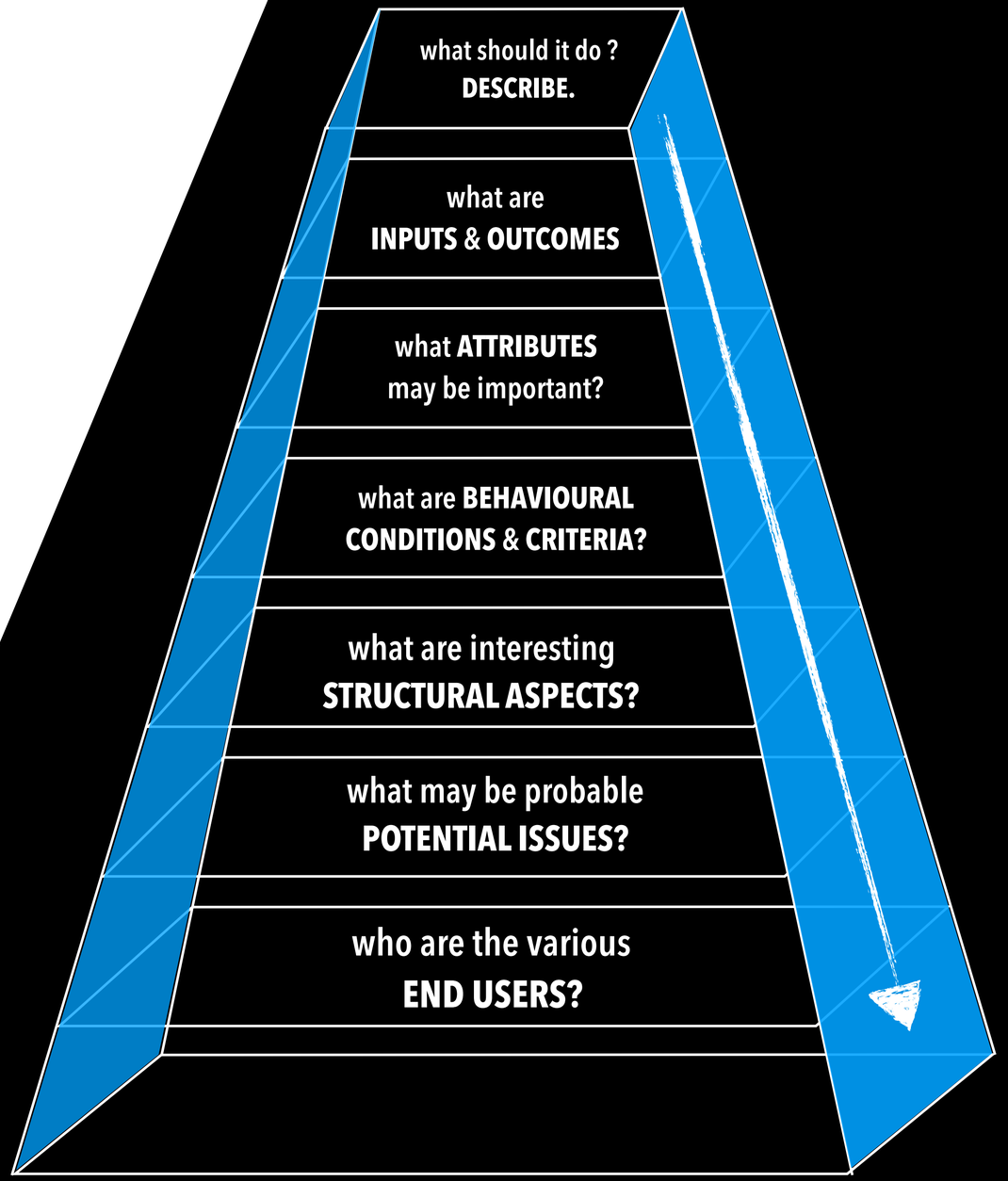
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



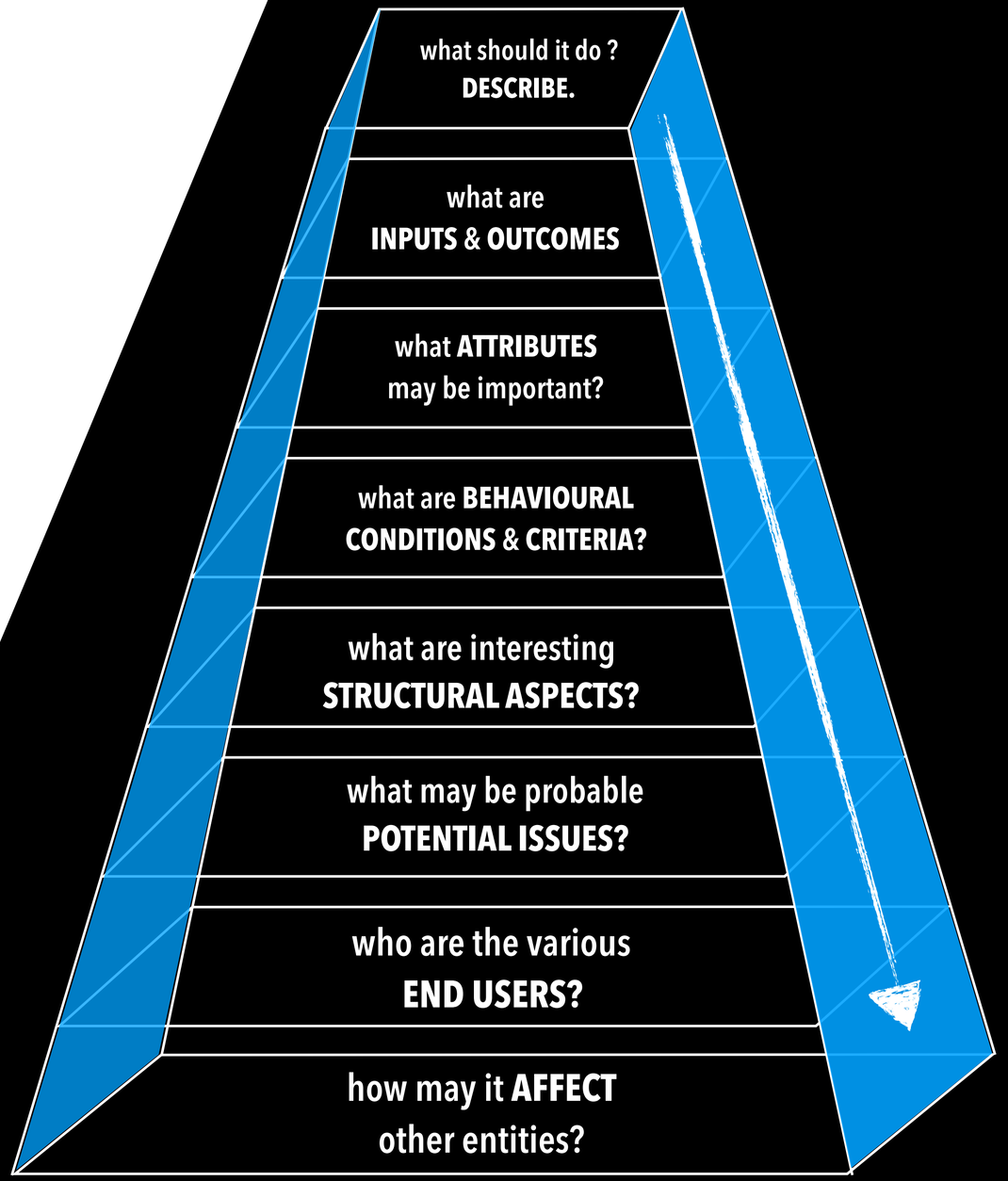
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



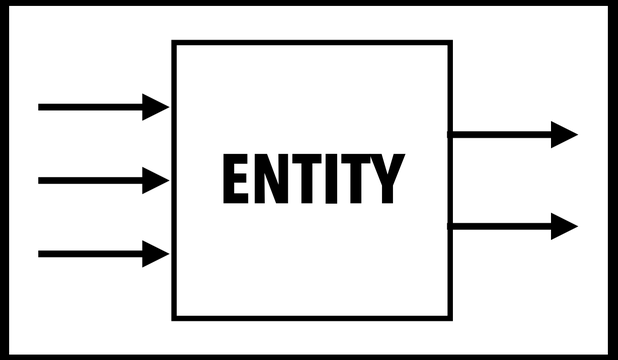
To design well,
to come up with good scenarios
do DEEP DIVE

