Quality Attribute Generic Scenarios

Different quality attributes

Availability is concerned with system failure and duration of system
failures. System failure means … when the system does not provide the service for which it was intended.

Modifiability is about the cost of change, both in time and money.

Performance is about timeliness. Events occur and the system must respond in a timely fashion.

Security is the ability of the system to prevent or resist unauthorized access while providing access to legitimate users. An attack is an attempt to breach security.

Testability refers to the ease with which the software can be made to demonstrate its faults or lack thereof. To be testable the system must control inputs and be able to observe outputs.

Usability is how easy it is for the user to accomplish tasks and what support the system provides for the user to accomplish this. Dimensions:

- Learning system features
- Using the system efficiently
- Minimizing the impact of errors
- Adapting the system to the user's needs
- Increasing confidence and satisfaction

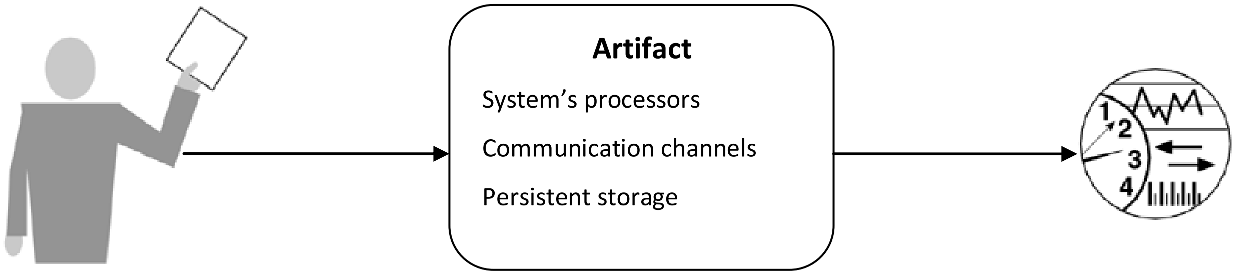
Quality attribute scenarios

A Quality Attribute Scenario is a quality-attribute-specific requirement.
There are 6 parts:

1. Source of stimulus (e.g., human, computer system, etc.)
2. Stimulus – a condition that needs to be considered
3. Environment - what are the conditions when the stimulus occurs?
4. Artifact – what elements of the system are stimulated.
5. Response – the activity undertaken after arrival of the stimulus
6. Response measure – when the response occurs it should be measurable so that the requirement can be tested.

Availability

General availability scenario



Artifact

System’s processors

Communication channels

Persistent storage

Source

Internal to system

External to system

Stimulus

Crash

Omission

Timing

No response

Incorrect response

Environment

Normal operation

Startup

Shutdown

Repair mode

Degraded (failsafe) mode

Overloaded operation

Response

Prevent the failure

Log the failure

Notify users / operators

Disable source of failure

Temporarily unavailable

Continue (normal / degraded)

