A Taxonomy of System and Software Testing Types

2015 FAA V&V Summit
Atlantic City, New Jersey
23 September 2015

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Donald G. Firesmith, Principal Engineer
20 September 2015

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

DM-0001886.

Topics

Relevant Testing Challenges

Goals of Presentation

What is Testing?

Presentation Scope

Testing Types

Conclusion

Relevant Testing Challenges

Many testers are only aware of a minority of types of testing, let alone know how to perform them.

Most testers know:

- A lot about a few types of testing (enough to perform)
- A little about some additional types of testing (not enough to perform)
- Nothing about a significant number of testing types

Test managers and developers are aware of even fewer testing types.

System acquirers typically know little about software test types.

The test strategies, project test plans, and test sections of system/software development plans tend to identify a very small number of types of testing (e.g., unit, integration, system, and acceptance testing).

- Not planned → Not performed

Goals of the Presentation

Make it clear that:

- There are ***many*** different types of testing.
- Testing is a complex discipline with its own technical jargon.
- There is a lot of overlap between different classes of testing types.
  - Think multiple classification (object-oriented design) or multiple inheritance (object-oriented programming).

Get you to take a look at your:

- Test strategies and test plans and ask yourselves “Are they sufficiently complete?”
- Testers and other testing stakeholders and ask yourselves “Do they need additional training in testing types.

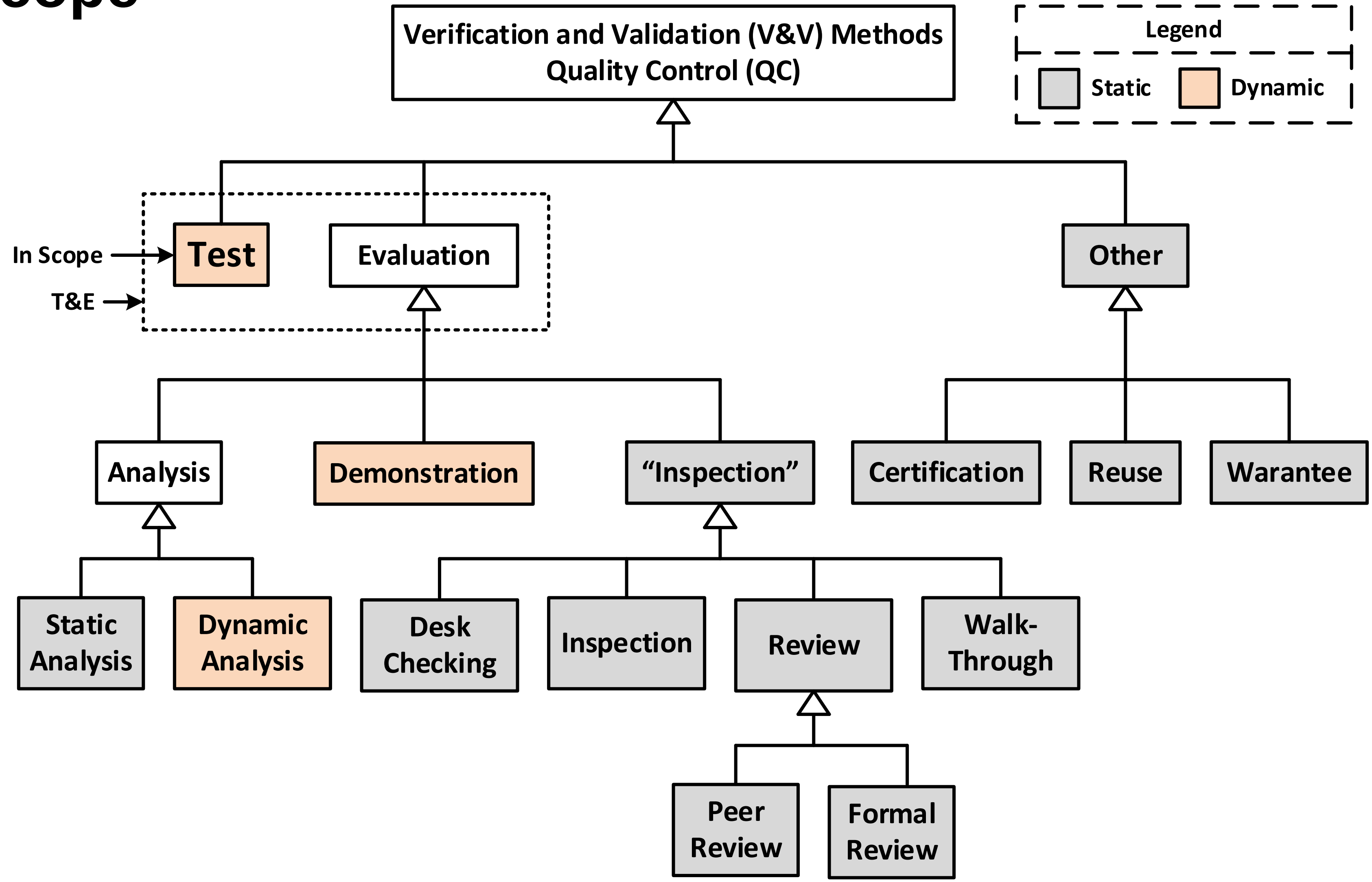
What is Testing?

Testing

The ***execution*** of an Object Under Test (OUT) under specific ***preconditions*** with specific ***stimuli*** so that its ***actual behavior*** can be compared with its ***expected or required behavior***

- *Preconditions:* pretest mode, states, stored data, or external conditions
- *Stimuli:*
  - Calls, commands, and messages (control flows)
  - Data inputs (data flows)
  - Trigger events such as state changes and temporal events
- *Actual Behavior:*
  - *During Test:*
    - Calls, commands, and messages (control flows)
    - Data outputs (data flows)
  - *Postconditions*: post-test mode, states, stored data, or external conditions

Presentation Scope



Types of Testing – Overview

A type of testing is:

- A specific way to perform testing
- A class or subclass of testing
- Much narrower in scope than a testing paradigm

Answer the 5W and 2H Questions

Test types can be classified and characterized (attributes) by how they answer the following classic 5W and 2H questions:

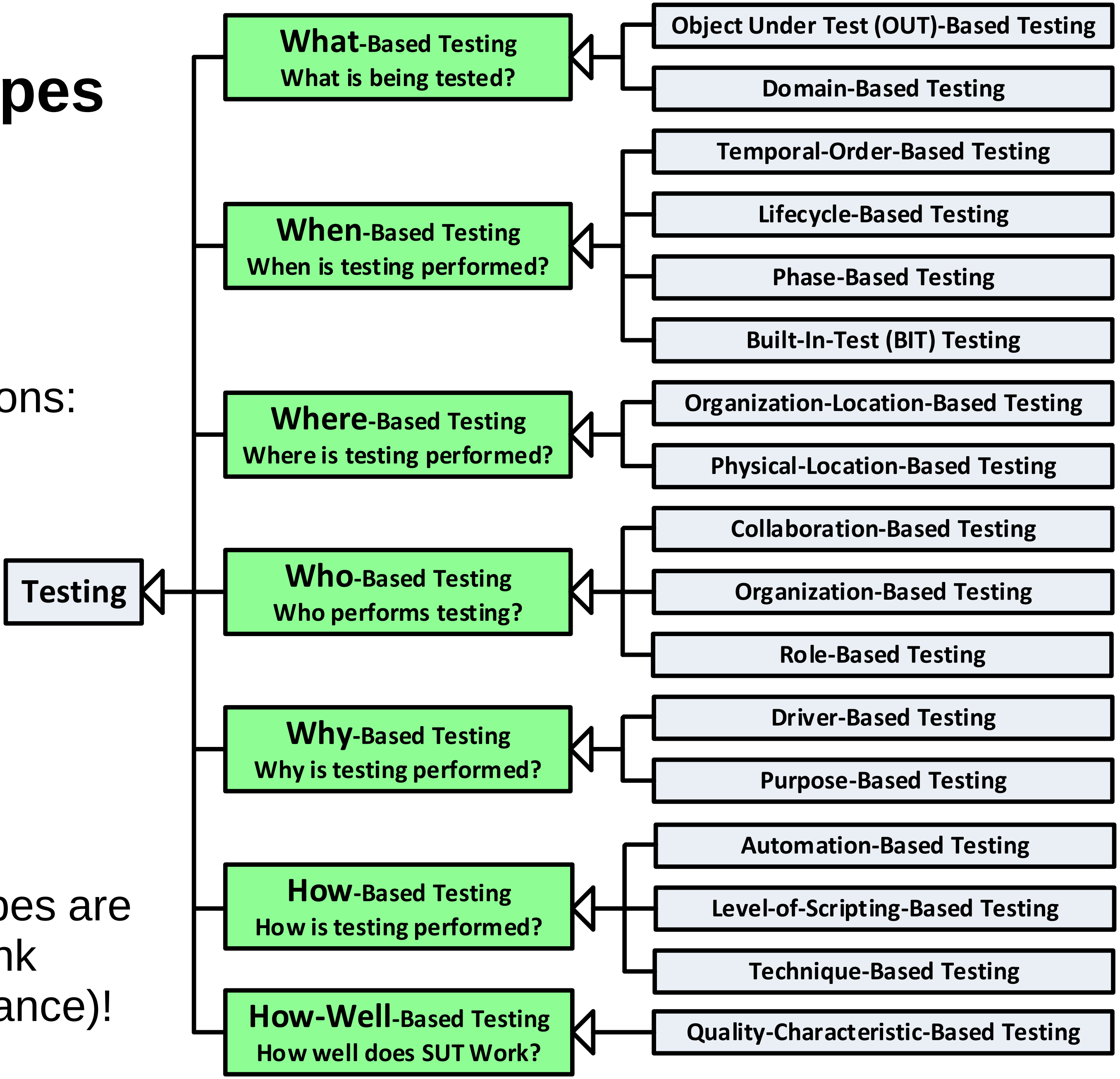
- **What** is being tested?
- **When** is the testing being performed?
- **Where** is the testing being performed?
- **Who** is performing the testing?
- **Why** is the testing being performed?
- **How** is the testing being performed?
- **How well** does the object under test perform its functions?

Testing Types

17 Categories Answering the
5W+2H Questions:

- **What?**
- **When?**
- **Where?**
- **Who?**
- **Why?**
- **How?**
- **How Well?**

These supertypes are not disjoint (think multiple inheritance)!



Testing

What-Based Testing
What is being tested?

Object Under Test (OUT)-Based Testing

Domain-Based Testing

When-Based Testing
When is testing performed?

Temporal-Order-Based Testing

Lifecycle-Based Testing

Phase-Based Testing

Built-In-Test (BIT) Testing

Where-Based Testing
Where is testing performed?

Organization-Location-Based Testing

Physical-Location-Based Testing

Who-Based Testing
Who performs testing?

Collaboration-Based Testing

Organization-Based Testing

Role-Based Testing

Why-Based Testing
Why is testing performed?

Driver-Based Testing

Purpose-Based Testing

How-Based Testing
How is testing performed?