an ENTITY can be a
small COMPONENT, FEATURE
or a big REQUIREMENT, FLOW



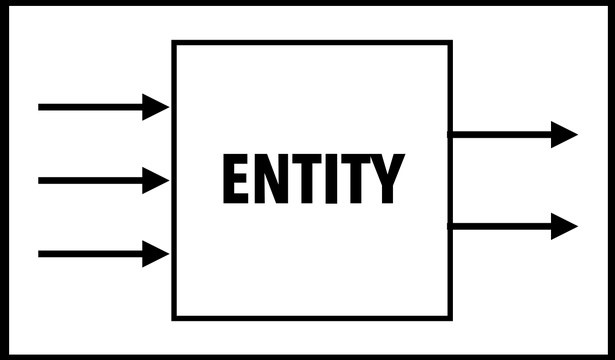
Exploring specification

Let’s explore specification of an entity



who

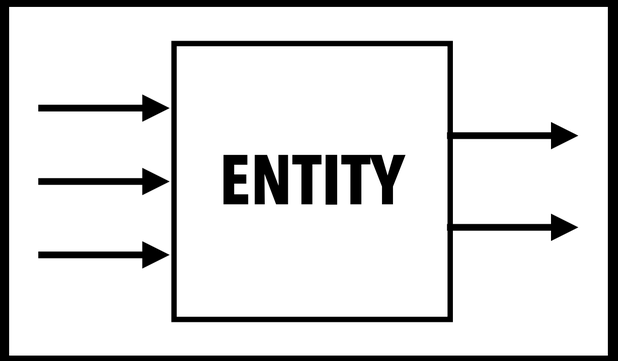
who will use this?
PERSONA



who + what

what should it do?
STEPS

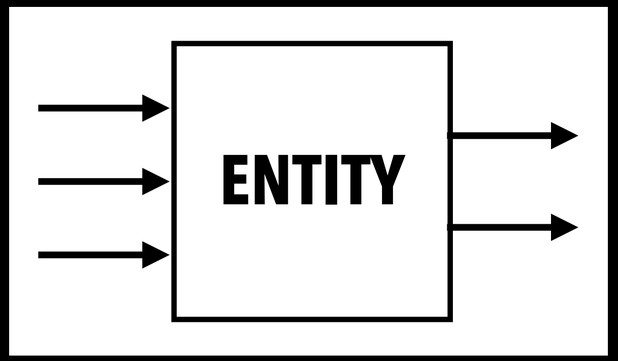
who will use this?
PERSONA



who + what + what-for

what should it do?
STEPS

who will use this?
PERSONA

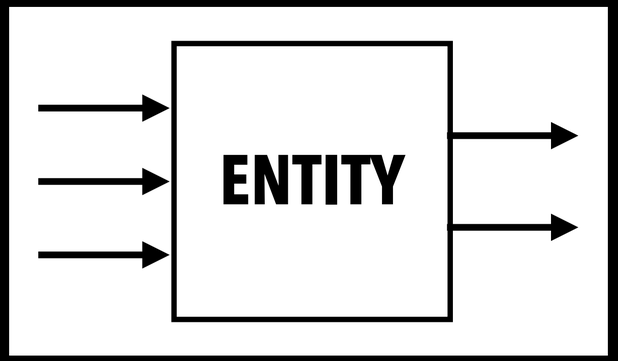


what should it satisfy?
A.CRITERIA/COS

who + what + what-for

what should it do?
STEPS

who will use this?
PERSONA



what should it satisfy?
A.CRITERIA/COS

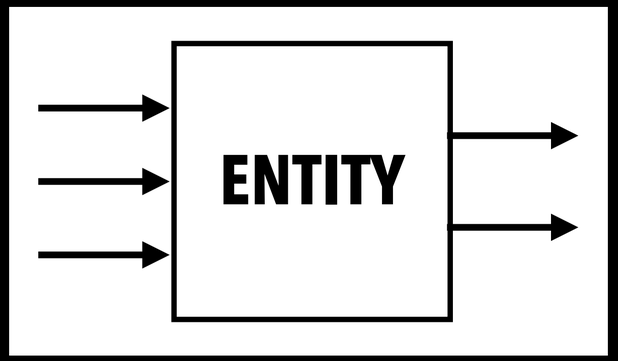
what should it do?
CONDITIONS

who + what + what-for

what should it do?
STEPS

DESCRIPTIVE
understand, implement

who will use this?
PERSONA



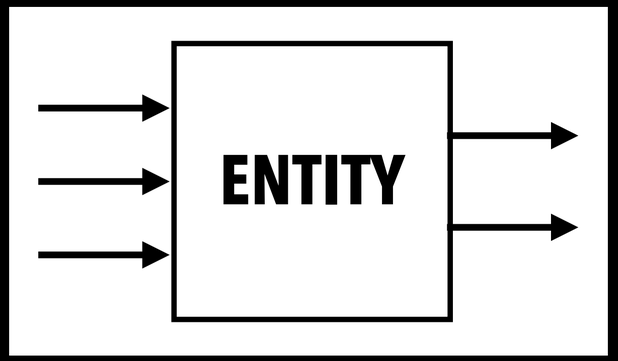
what should it satisfy?
A.CRITERIA/COS

what should it do?
CONDITIONS

who + what + what-for

what should it do?
STEPS

who will use this?
PERSONA



what should it satisfy?
A.CRITERIA/COS

what should it do?
CONDITIONS

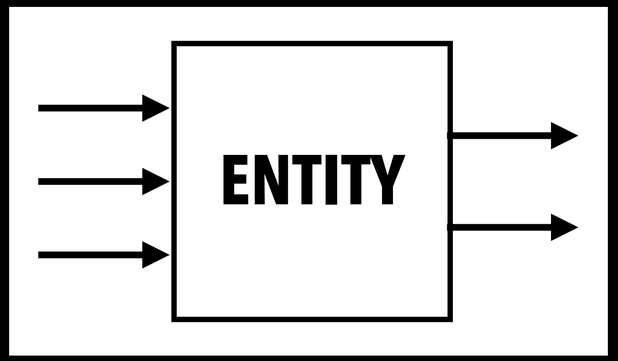
PRESCRIPTIVE
evaluate

who + what + what-for

what should it do?
STEPS

DESCRIPTIVE
understand, implement

who will use this?
PERSONA



what should it satisfy?
A.CRITERIA/COS

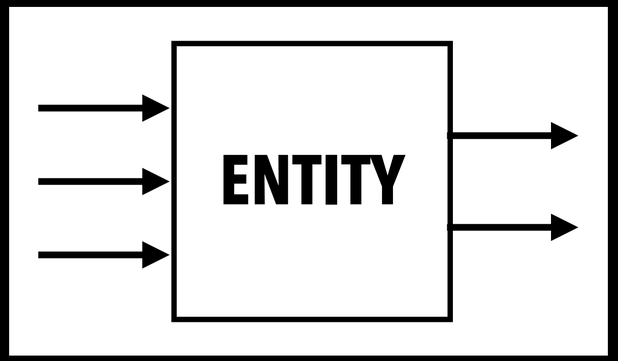
what should it do?
CONDITIONS

PRESCRIPTIVE
evaluate

USE CASE

what should it do?
STEPS

who will user this?
PERSONA



what should it satisfy?
A.CRITERIA/COS

what should it do?
CONDITIONS

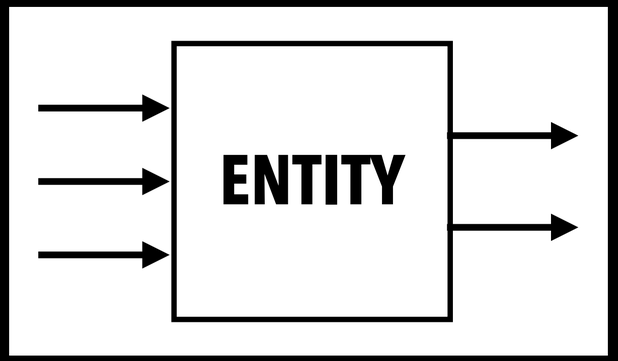
DESCRIPTIVE

understand, implement

Use Cases may be more granular,
and describe how your system will act.

what should it do?
STEPS

who will user this?
PERSONA



what should it satisfy?
A.CRITERIA/COS

what should it do?
CONDITIONS

USER STORY

PRESCRIPTIVE

evaluate

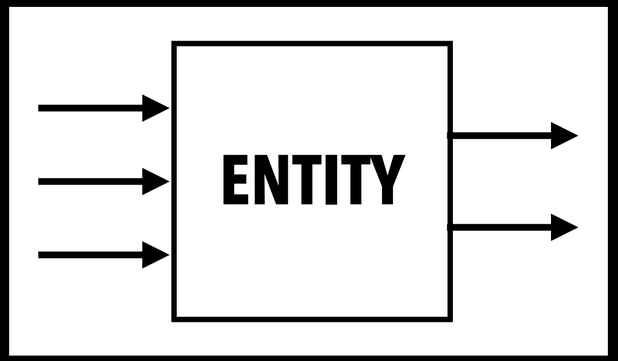
User Stories are centered on result and benefit of the thing you're describing,

