AMPHINICY TECHNOLOGIES

Nothing is Impossible.®

Groovy Integration Testing

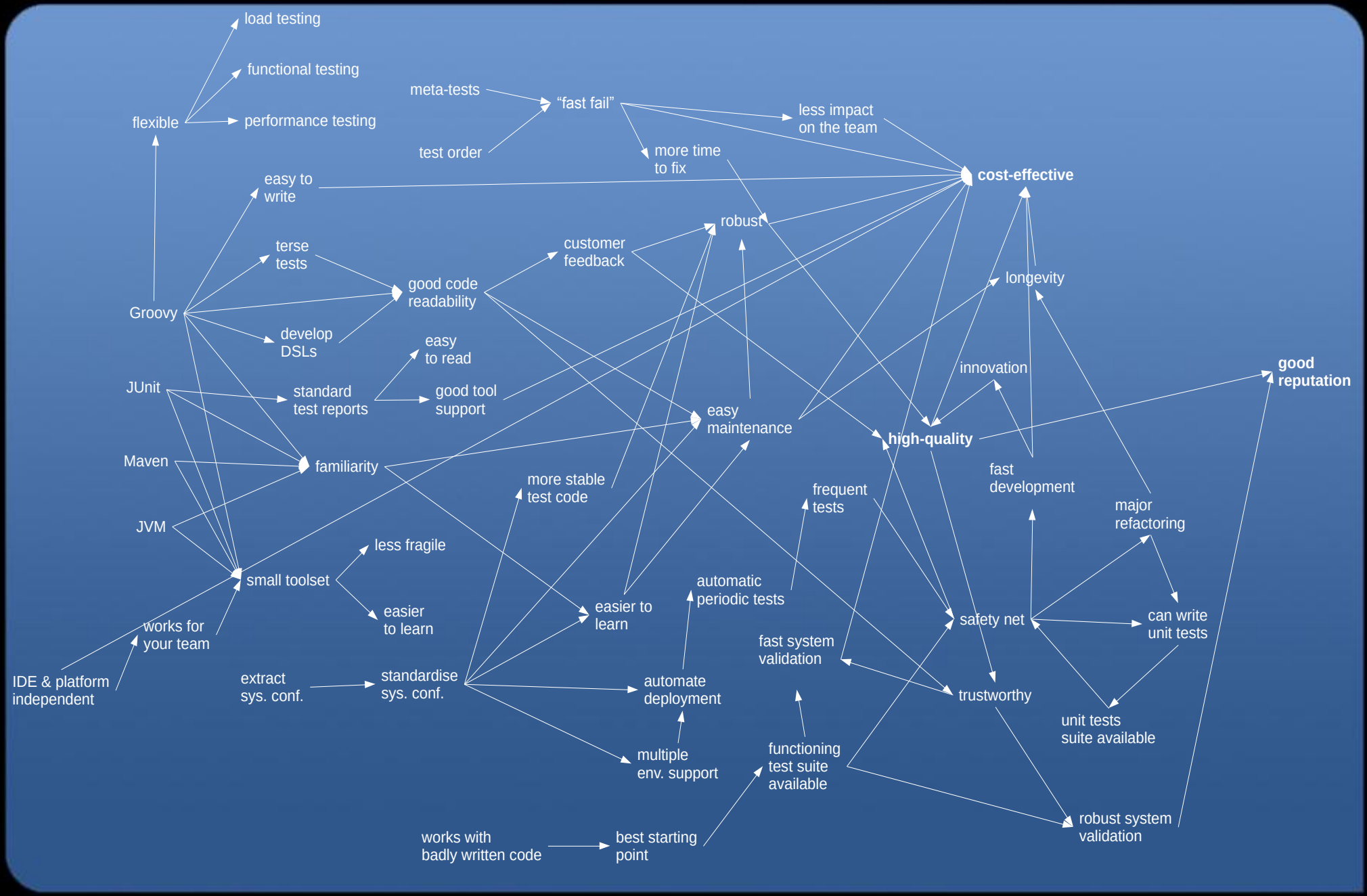
Tomislav Nakić-Alfirević
Amphinicy Technologies

Overview

- What is it?
- Why it matters?
- How to go about it?