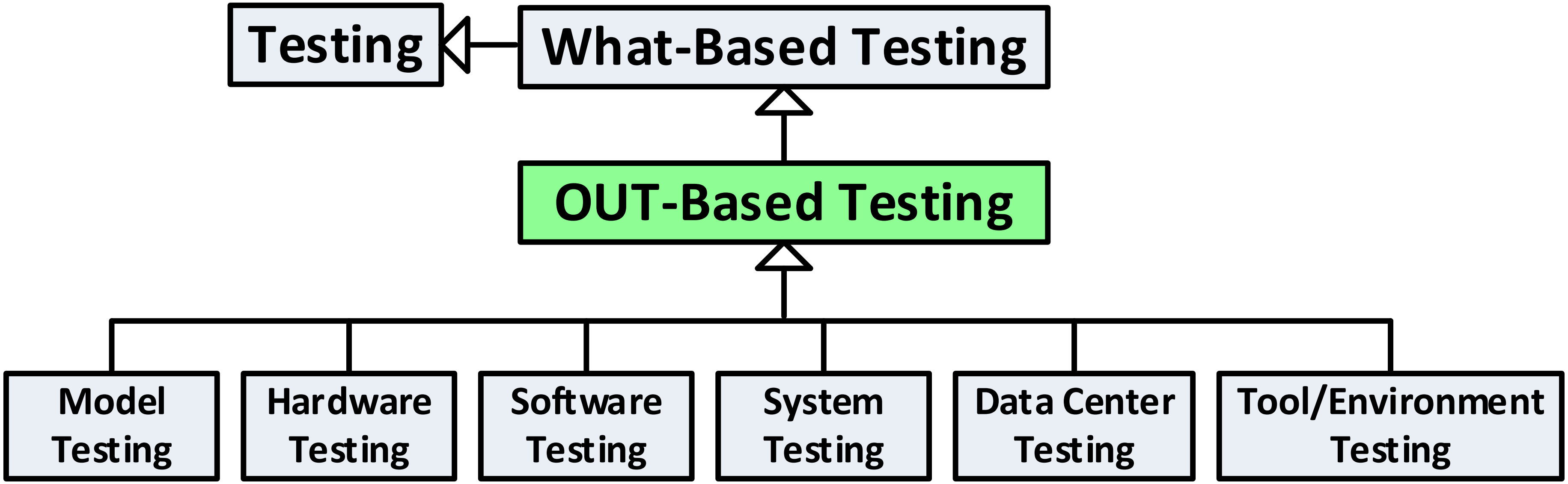
Automation-Based Testing

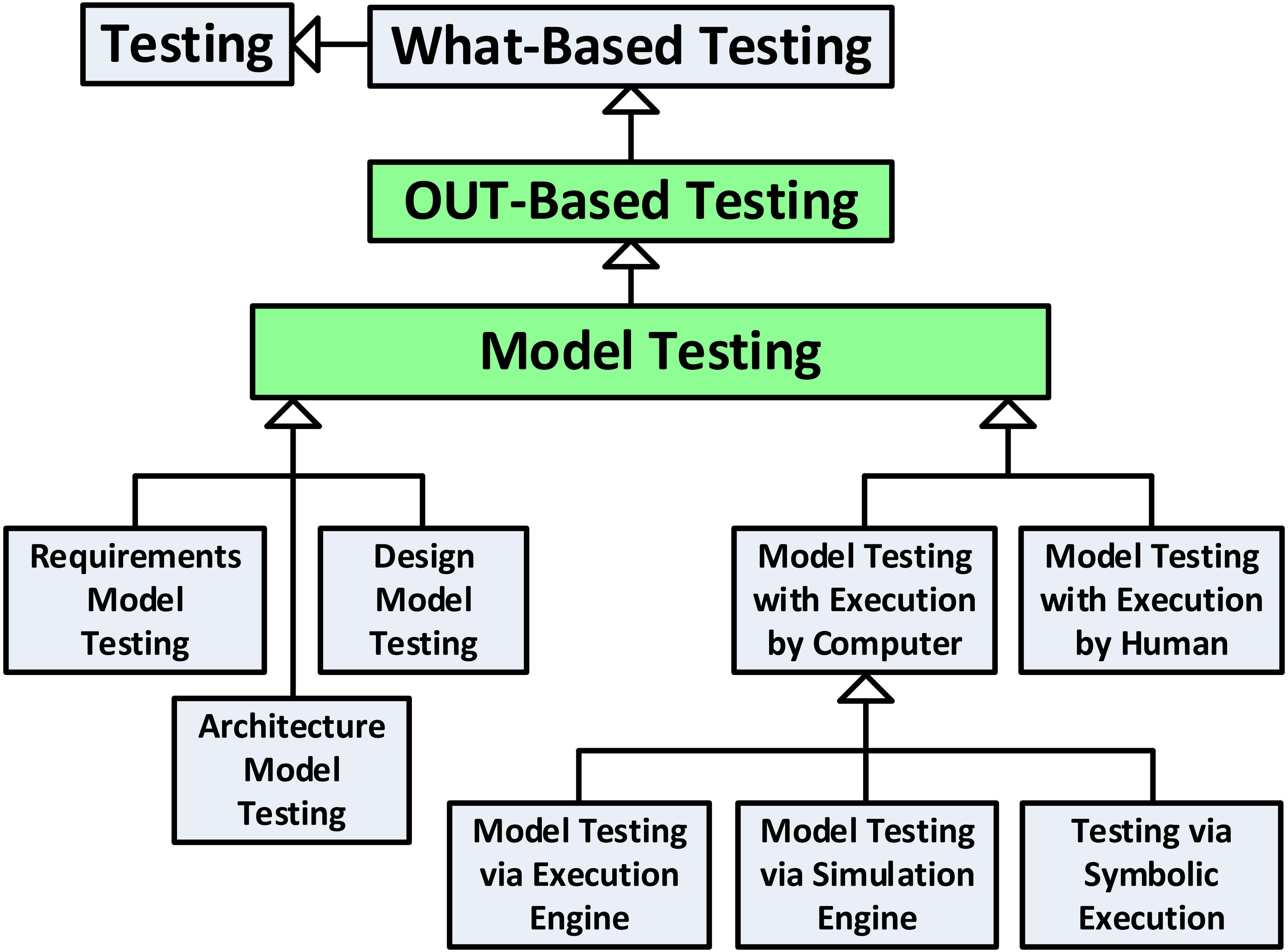
Level-of-Scripting-Based Testing

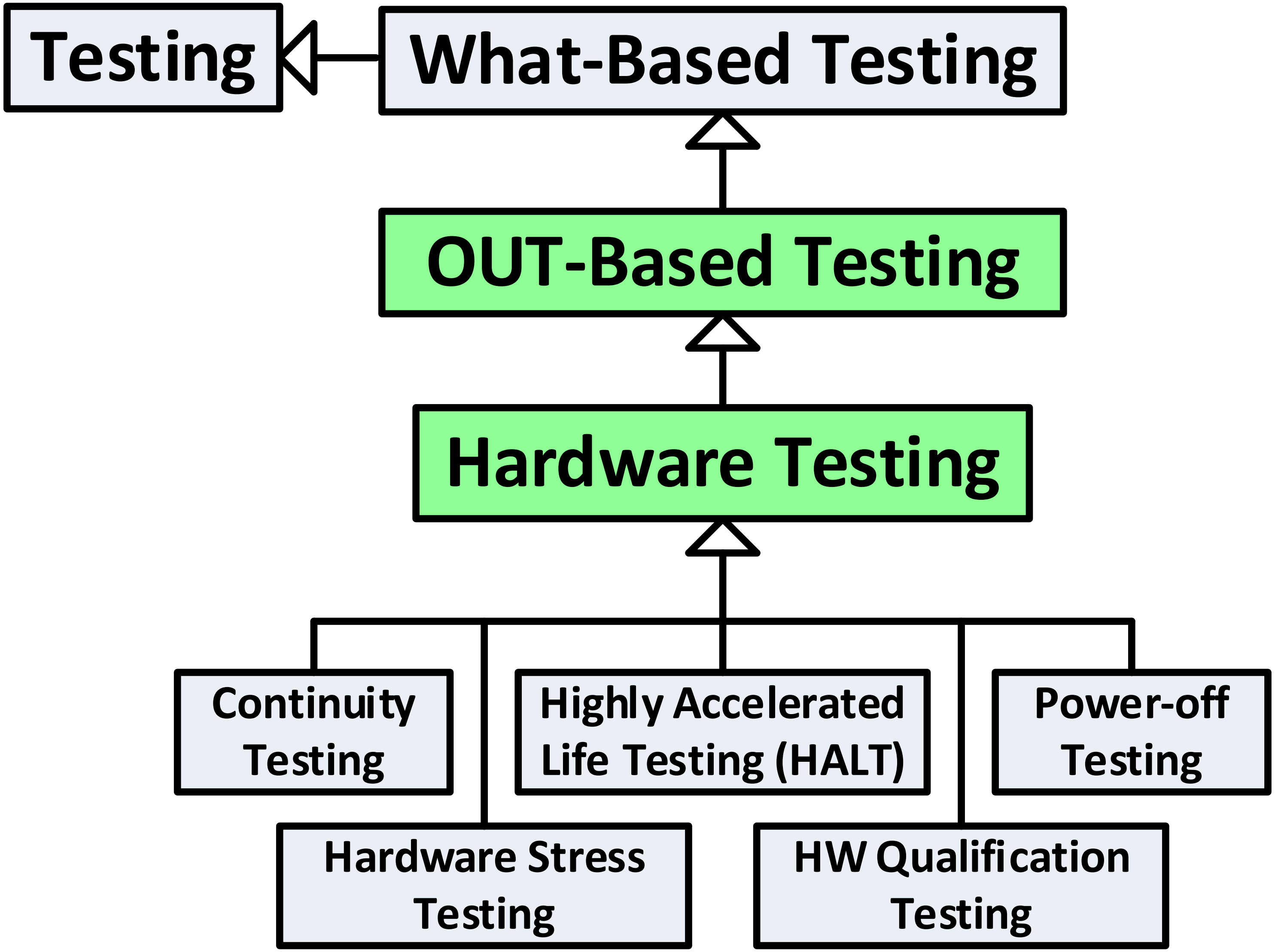
Technique-Based Testing

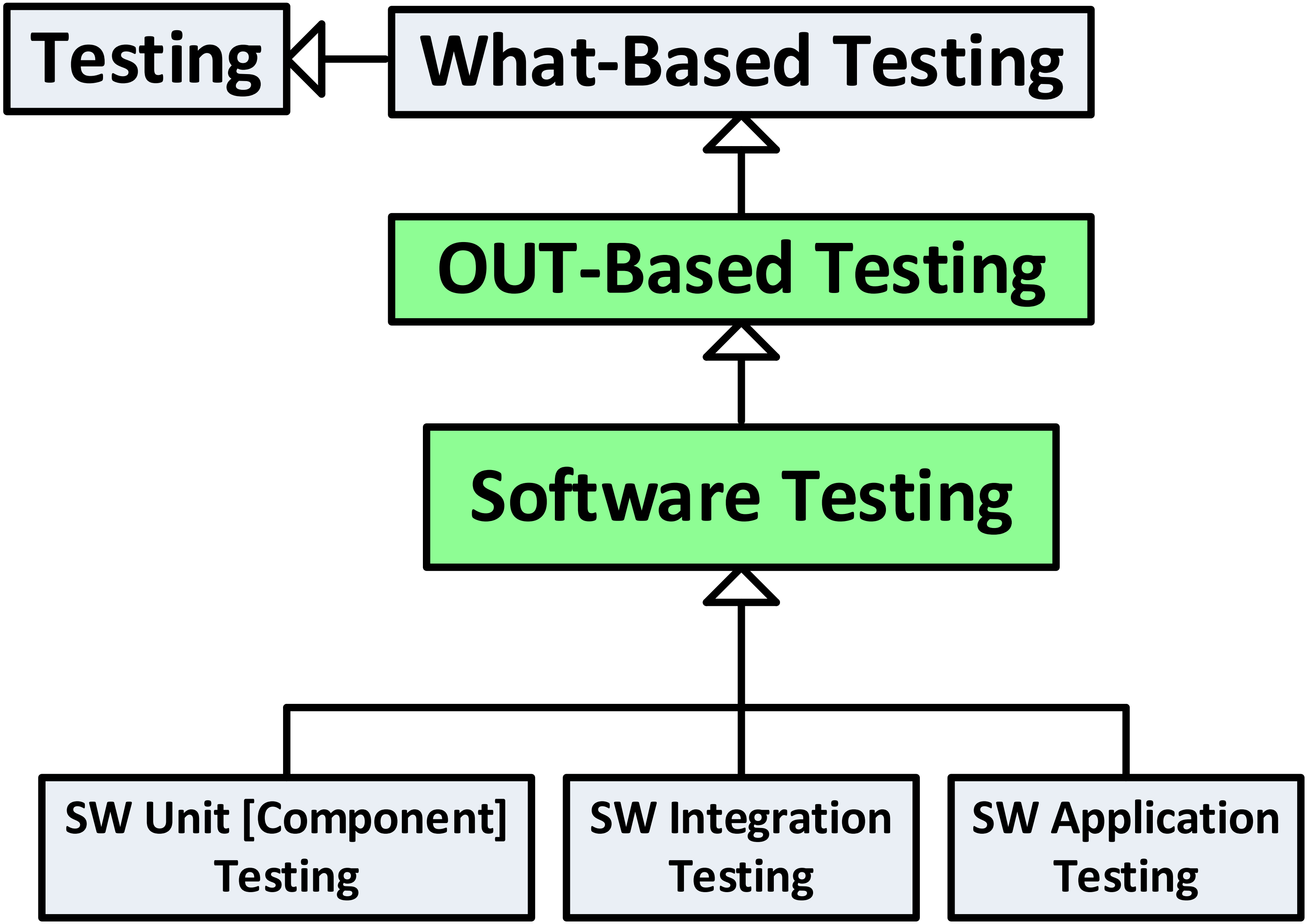
How-Well-Based Testing
How well does SUT Work?

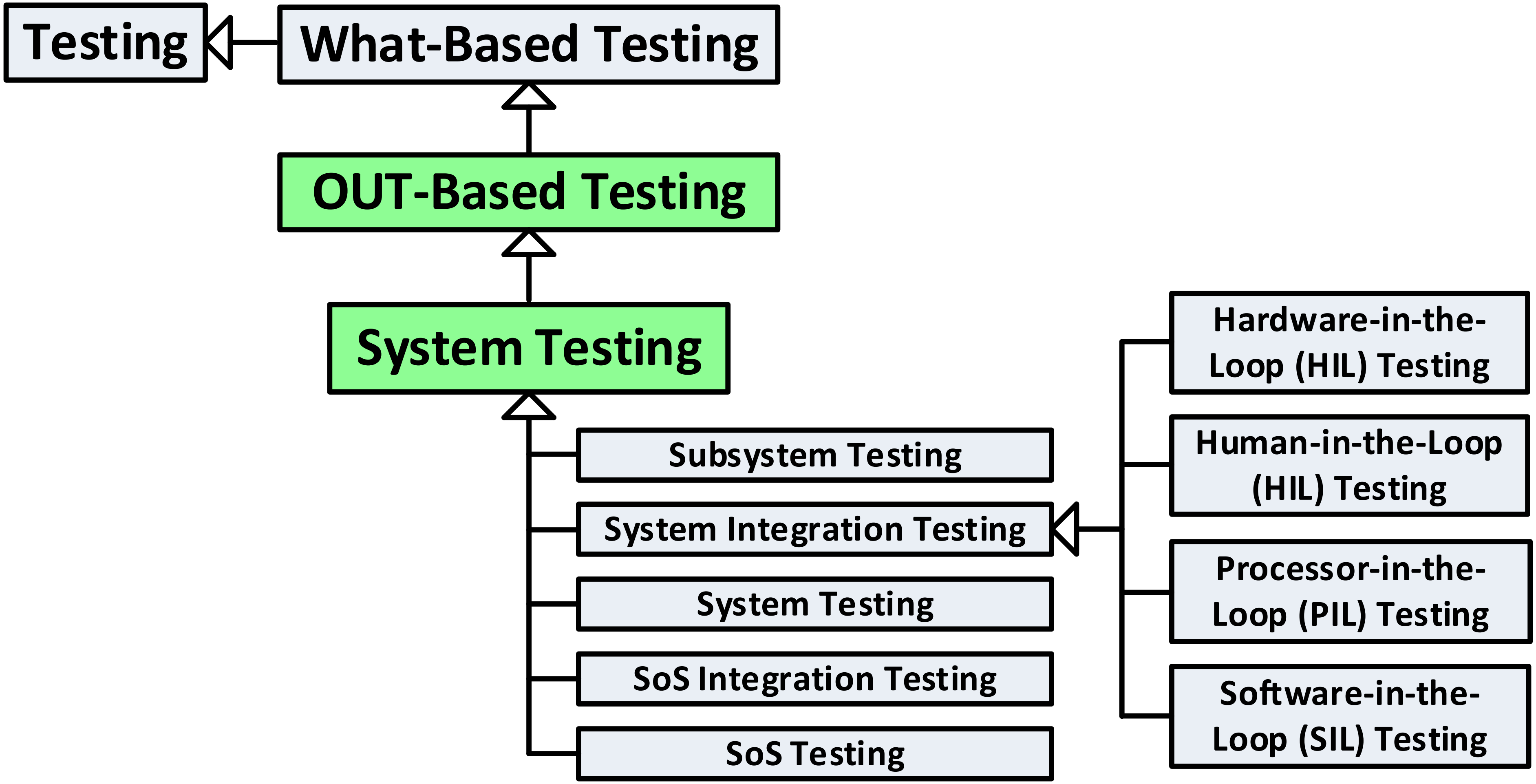
Quality-Characteristic-Based Testing

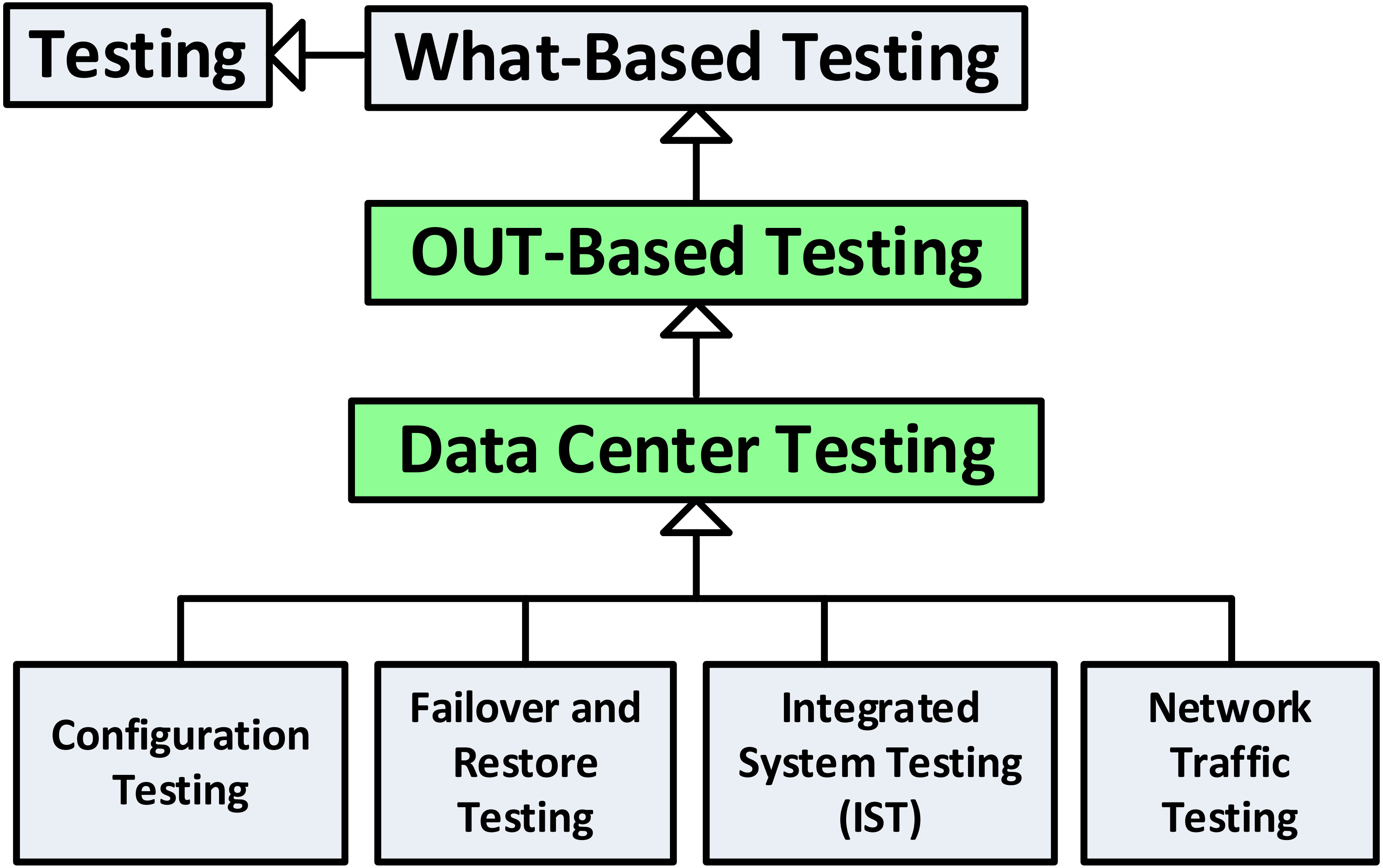

Testing
What-Based Testing
OUT-Based Testing
Model Testing
Hardware Testing
Software Testing
System Testing
Data Center Testing
Tool/Environment Testing

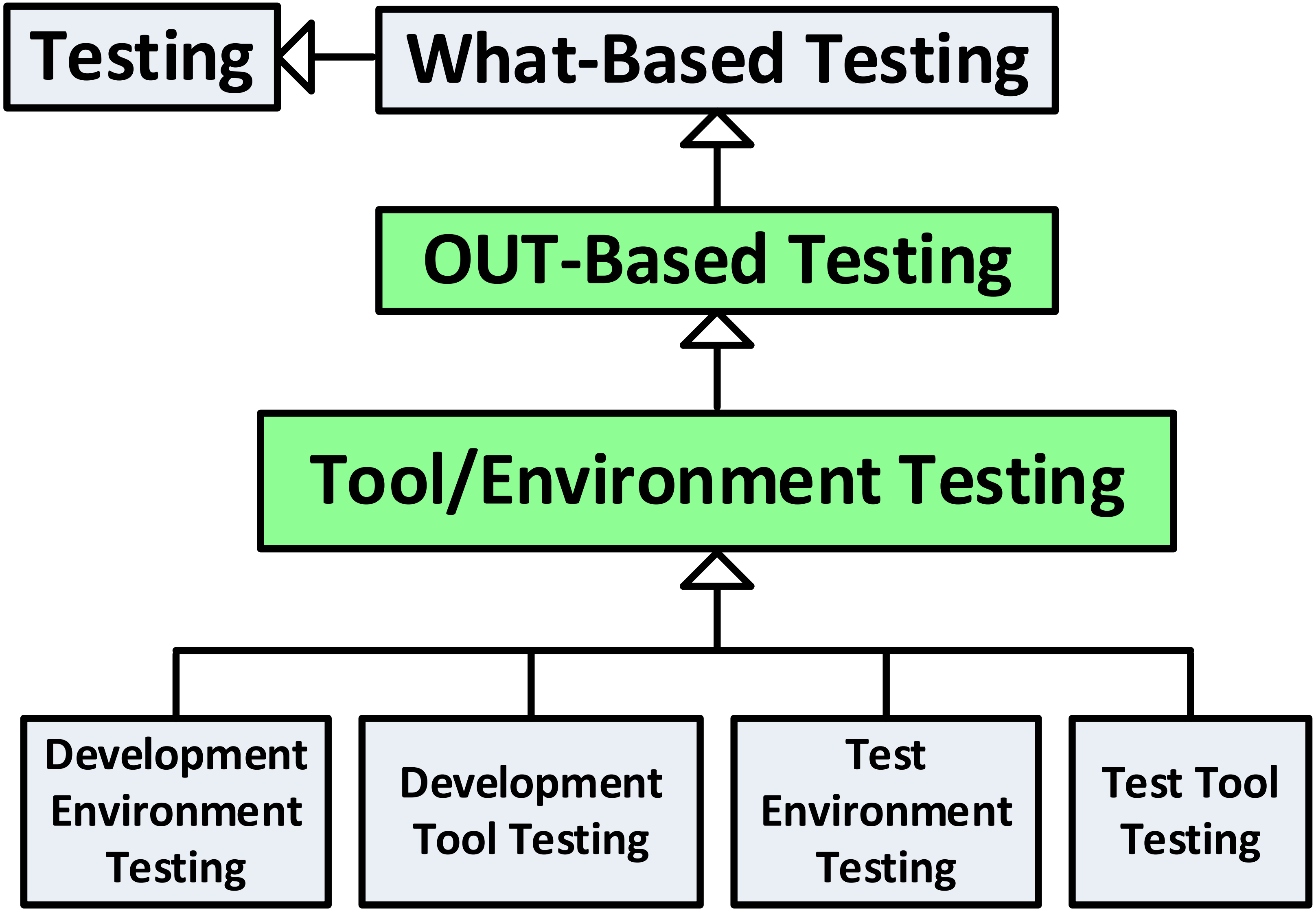

Testing
What-Based Testing
OUT-Based Testing
Model Testing
Requirements Model Testing
Design Model Testing
Architecture Model Testing
Model Testing with Execution by Computer
Model Testing with Execution by Human
Model Testing via Execution Engine
Model Testing via Simulation Engine
Testing via Symbolic Execution

