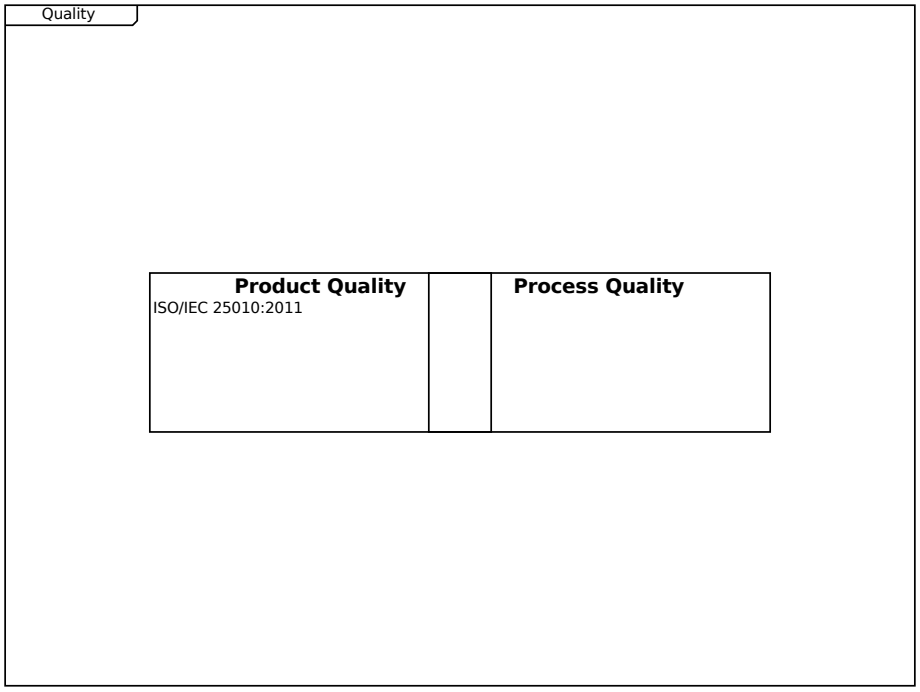
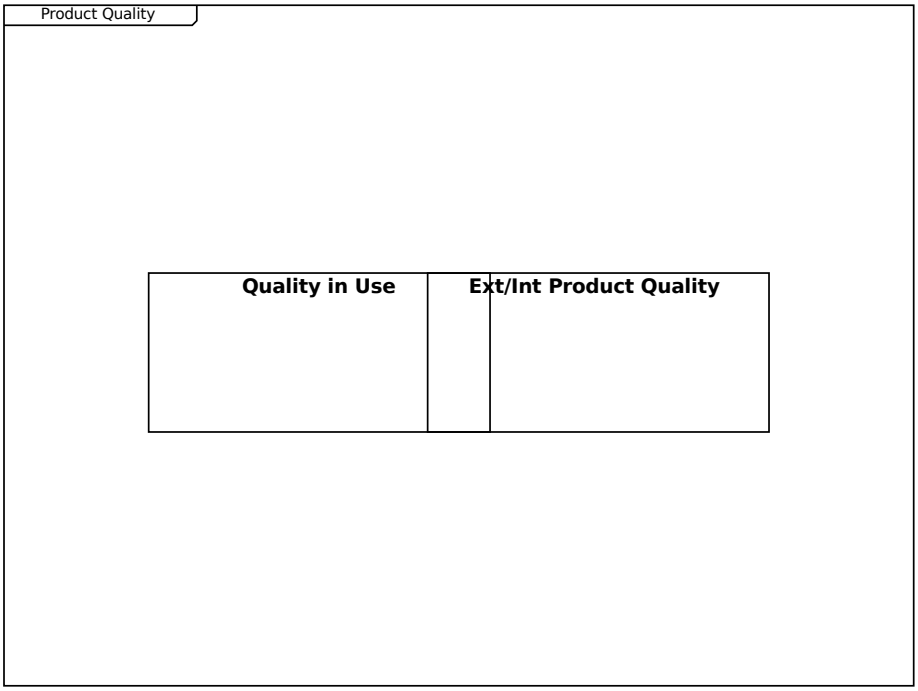
1 Quality Example