AMPHINICY TECHNOLOGIES

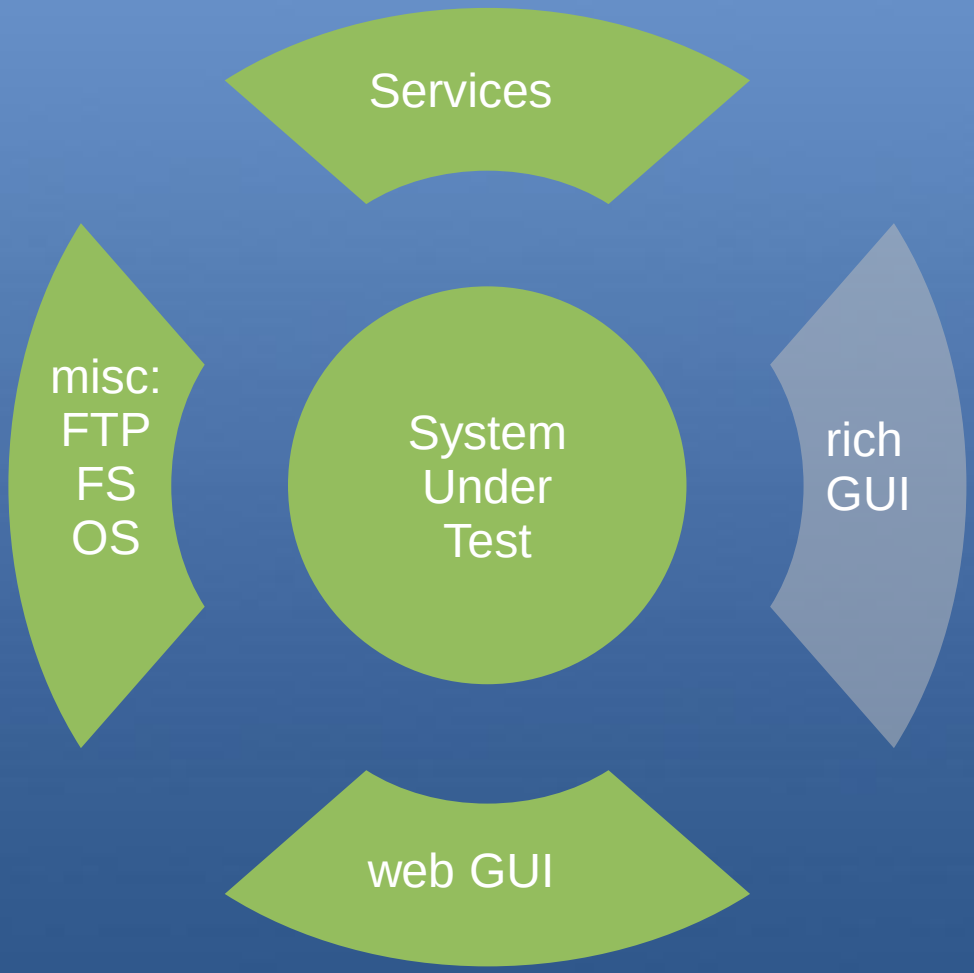
Nothing is Impossible.®

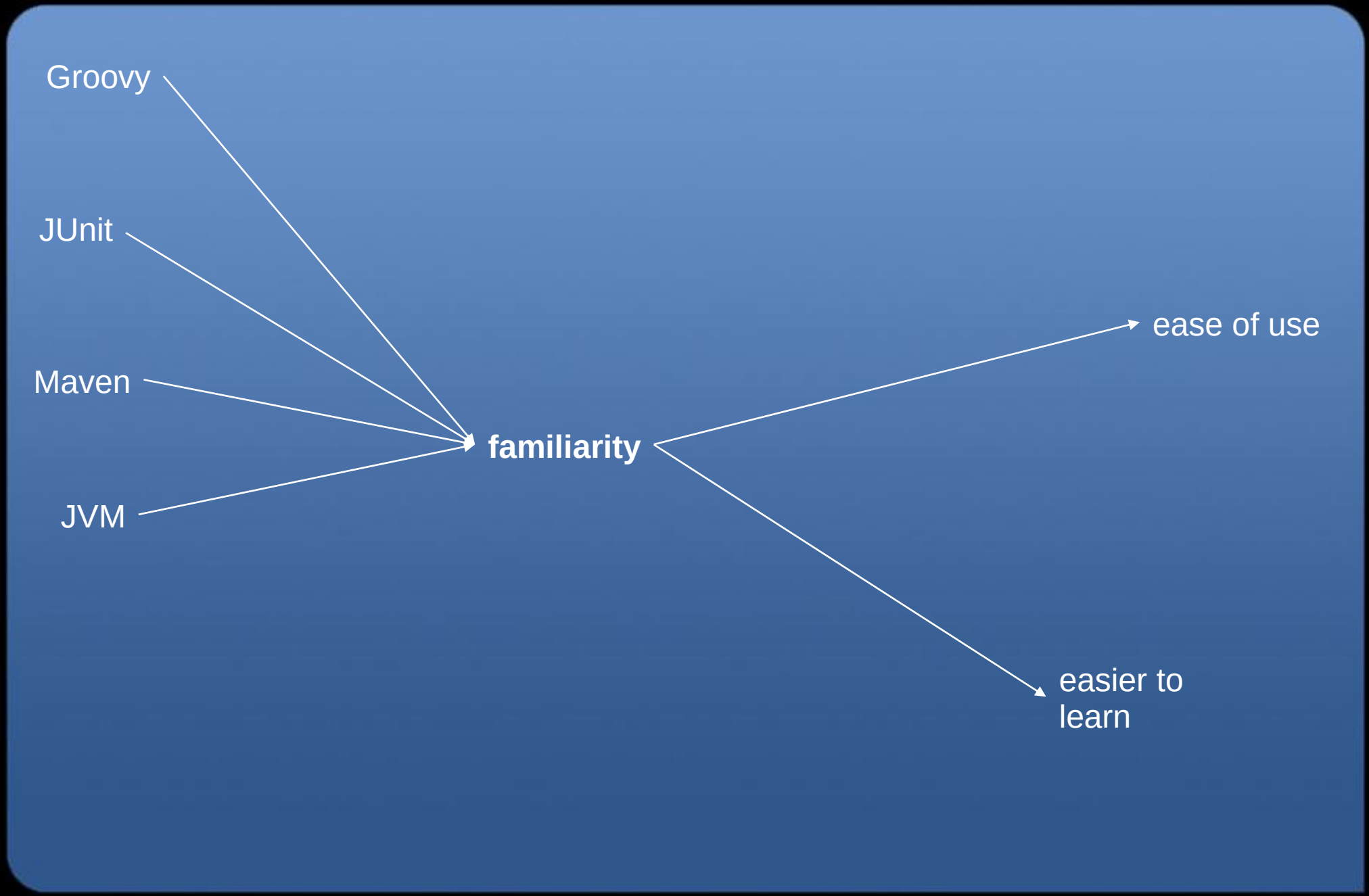