Measure

Time interval available

Availability %

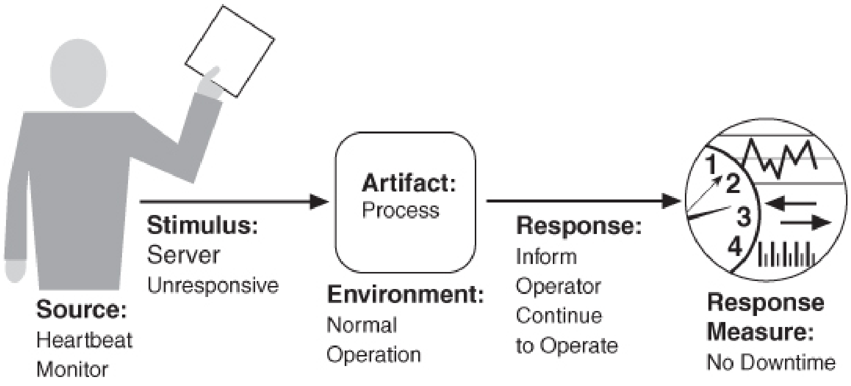
Detection time

Repair time

Degraded mode time interval

Unavailability time interval

Sample availability scenario



Source: Heartbeat Monitor

Stimulus: Server Unresponsive

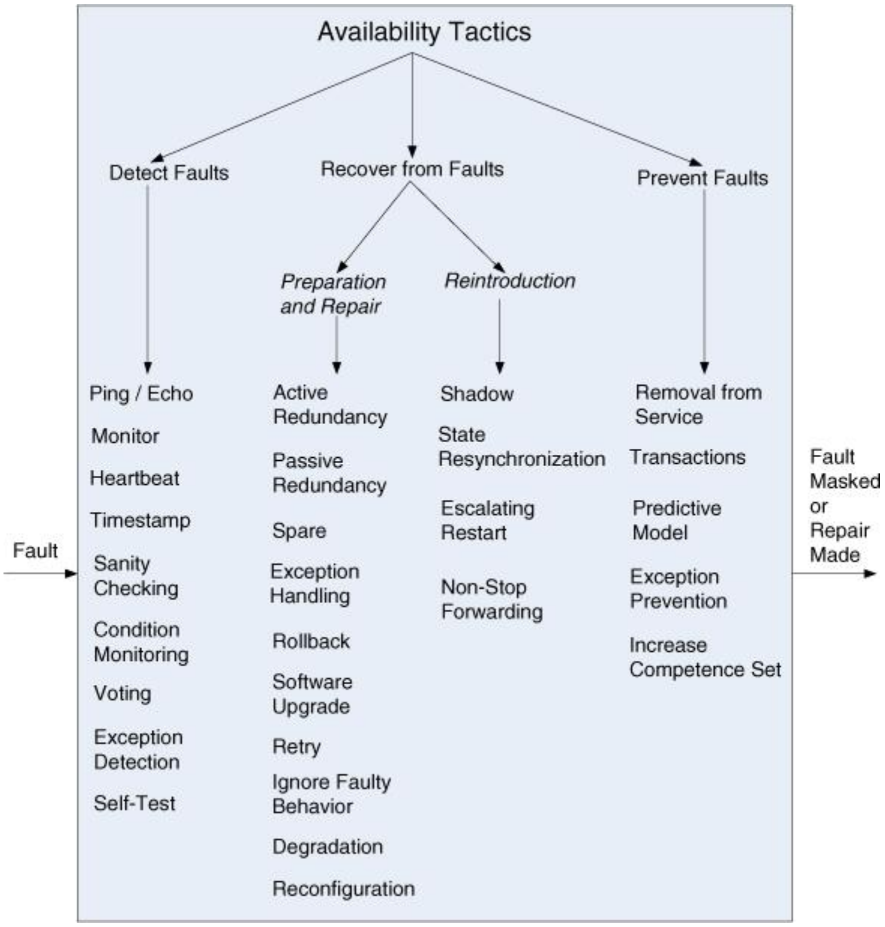
Artifact: Process

Environment: Normal Operation

Response: Inform Operator Continue to Operate

Response Measure: No Downtime

Availability tactics



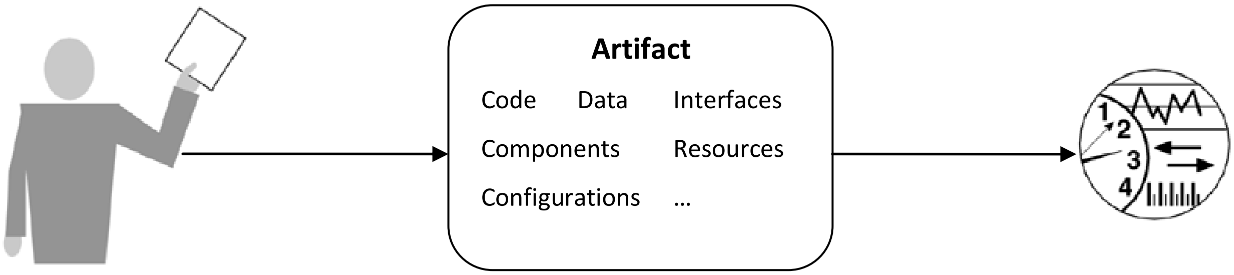
More information

You can find more information in the book “Software architecture in Practice”. Please note that the website of the book contains an outdated version (version 2) while this document is based on version 3.

Availability: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch04lev1sec4.html
Availability tactics: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch05lev1sec2.html

