Connecting your Coffee-Shop Laptop to a Life-critical System

Jean Arlat, Yves Deswarte, Youssef Laarouchi,
Éric de Nadai, David Powell

57th IFIP 10.4 working group meeting, Ishigaki, Japan, 21-25 January 2010

LAAS-CNRS

AIRBUS

Outline

- Introduction
- Levels of confidence
- Multi-level confidence models
- Platform virtualization
- Laptop prototype
- Conclusion

Outline

- Introduction
- Levels of confidence
- Multi-level confidence models
- Platform virtualization
- Laptop prototype
- Conclusion

Case study 1: electronic flight-book

Take-off computation

Onboard computer

Airport

Pilot

BRITISH AIRWAYS

الخطوط التونسية TUNISAIR

AIR FRANCE

Airline Company

Onboard equipment

Case study 1: electronic flight-book

Take-off computation

Onboard computer

Airport

Pilot

BRITISH AIRWAYS

الخطوط التونسية TUNISAIR

AIR FRANCE

Compagnies aériennes

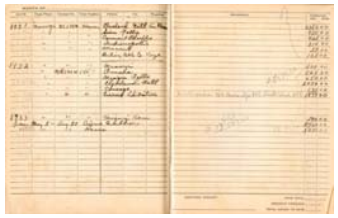
Airline Company

Onboard equipment

Case study 2: maintenance laptop

Pilot

Onboard equipment



Flight logbook

Maintenance engineer

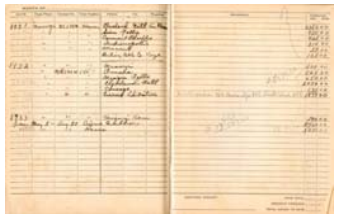
Maintenance terminal

Paper manuals

Electronic manuals

Case study 2: maintenance laptop

Pilot



Flight logbook

Maintenance engineer

Onboard equipment

Maintenance terminal

Paper manuals

Maintenance laptop

Motivations

- **Less manual intervention**
  - reduce stopover time and delays
- **Laptop**
  - flexibility and convenience
  - single mobile interface
- **COTS hardware and operating system**
  - economic
  - genericity and flexibility

Enabling technologies

- **Totel *et al*'s "multi-level integrity" model [FTCS-28]**
  - framework for executing tasks of different criticality levels in a single system
  - requires a trusted computing base (TCB) to isolate levels and mediate the flow of data
  - applies fault-tolerance techniques to allow data to flow from low levels to higher levels
- **Platform virtualization techniques**
  - provide isolation and mediation between virtual machines
  - attractive approach for implementing TCB

Outline

- Introduction
- **Levels of confidence**
- Multi-level confidence models
- Platform virtualization
- Laptop prototype
- Conclusion

Criticality and confidence

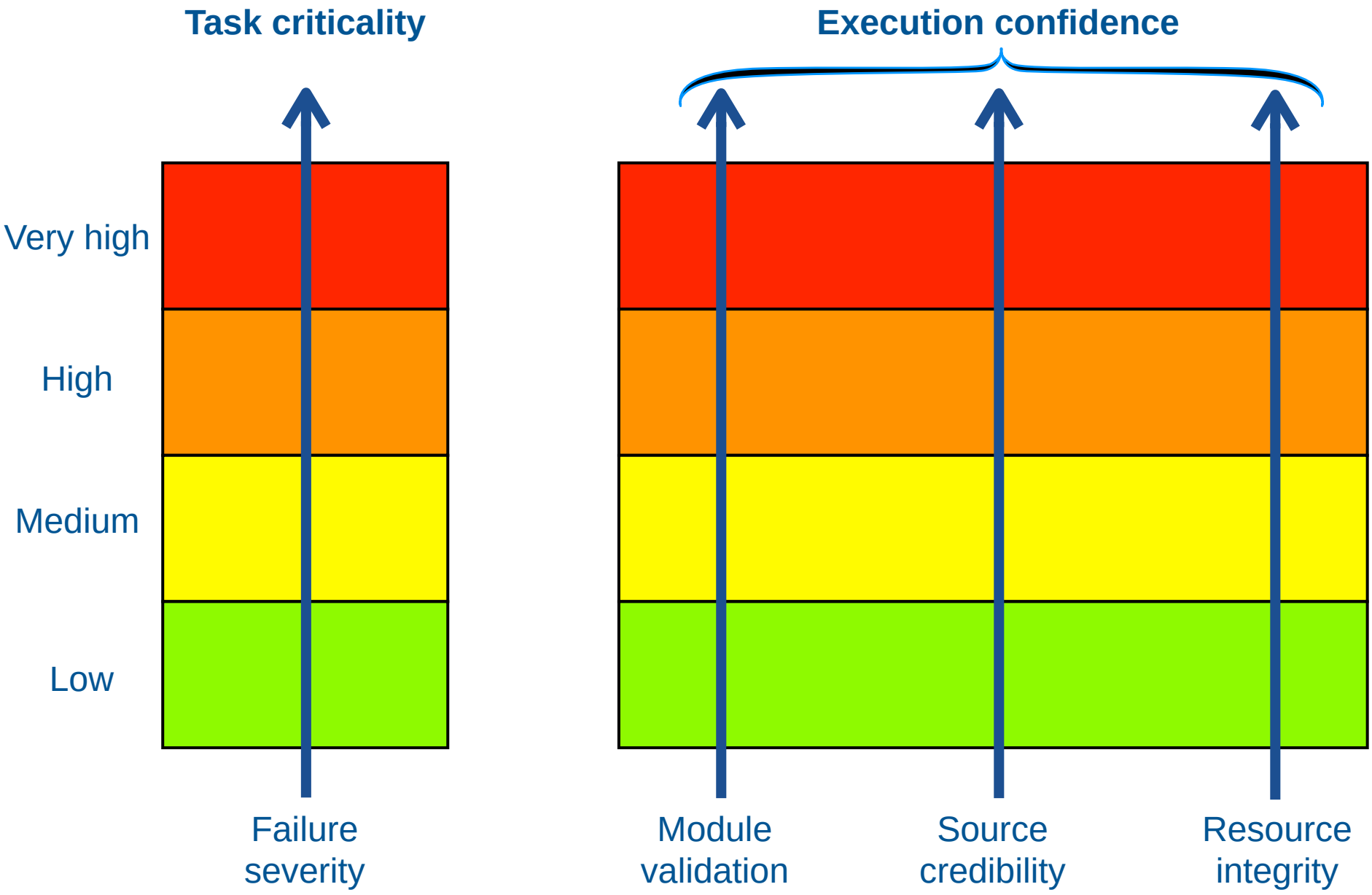
- **Criticality**
  - ↗ Severity of task failure ⇒ ↗ Criticality of task
  - Categorize severity (& criticality) in discrete levels according to consequence of failure