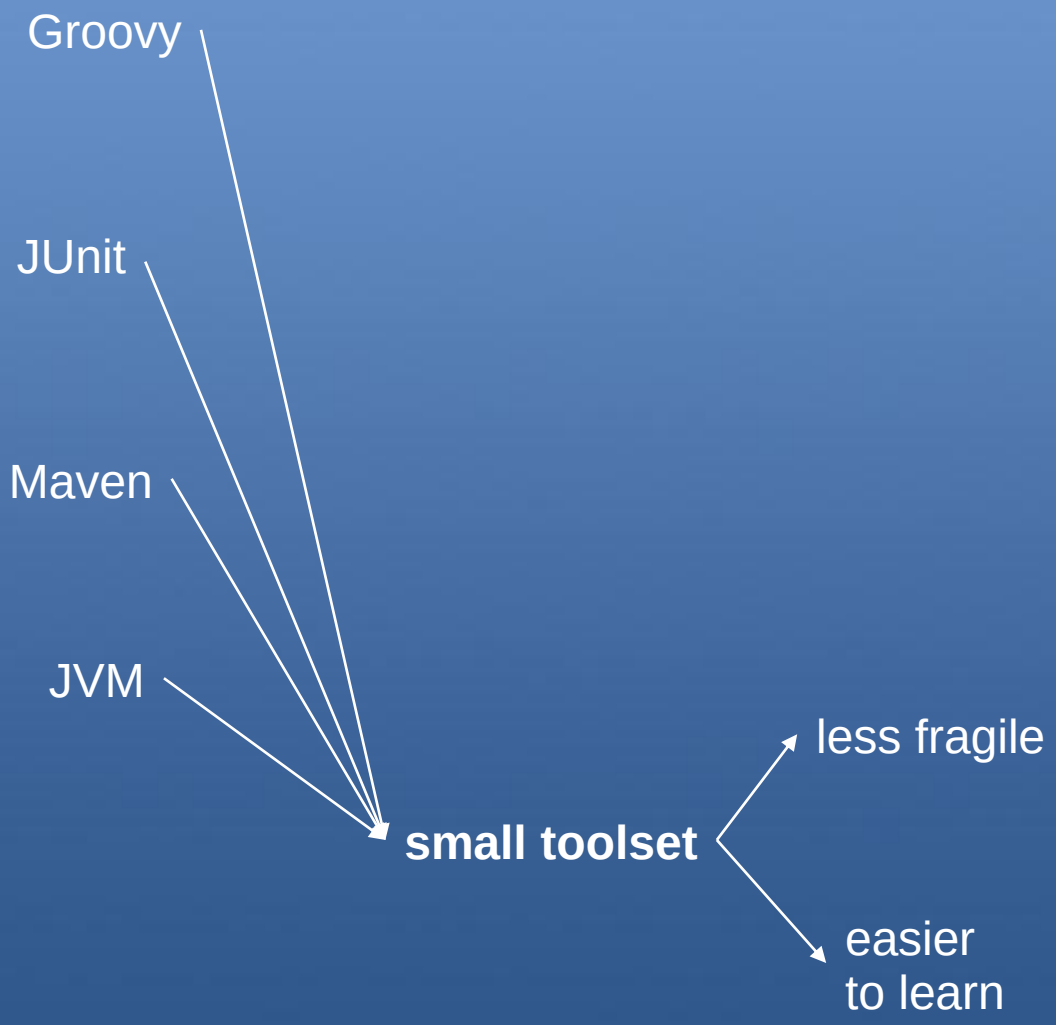
Functional Testing

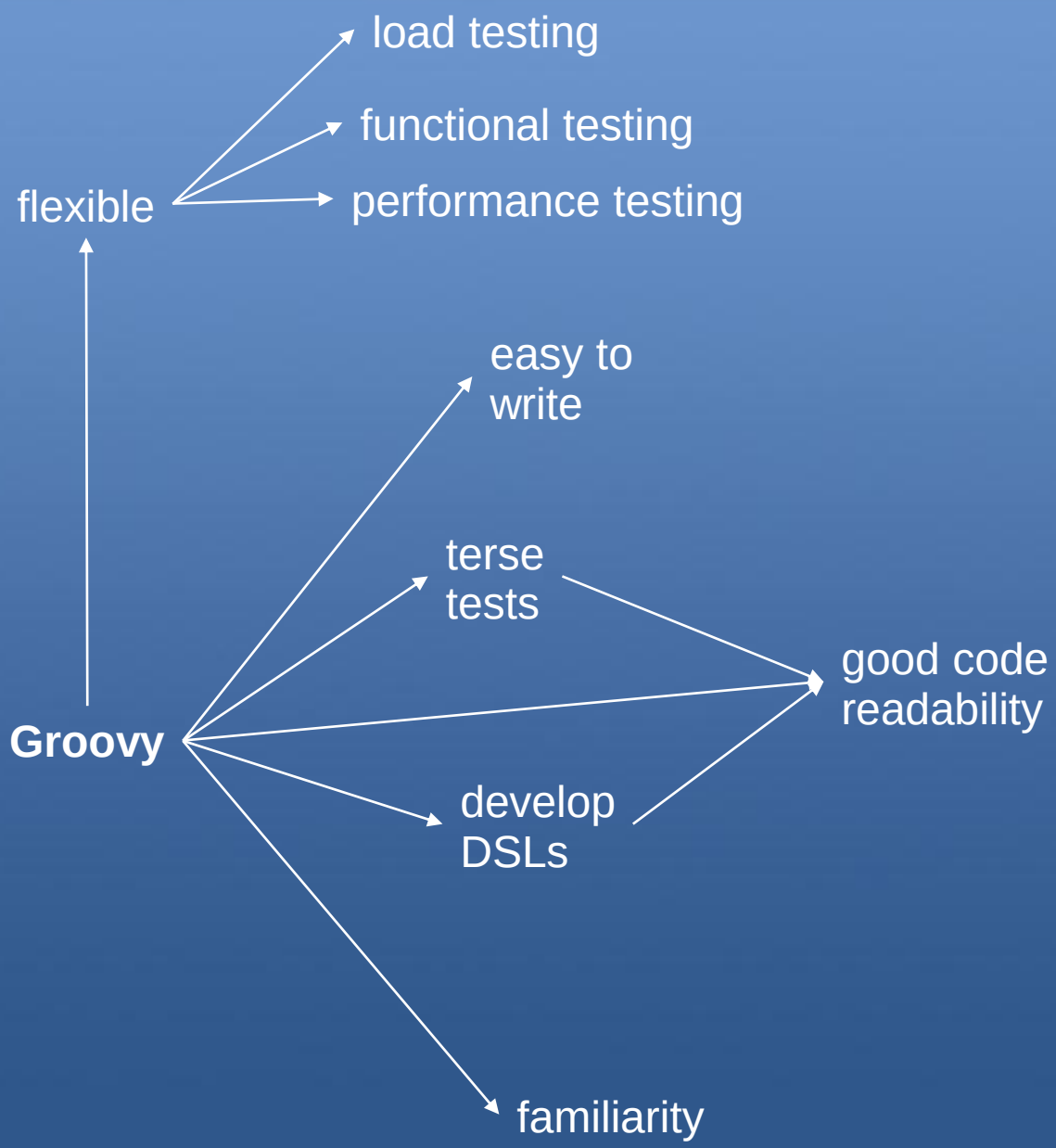
- “Integration testing, noun: functional testing”
- Black-box
- Does the system do what it should?