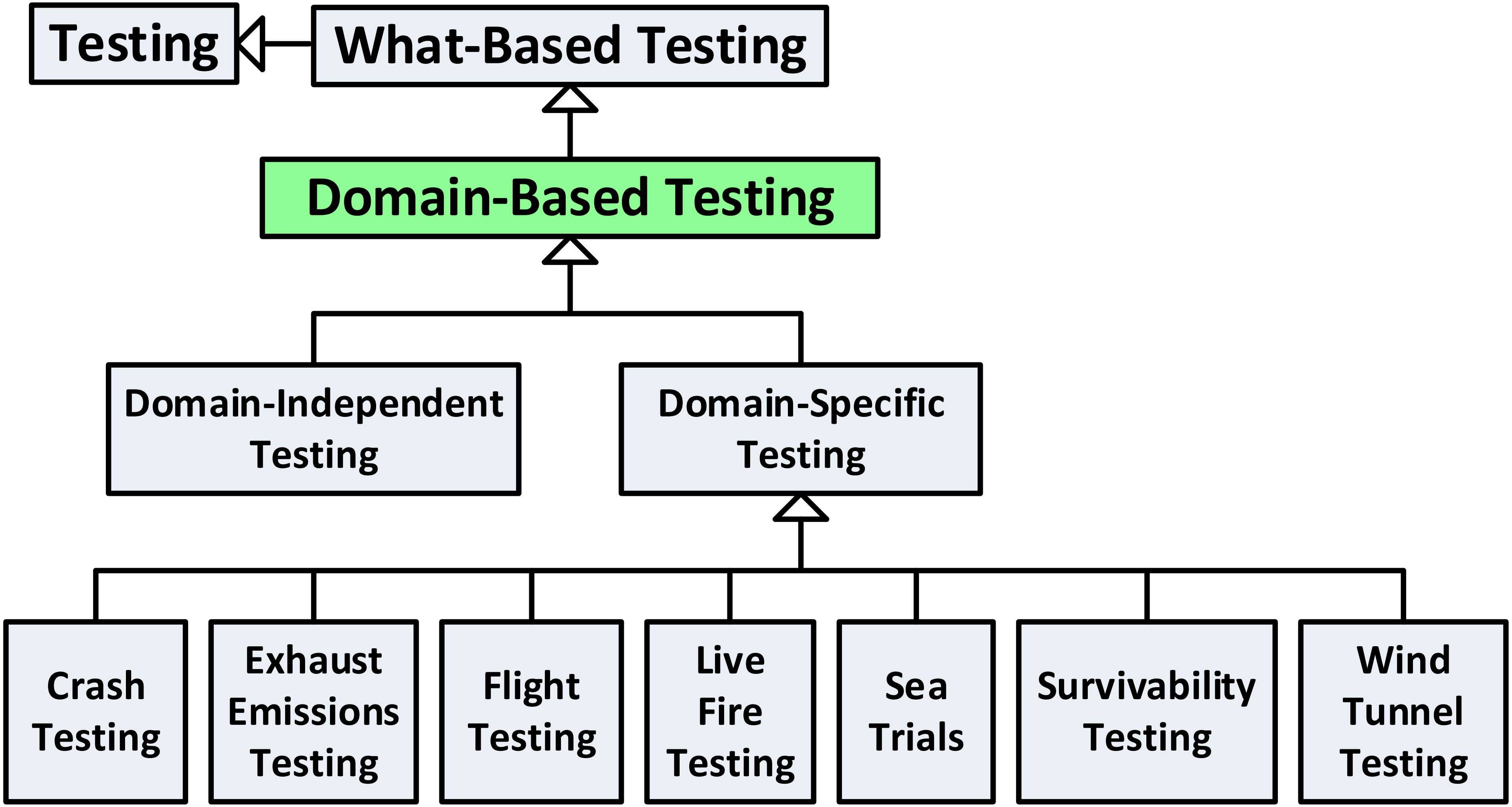

Testing
What-Based Testing
OUT-Based Testing
Hardware Testing
Continuity Testing
Highly Accelerated Life Testing (HALT)
Power-off Testing
Hardware Stress Testing
HW Qualification Testing


Testing
What-Based Testing
OUT-Based Testing
Software Testing
SW Unit [Component] Testing
SW Integration Testing
SW Application Testing


Testing
What-Based Testing
OUT-Based Testing
System Testing
Subsystem Testing
System Integration Testing
System Testing
SoS Integration Testing
SoS Testing
Hardware-in-the-Loop (HIL) Testing
Human-in-the-Loop (HIL) Testing
Processor-in-the-Loop (PIL) Testing
Software-in-the-Loop (SIL) Testing


Testing
What-Based Testing
OUT-Based Testing
Data Center Testing
Configuration Testing
Failover and Restore Testing
Integrated System Testing (IST)
Network Traffic Testing


Testing
What-Based Testing
OUT-Based Testing
Tool/Environment Testing
Development Environment Testing
Development Tool Testing
Test Environment Testing
Test Tool Testing



Testing

What-Based Testing

Domain-Based Testing

Domain-Independent Testing

Domain-Specific Testing

Crash Testing

Exhaust Emissions Testing

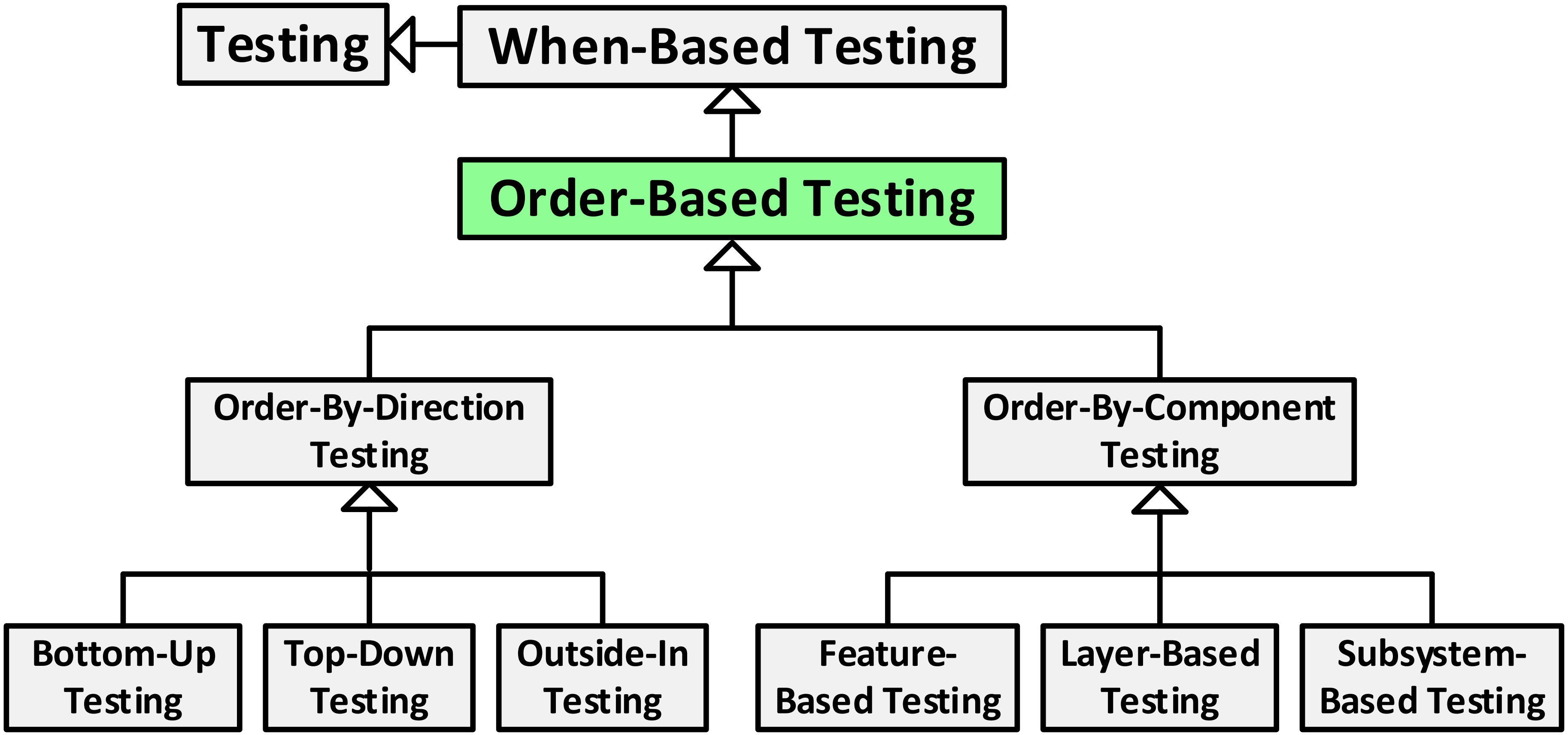
Flight Testing

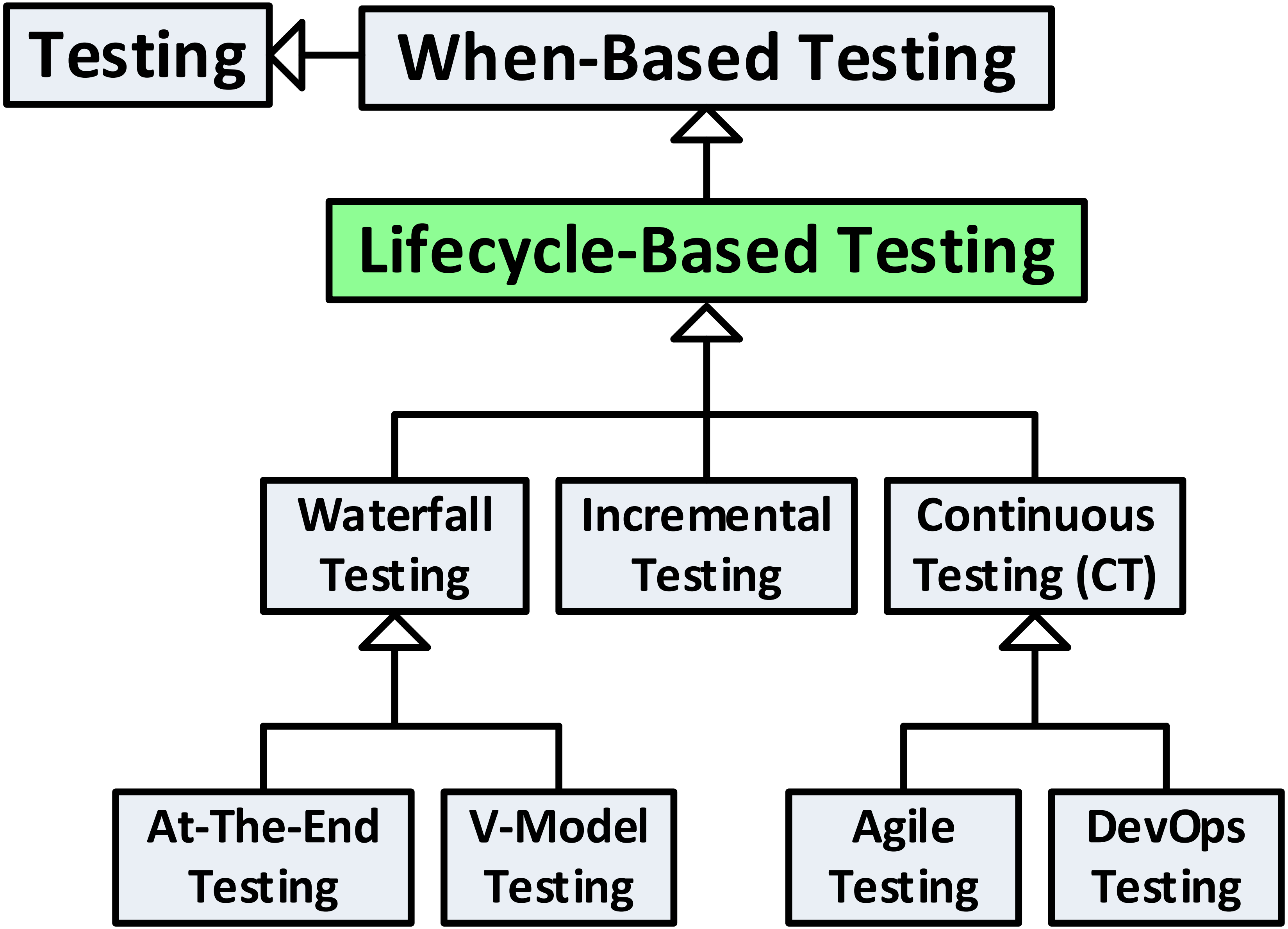
Live Fire Testing

Sea Trials

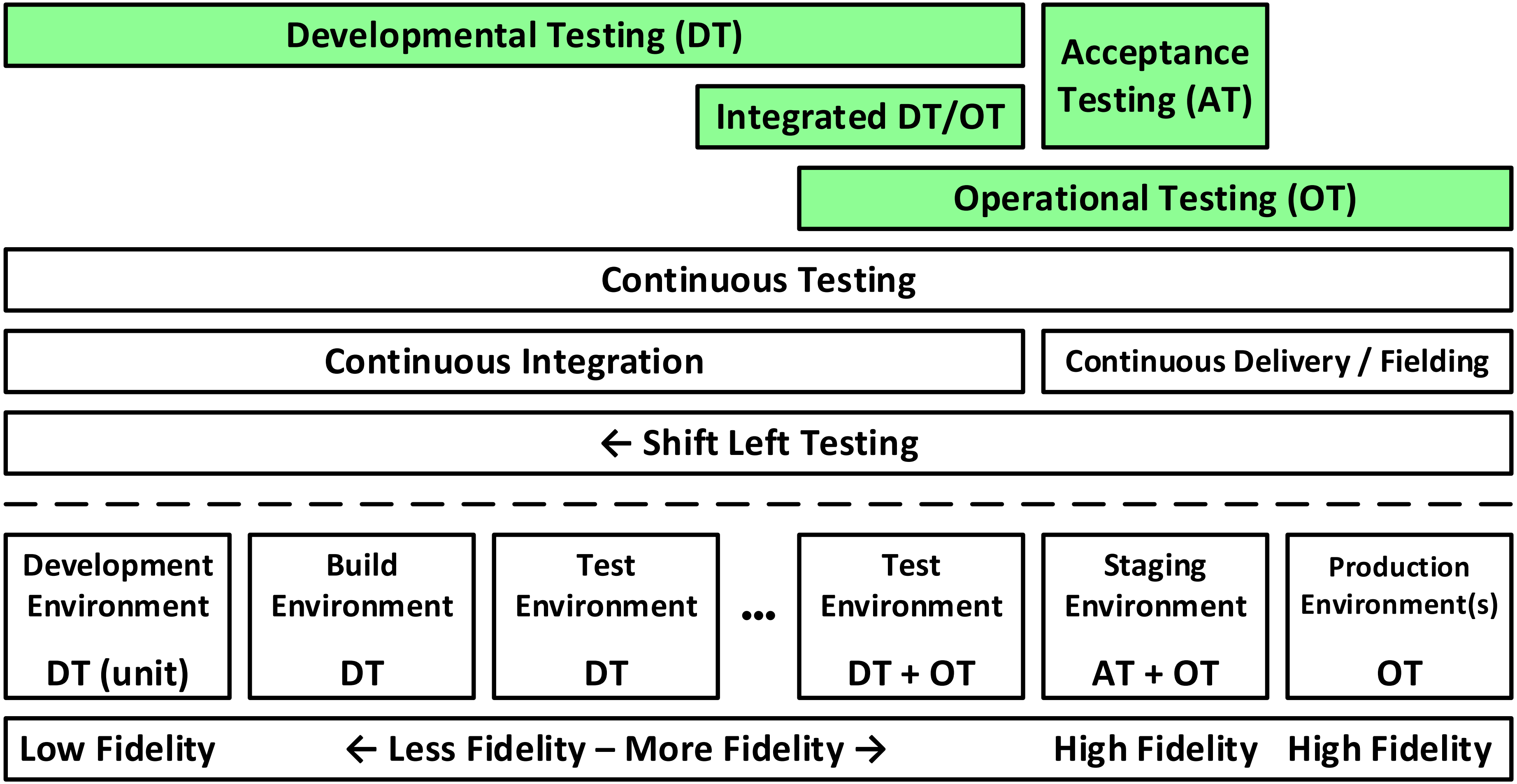
Survivability Testing

Wind Tunnel Testing


Testing
When-Based Testing
Order-Based Testing
Order-By-Direction Testing
Order-By-Component Testing
Bottom-Up Testing
Top-Down Testing
Outside-In Testing
Feature-Based Testing
Layer-Based Testing
Subsystem-Based Testing


Testing
When-Based Testing
Lifecycle-Based Testing
Waterfall Testing
Incremental Testing
Continuous Testing (CT)
At-The-End Testing
V-Model Testing
Agile Testing
DevOps Testing

Test Environments and System Lifecycle Phases



Developmental Testing (DT)

Integrated DT/OT

Acceptance Testing (AT)

Operational Testing (OT)

Continuous Testing

Continuous Integration

Continuous Delivery / Fielding

← Shift Left Testing

Development Environment

DT (unit)