    e.g., none, minor, major, dangerous, catastrophic
- **Confidence**
  - ↗ Criticality of task ⇒ ↗ Confidence in task execution
  - Convenient to categorize confidence in discrete levels that correspond with levels of criticality

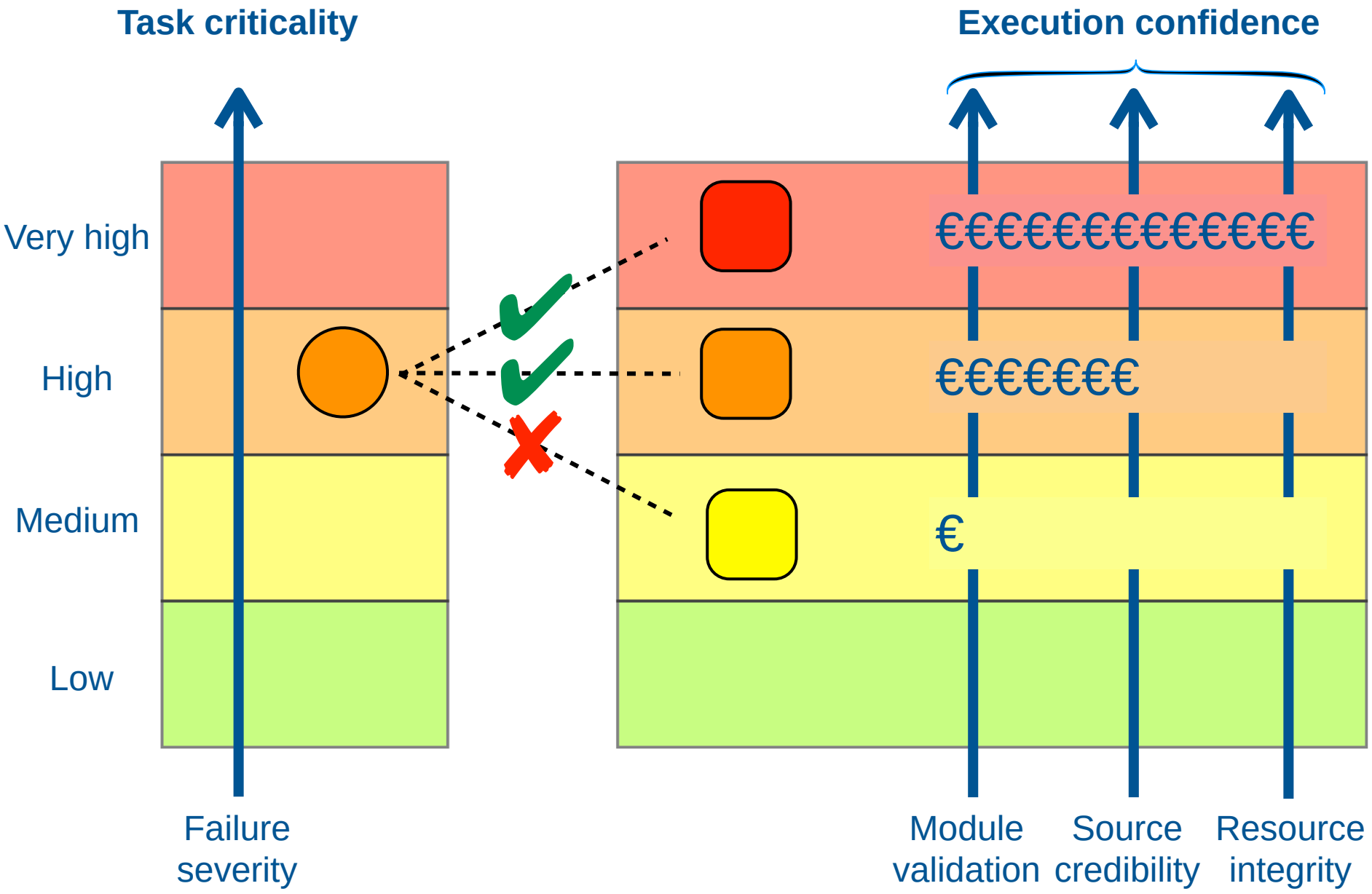
Confidence attributes

- **Validation (of a module)**
  - effort deployed in assuring that a module meets its specifications
    - e.g., DO-178B for software, DO-254 for hardware
- **Credibility (of sources)**
  - belief in source(s) of data input to a module
    - e.g., expertise of human operator
    - e.g., reliability and accuracy of data sensor
- **Integrity (of resources)**
  - degree of trust that module's code, data and other resources, are free from corruption