Interfaces



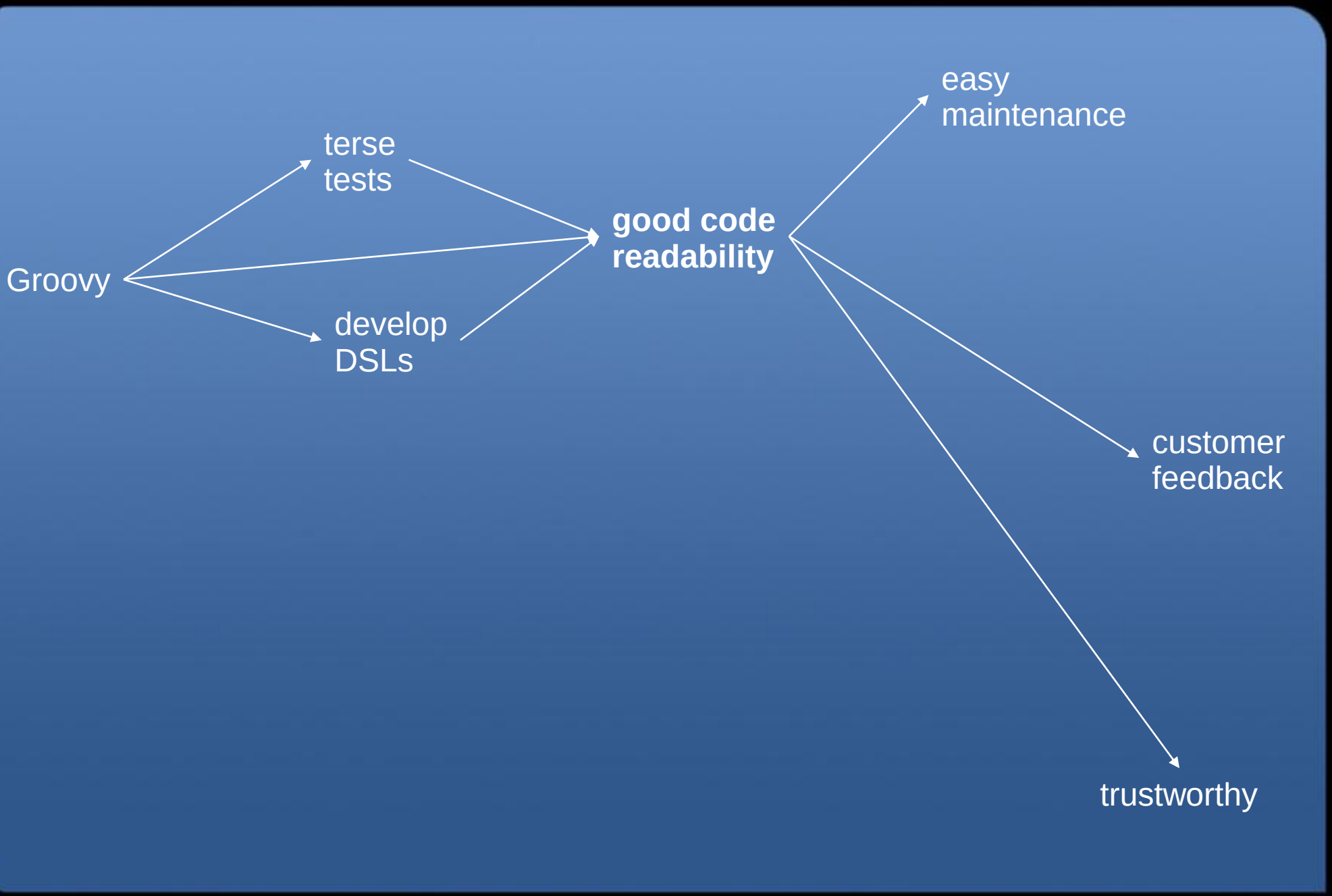

Groovy
JUnit
Maven
JVM
familiarity
ease of use
easier to learn

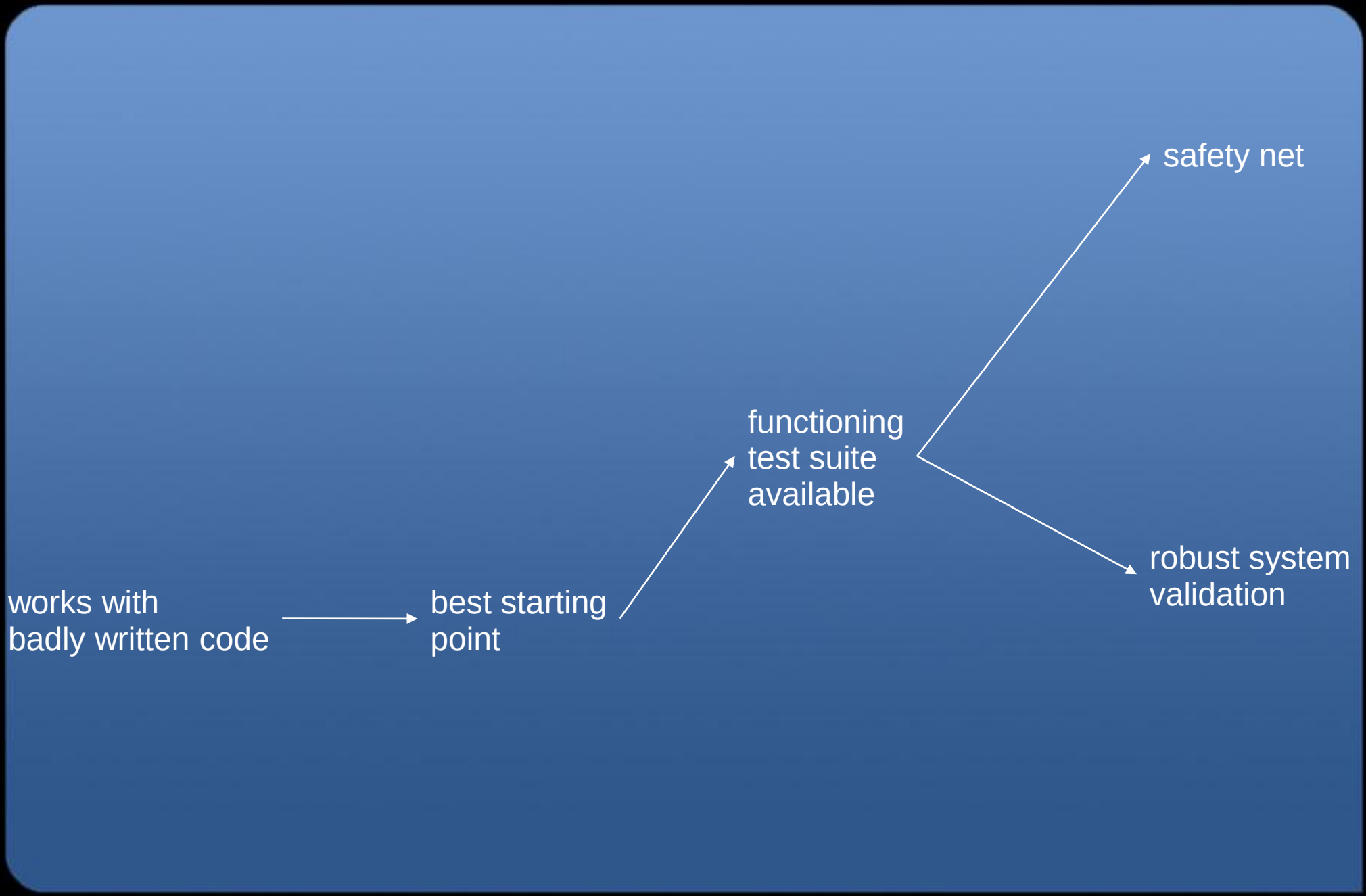

Groovy
JUnit
Maven
JVM
small toolset
less fragile