Modifiability

General modifiability scenario



Artifact

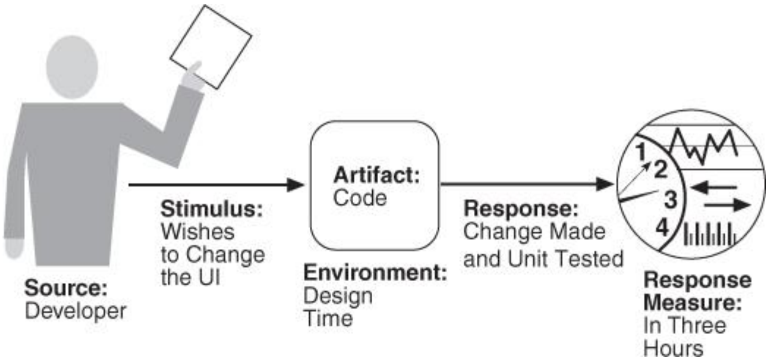
Code Data Interfaces

Components Resources

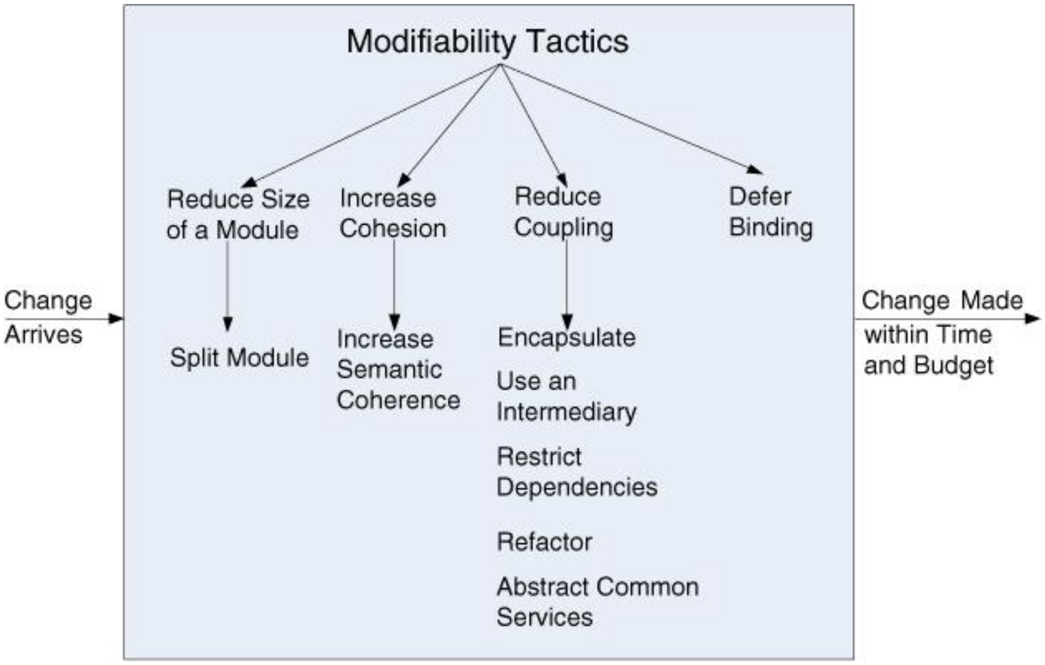
Configurations …

Source	Stimulus	Environment	Response	Measure
End-user Developer System-administrator	Add / delete / modify functionality, quality attribute, capacity or technology	Runtime Compile time Build time Initiation time Design time	Make modification Test modification Deploy modification	Cost in effort Cost in money Cost in time Cost in number, size, complexity of affected artifacts Extent affects other system functions or qualities New defects introduced

Sample modifiability scenario


Stimulus: Wishes to Change the UI
Artifact: Code
Response: Change Made and Unit Tested
Source: Developer
Environment: Design Time
Response Measure: In Three Hours
1
2
3
4

Modifiability tactics



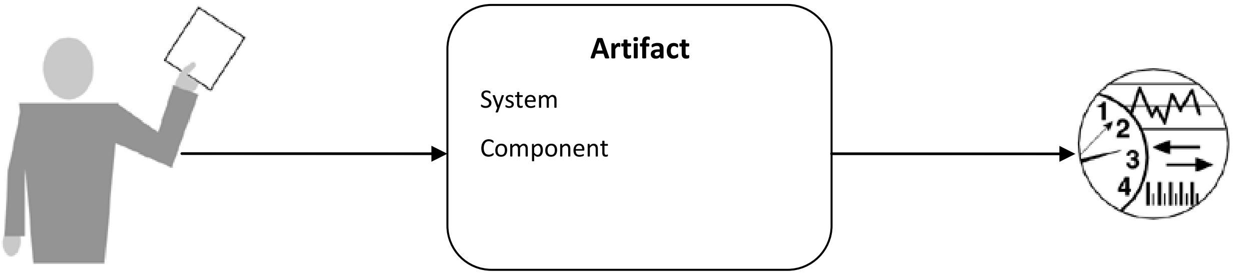
More information

You can find more information in the book "Software architecture in Practice". Please note that the website of the book contains an outdated version (version 2) while this document is based on version 3.

Modifiability: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch04lev1sec4.html
Modifiability tactics: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch05lev1sec3.html