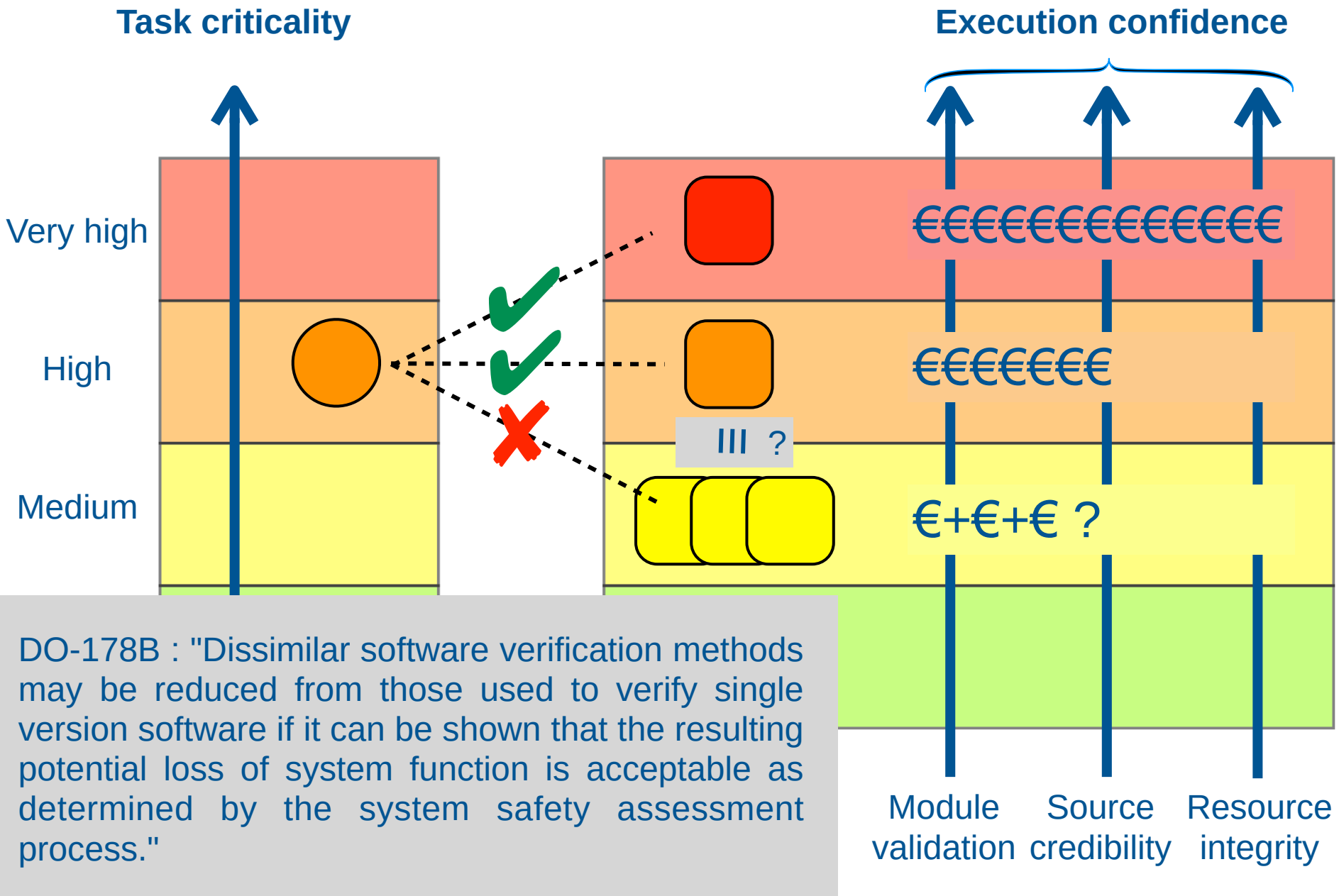
Levels of criticality and confidence



Levels of criticality and confidence



Levels of criticality and confidence



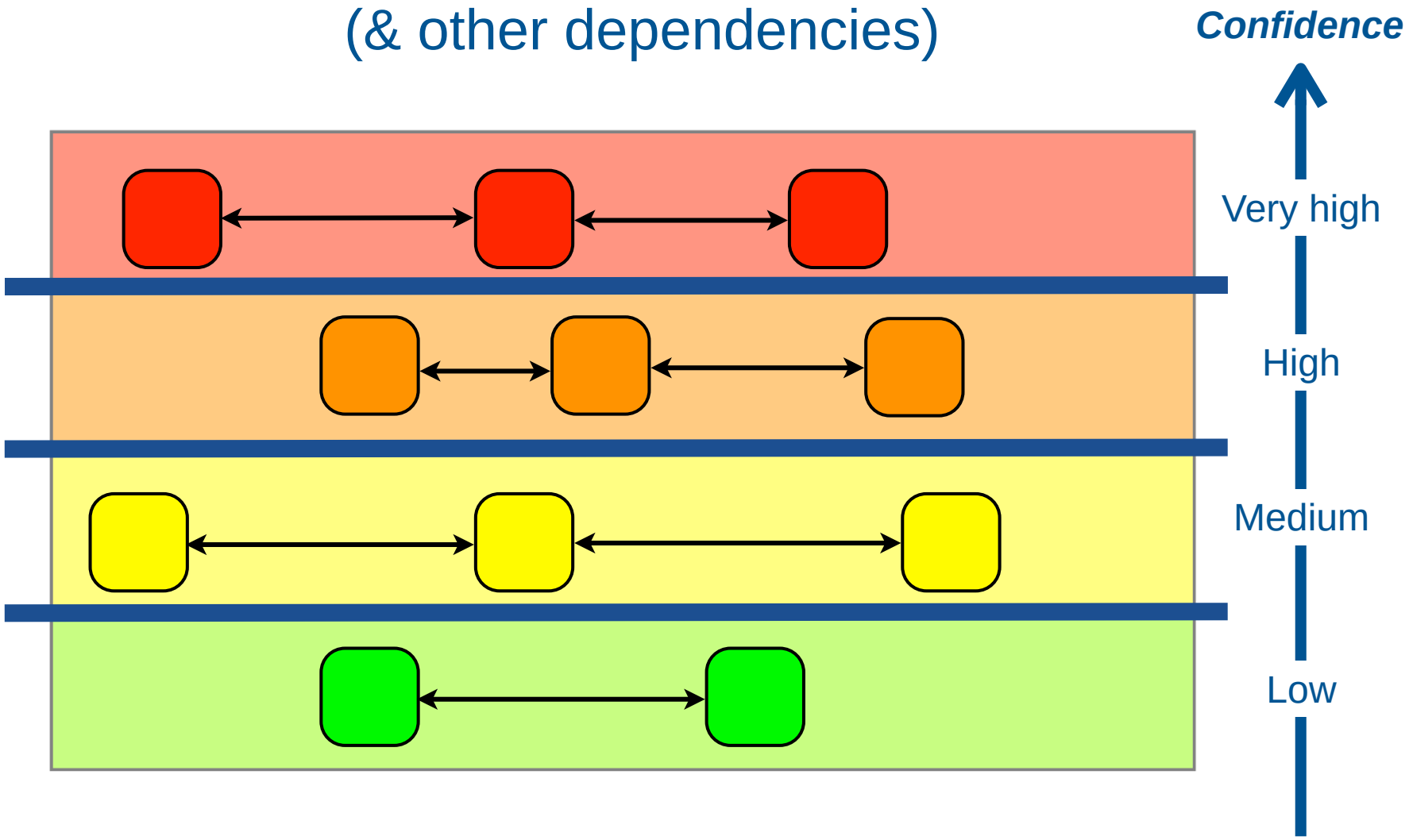
DO-178B : "Dissimilar software verification methods may be reduced from those used to verify single version software if it can be shown that the resulting potential loss of system function is acceptable as determined by the system safety assessment process."