Build Environment

DT

Test Environment

DT

…

Test Environment

DT + OT

Staging Environment

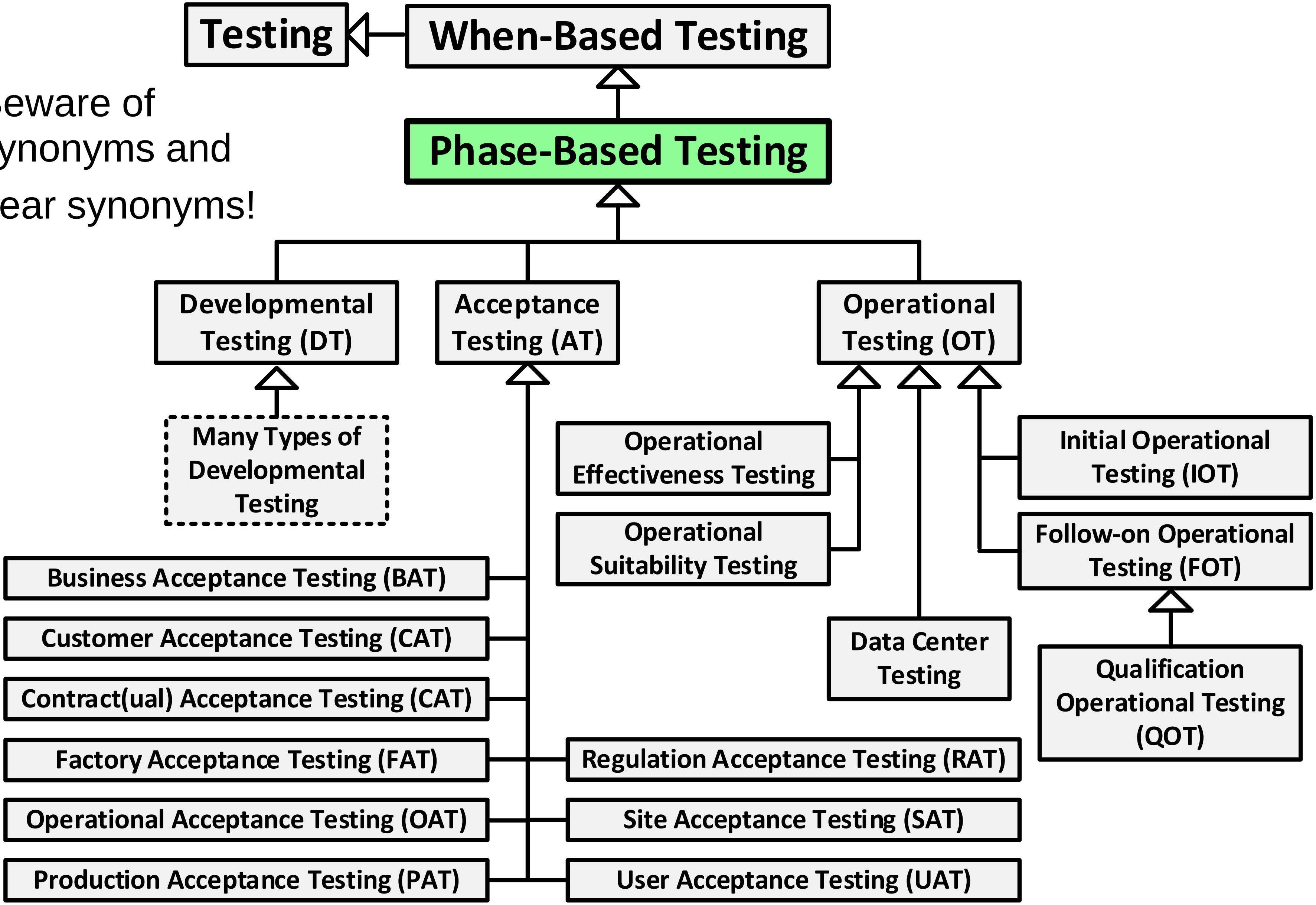
AT + OT

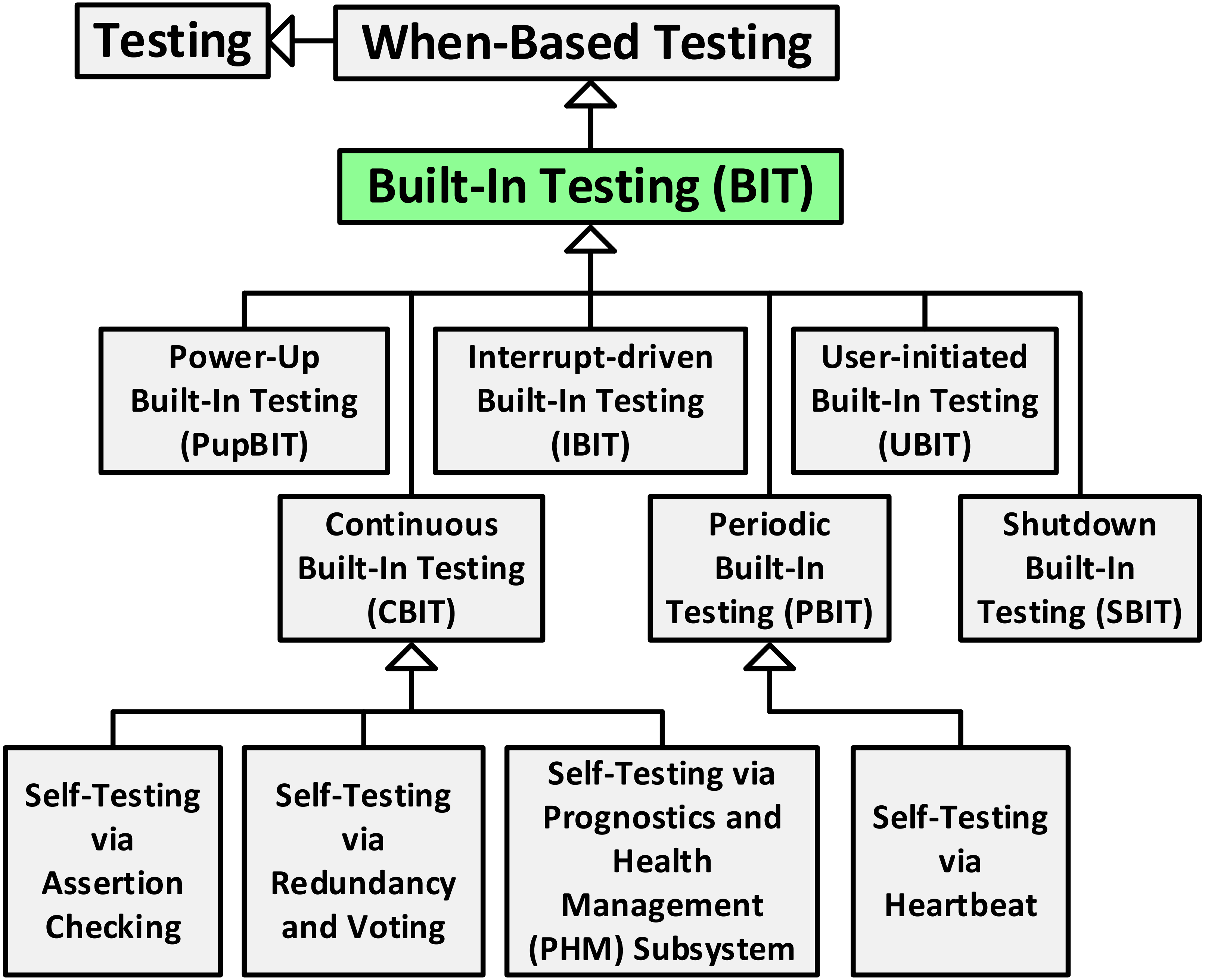
Production Environment(s)

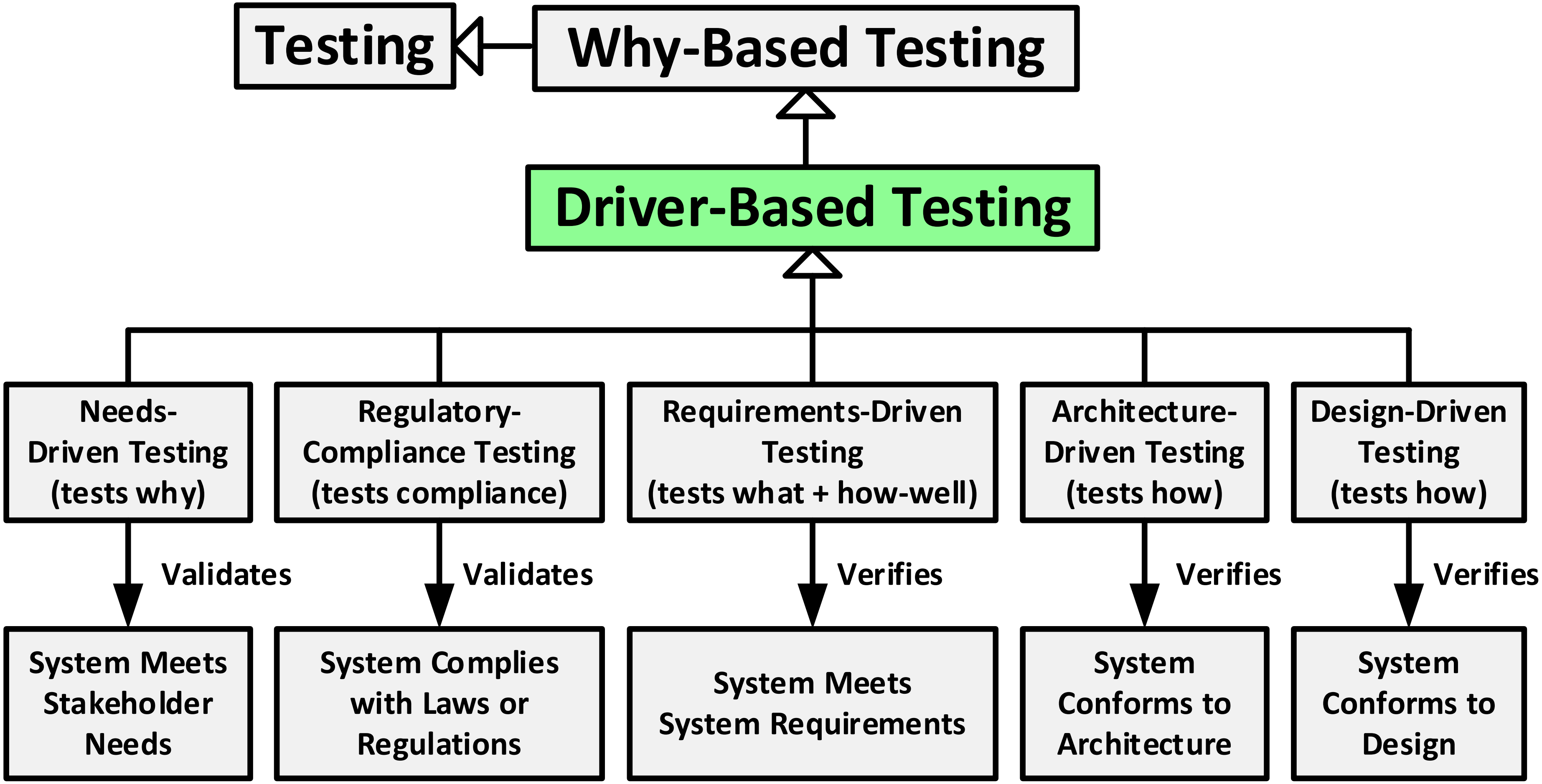
OT

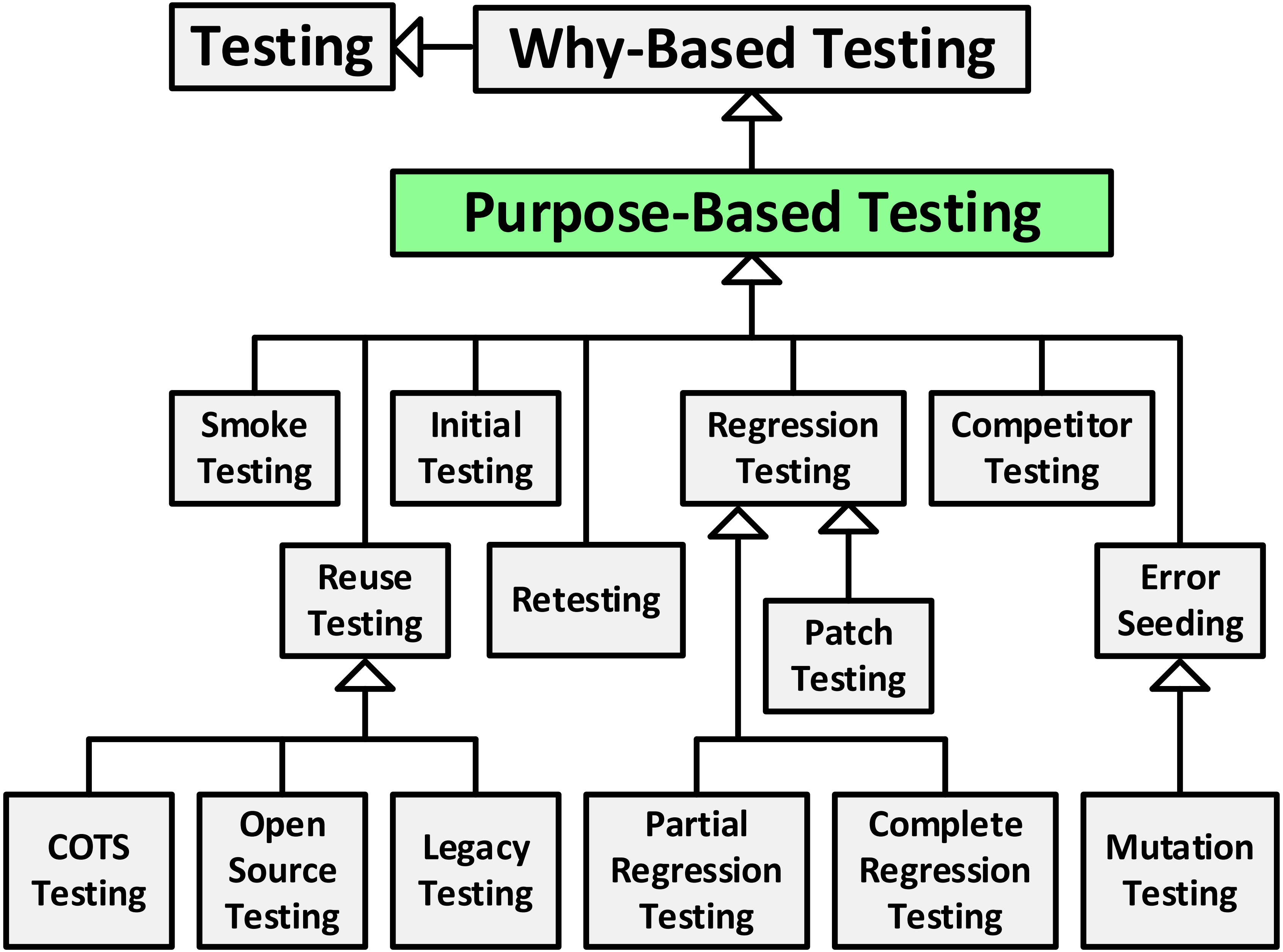
Low Fidelity ← Less Fidelity – More Fidelity → High Fidelity High Fidelity

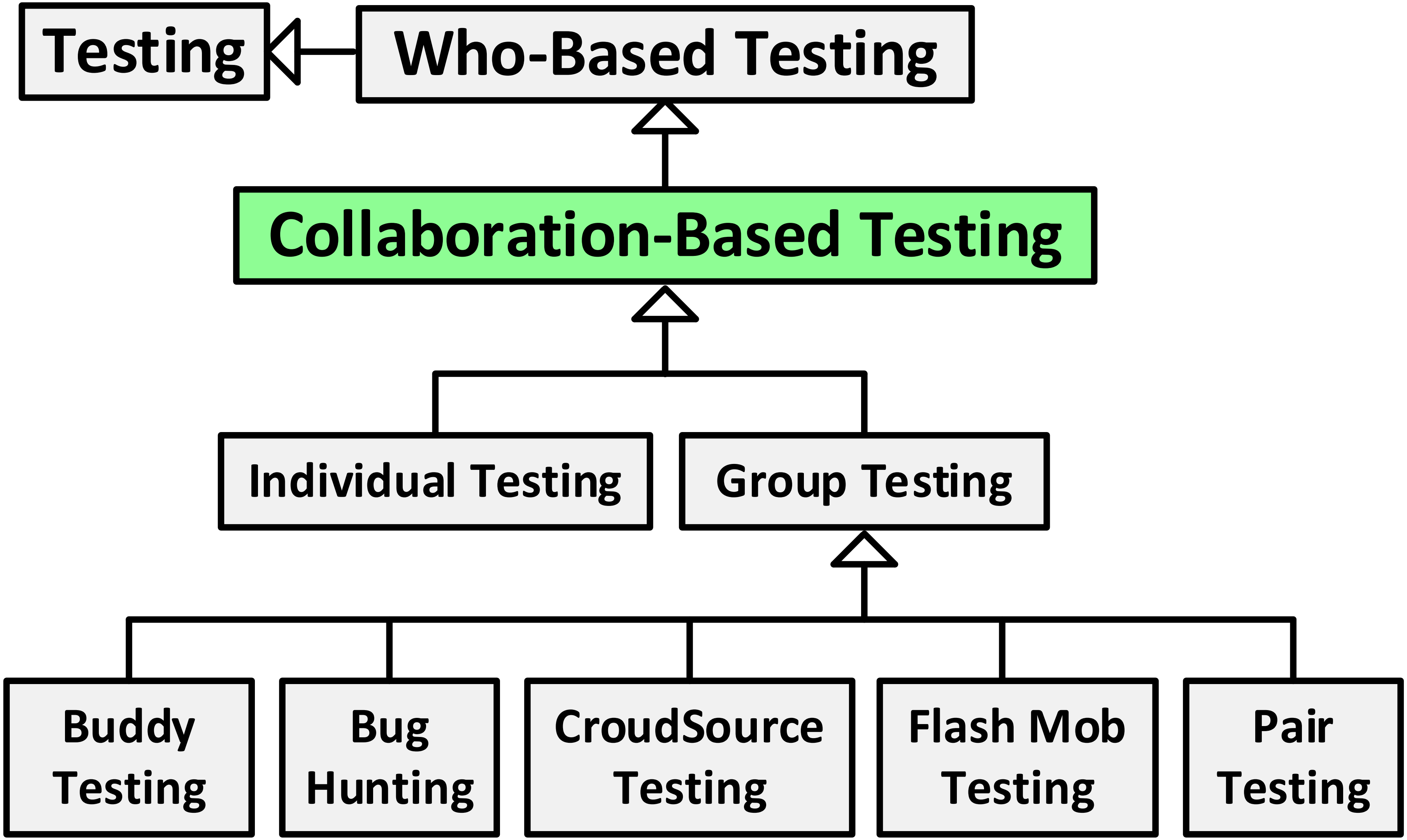
Beware of synonyms and near synonyms!