easier
to learn

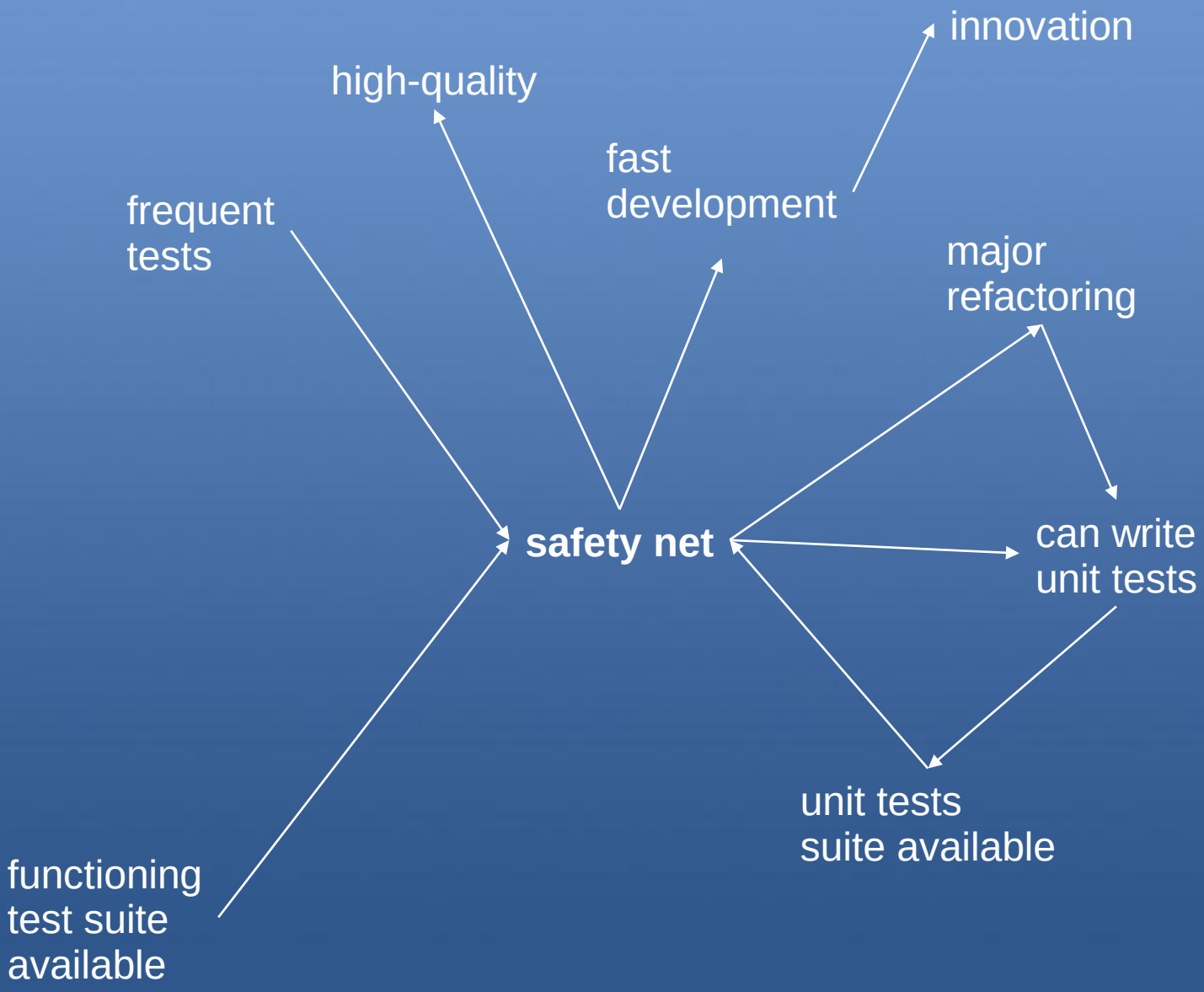

load testing
functional testing
performance testing
flexible
easy to write
terse tests
good code readability
Groovy
develop DSLs
familiarity