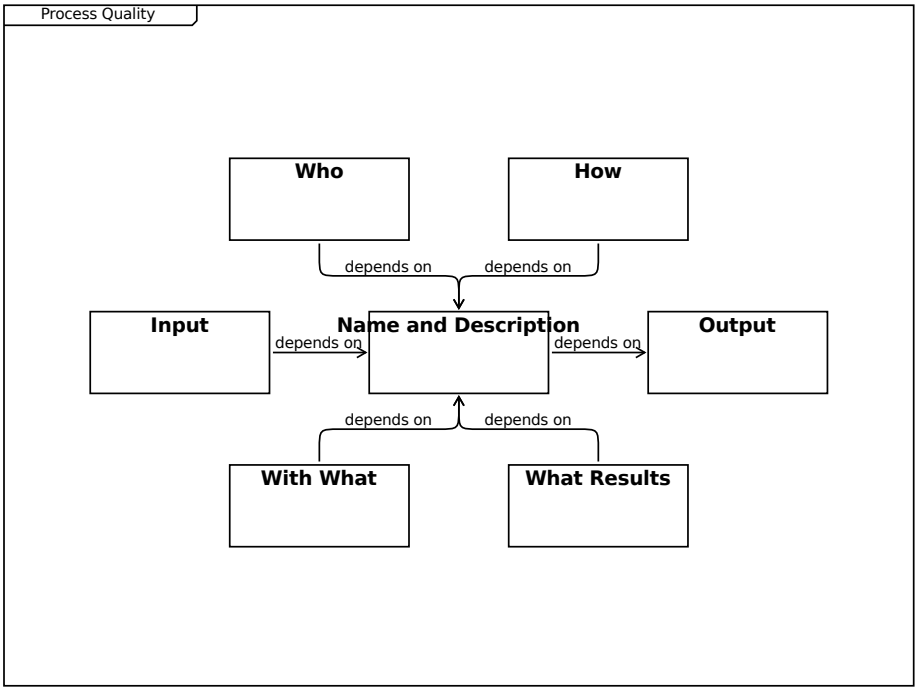

```
Quality:


Product Quality:

  ISO/IEC 25010:2011:


Process Quality:
```



Product Quality:

Quality in Use:

Ext/Int Product Quality :



Process Quality:
| The turtle diagram shows the elements of a process.

Who:
| Roles,
| Skills, Knowledge,
| Trainings

  depends on --> Name and Description:

How:
| Guidelines, Checklists,
| Templates

  depends on --> Name and Description:

Input:

  depends on --> Name and Description:

Name and Description:

  depends on --> Output:

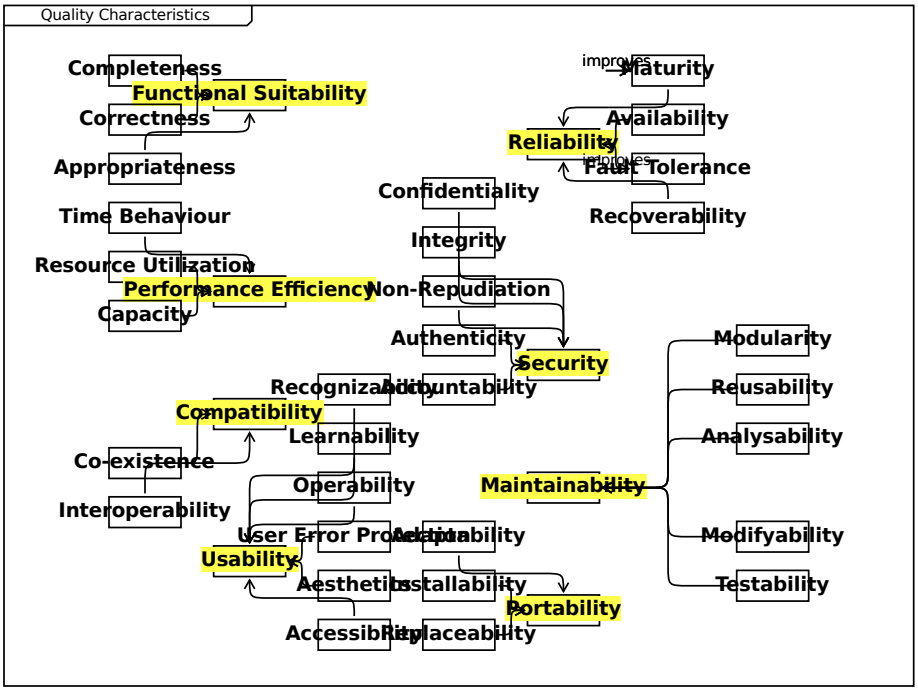
Output:
| Process output,
| Evidence on performed process