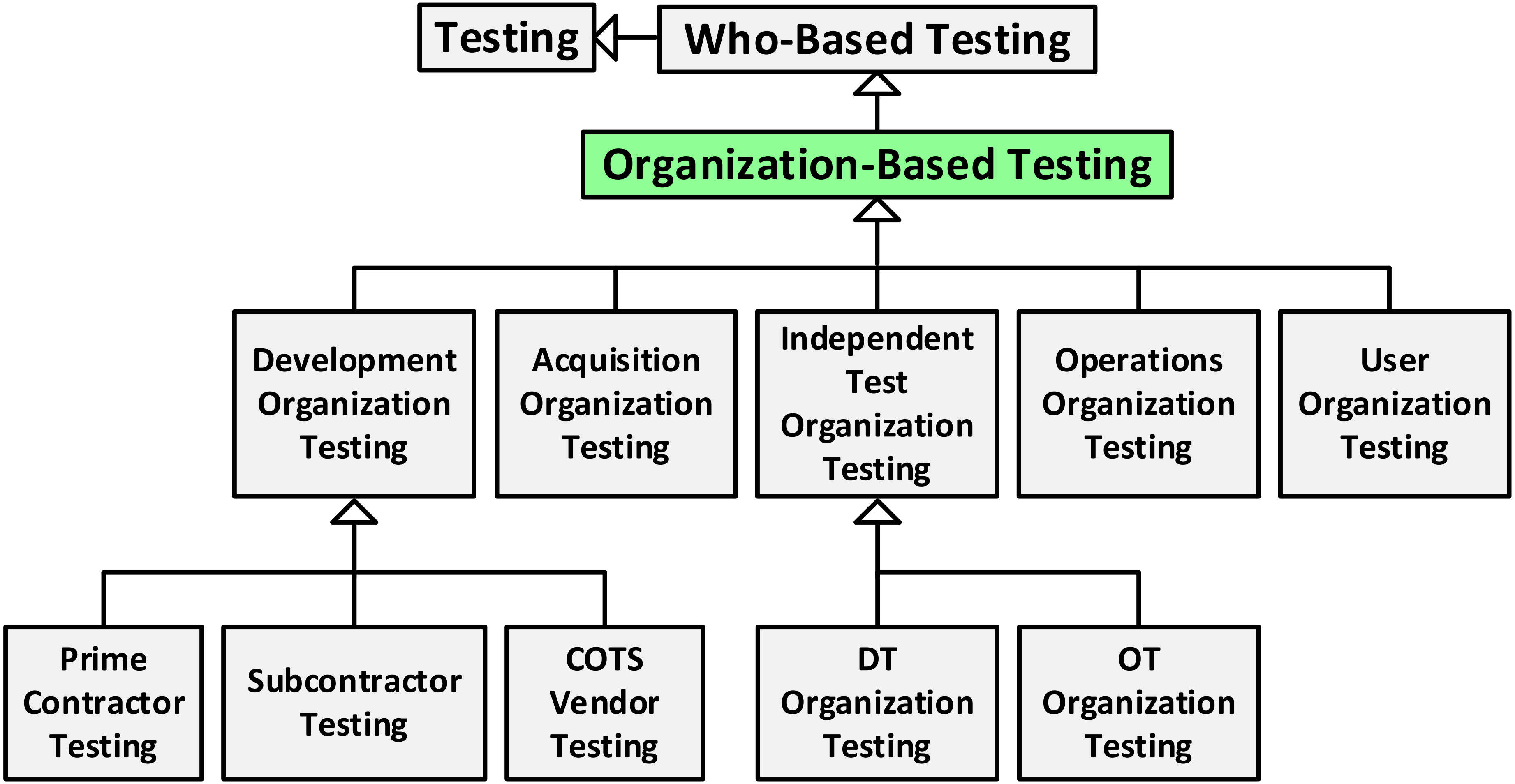

Testing
When-Based Testing
Phase-Based Testing
Developmental Testing (DT)
Acceptance Testing (AT)
Operational Testing (OT)
Many Types of Developmental Testing
Operational Effectiveness Testing
Operational Suitability Testing
Initial Operational Testing (IOT)
Follow-on Operational Testing (FOT)
Data Center Testing
Qualification Operational Testing (QOT)
Business Acceptance Testing (BAT)
Customer Acceptance Testing (CAT)
Contract(ual) Acceptance Testing (CAT)
Factory Acceptance Testing (FAT)
Operational Acceptance Testing (OAT)
Production Acceptance Testing (PAT)
Regulation Acceptance Testing (RAT)
Site Acceptance Testing (SAT)
User Acceptance Testing (UAT)

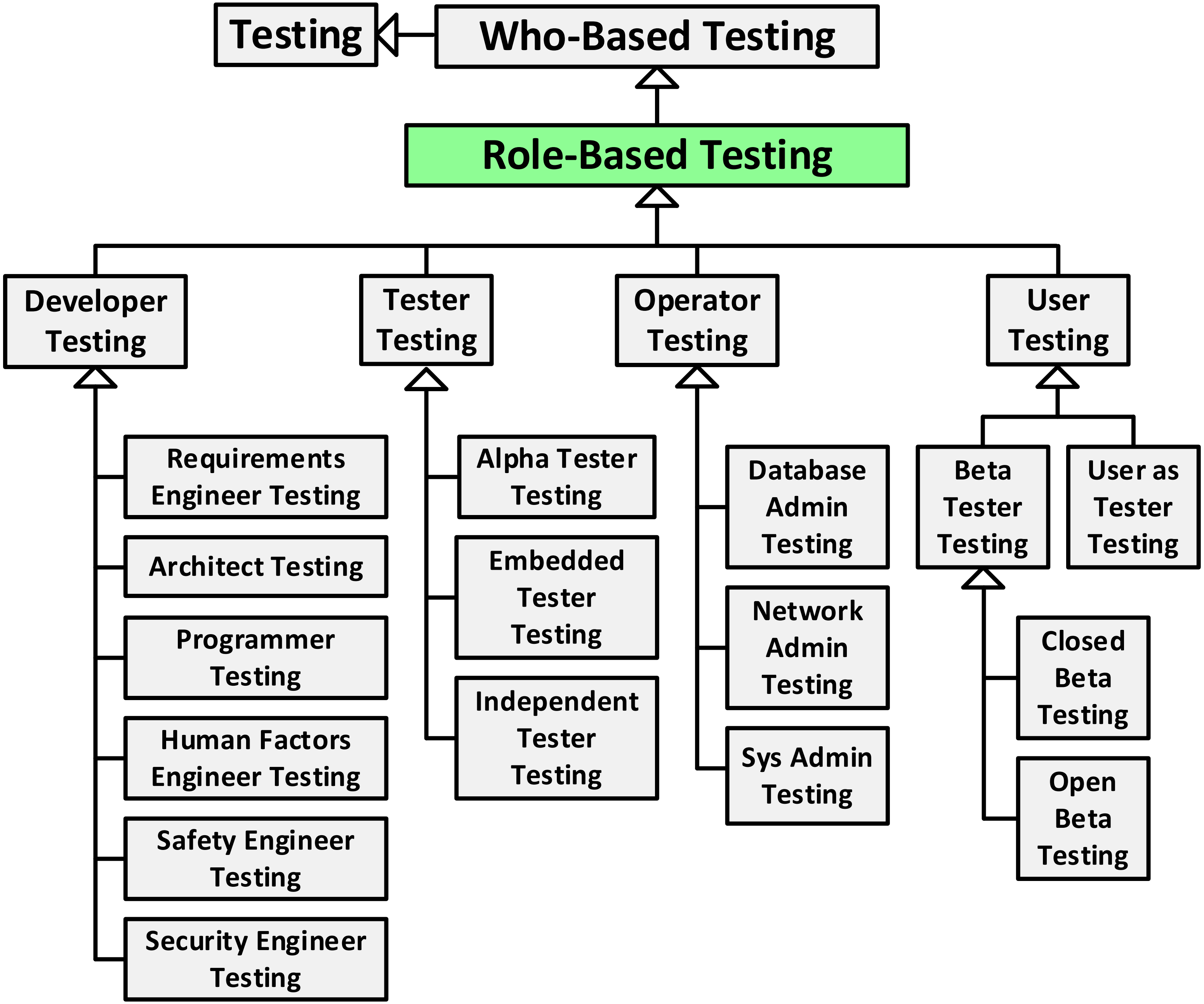

Testing
When-Based Testing
Built-In Testing (BIT)
Power-Up Built-In Testing (PupBIT)
Interrupt-driven Built-In Testing (IBIT)
User-initiated Built-In Testing (UBIT)
Continuous Built-In Testing (CBIT)
Periodic Built-In Testing (PBIT)
Shutdown Built-In Testing (SBIT)
Self-Testing via Assertion Checking
Self-Testing via Redundancy and Voting
Self-Testing via Prognostics and Health Management (PHM) Subsystem
Self-Testing via Heartbeat

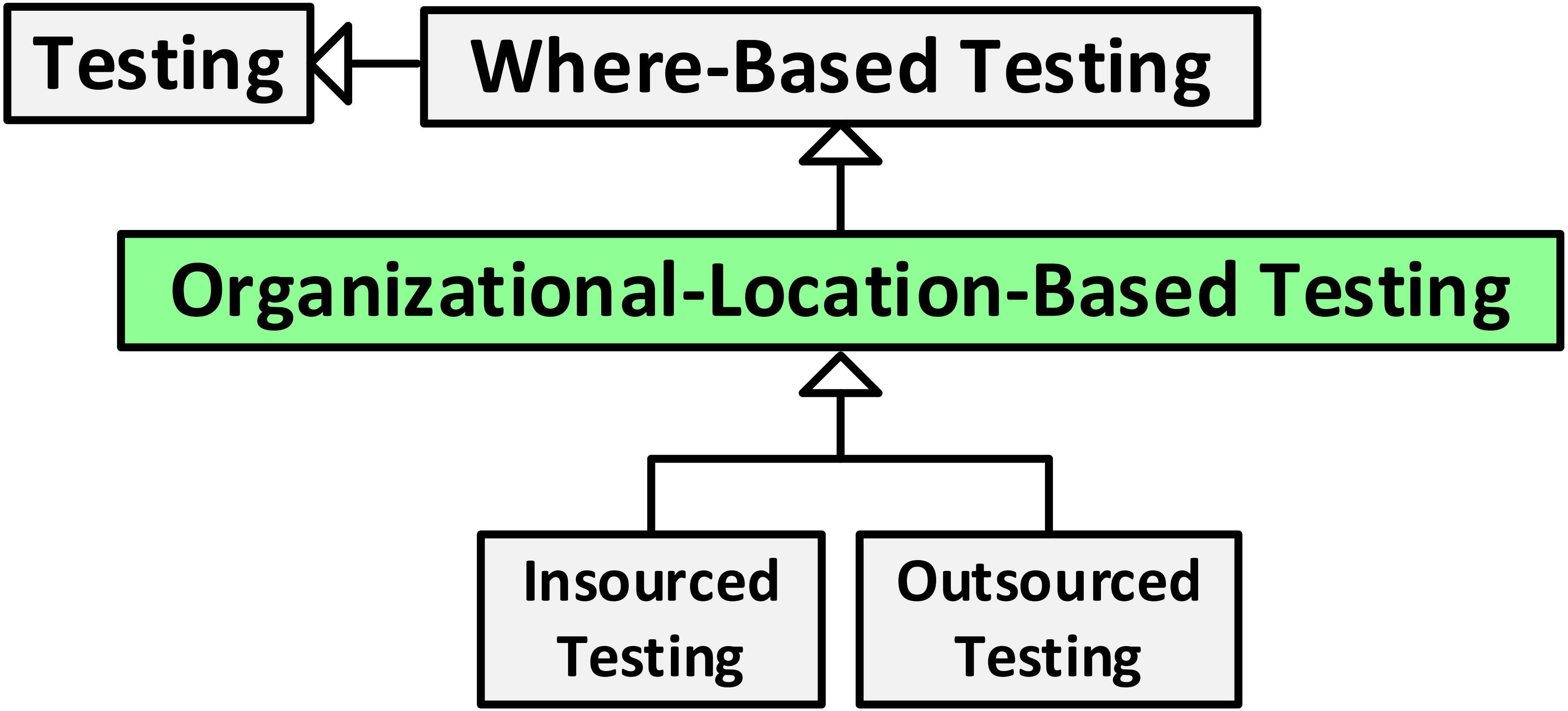

Testing
Why-Based Testing
Driver-Based Testing
Needs-Driven Testing (tests why)
Regulatory-Compliance Testing (tests compliance)
Requirements-Driven Testing (tests what + how-well)
Architecture-Driven Testing (tests how)
Design-Driven Testing (tests how)
Validates
Validates
Verifies
Verifies
Verifies
System Meets Stakeholder Needs
System Complies with Laws or Regulations
System Meets System Requirements
System Conforms to Architecture
System Conforms to Design

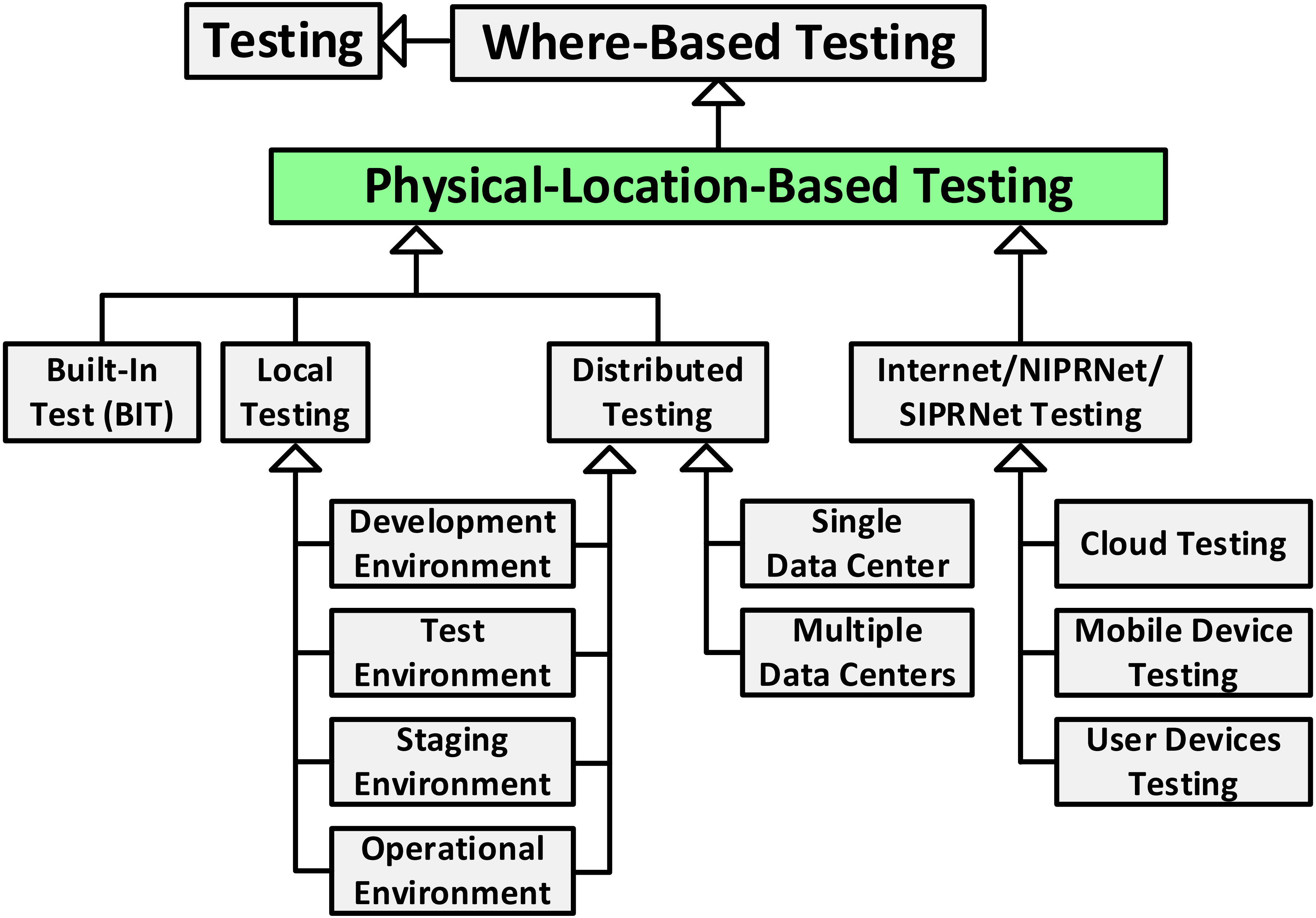

Testing
Why-Based Testing
Purpose-Based Testing
Smoke Testing
Initial Testing
Regression Testing
Competitor Testing
Reuse Testing
Retesting
Patch Testing
Error Seeding
COTS Testing
Open Source Testing
Legacy Testing
Partial Regression Testing
Complete Regression Testing
Mutation Testing

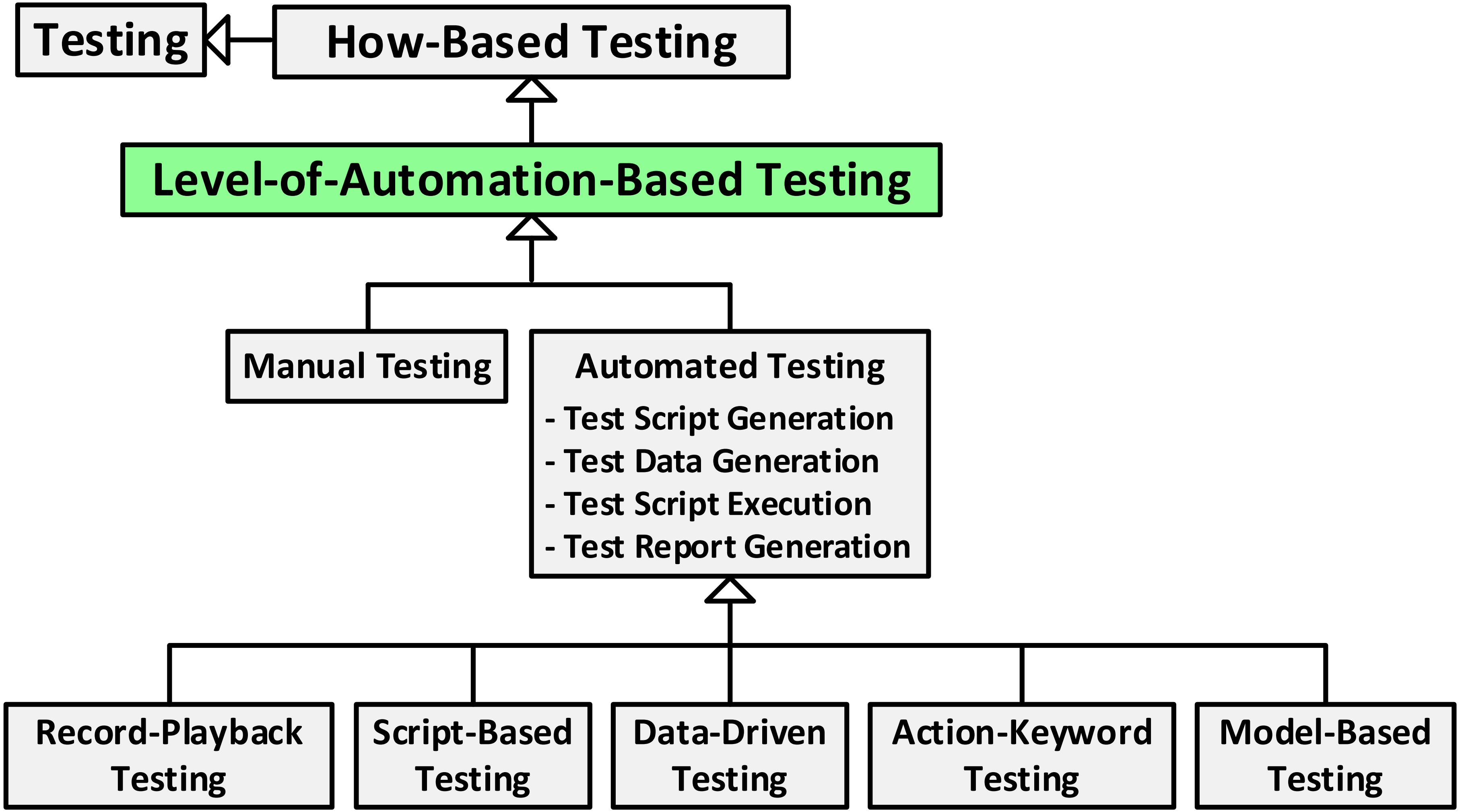

Testing
Who-Based Testing
Collaboration-Based Testing
Individual Testing
Group Testing
Buddy Testing
Bug Hunting
CroudSource Testing
Flash Mob Testing
Pair Testing