USE CASE

what should it do?
STEPS

DESCRIPTIVE
understand, implement

who will user this?
PERSONA



what should it satisfy?
A.CRITERIA/COS

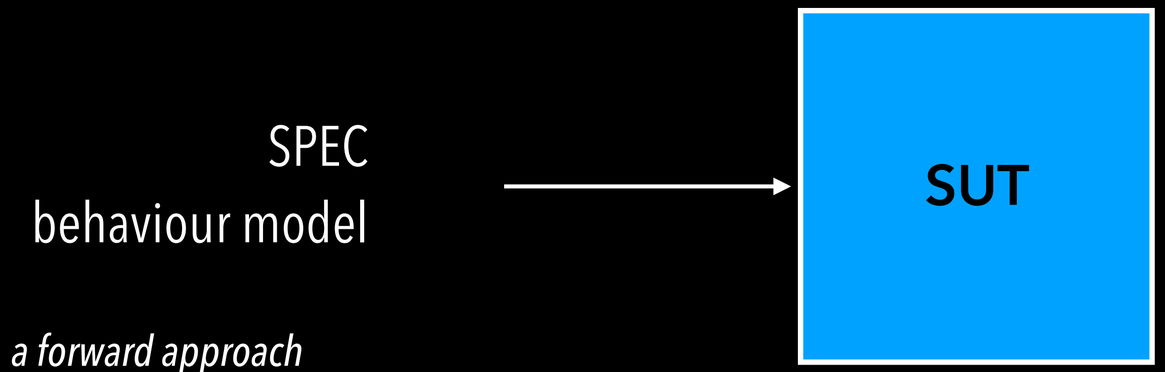
what should it do?
CONDITIONS

PRESCRIPTIVE
evaluate

USER STORY

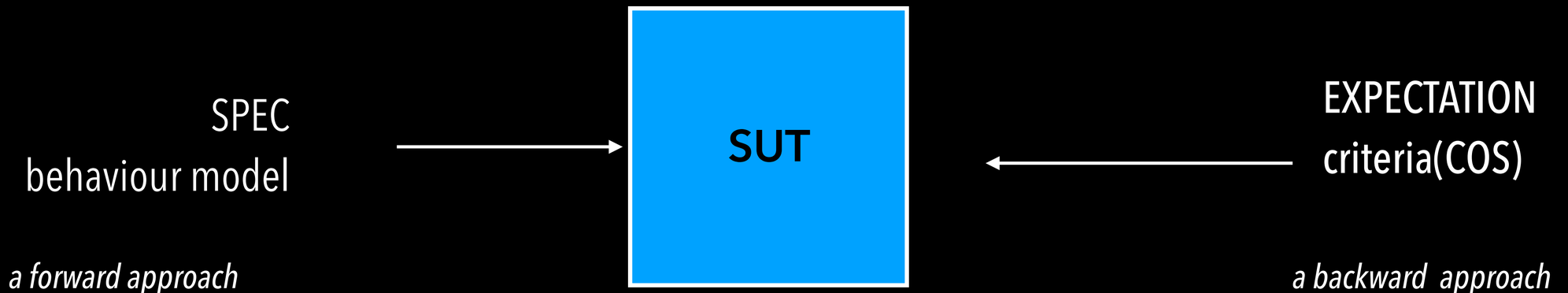
Design approaches

Design method



a forward approach
using given spec or discovered spec

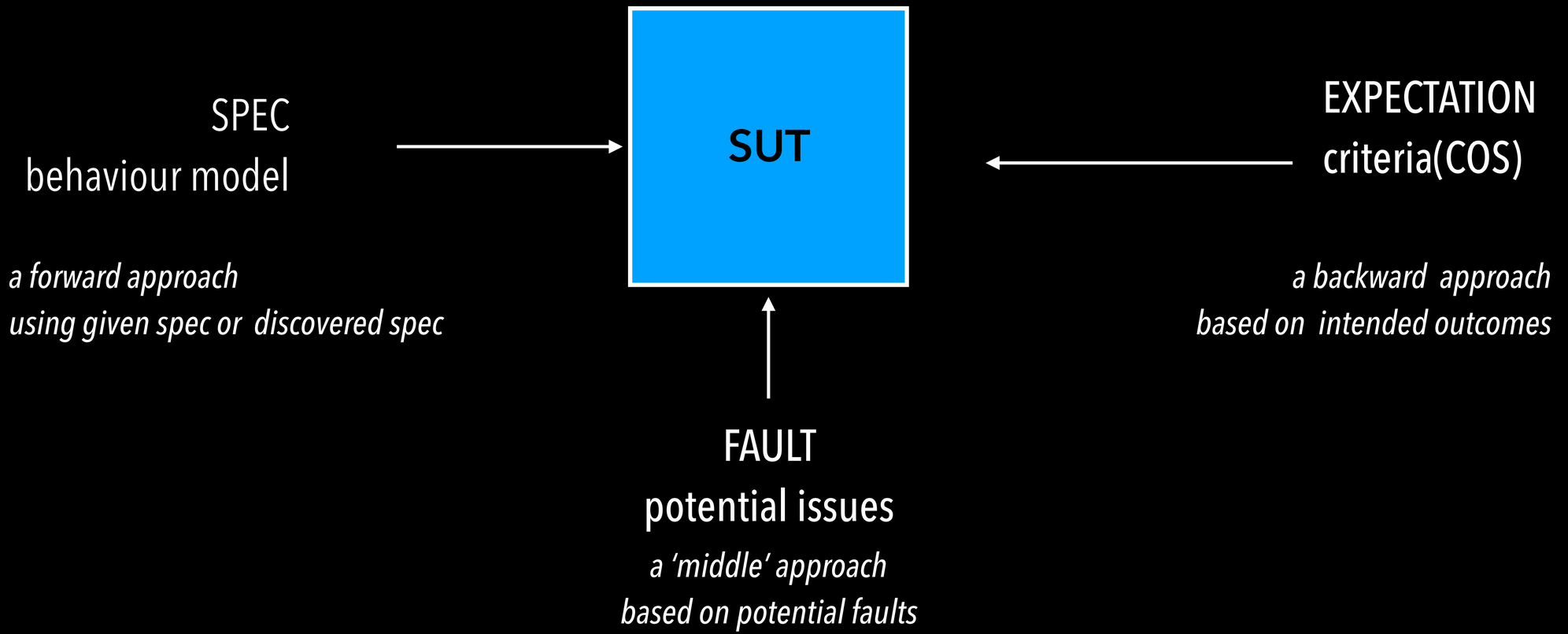
Design method



a forward approach
using given spec or discovered spec

a backward approach
based on intended outcomes

Design method



a forward approach
using given spec or discovered spec

a backward approach
based on intended outcomes

a 'middle' approach
based on potential faults

SmartDesign

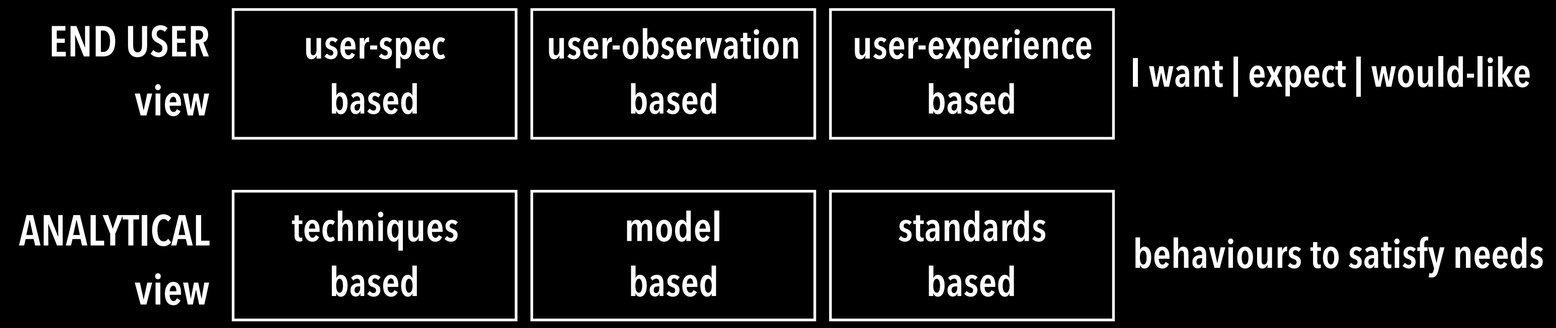
scenarios from
different views

SmartDesign

scenarios from
different views

SmartDesign

scenarios from different views



SmartDesign

scenarios from different views

END USER view	user-spec based	user-observation based	user-experience based	I want | expect | would-like
ANALYTICAL view	techniques based	model based	standards based	behaviours to satisfy needs
CONSTRUCTION view	coverage based	technology based	architecture based	that are implemented well

SmartDesign

scenarios from different views

END USER view	user-spec based	user-observation based	user-experience based		I want | expect | would-like
ANALYTICAL view	techniques based	model based	standards based		behaviours to satisfy needs
CONSTRUCTION view	coverage based	technology based	architecture based		that are implemented well
TEST /QUALITY view	fault oriented	robustness oriented	failure mode based	exploration based	and comprehensively covered questioning driven

SmartDesign

scenarios from different views

END USER view	user-spec based	user-observation based	user-experience based		I want | expect | would-like
ANALYTICAL view	techniques based	model based	standards based		behaviours to satisfy needs
CONSTRUCTION view	coverage based	technology based	architecture based		that are implemented well
TEST /QUALITY view	fault oriented	robustness oriented	failure mode based	exploration based	and comprehensively covered questioning driven
EXPERIENCE view	support issue(s) based	fault patterns based			repeating no prior mistakes

SmartDesign

scenarios from different views

END USER view	user-spec based	user-observation based	user-experience based		I want | expect | would-like
ANALYTICAL view	techniques based	model based	standards based		behaviours to satisfy needs
CONSTRUCTION view	coverage based	technology based	architecture based		that are implemented well
TEST /QUALITY view	fault oriented	robustness oriented	failure mode based	exploration based	and comprehensively covered questioning driven
EXPERIENCE view	support issue(s) based	fault patterns based			repeating no prior mistakes
OPERATIONAL view	usage profile based	deployment oriented			to help me do well on my environments

SmartDesign

scenarios from different views

END USER view	user-spec based	user-observation based	user-experience based		I want | expect | would-like
ANALYTICAL view	techniques based	model based	standards based		behaviours to satisfy needs
CONSTRUCTION view	coverage based	technology based	architecture based		that are implemented well
TEST /QUALITY view	fault oriented	robustness oriented	failure mode based	exploration based	and comprehensively covered questioning driven
EXPERIENCE view	support issue(s) based	fault patterns based			repeating no prior mistakes
OPERATIONAL view	usage profile based	deployment oriented			to help me do well on my environments
EVOLUTION view	code-change propagation	env-change impact			with no side effects