Outline

- Introduction
- Levels of confidence
- **Multi-level confidence models**
- Platform virtualization
- Laptop prototype
- Conclusion

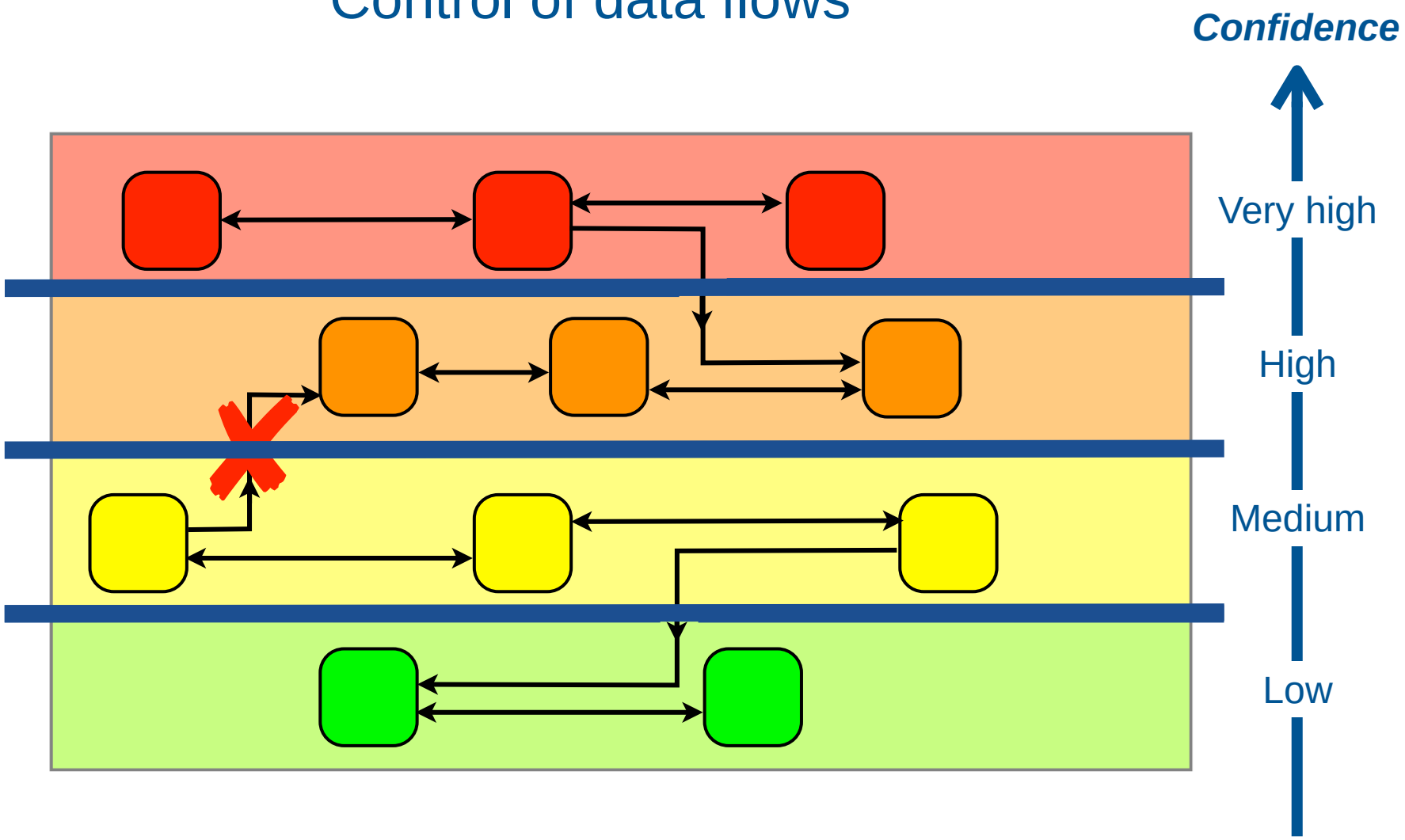
Isolation

Separation of data flows
(& other dependencies)