Testing
Who-Based Testing
Organization-Based Testing
Development Organization Testing
Acquisition Organization Testing
Independent Test Organization Testing
Operations Organization Testing
User Organization Testing
Prime Contractor Testing
Subcontractor Testing
COTS Vendor Testing
DT Organization Testing
OT Organization Testing


Testing
Who-Based Testing
Role-Based Testing
Developer Testing
Tester Testing
Operator Testing
User Testing
Requirements Engineer Testing
Architect Testing
Programmer Testing
Human Factors Engineer Testing
Safety Engineer Testing
Security Engineer Testing
Alpha Tester Testing
Embedded Tester Testing
Independent Tester Testing
Database Admin Testing
Network Admin Testing
Sys Admin Testing
Beta Tester Testing
User as Tester Testing
Closed Beta Testing
Open Beta Testing


Testing
Where-Based Testing
Organizational-Location-Based Testing
Insourced Testing
Outsourced Testing


Testing
Where-Based Testing
Physical-Location-Based Testing
Built-In Test (BIT)
Local Testing
Distributed Testing
Internet/NIPRNet/ SIPRNet Testing
Development Environment
Test Environment
Staging Environment
Operational Environment
Single Data Center
Multiple Data Centers
Cloud Testing
Mobile Device Testing
User Devices Testing



Testing

How-Based Testing

Level-of-Automation-Based Testing

Manual Testing

Automated Testing

- Test Script Generation
- Test Data Generation
- Test Script Execution
- Test Report Generation

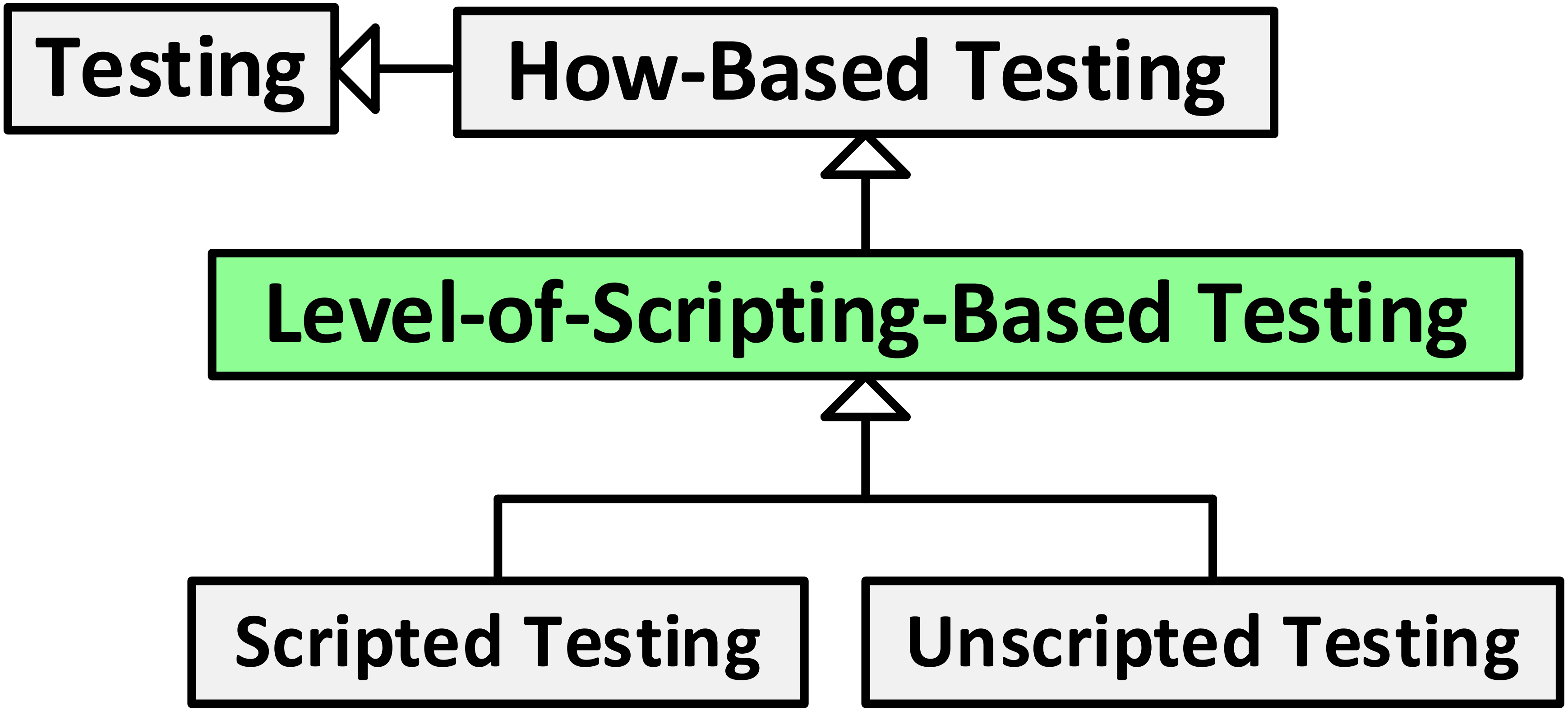
Record-Playback Testing

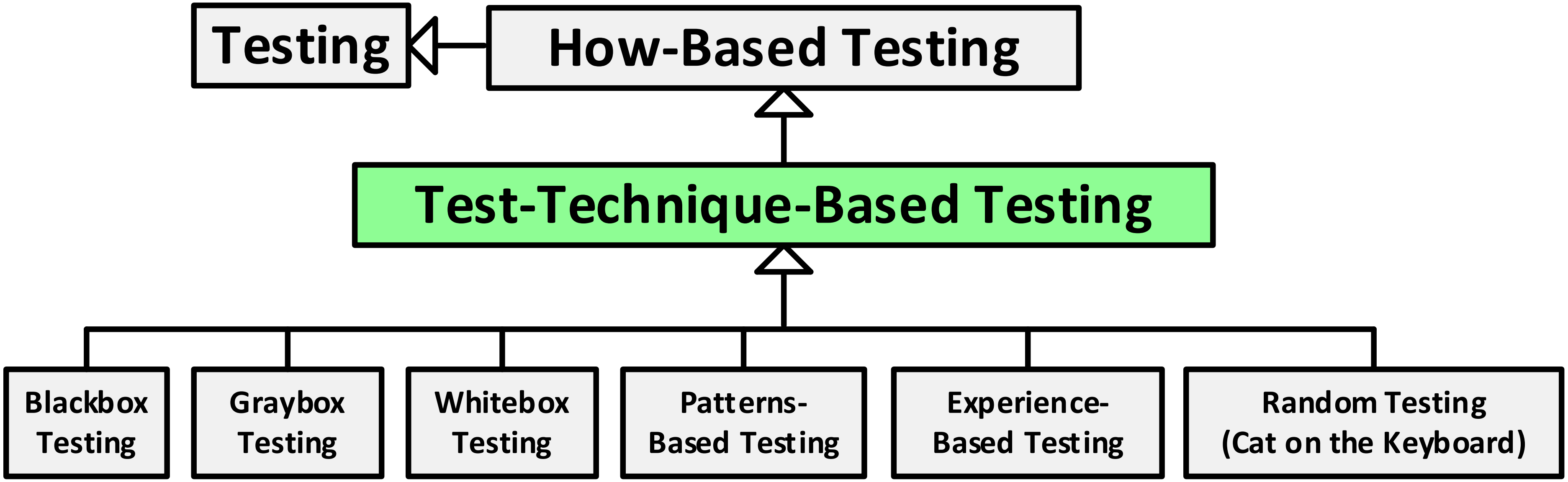
Script-Based Testing

Data-Driven Testing

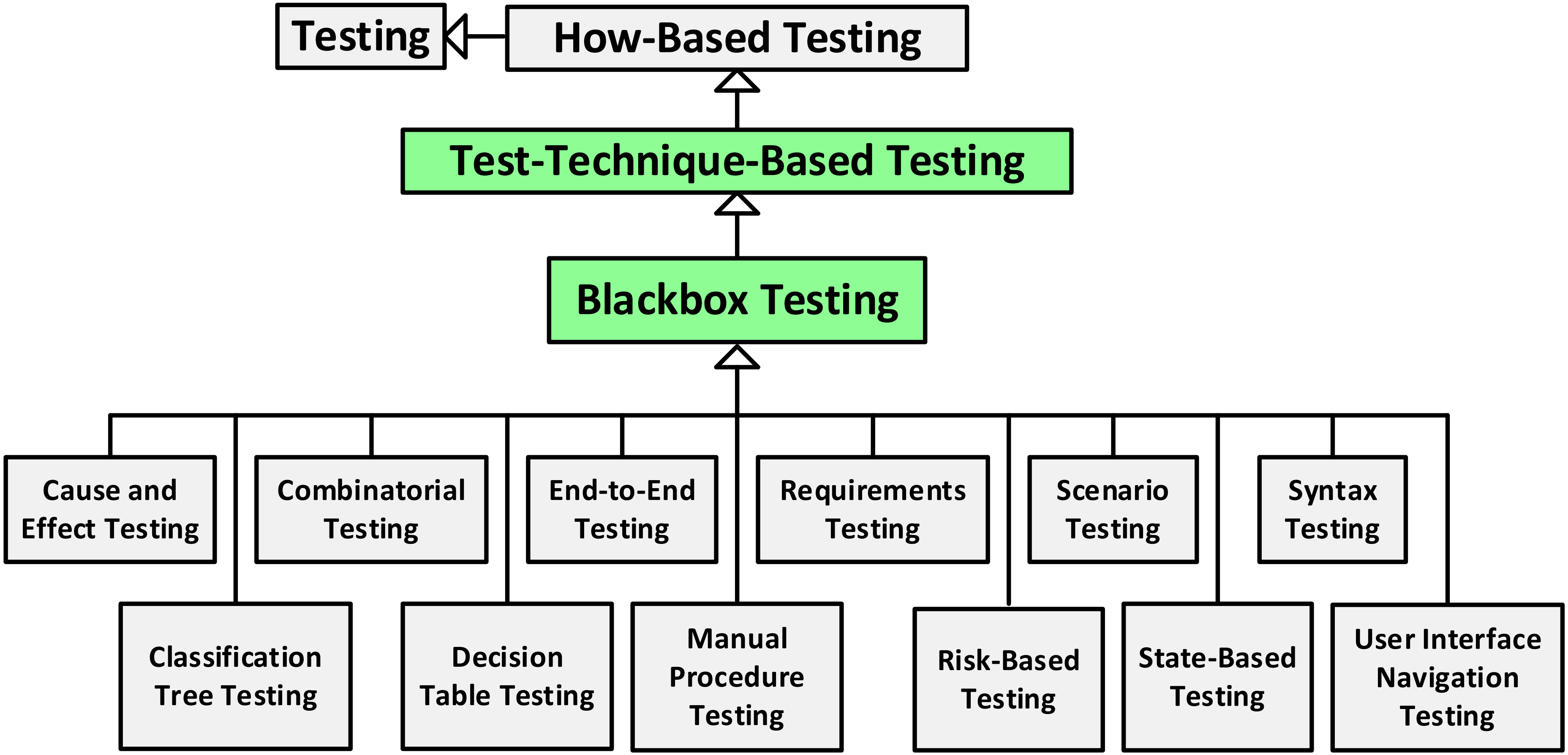
Action-Keyword Testing

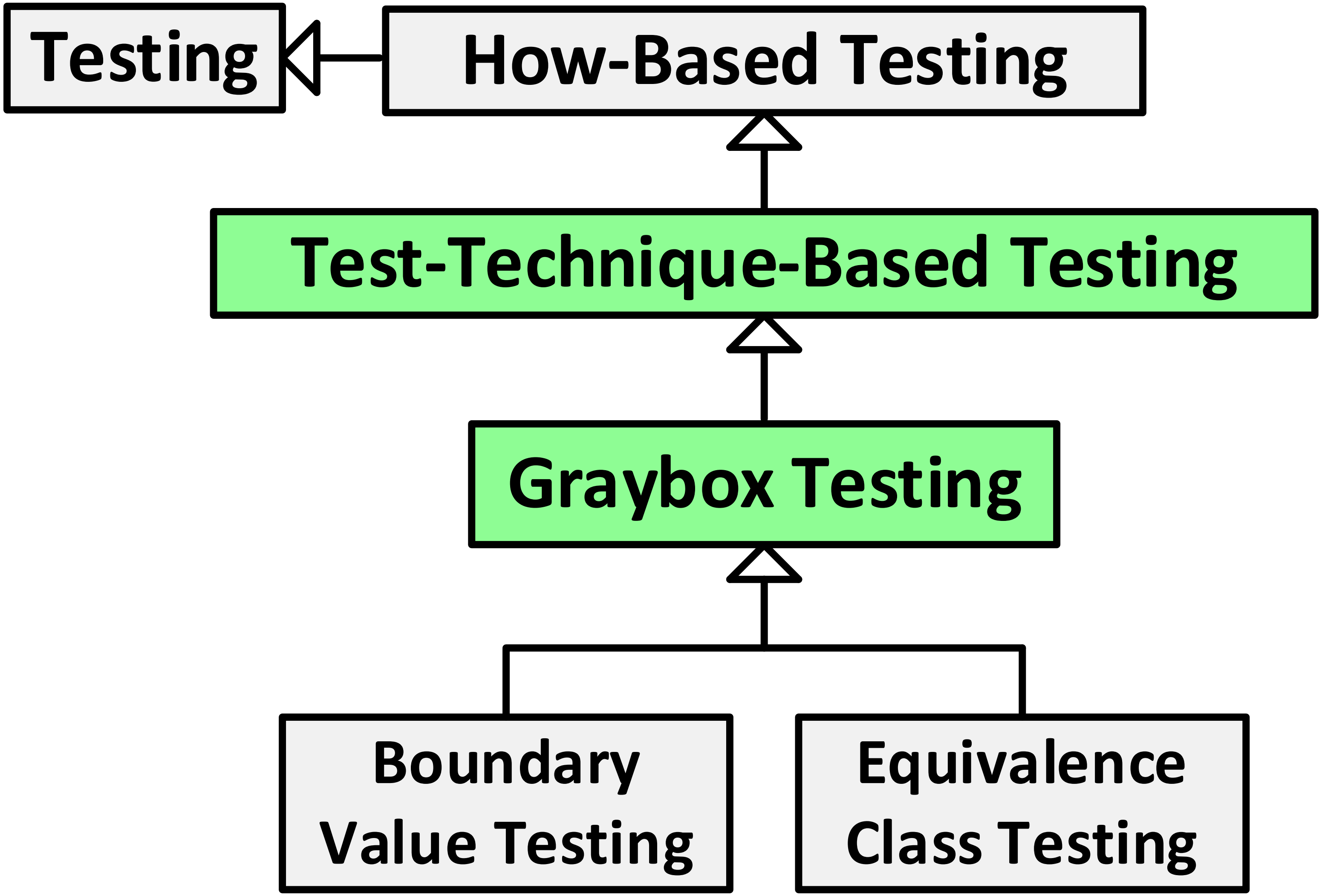
Model-Based Testing

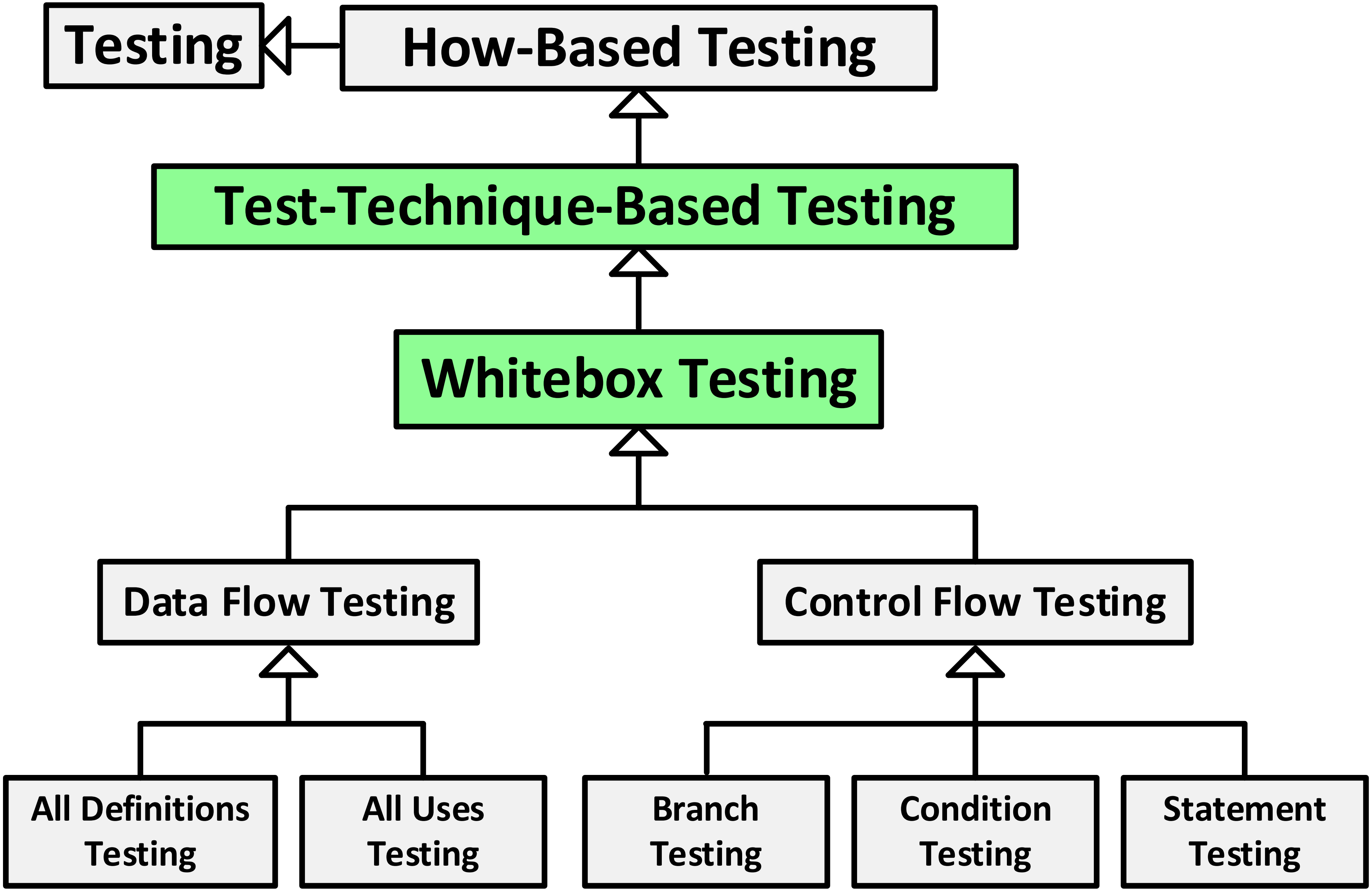

Testing
How-Based Testing
Level-of-Scripting-Based Testing
Scripted Testing
Unscripted Testing