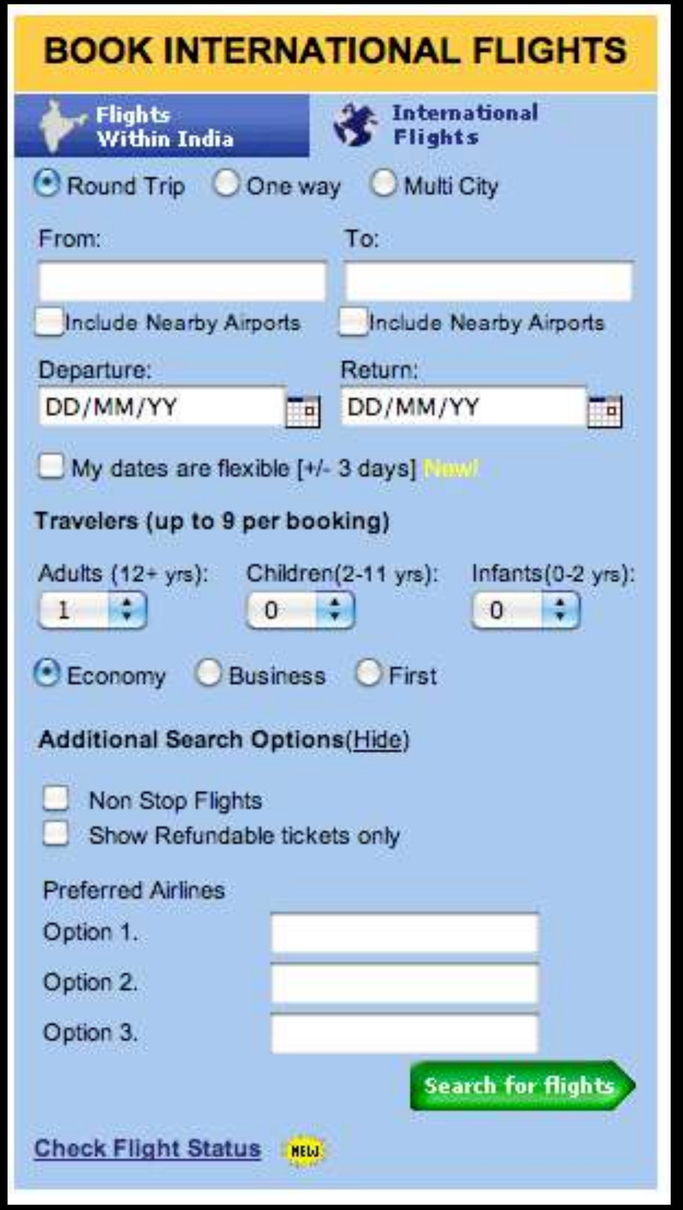
SmartDesign

scenarios from different views

END USER view	user-spec based	user-observation based	user-experience based		I want | expect | would-like
ANALYTICAL view	techniques based	model based	standards based		behaviours to satisfy needs
CONSTRUCTION view	coverage based	technology based	architecture based		that are implemented well
TEST /QUALITY view	fault oriented	robustness oriented	failure mode based	exploration based	questioning driven	and comprehensively covered
EXPERIENCE view	support issue(s) based	fault patterns based			repeating no prior mistakes
OPERATIONAL view	usage profile based	deployment oriented			to help me do well on my environments
EVOLUTION view	code-change propagation	env-change impact			with no side effects

"I want | expect | would-like behaviours to satisfy needs that are implemented well and comprehensively covered to help me do well on my environments with no side effects"

Level based design & evaluation



L1 Input correctness

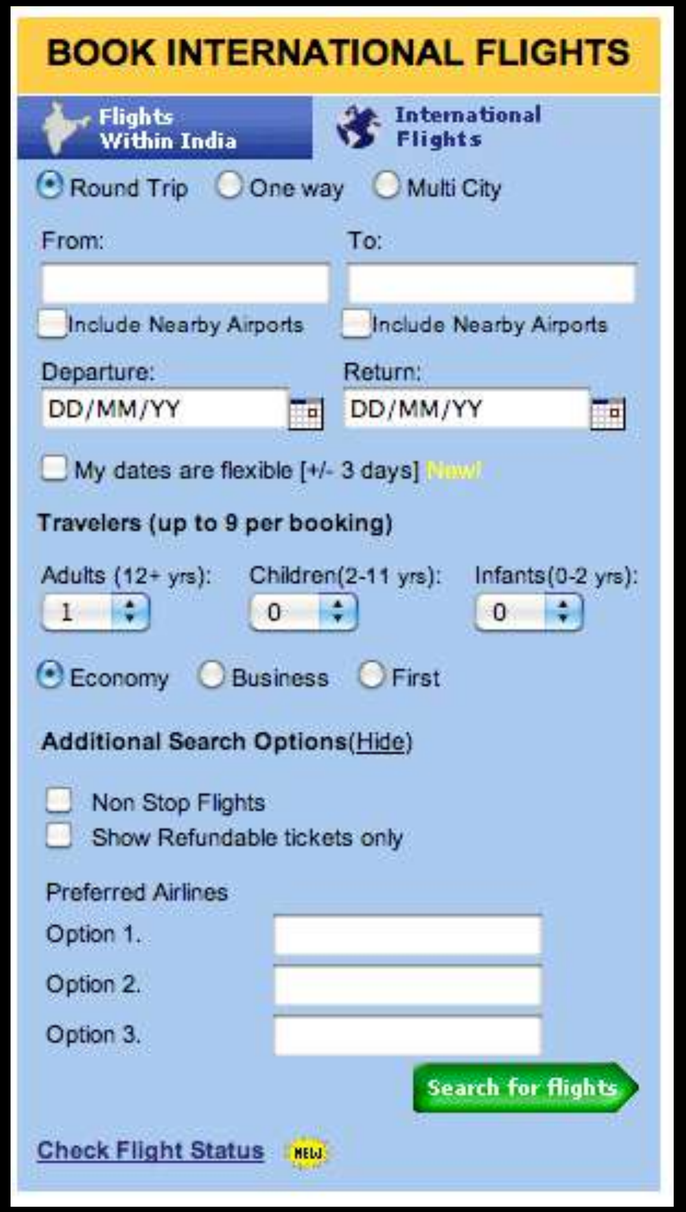
Boundaries, conditions, limits

L2 Interface correctness

defaults, tab order, layout..

L1 Input correctness

Boundaries, conditions, limits



L3 Structural correctness

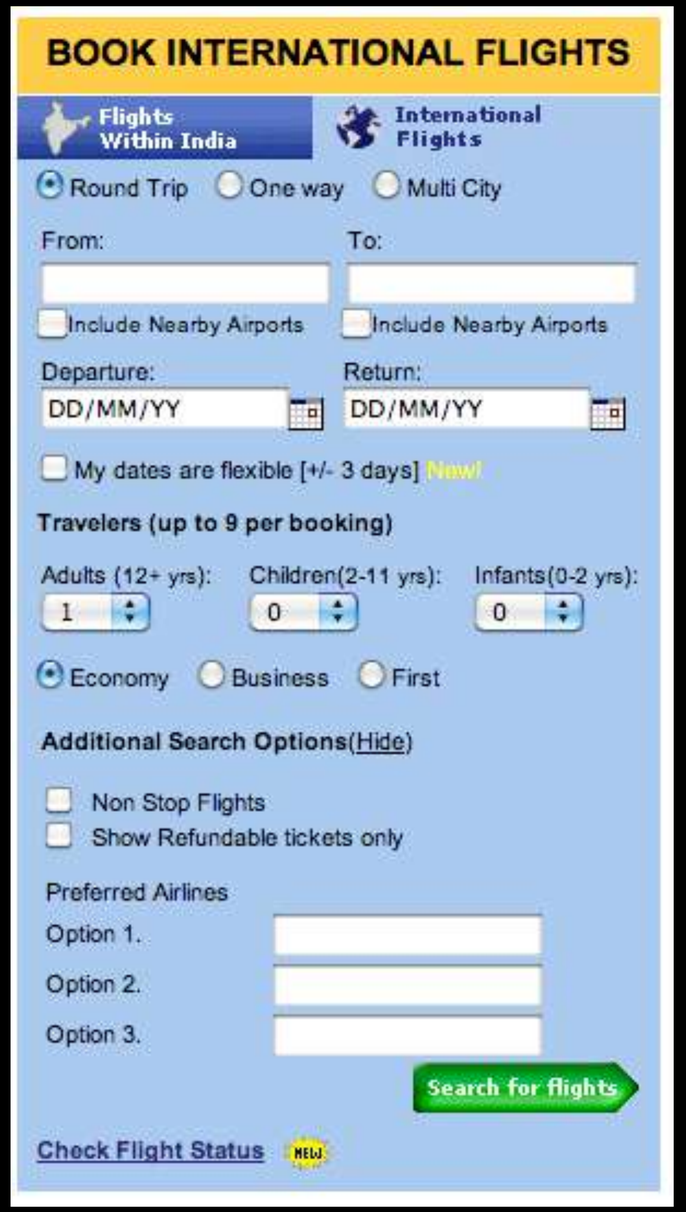
L2 Interface correctness

L1 Input correctness

error handling(connection)…

defaults, tab order, layout..

Boundaries, conditions, limits



L4 Behaviour correctness

L3 Structural correctness

L2 Interface correctness

L1 Input correctness

functional correctness

error handling(connection)…

defaults, tab order, layout..