Performance

General performance scenario



Artifact

System

Component

Source

Internal to the system

External to the system

Stimulus

Periodic events

Sporadic events

Bursty events

Stochastic events

Environment

Normal mode

Overload mode

Reduced capacity mode

Emergency mode

Peak mode

Response

Process events

Change level of service

Measure

Latency

Deadline

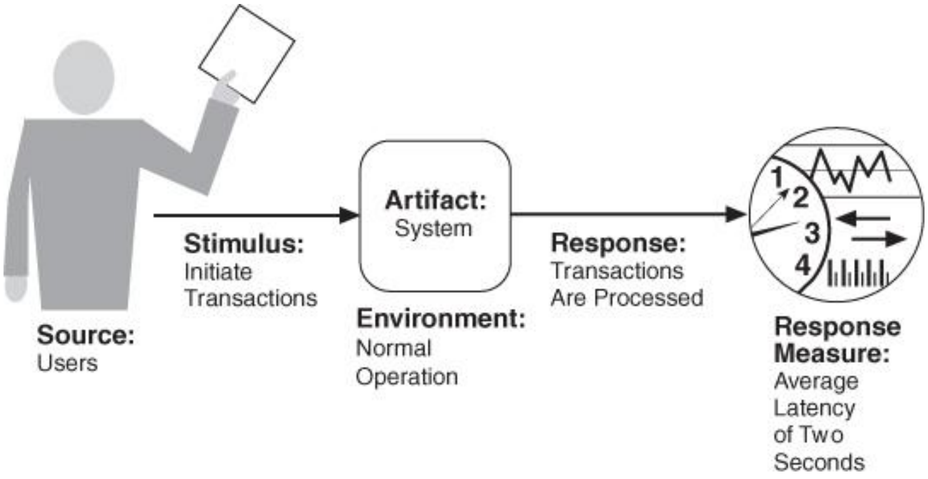
Throughput

Jitter

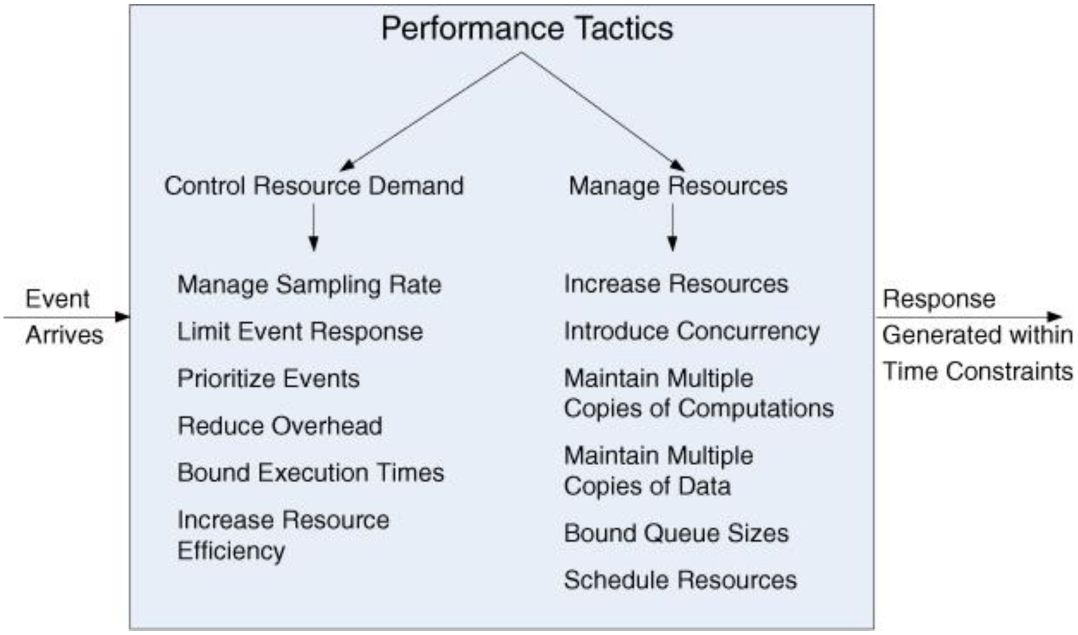
Miss rate

Data loss

Sample performance scenario



Performance tactics



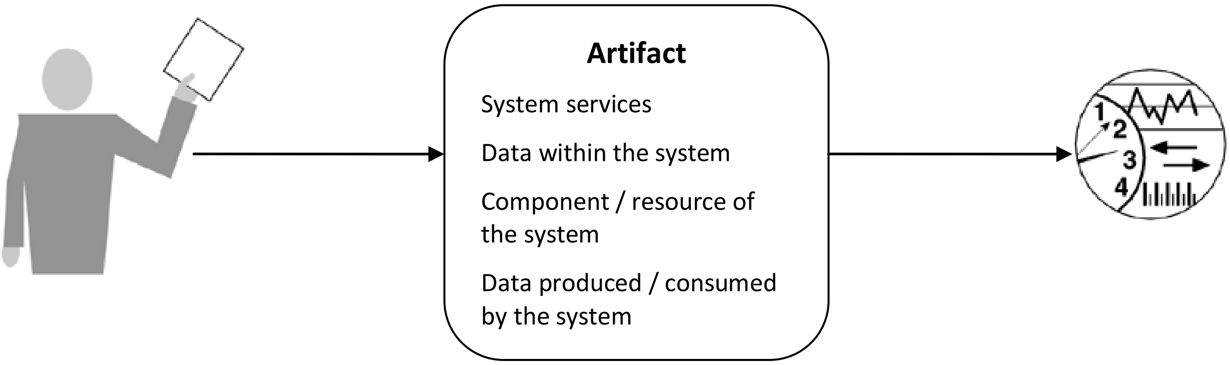
More information

You can find more information in the book “Software architecture in Practice”. Please note that the website of the book contains an outdated version (version 2) while this document is based on version 3.