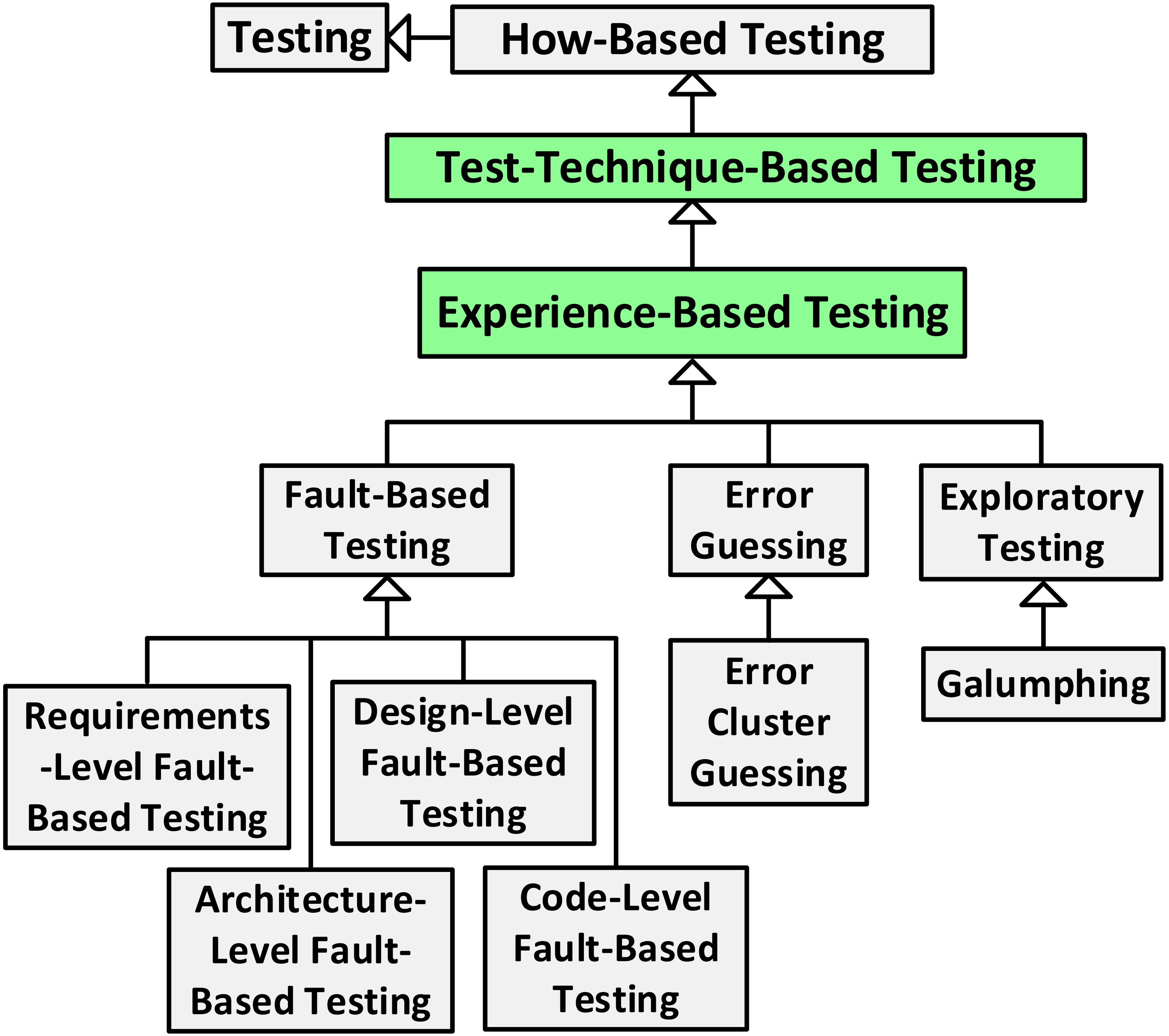

Testing
How-Based Testing
Test-Technique-Based Testing
Blackbox Testing
Graybox Testing
Whitebox Testing
Patterns-Based Testing
Experience-Based Testing
Random Testing (Cat on the Keyboard)

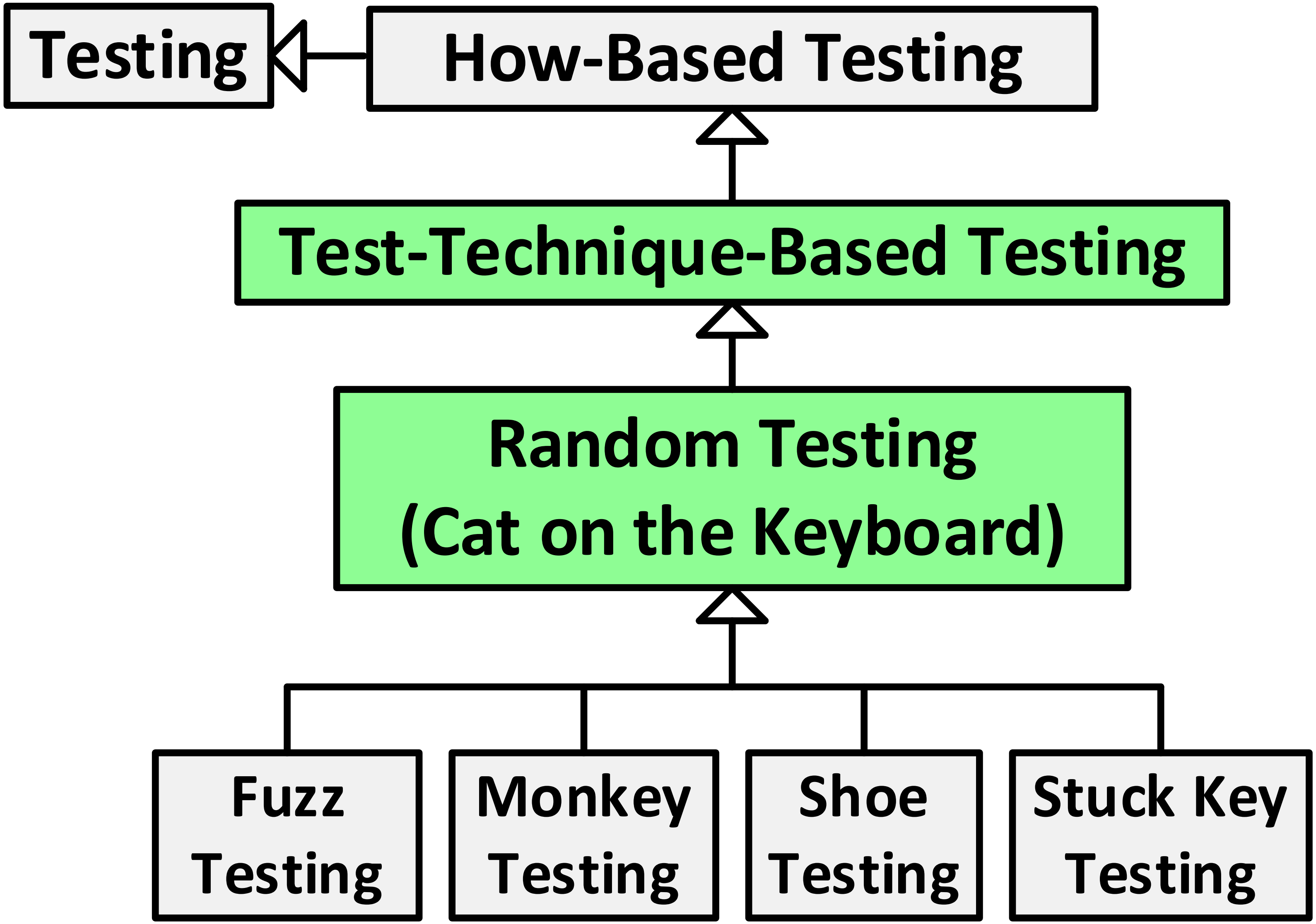
How: by Technique - Blackbox Testing




Testing
How-Based Testing
Test-Technique-Based Testing
Graybox Testing
Boundary Value Testing
Equivalence Class Testing


Testing
How-Based Testing
Test-Technique-Based Testing
Whitebox Testing
Data Flow Testing
Control Flow Testing
All Definitions Testing
All Uses Testing
Branch Testing
Condition Testing
Statement Testing

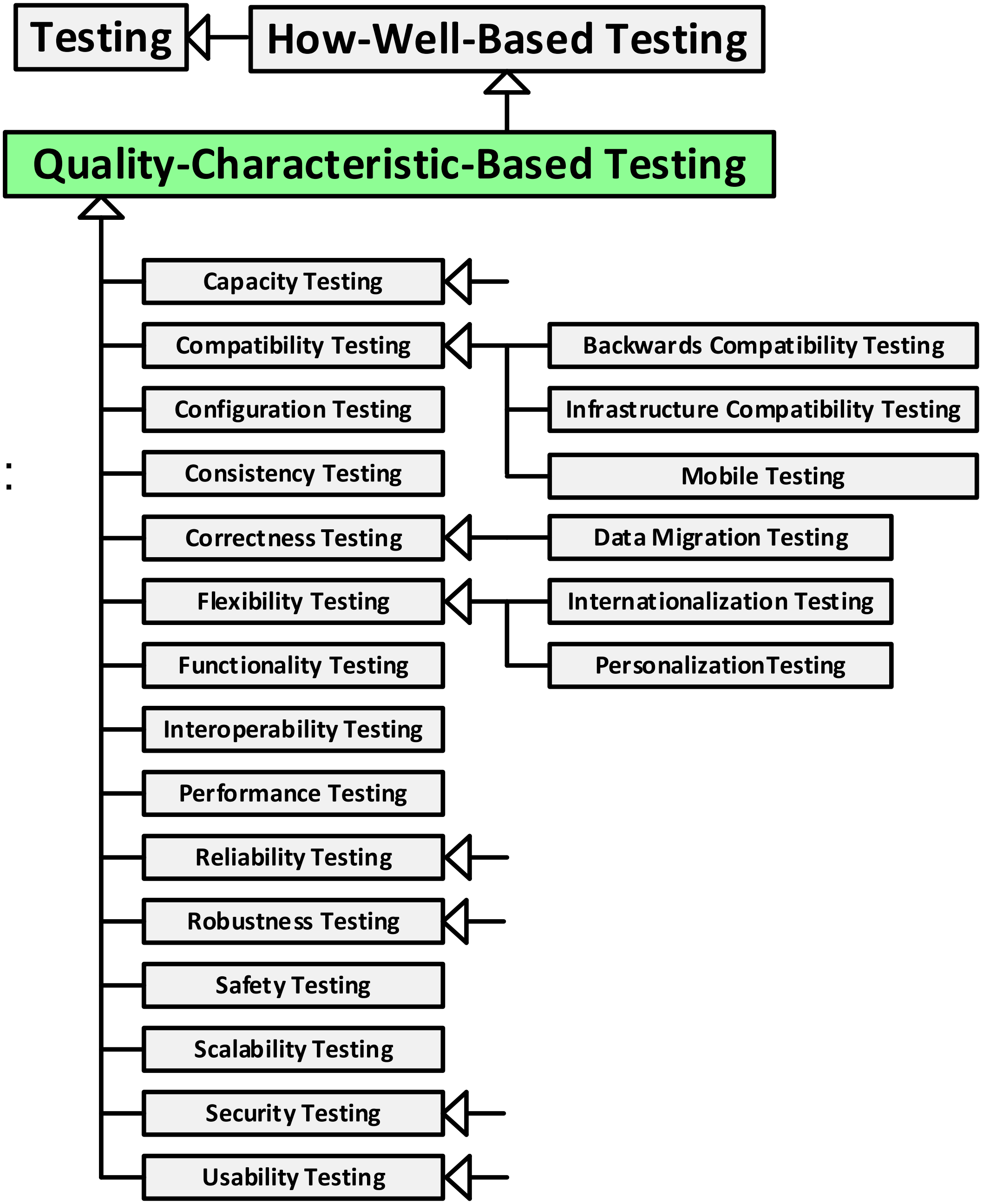

Testing
How-Based Testing
Test-Technique-Based Testing
Experience-Based Testing
Fault-Based Testing
Error Guessing
Exploratory Testing
Requirements-Level Fault-Based Testing
Design-Level Fault-Based Testing
Architecture-Level Fault-Based Testing
Code-Level Fault-Based Testing
Error Cluster Guessing
Galumphing

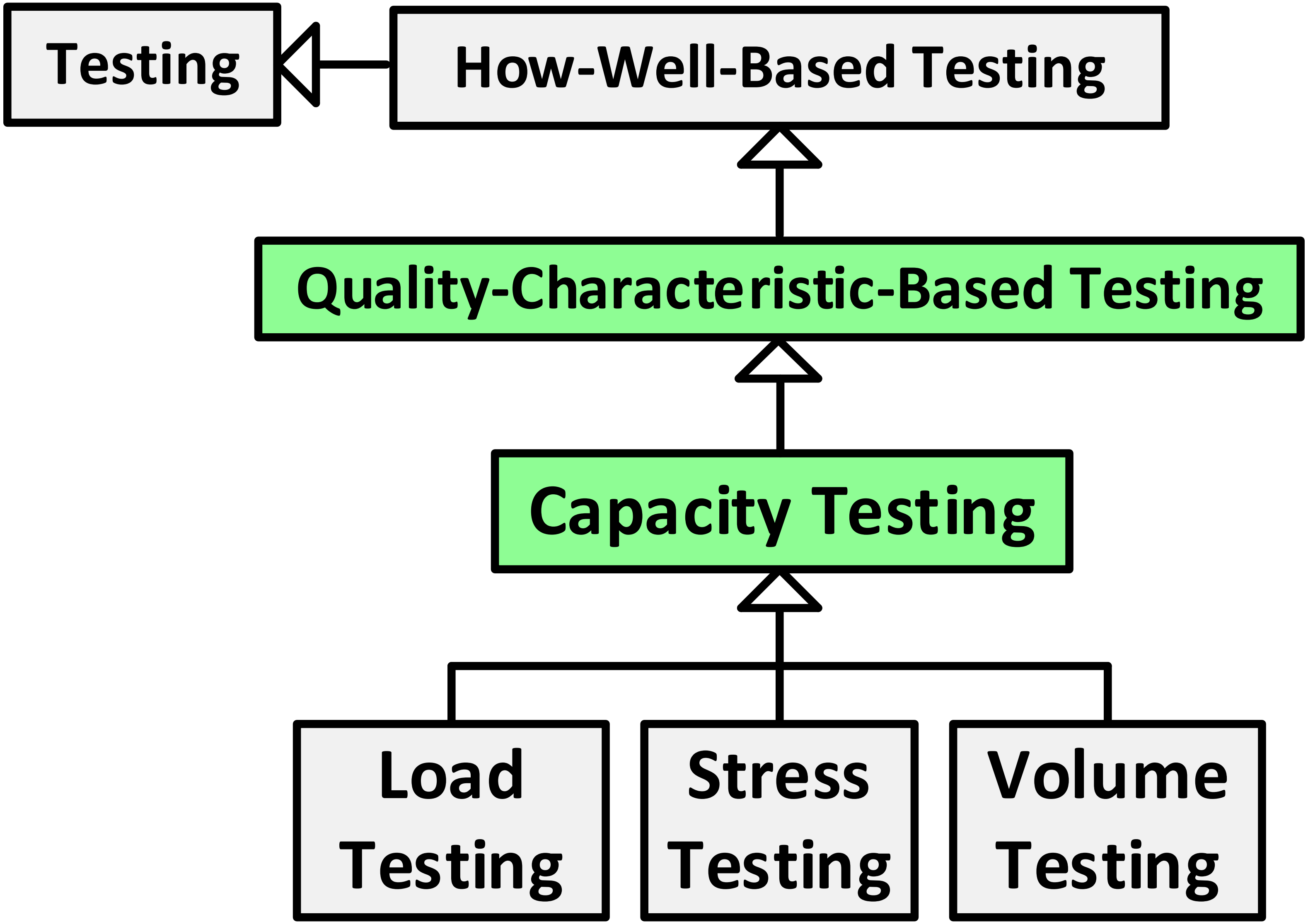

Testing
How-Based Testing
Test-Technique-Based Testing
Random Testing
(Cat on the Keyboard)
Fuzz Testing
Monkey Testing
Shoe Testing
Stuck Key Testing