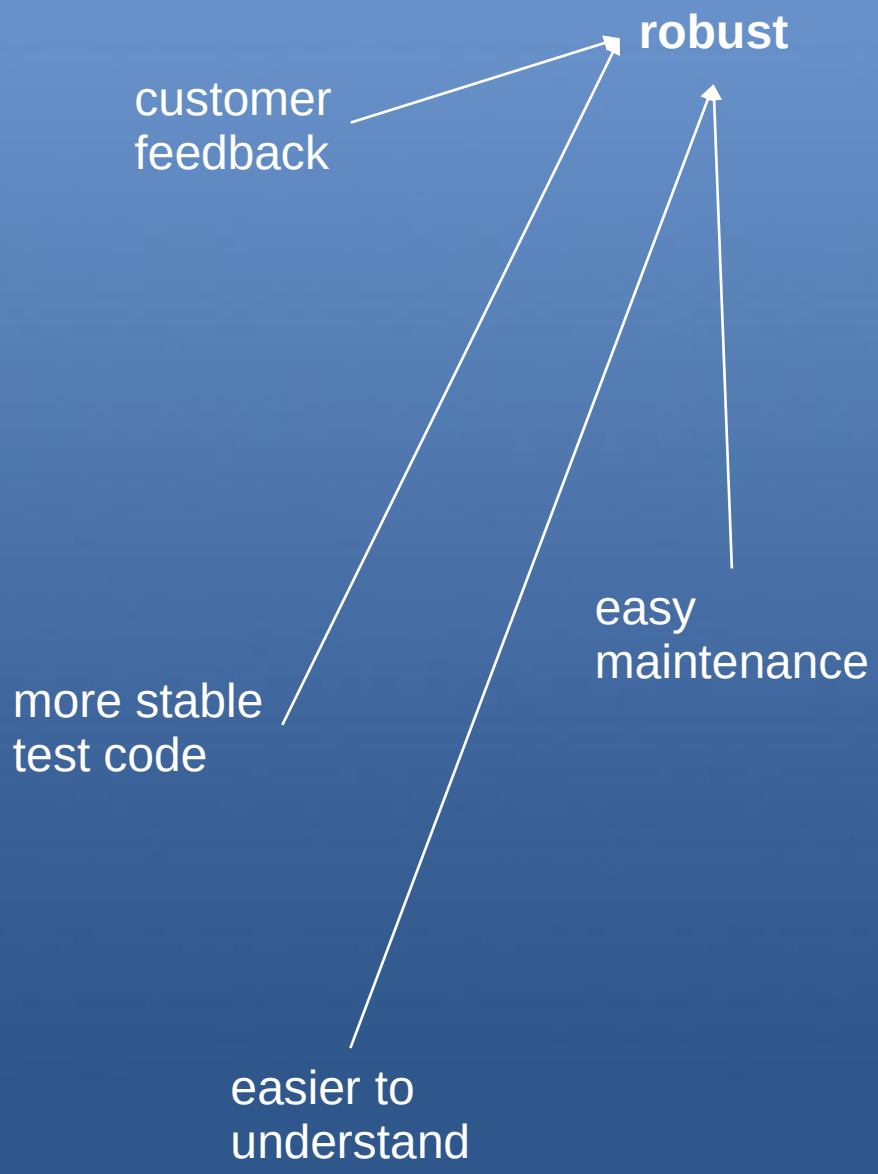
Taste of Groovy

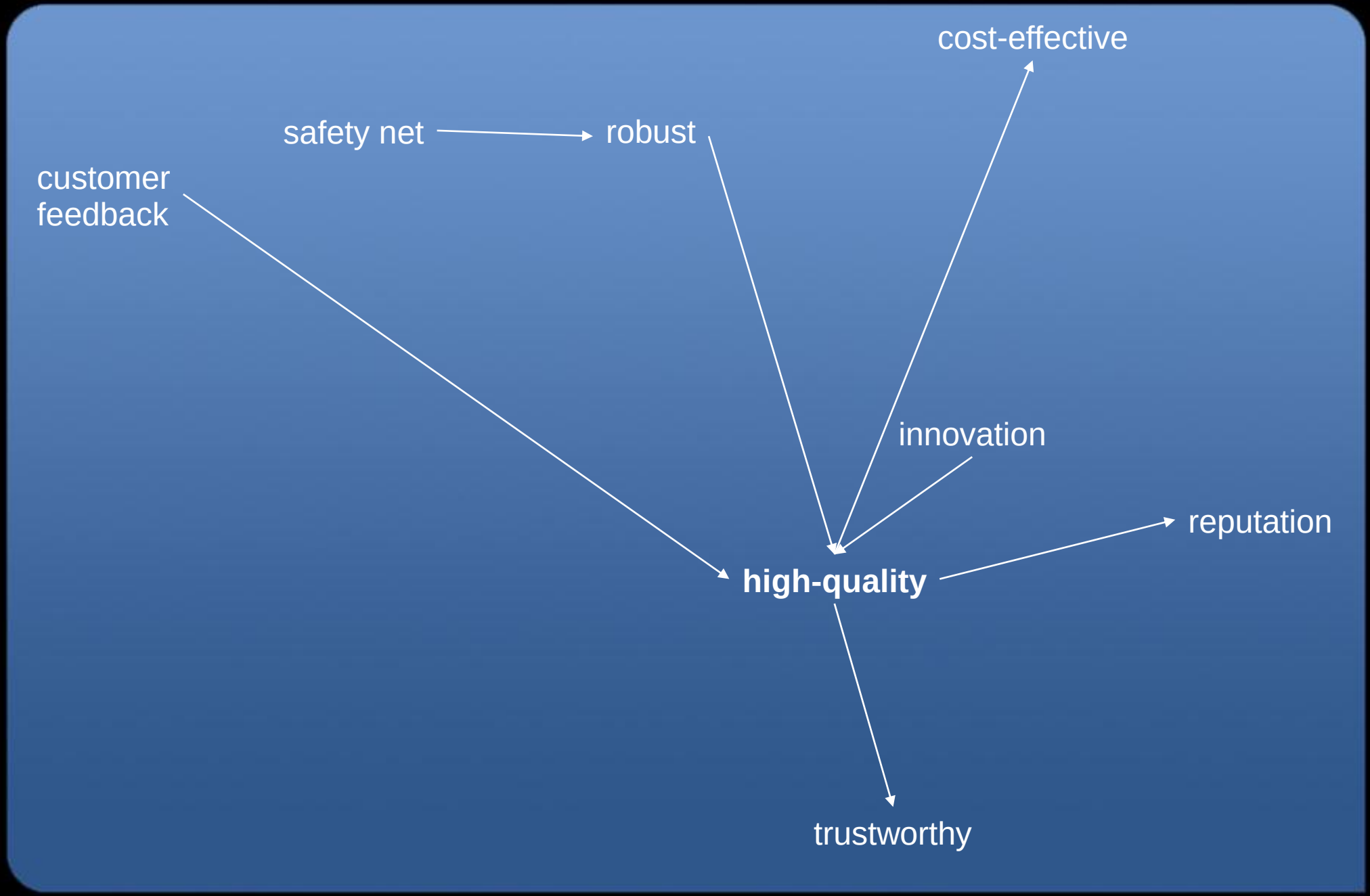
- file contents:
```
content = new File("myfile.txt").text
```
- map init:
```
Map map = [red:"wait", yellow:"get ready", green:"go!"]
```
- number of chars per traffic light:
```
map.values().collect { println it.size() }
```
- count to 128:
```
128.times { println it }
```
- traversing objects regardless of null values:
```
user?.address?.streetName
```
- domain-specific language support:
```
30.km/h + 2.m/s * 2
3 * 3.mg/L
1/2.s - 2.Hz
myAccount += 400.euros
```