Performance: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch04lev1sec4.html
Performance tactics: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch05lev1sec4.html

Security

General security scenario



Artifact

System services

Data within the system

Component / resource of the system

Data produced / consumed by the system

Source

Identified user

Unknown user

Hacker from outside the organization

Hacker from inside the organization

Stimulus

Attempt to display data

Attempt to modify data

Attempt to delete data

Access system services

Change system's behavior

Reduce availability

Environment

Normal mode

Overload mode

Reduced capacity mode

Emergency mode

Peak mode

Response

Process events

Change level of service

Measure

Latency

Deadline

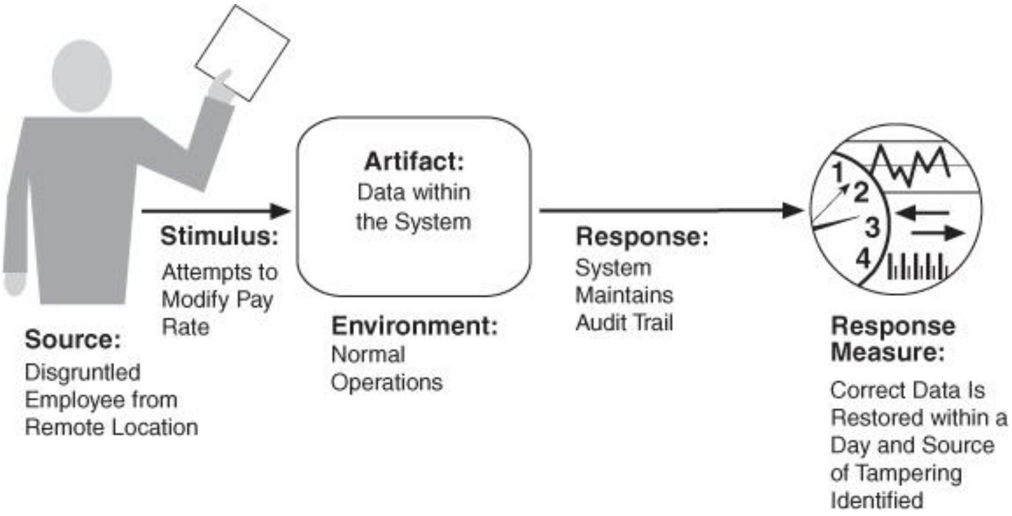
Throughput

Jitter

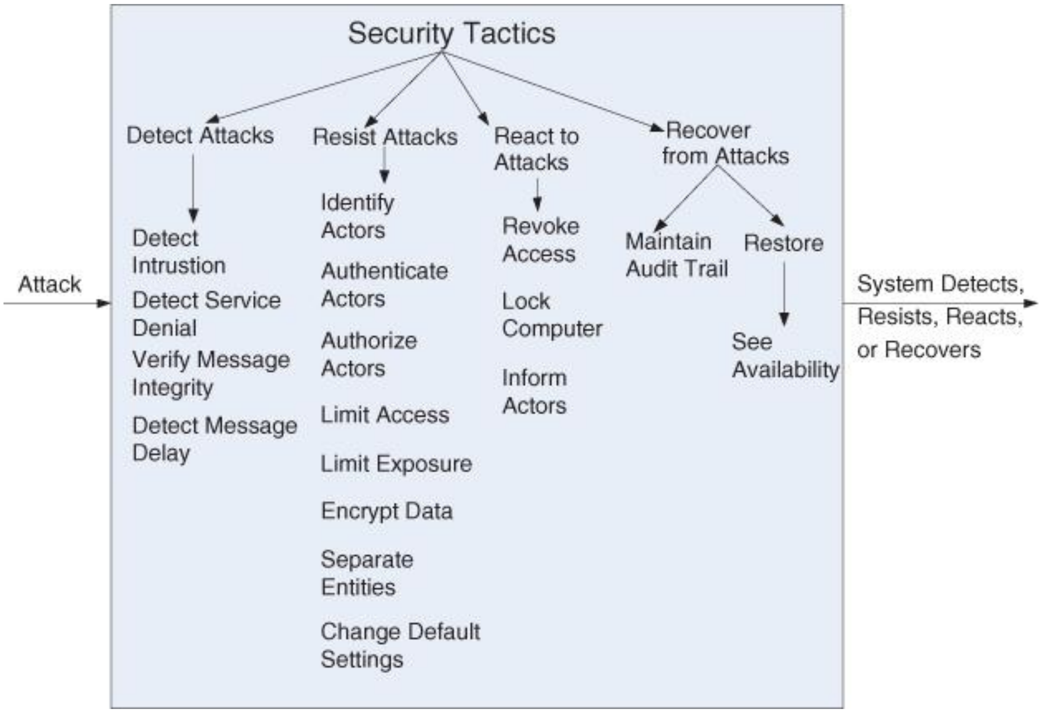
Miss rate

Data loss

Sample security scenario


Stimulus:
Attempts to Modify Pay Rate
Source:
Disgruntled Employee from Remote Location
Artifact:
Data within the System
Environment:
Normal Operations
Response:
System Maintains Audit Trail
Response Measure:
Correct Data Is Restored within a Day and Source of Tampering Identified