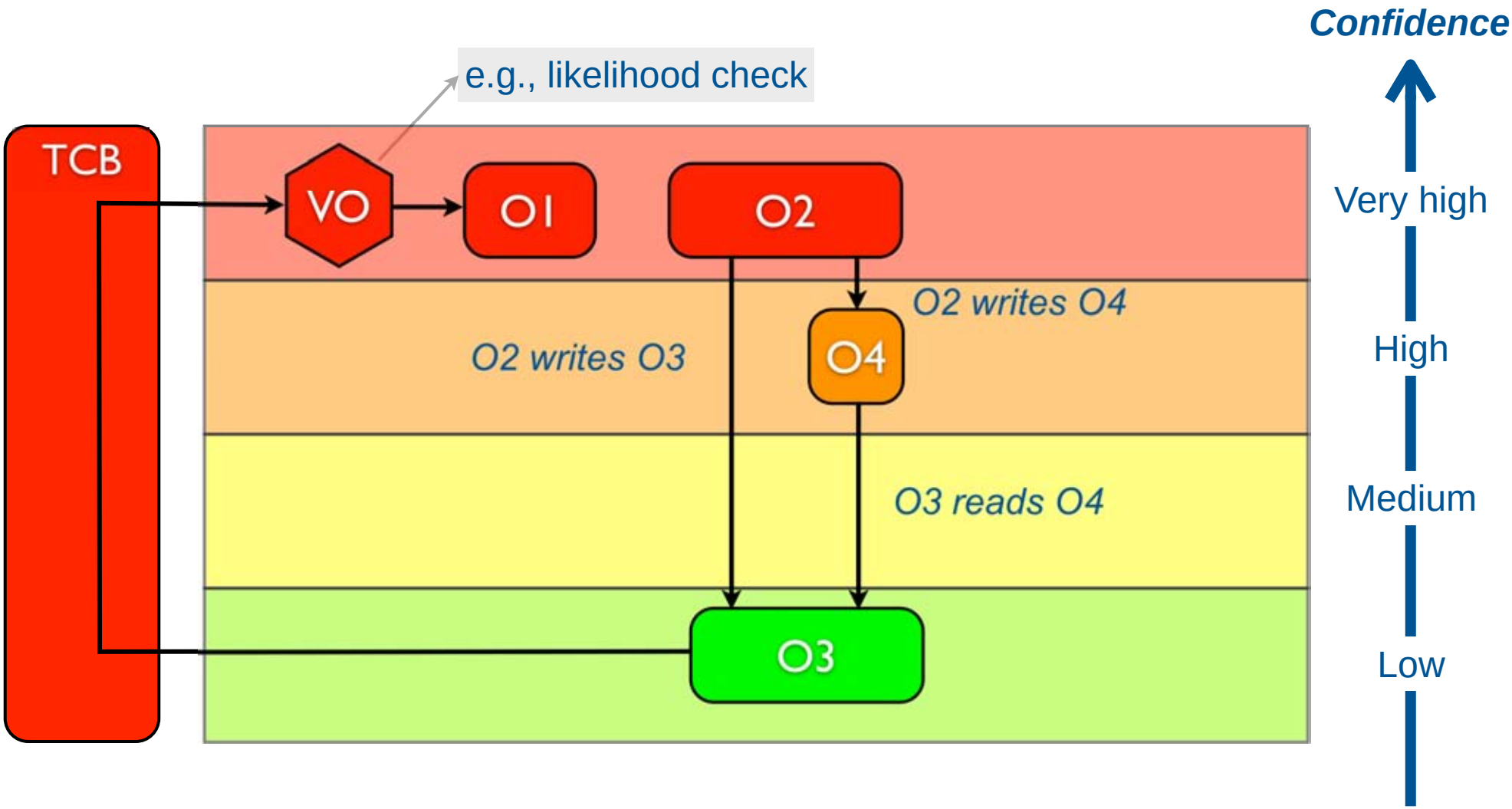
Mediation

Control of data flows



Totel's model

Allows controlled updward data flows

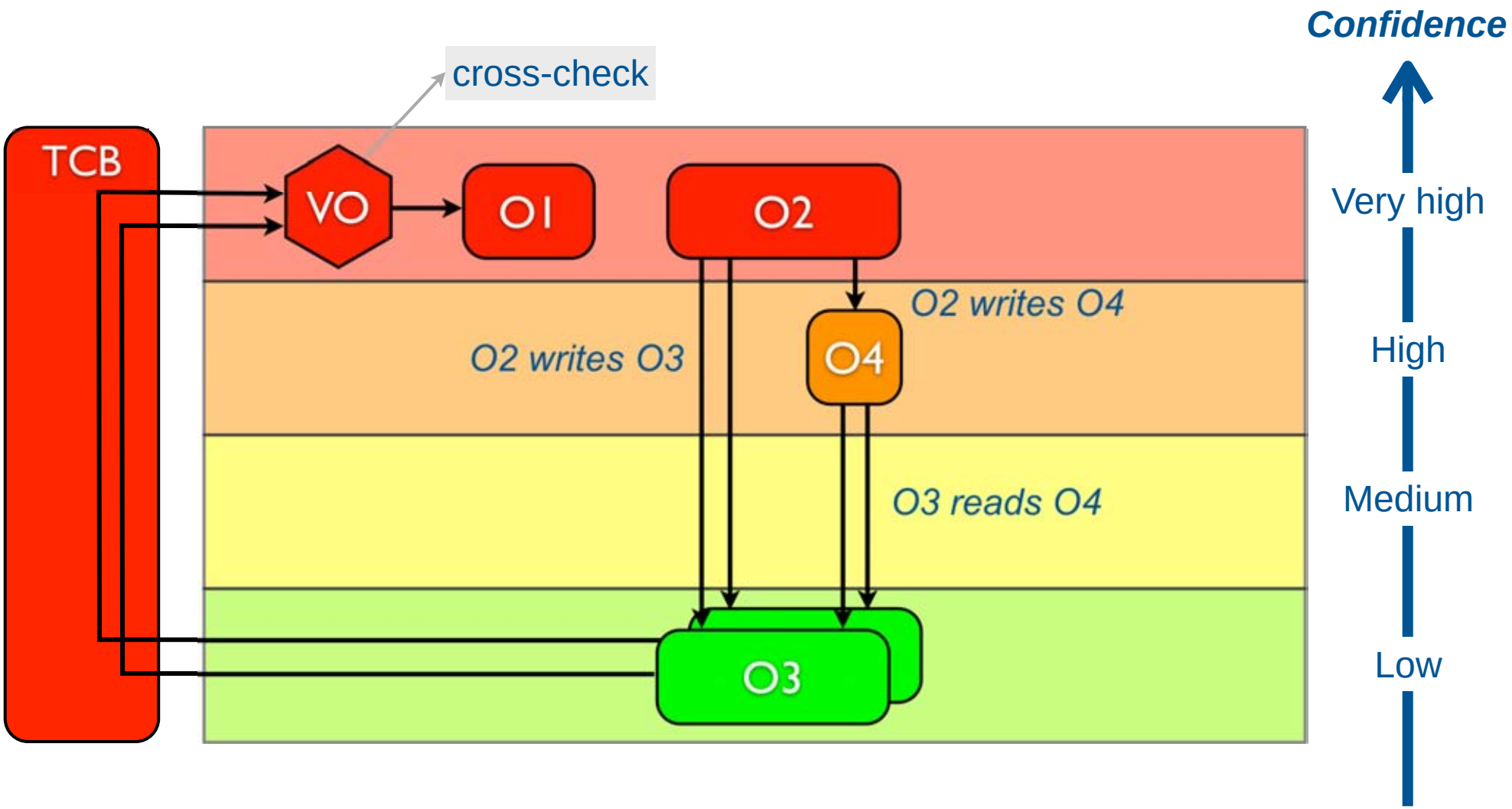


TCB: Trusted Computing Base