Boundaries, conditions, limits

Trip Type 3

Unique From/To 2

Nearby airports? 2

Ret >= Dep >=Today 2

Flexible dates? 2

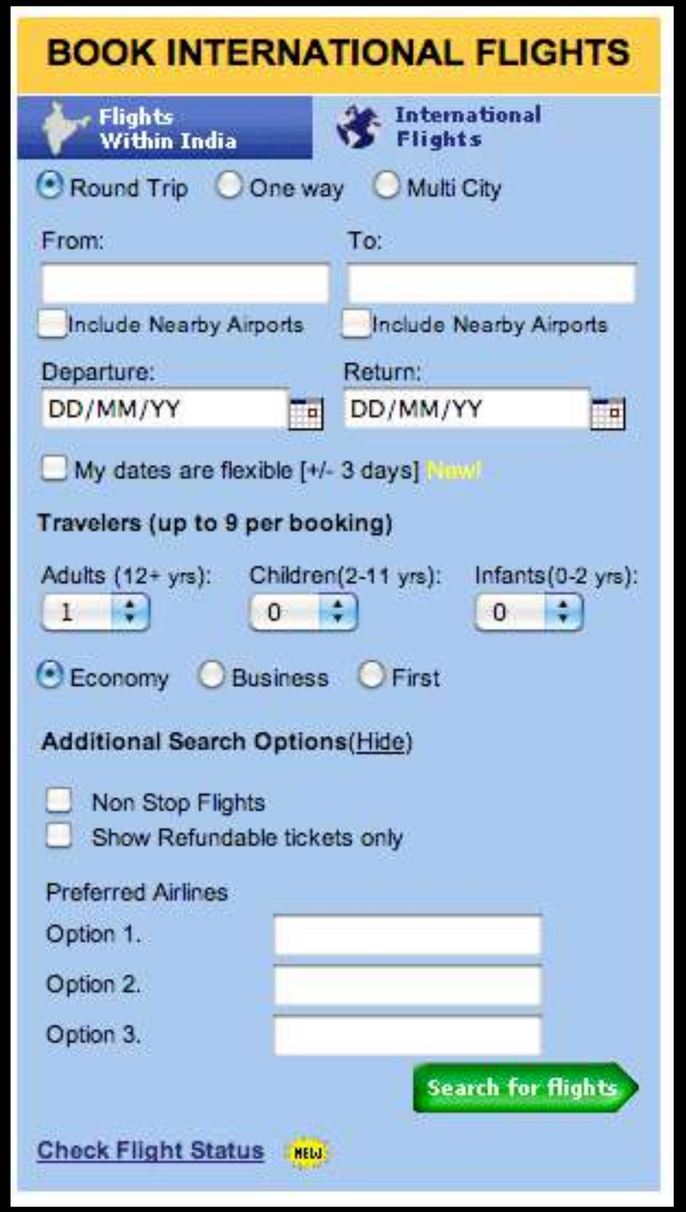
#Travelers valid? 2

Class? 3

Non stop? 2

Refundable? 2

Preferred airlines? 2



#TS (Functional) 13 - 1000+

L8 Deployment correctness

L7 Attribute correctness

L6 Environment correctness

L5 Flow correctness

L4 Behaviour correctness

L3 Structural correctness

L2 Interface correctness

L1 Input correctness

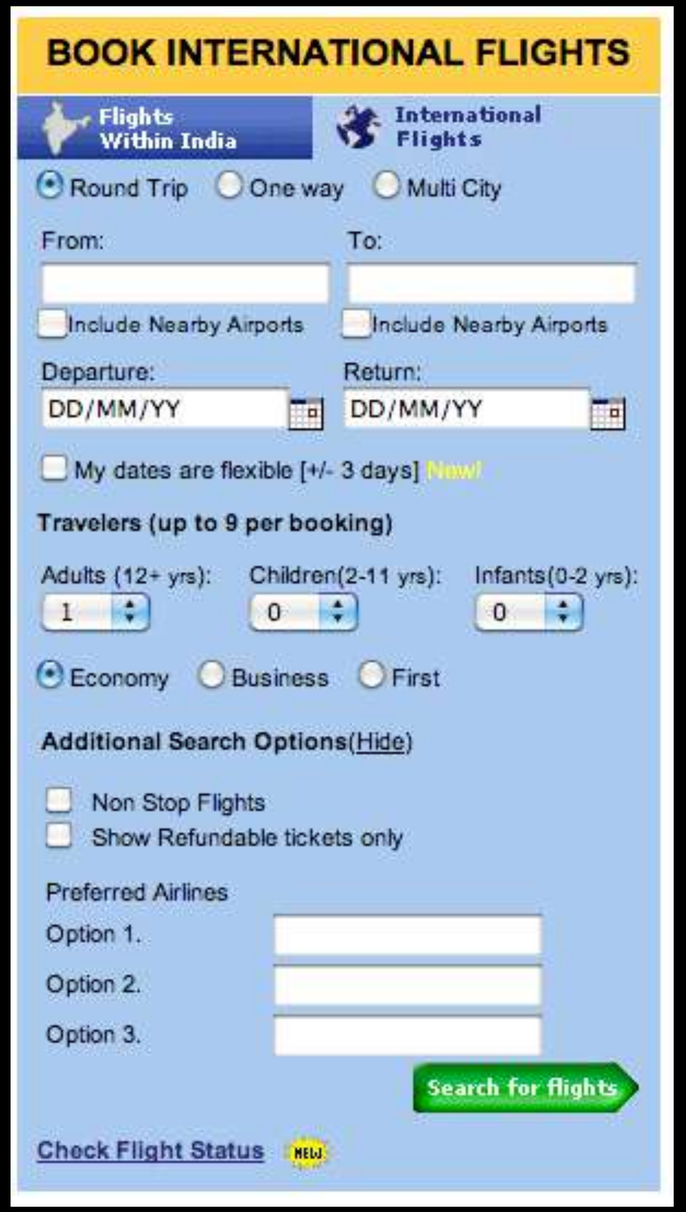
Shift Left?
performance, load,
responsiveness..

functional correctness

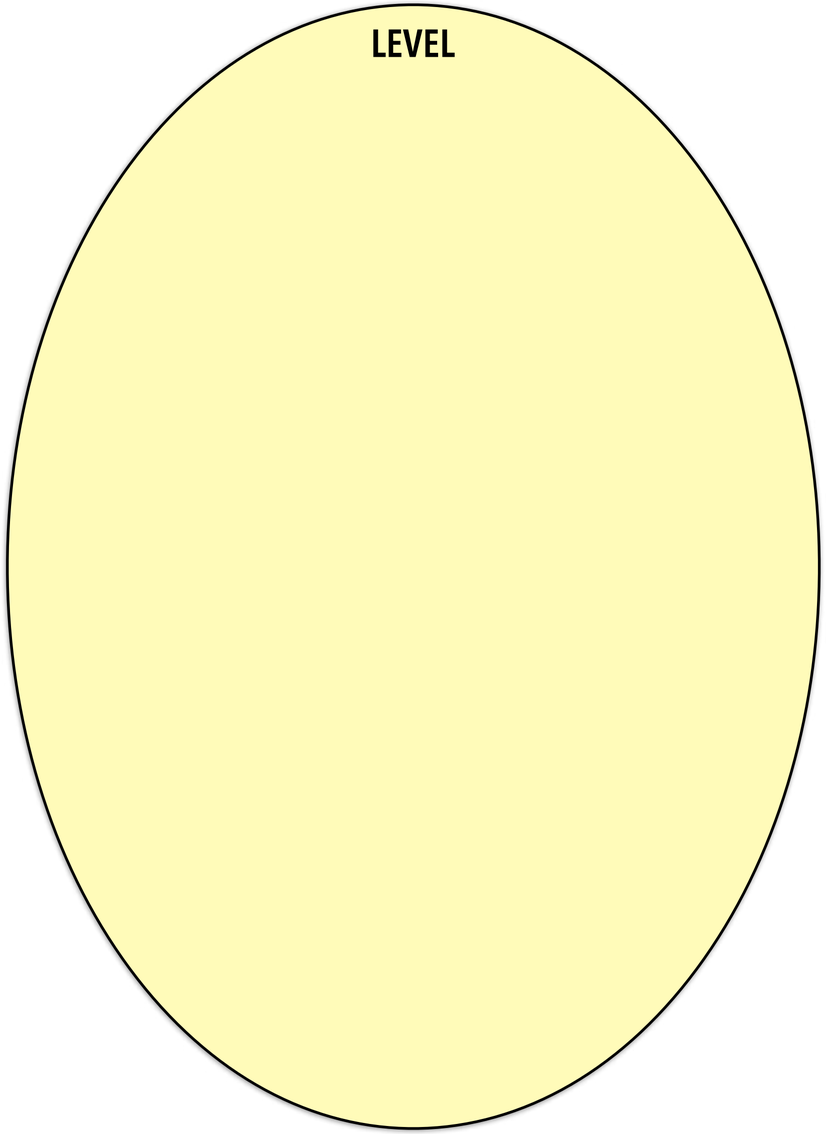
error handling(connection)…

defaults, tab order, layout..