Security tactics



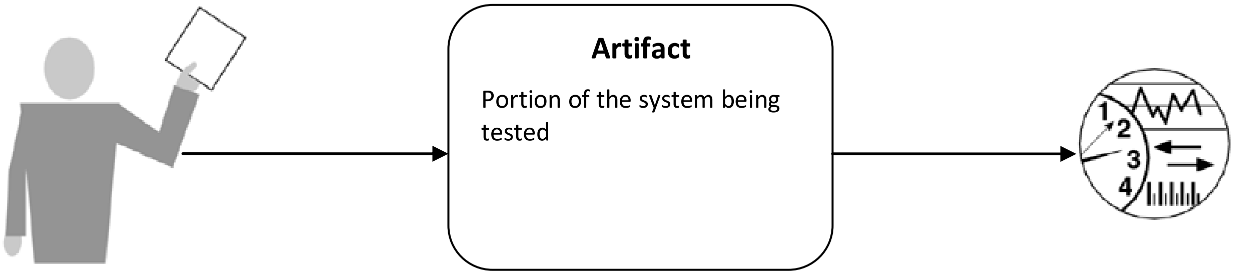
More information

You can find more information in the book "Software architecture in Practice". Please note that the website of the book contains an outdated version (version 2) while this document is based on version 3.

Security: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch04lev1sec4.html
Security tactics: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch05lev1sec5.html

Testability

General testability scenario



Artifact

Portion of the system being tested

Source

- Unit tester
- Integration tester
- System tester
- Acceptance tester
- End user
- Automated testing tools

Stimulus

- Execution of tests due to completion of code increment

Environment

- Design time
- Development time
- Compile time
- Integration time
- Deployment time
- Run time

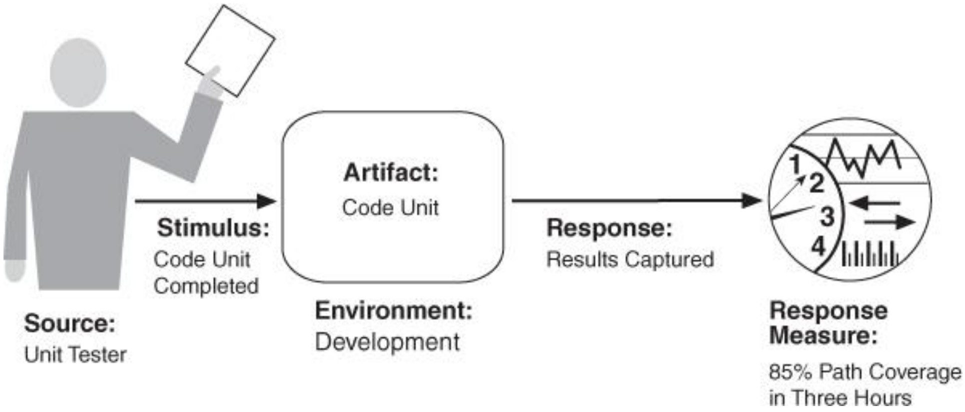
Response

- Execute test suite & capture results
- Capture cause of fault
- Control & monitor state of the system