VO: Validation Object

Totel's model

Allows controlled updward data flows

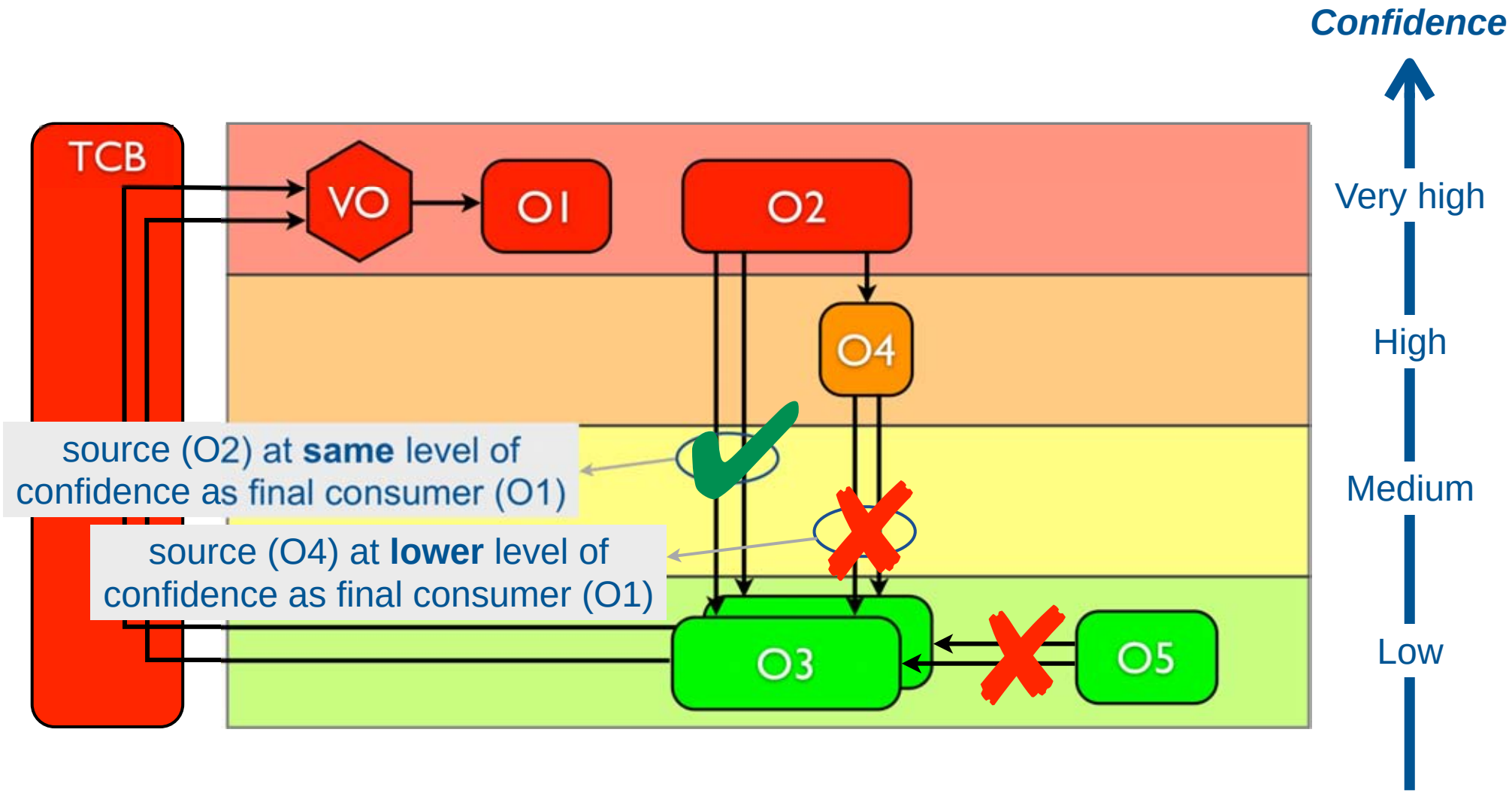


TCB: Trusted Computing Base

VO: Validation Object

Common sources

Potential common-mode fault?