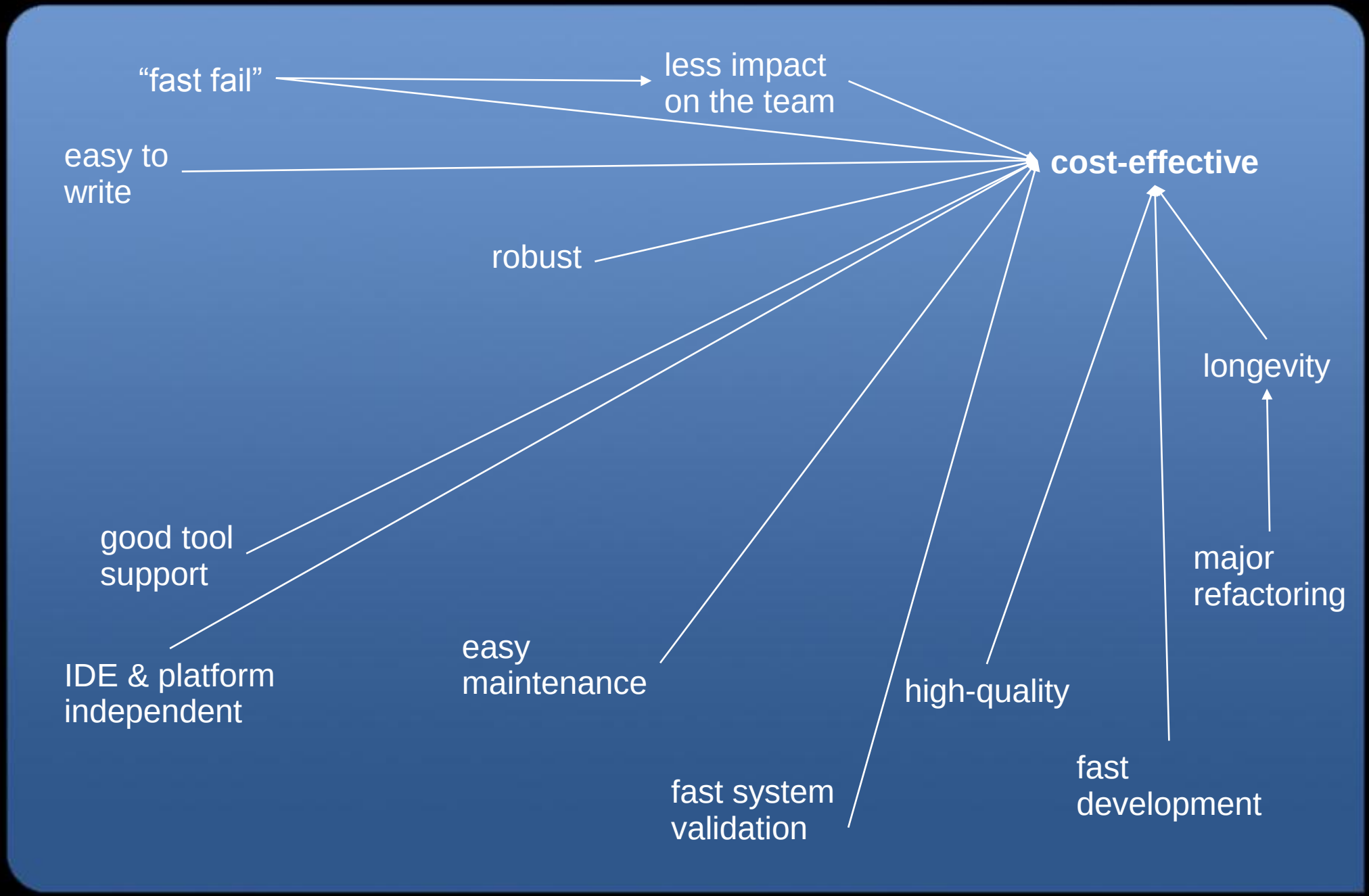

Groovy
terse tests
develop DSLs
good code readability
easy maintenance
customer feedback
trustworthy


safety net
functioning
test suite
available
robust system
validation
works with
badly written code
best starting
point


innovation
high-quality
fast development
frequent tests
major refactoring
safety net
can write unit tests
unit tests suite available
functioning test suite available


robust
customer feedback
more stable test code
easy maintenance
easier to understand


cost-effective
safety net
robust
customer feedback
innovation
reputation
high-quality
trustworthy


“fast fail”
less impact on the team
easy to write
cost-effective
robust
longevity
good tool support
major refactoring
IDE & platform independent
easy maintenance
high-quality
fast system validation
fast development

Examples

Test prototyping

- Groovy shell
- Web services: WSClient, groovy-wslite
- Web UI: htmlunit, webtest...

In summary…

Groovy, Maven, JDK, JUnit

easy

rapid

high-quality

efficient

safety net