Boundaries, conditions, limits

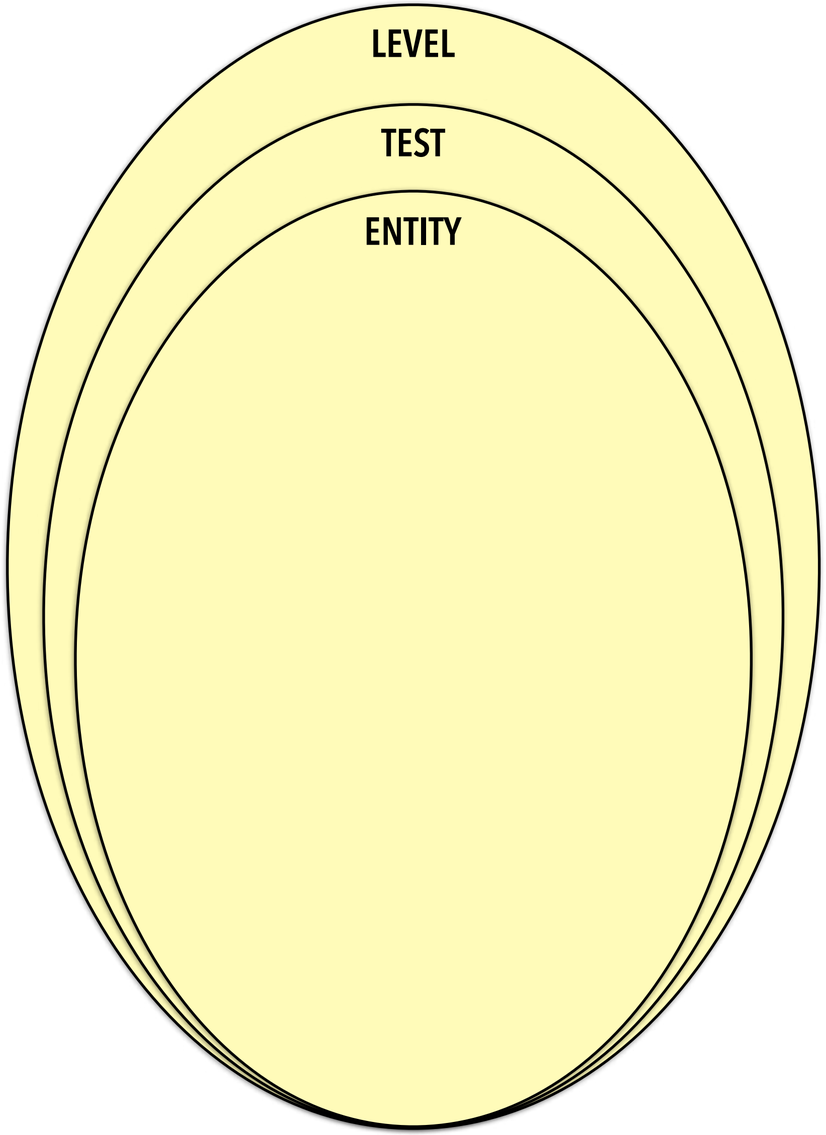


Test case architecture



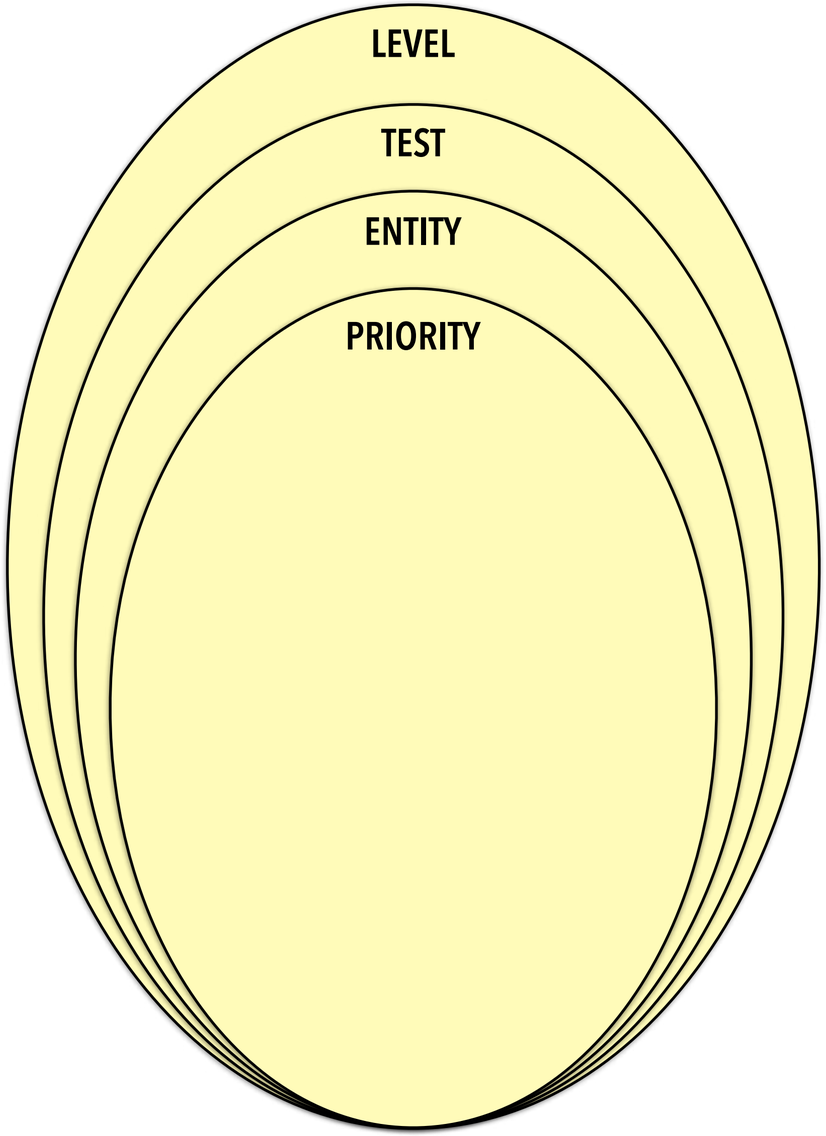
TS

Arranged
by LEVELS,



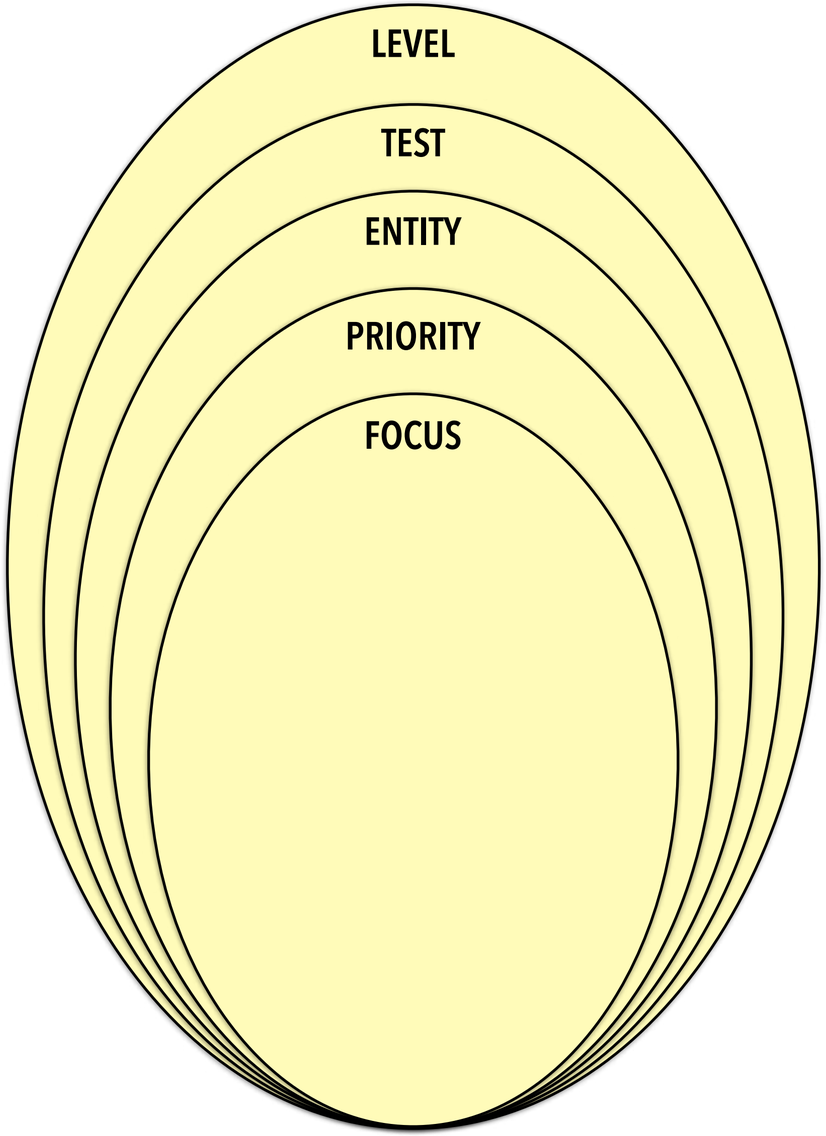
TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,



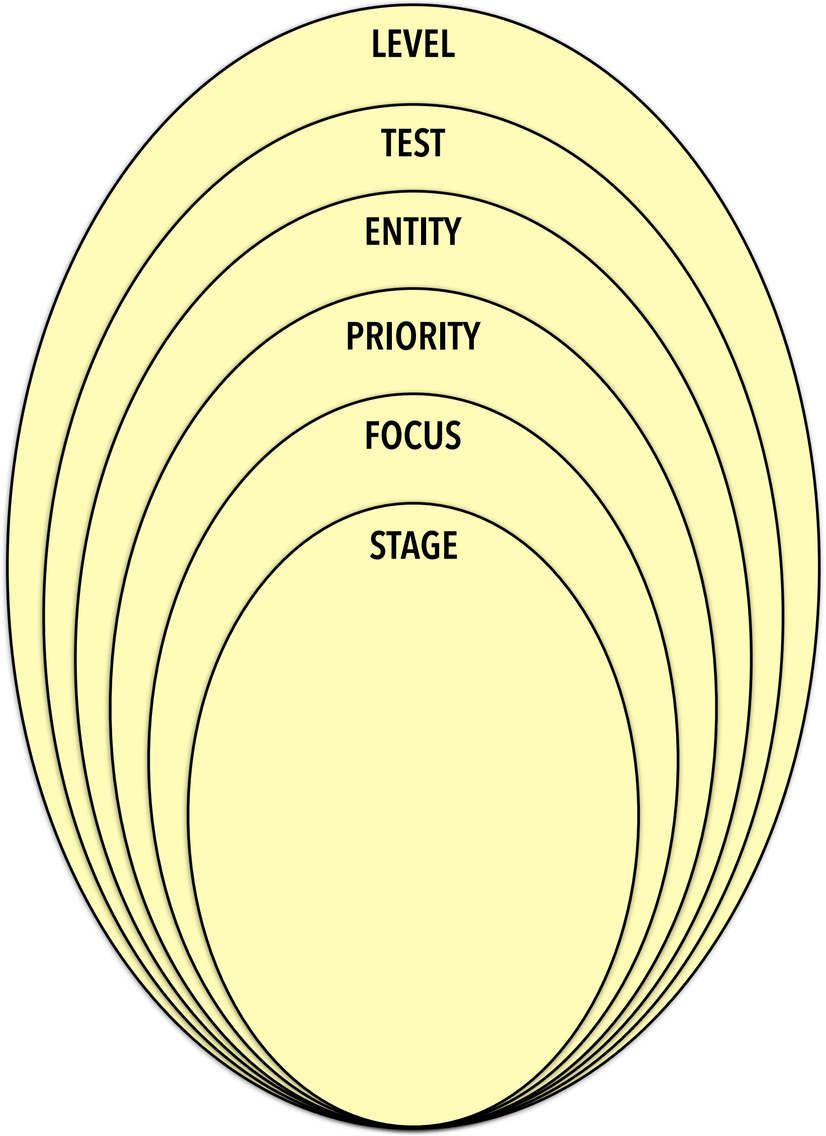
TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,