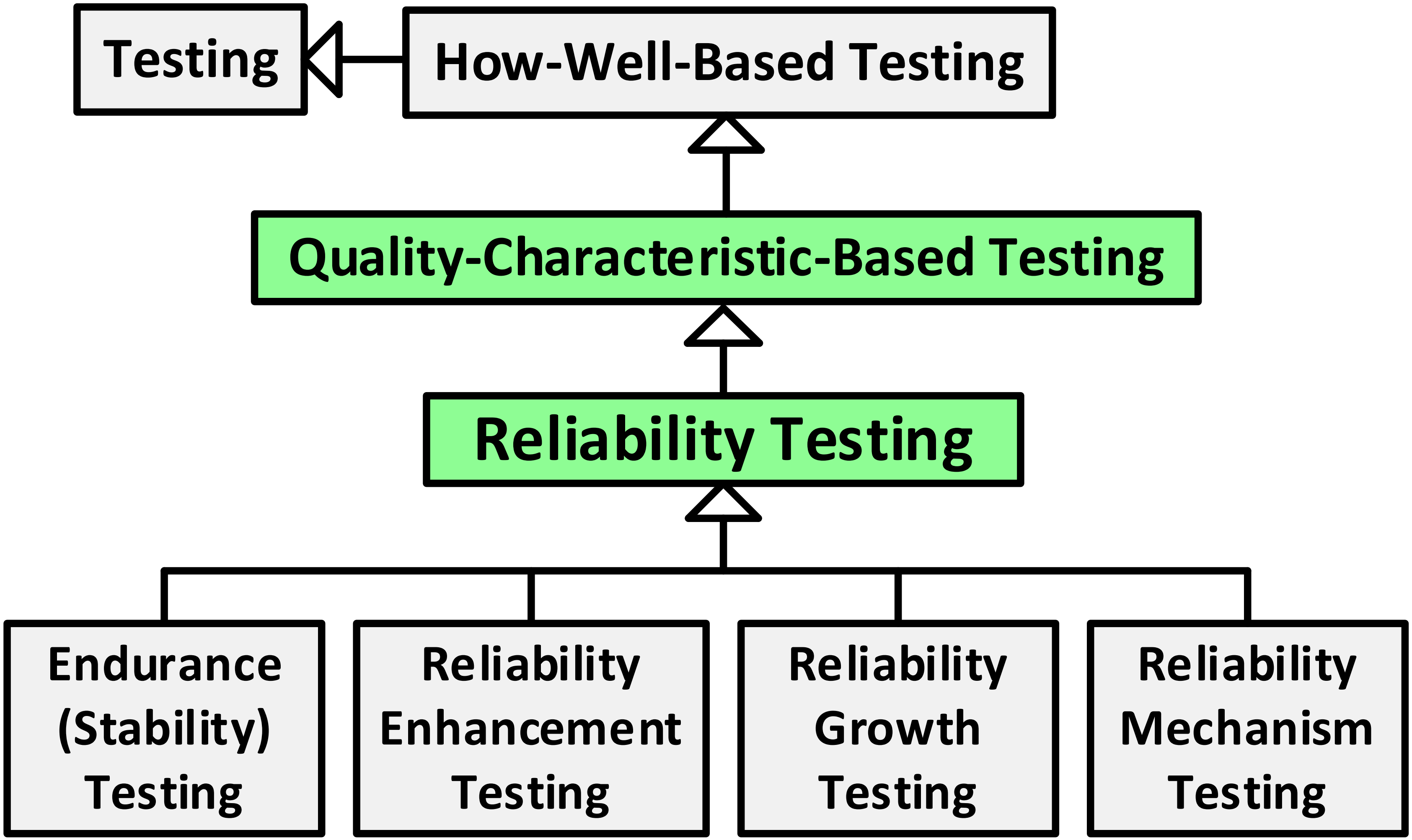
How Well: by Quality Characteristic

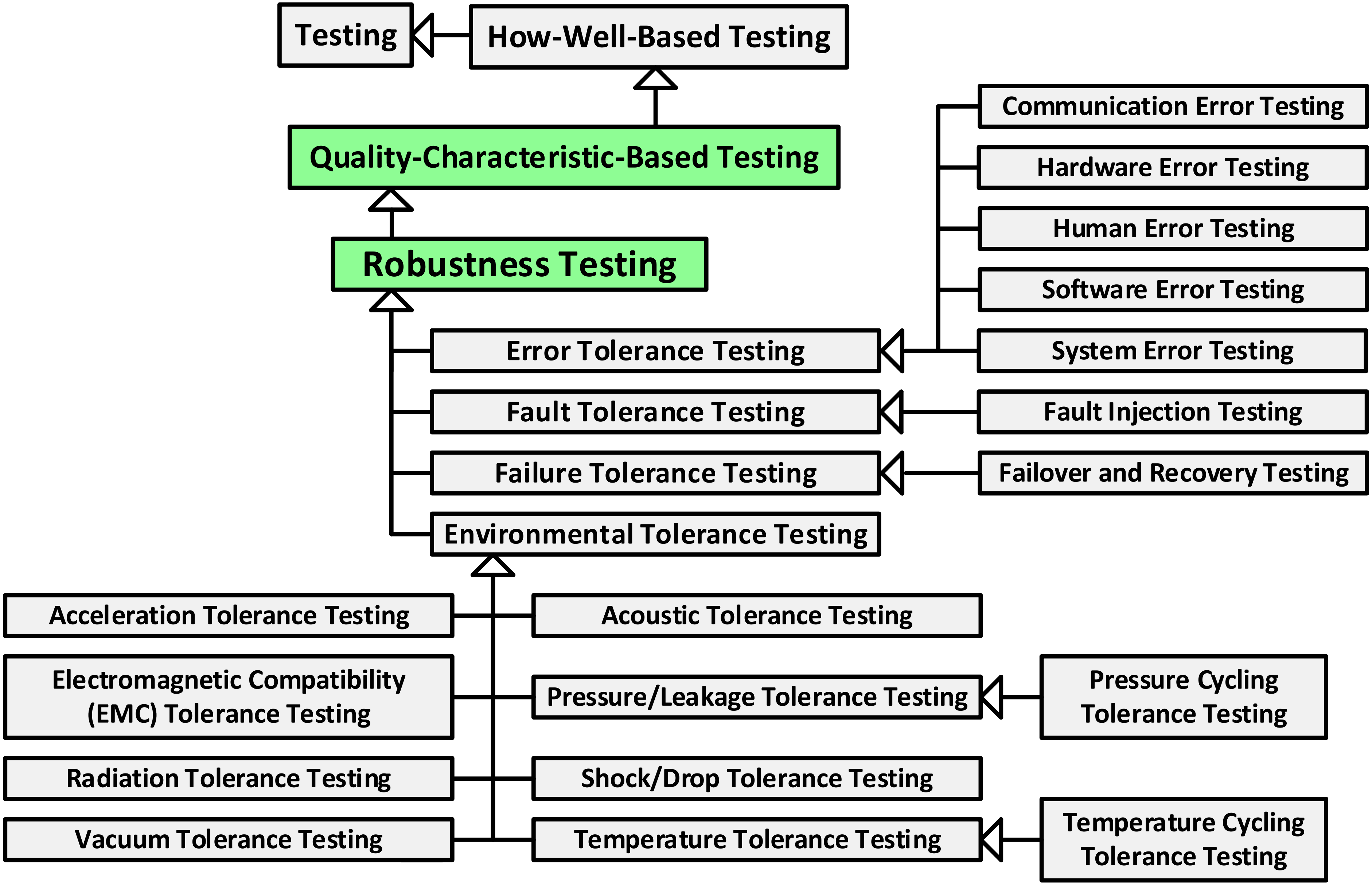
Based on the associated quality characteristic and its associated quality attributes:

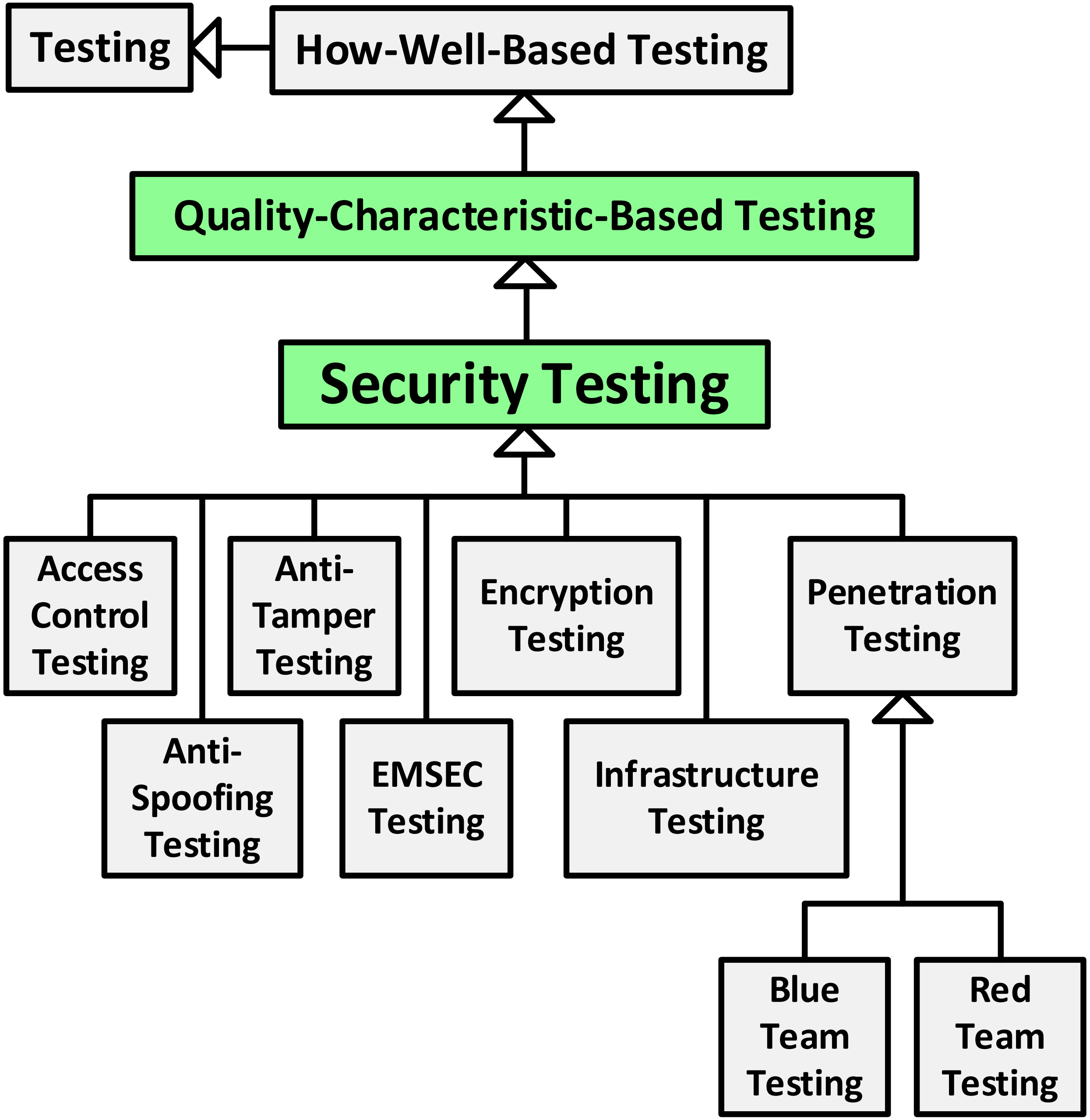
- Uncover related defects
- Determine level of quality

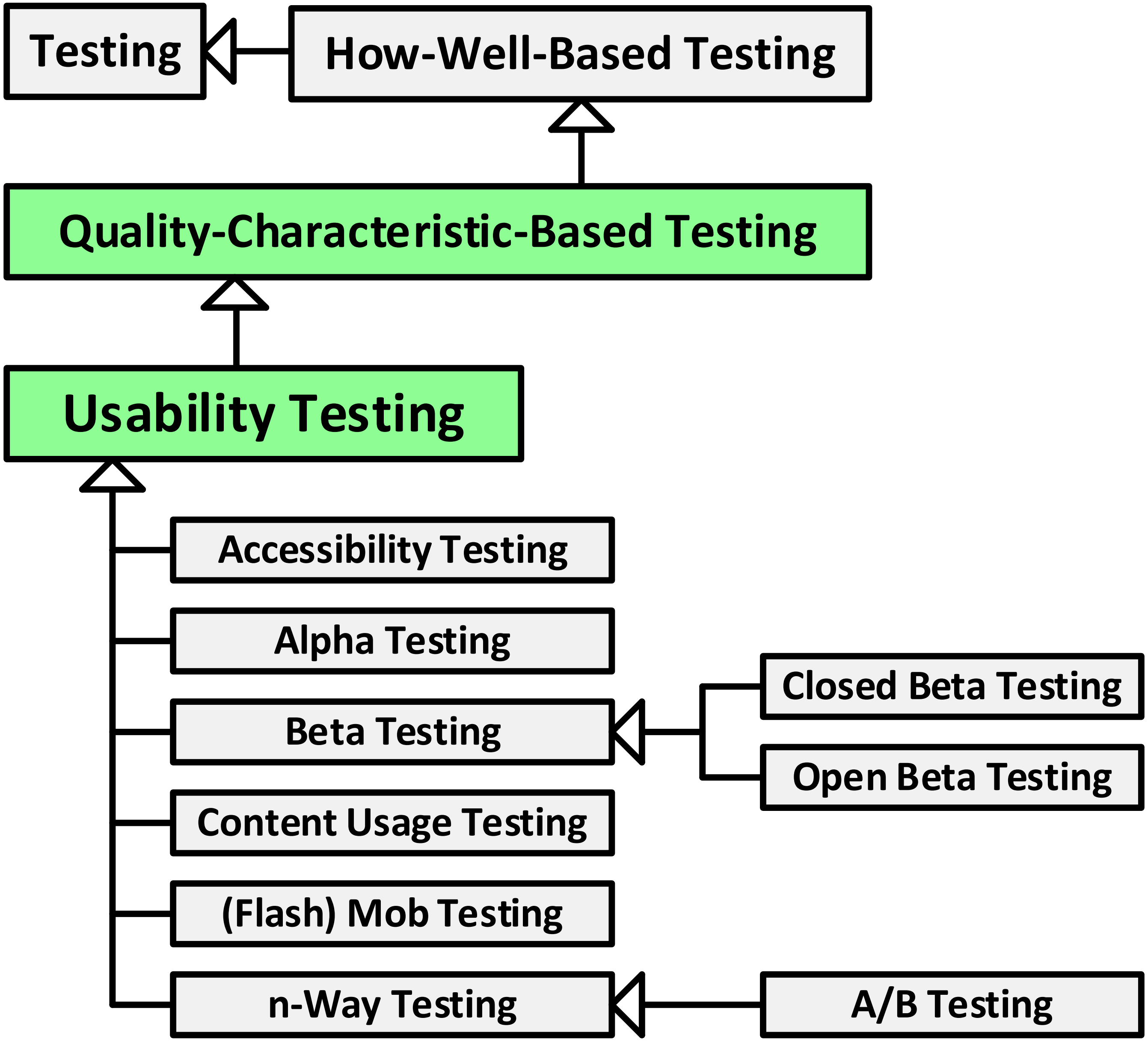



Testing
How-Well-Based Testing
Quality-Characteristic-Based Testing
Capacity Testing
Load Testing
Stress Testing
Volume Testing


Testing
How-Well-Based Testing
Quality-Characteristic-Based Testing
Reliability Testing
Endurance (Stability) Testing
Reliability Enhancement Testing
Reliability Growth Testing
Reliability Mechanism Testing


Testing
How-Well-Based Testing
Quality-Characteristic-Based Testing
Robustness Testing
Communication Error Testing
Hardware Error Testing
Human Error Testing
Software Error Testing
System Error Testing
Error Tolerance Testing
Fault Tolerance Testing
Fault Injection Testing
Failure Tolerance Testing
Failover and Recovery Testing
Environmental Tolerance Testing
Acceleration Tolerance Testing
Acoustic Tolerance Testing
Electromagnetic Compatibility (EMC) Tolerance Testing
Pressure/Leakage Tolerance Testing
Pressure Cycling Tolerance Testing
Radiation Tolerance Testing
Shock/Drop Tolerance Testing
Vacuum Tolerance Testing
Temperature Tolerance Testing
Temperature Cycling Tolerance Testing


Testing
How-Well-Based Testing
Quality-Characteristic-Based Testing
Security Testing
Access Control Testing
Anti-Tamper Testing
Encryption Testing
Penetration Testing
Anti-Spoofing Testing
EMSEC Testing
Infrastructure Testing
Blue Team Testing
Red Team Testing


Testing
How-Well-Based Testing
Quality-Characteristic-Based Testing
Usability Testing
Accessibility Testing
Alpha Testing
Beta Testing
Closed Beta Testing
Open Beta Testing
Content Usage Testing
(Flash) Mob Testing
n-Way Testing
A/B Testing

Conclusion

Most systems require quite a few different types of testing.

Most testers are not aware of the majority of the different types of testing.

- If you are not aware that it exists, then you don't know whether you need it.

These types of testing can be organized into a taxonomy by the 5W + 2H questions.

This taxonomy has several uses:

- Ensure the test strategy is sufficiently complete with no important type of testing overlooked.
- Organize testing types to make them and their relationships more understandable.
- Augment test training materials.
- Help categorize and understand limitations of testing tools.

Further Information

Conference Presentations:

- *12th Next Gen Testing Conference* in London, England on 25 June 2015
- FAA V&V Summit in Atlantic City, New Jersey on 23 September 2015
- Software Solutions Conference (SSC) in Arlington, Virginia on 17 November 2015

More information can be found at the following locations:

- SEI Webinar (one hour): https://www.webcaster4.com/Webcast/Page/139/10188
- SEI Podcast (15 minutes): http://www.sei.cmu.edu/podcasts/podcast_episode.cfm?episodeid=444938
- SEI Blog Series:
  1. https://insights.sei.cmu.edu/sei_blog/2015/08/a-taxonomy-of-testing.html
  2. Second on to be posted on 21 September 2015
- SEI wiki with definitions, descriptions, 5+2 parameters, etc. (Just Started)

Contact Information Slide Format

Donald G. Firesmith
Principal Engineer
Software Solutions Division
Telephone: +1 412-268-6874
Email: dgf@sei.cmu.edu

Web
www.sei.cmu.edu
www.sei.cmu.edu/contact.cfm

U.S. Mail
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Customer Relations
Email: info@sei.cmu.edu
Telephone: +1 412-268-5800
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257