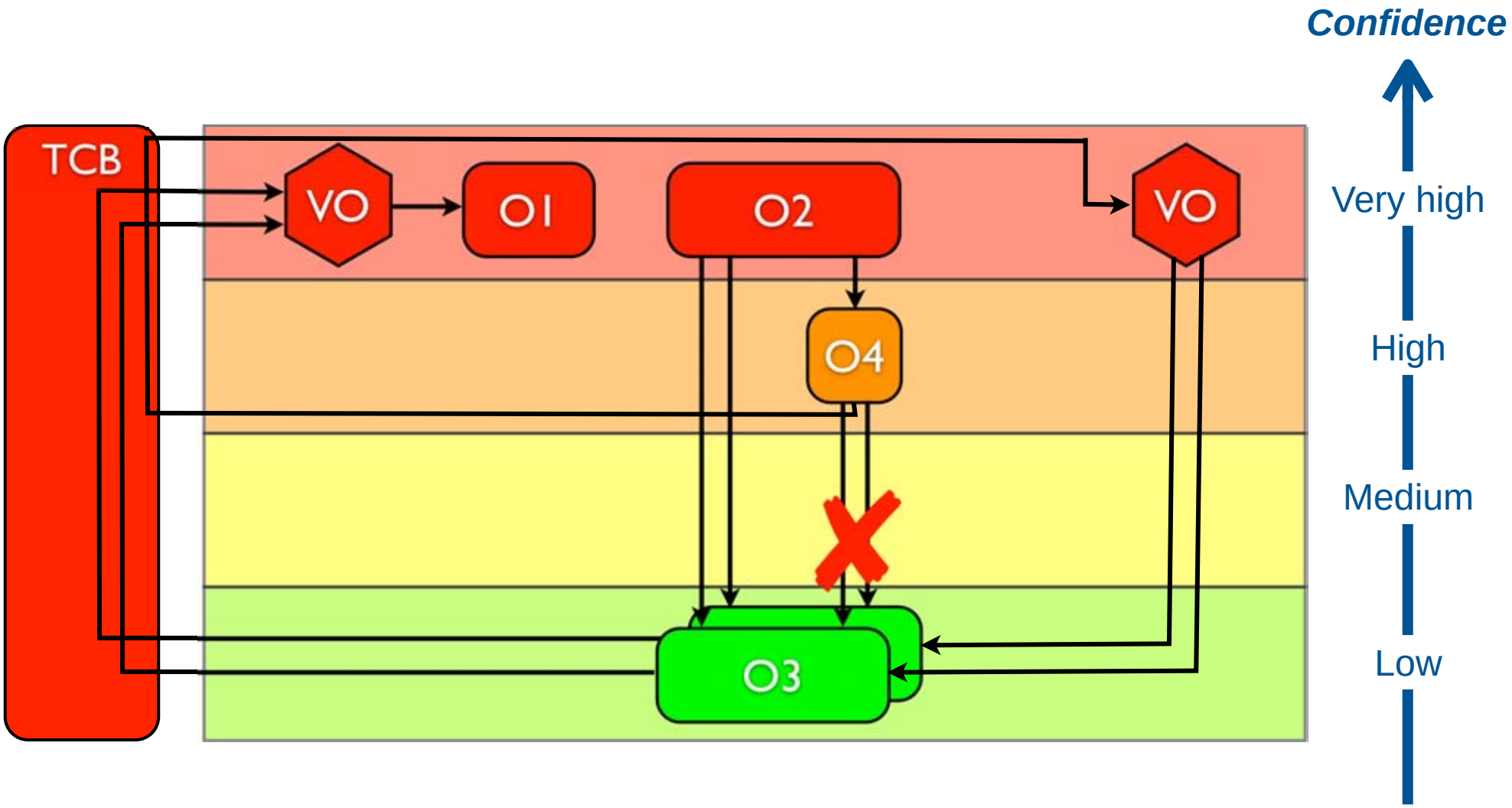


TCB: Trusted Computing Base

VO: Validation Object

Common sources

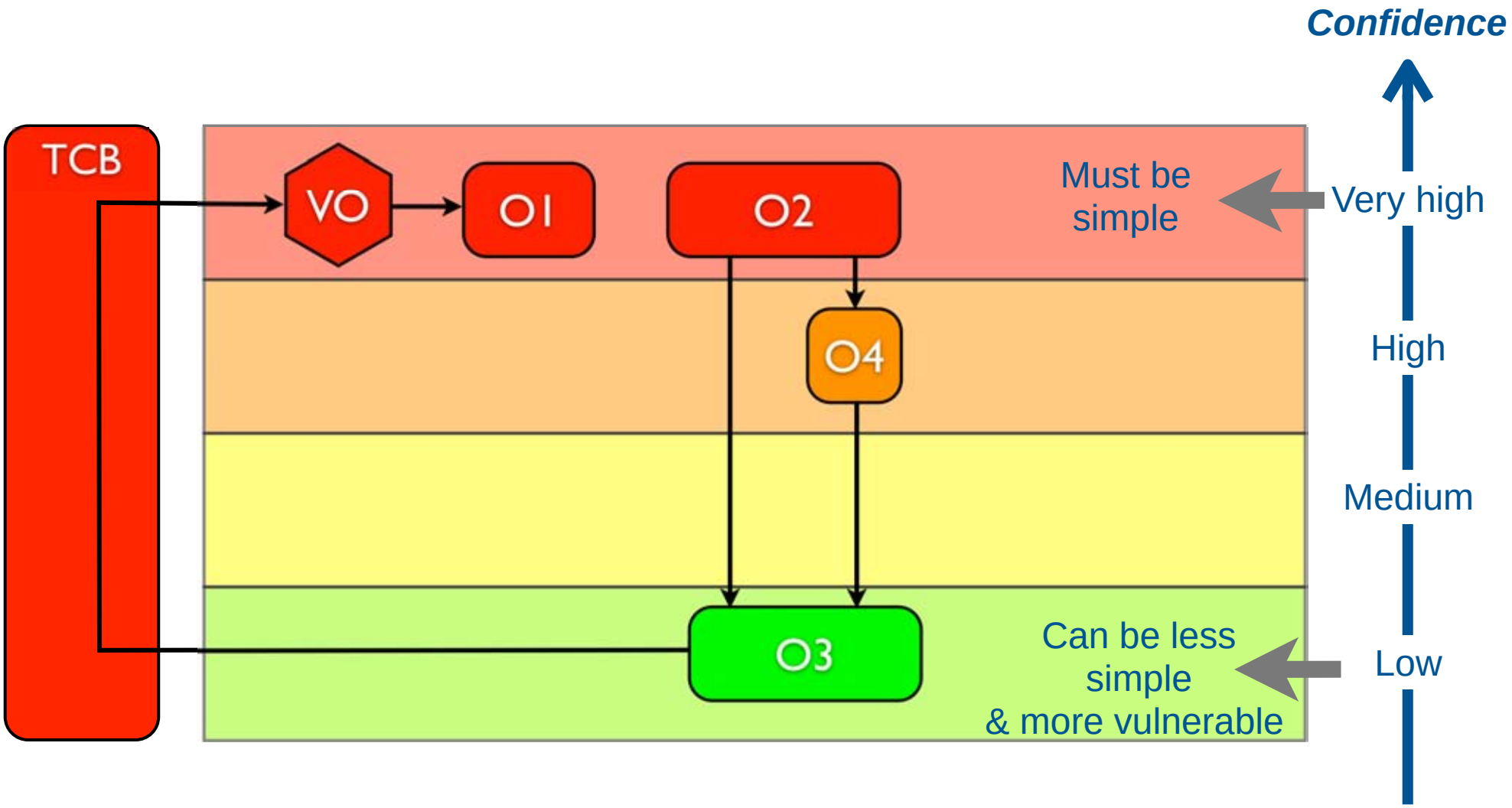
Potential common-mode fault



TCB: Trusted Computing Base

VO: Validation Object