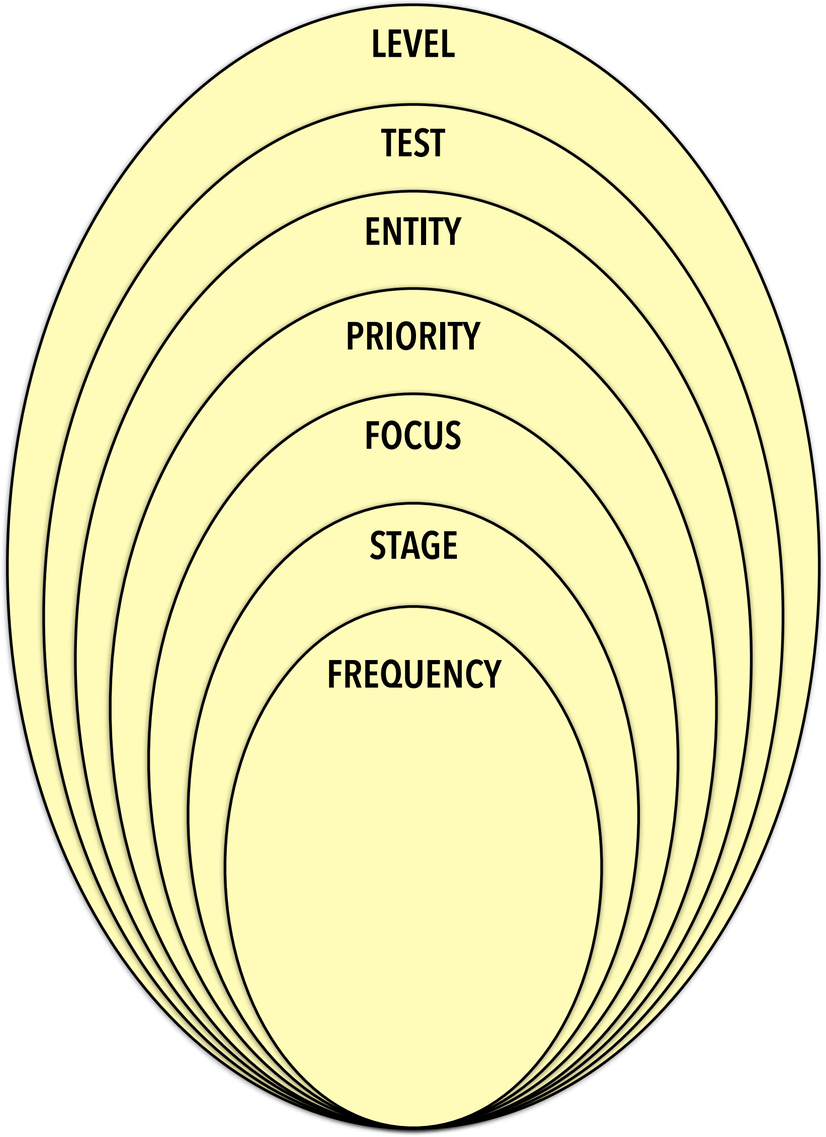
TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,
categorised by FOCUS,



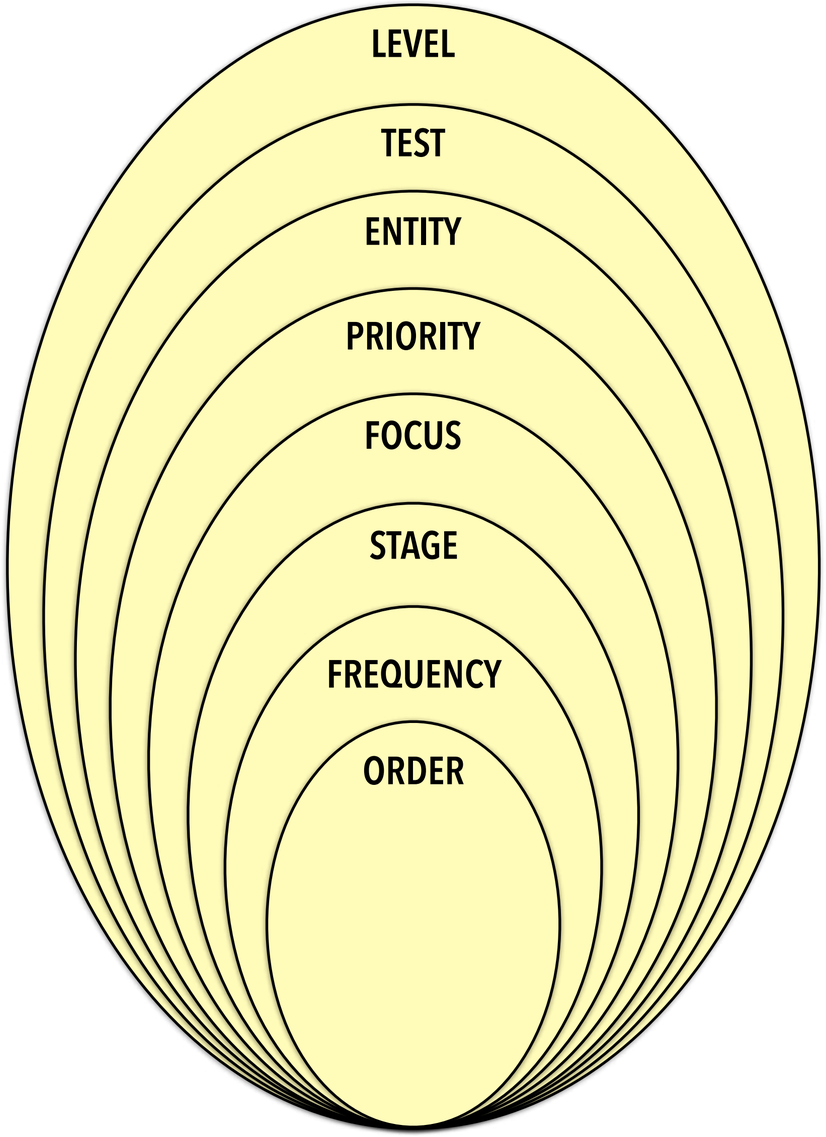
TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,
categorised by FOCUS,
classified by evaluation STAGE,



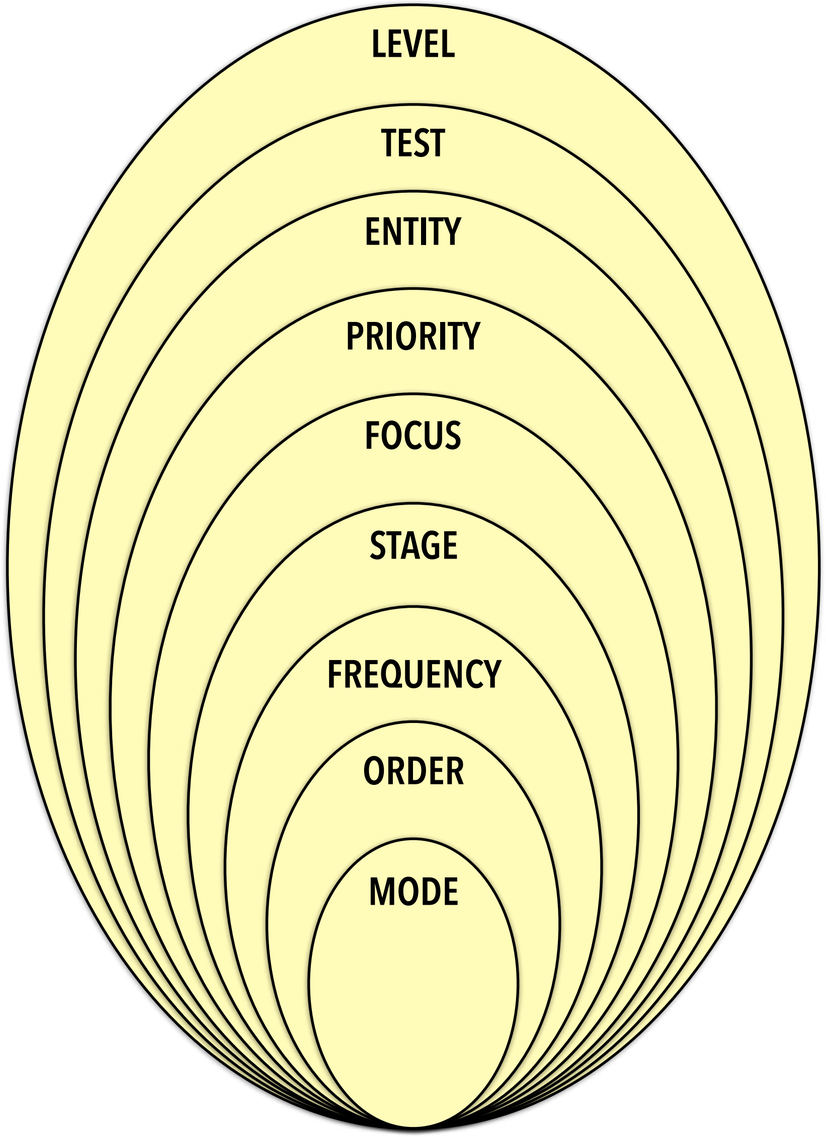
TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,
categorised by FOCUS,
classified by evaluation STAGE,
tagged with execution FREQUENCY,



TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,
categorised by FOCUS,
classified by evaluation STAGE,
tagged with execution FREQUENCY,
linked by optimal run ORDER,