With What:

  depends on --> Name and Description:

What Results:

  depends on --> Name and Description:



Quality Characteristics:
| according to ISO 25010

Completeness:

--> Functional Suitability:

Maturity:

--> Reliability:

Functional Suitability:

Correctness:

--> Functional Suitability:

Availability:

--> Reliability:

Reliability:

Appropriateness:

--> Functional Suitability:

Fault Tolerance:

--> Reliability:

Confidentiality:

--> Security:

Time Behaviour:

--> Performance Efficiency:

Recoverability:

--> Reliability:

Integrity:

--> Security:

Resource Utilization:

--> Performance Efficiency:

Performance Efficiency:

Non-Repudiation:

--> Security:

Capacity:

--> Performance Efficiency:

Authenticity:

--> Security:

Modularity:

--> Maintainability:

Security:

Recognizability:

--> Usability:

Accountability:

--> Security:

Reusability:

--> Maintainability:

Compatibility:

Learnability:

--> Usability:

Analysability:

--> Maintainability:

Co-existence:

--> Compatibility:

Operability:

--> Usability:

Maintainability:

Interoperability:

--> Compatibility:

User Error Protection:

--> Usability:

Adaptability:

   --> Portability:

Modifyability:

   --> Maintainability:

Usability:

Aesthetics:

   --> Usability:

Installability:

   --> Portability:

Testability:

   --> Maintainability:

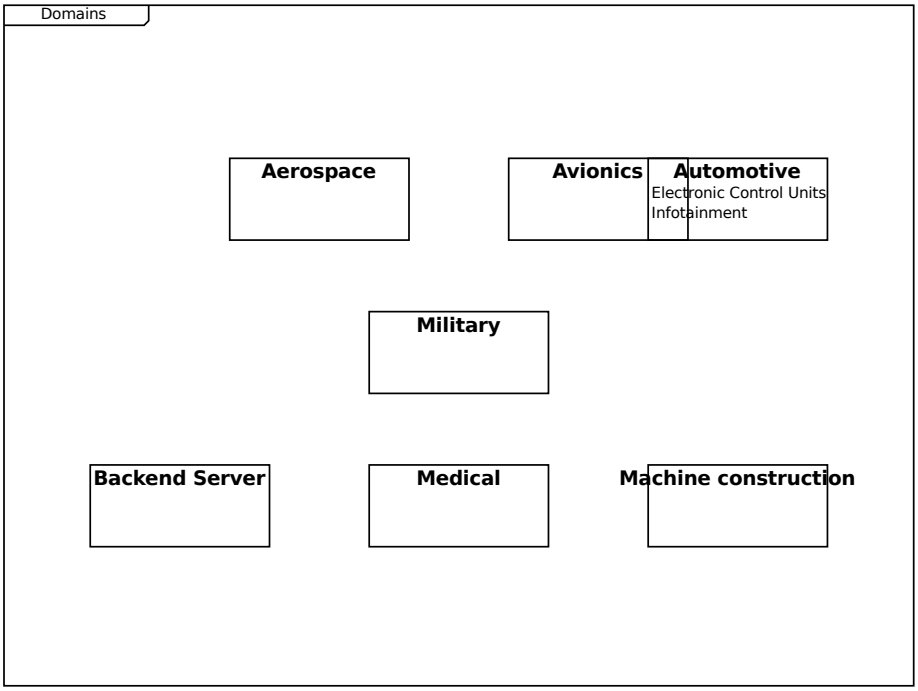
Portability:

Accessibility:

   --> Usability:

Replaceability:

   --> Portability:



Domains:

Aerospace:

Avionics:

Automotive:

  Electronic Control Units:

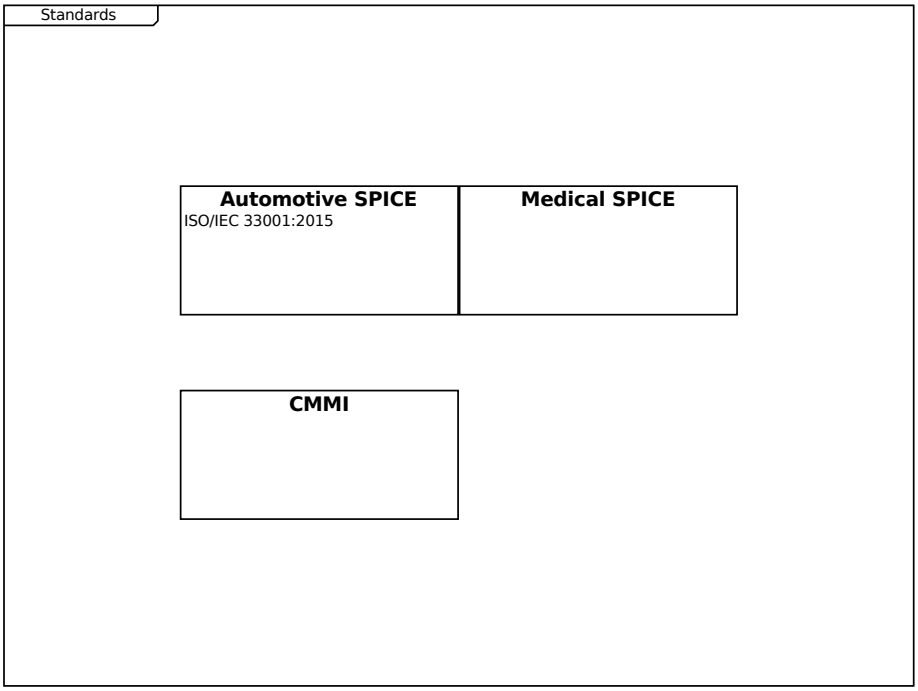
  Infotainment:

Military:

Backend Server:

Medical:

Machine construction:



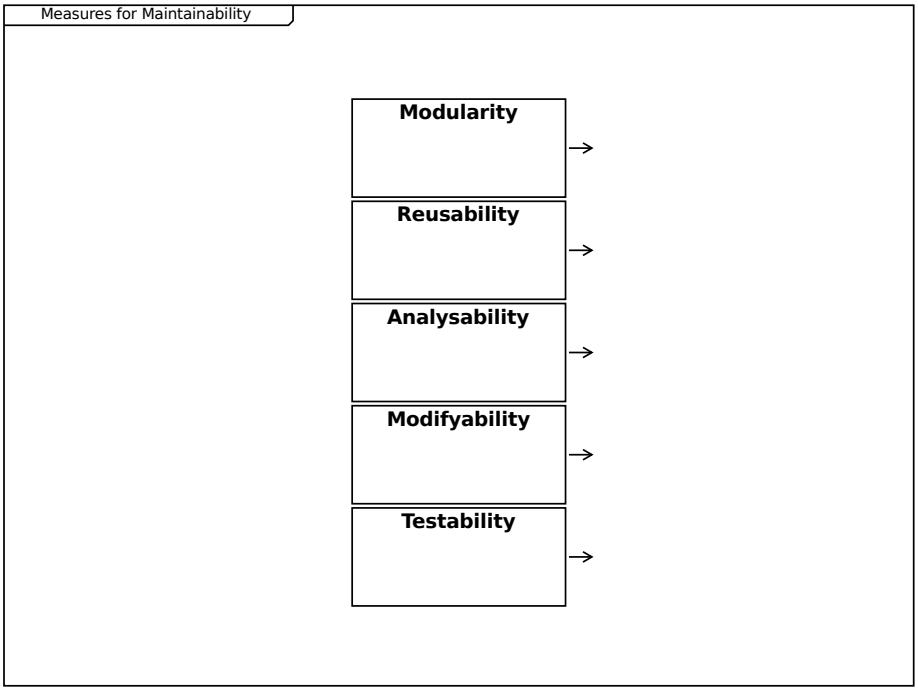
Standards:

Automotive SPICE:

ISO/IEC 33001:2015:

Medical SPICE:

CMMI:



Measures for Maintainability:

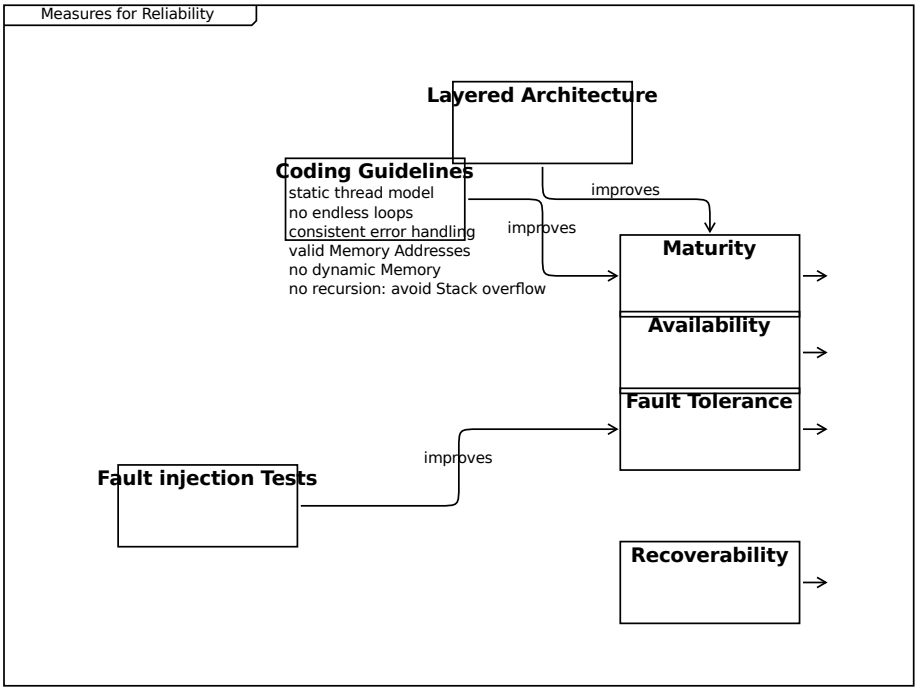
Modularity:

Reusability:

Analysability:

Modifyability:

Testability:



Measures for Reliability:

Layered Architecture:

  improves --> Maturity:

Coding Guidelines:

  static thread model:
  | Execution threads shall not be started/stopped dynamically

  no endless loops:
  | Every loop shall have a counter that ensures that after a predefined maximum value after

  consistent error handling:
  | Inconsistencies in error handling make code vulnerabile to bugs in error handling

  valid Memory Addresses:
  | Only valid memory addresses may be read/written.
  | E.g. Java solves this by prohibiting pointers

```
  no dynamic Memory:

  no recursion: avoid Stack overflow:

  improves --> Maturity:

Maturity:

Availability:

Fault Tolerance:

Fault injection Tests:

  improves --> Fault Tolerance:

Recoverability:
```