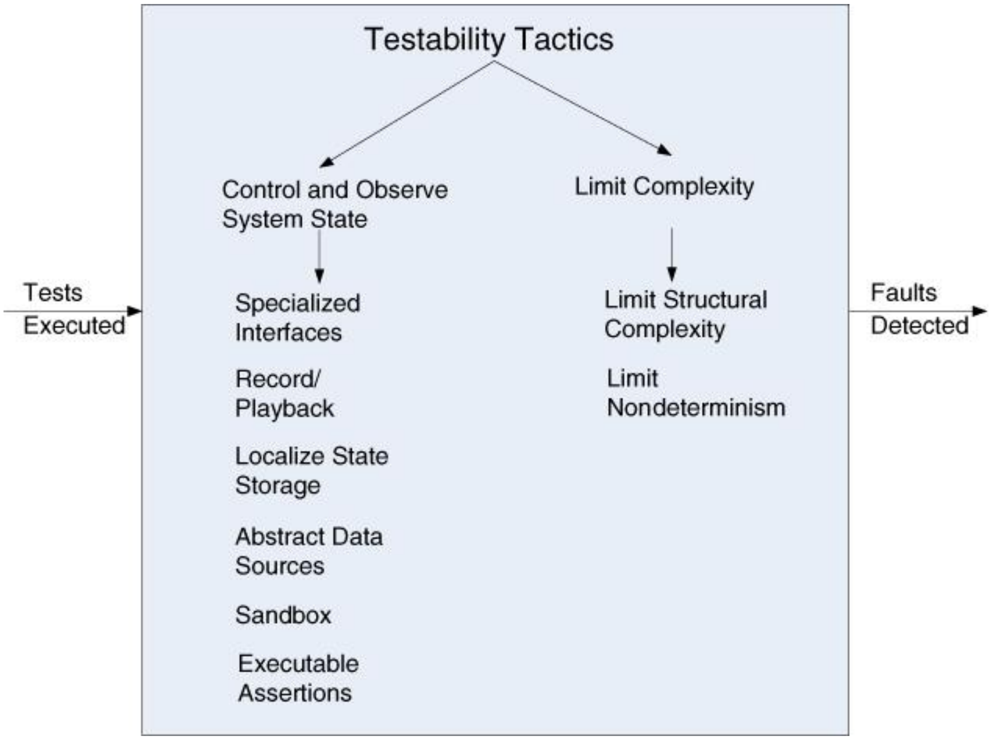
Measure

- Effort to find fault
- Effort to achieve coverage %
- Probability of fault being revealed by next test
- Time to perform tests
- Effort to detect faults
- Length of longest dependency chain
- Time to prepare test environment
- Reduction in risk exposure

Sample testability scenario



Testibility tactics



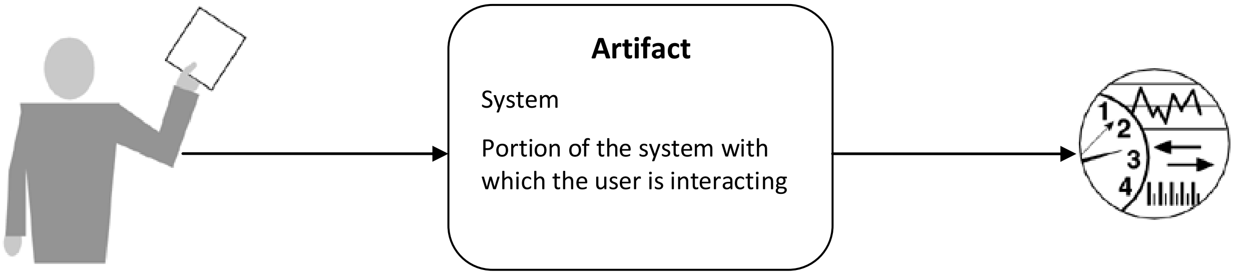
More information

You can find more information in the book “Software architecture in Practice”. Please note that the website of the book contains an outdated version (version 2) while this document is based on version 3.

Testability: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch04lev1sec4.html
Testability tactics: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch05lev1sec6.html

Usability

General usability scenario



Source

End user (possibly special role)