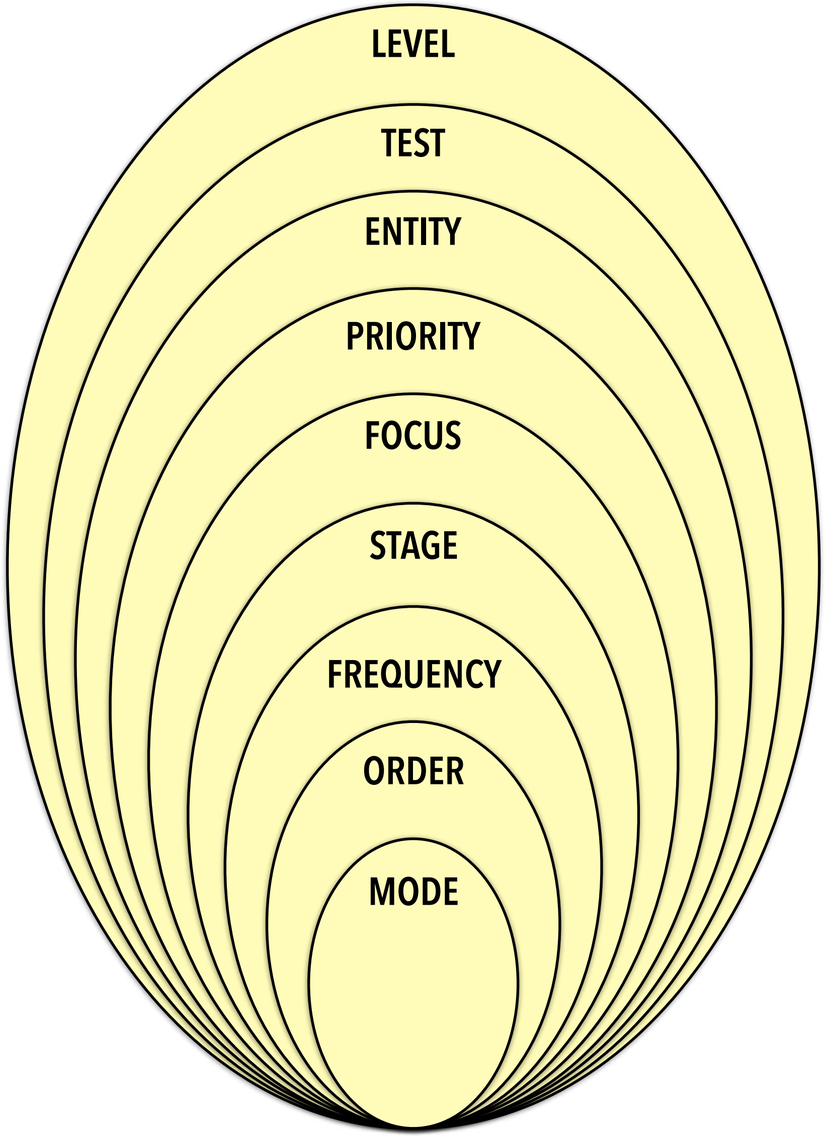


HBT test architecture (stagsoftware.com/hbtcentral)

TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,
categorised by FOCUS,
classified by evaluation STAGE,
tagged with execution FREQUENCY,
linked by optimal run ORDER,
collated by execution MODE.

A well architected set of test cases is like a effective bait that can 'attract' defects in the system.



HBT test architecture (stagsoftware.com/hbtcentral)

See aesthetics in this arrangement?

TS

Arranged
by LEVELS,
with ENTITIES
grouped by TEST,
ranked by PRIORITY,
categorised by FOCUS,
classified by evaluation STAGE,
tagged with execution FREQUENCY,
linked by optimal run ORDER,
collated by execution MODE.

A well architected set of test cases is like a effective bait that can 'attract' defects in the system.

“Role of test documentation styles and organisation”

Test documentation is often weary, dreary and lengthy.

Does writing style and organisation impact test effectiveness?
Does it have a bearing on the ability to be agile?
Should it be structured? or free format?
How can we be structured yet be creative?

Your comments/thoughts please.

Test writing style

Contents of test case document

Test objective

This is really a TEST SCENARIO

Pre-requisites

What is to be done before we stimulate this behaviour

Test data

INPUT-1	INPUT-2	INPUT-3	EXP. RESULT
A1	B1	C1	ER1
A2	B2	C2	ER2

Test steps

Actions to be done to execute this behaviour

Expected result

List ‘what to check for’ to assess correctness

Contents of test case document - How much do we need?

Test objective

This is really a TEST SCENARIO

Pre-requisites

What is to be done before we stimulate this behaviour

Test data

INPUT-1	INPUT-2	INPUT-3	EXP. RESULT
A1	B1	C1	ER1
A2	B2	C2	ER2

Test steps

Actions to be done to execute this behaviour

Expected result

List 'what to check for' to assess correctness

Typical contents of test documentation

Entity #1 (say Feature 1)
Entity #2 (say Feature 2)
Entity #3 (say Feature 3)

1. Do test wise
2. For each test, entity-wise
3. For each entity, Scenario-wise
4. For each scenario, list pre-requisites
5. For each scenario, list test cases using tabular notation
6. For each scenario, list detailed steps as needed
7. Tag scenario tag :
    a.to type Conformance (+ve)/ Robustness (-ve)
    b.with PRIORITY (say 1/2/3)
    c.to PDT as appropriate

Entity #1

Scenario#1

Pre-requisites

…

Test steps

1.

2.

Test Cases			

Scenario#2

…

Writing a test scenario - One liner

A test scenario reflects a behavior and is therefore “the path from beginning to end”

Write this in ONE sentence using IMPERATIVE style

Ensure that
the system does …”

+ve

Ensure that
the system does NOT …”

-ve

“Ensure that email is delivered when sender and receiver information is complete,
PDF file attached, payment method selected and terms/conditions agreed.”

...add attributes to test scenario like TYPE, PRIORITY

Degree of importance, maybe rank as 1/2/3

What type of scenario? Positive or Negative

	Objective		
TS1	Ensure that email is delivered when sender and receiver information is complete, PDF file attached, payment method selected and terms/conditions agreed.	+ve	1

TC ID	SEND By	Pgs2Print	PaperSize	Two Sided?	Colour?	EXP. RESULT
TS1.TC1	Normal Post	1	A4	Yes	Yes	Successful
TS1.TC2	Speed Post	2	A5	No	No (B/W)	Successful

Writing test cases - Tabular

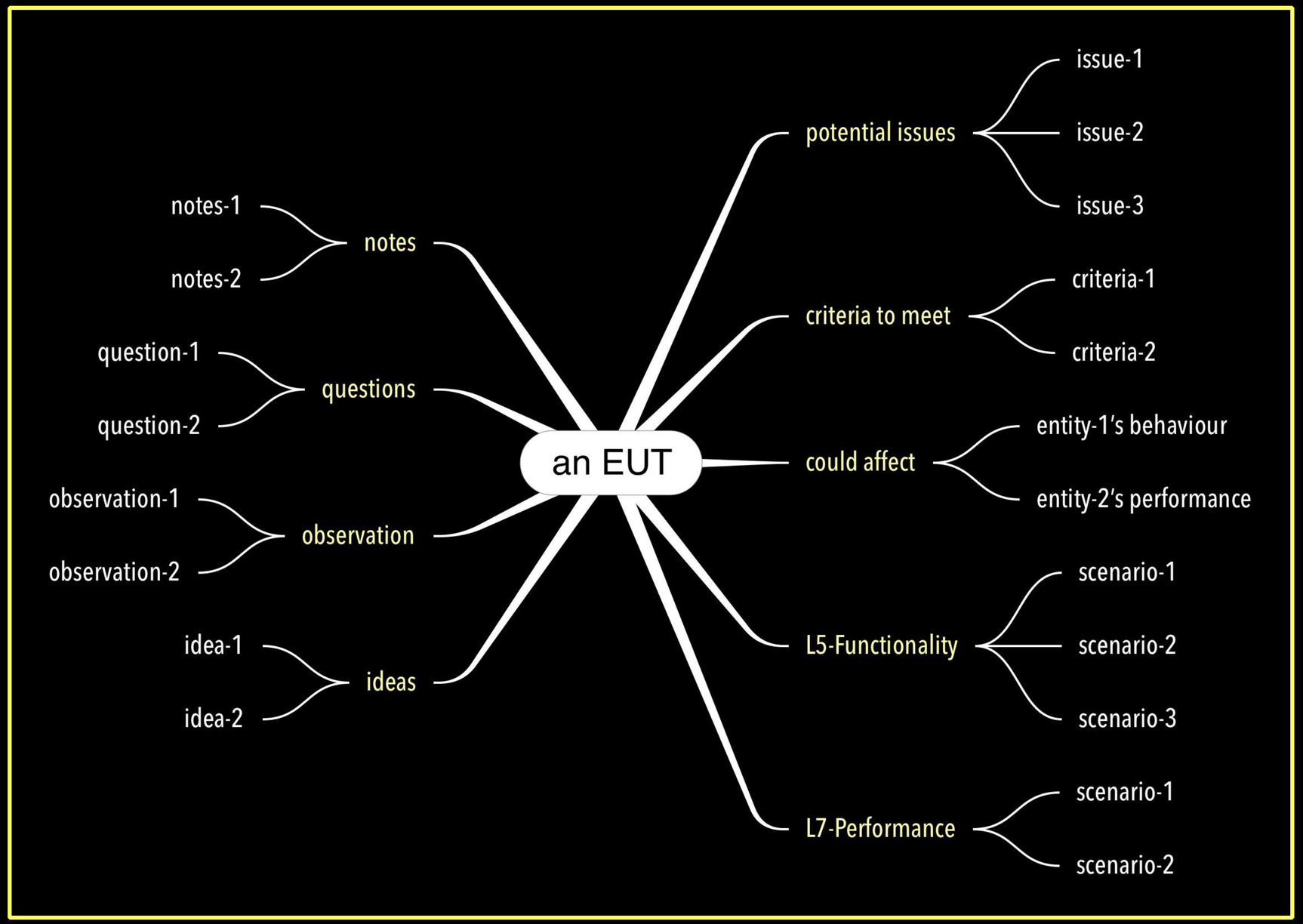
A test case is an unique combination of a set of inputs to stimulate a behavior

Consider the example where the following inputs vary for this scenario

INPUT	Values
SendBy	SpeedPost, NormalPost
Pgs2Print	1, 2
PaperSize	A4, A5
Two-sided?	Yes/No
Colour?	C/BW

TC ID	SEND By	Pgs2Print	PaperSize	Two Sided?	Colour?	EXP. RESULT
TS1.TC1	Normal Post	1	A4	Yes	Yes	Successful
TS1.TC2	Speed Post	2	A5	No	No (B/W)	Successful

Could we ideate & document non-linearly?



SmartQA

Thank you.

www.stagsoftware.com