Stimulus

Use the system efficiently

Learn to use the system

Minimize impact of errors

Adapt the system

Configure the system

Environment

Runtime

Configuration time

Response

Provide features needed

Anticipate the user's needs

Measure

Task time

Number of errors

Number of tasks accomplished

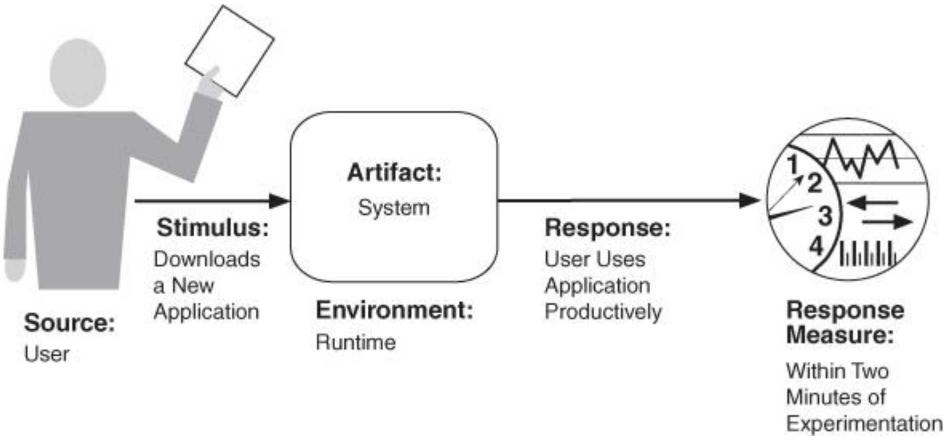
User satisfaction

Gain of user knowledge

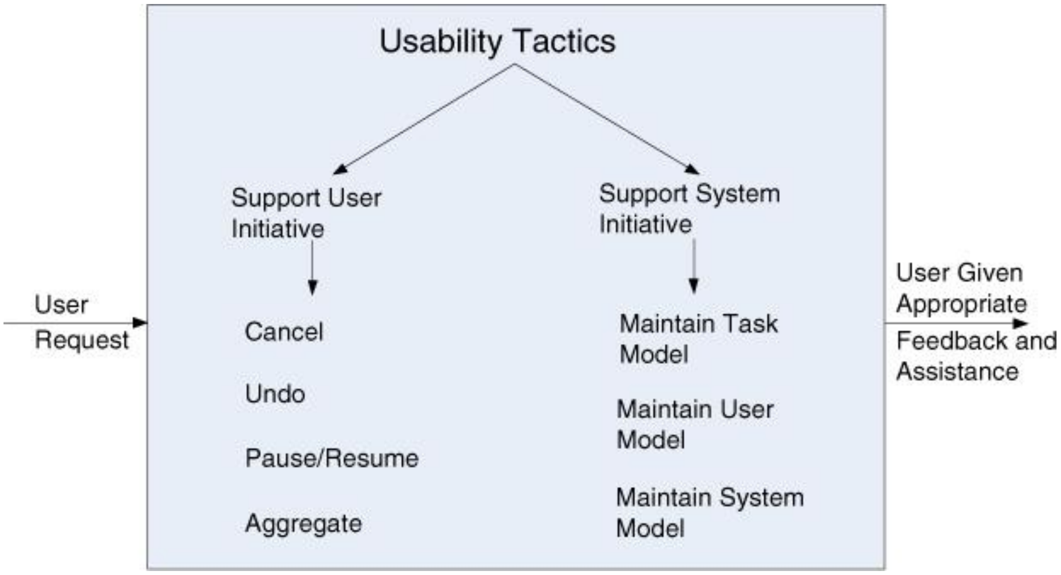
Ratio of successful operations to total operations

Amount of time / data lost when error occurs

Sample usability scenario


Stimulus:
Downloads a New Application
Artifact:
System
Environment:
Runtime
Response:
User Uses Application Productively
Source:
User
Response Measure:
Within Two Minutes of Experimentation
1
2
3
4

Usability tactics



More information

You can find more information in the book "Software architecture in Practice". Please note that the website of the book contains an outdated version (version 2) while this document is based on version 3.

Usability: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch04lev1sec4.html
Usability tactics: http://www.ece.ubc.ca/~matei/EECE417/BASS/ch05lev1sec7.html

Table 1 – Availability Generic Scenario.

Availability	
Source	Internal to system or external to system
Stimulus	Crash, omission, timing, no response, incorrect response
Artifact	System’s processors, communication channels, persistent storage
Environment	Normal operation; degraded (failsafe) mode
Response	Log the failure, notify users/operators, disable source of failure, continue (normal/degraded)
Response Measure	Time interval available, availability%, repair time, unavailability time interval

Table 2 – Modifiability Generic Scenario.

MODIFIABILITY	
Source	End-user, developer, system-administrator
Stimulus	Add/delete/modify functionality or quality attr.
Artifact	System user interface, platform, environment
Environment	At runtime, compile time, build time, design-time
Response	Locate places in architecture for modifying, modify, test modification, deploys modification
RespMeasure	Cost in effort, money, time, extent affects other system functions or qualities

Table 3 – Performance Generic Scenario.

Performance	
Source	A number of sources both external and internal
Stimulus	Periodic events, sporadic events, bursty, stochastic events
Artifact	System, or possibly a component
Environment	Normal mode; overload mode ; reduced capacity mode
Response	Process stimuli; change level of service
RespMeasure	Latency, deadline, throughput, capacity jitter, miss rate, data loss

Table 4 – Security Generic Scenario.

Scenario Portion	Possible Values
Source	User/system who is legitimate/imposter/unknown with full/limited access
Stimulus	Attempt to display/modify data; access services
Artifact	System services, data
Environment	Normal operation; degraded (failsafe) mode
Response	Authenticate user; hide identity of user; grant/block access; encrypt data; detect excessive demand…
RespMeasure	Time /effort/resources to circumvent security measures with probability of success

Table 5 – Testability Generic Scenario.

Scenario Portion	Possible Values
Source	Unit developer, increment integrator, system verifier, client acceptance tester, system user
Stimulus	Analysis, architecture, design, class, subsystem integration, system delivered
Artifact	Piece of design, piece of code, complete system
Environment	At design time, at development time, at compile time, at deployment time
Response	Provide access to state data values, observes results, compares
RespMeasure	% coverage; prob. of failure; time to perform tests; length of time to prepare test environment

Table 6 – Usability Generic Scenario.

Scenario Portion	Possible Values
Source	End user
Stimulus	Wants to: learn system, use system, recover from errors, adapt system, feel comfortable
Artifact	System
Environment	At runtime, or configure time, install-time
Response	(see below)
RespMeasure	Task time, number of errors, number of tasks accomplished, user satisfaction, gain of user knowledge, amount of time/data lost