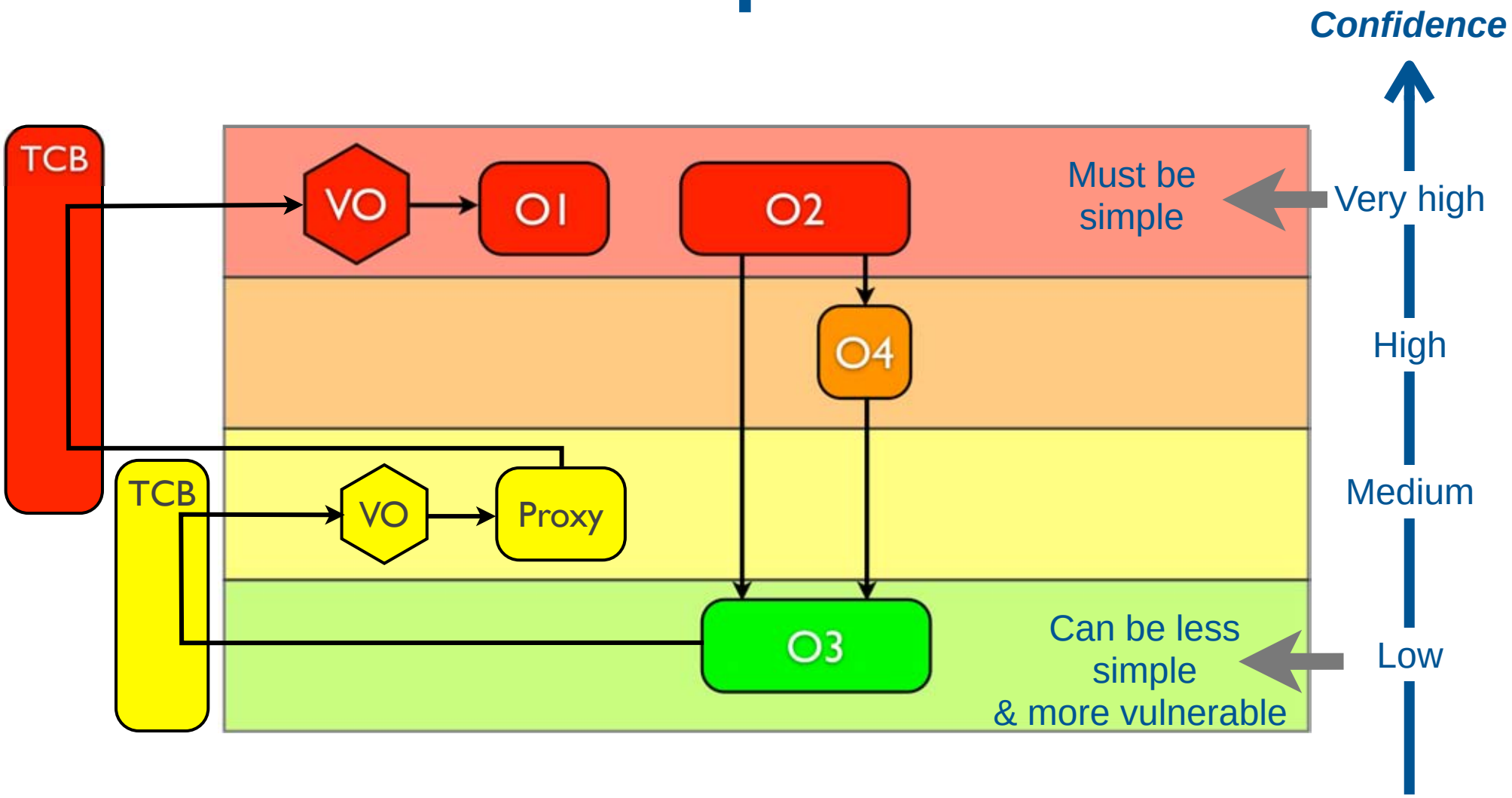
Bridging the complexity gap...



TCB: Trusted Computing Base

VO: Validation Object

Bridging the complexity gap...
...with proxies



TCB: Trusted Computing Base

VO: Validation Object

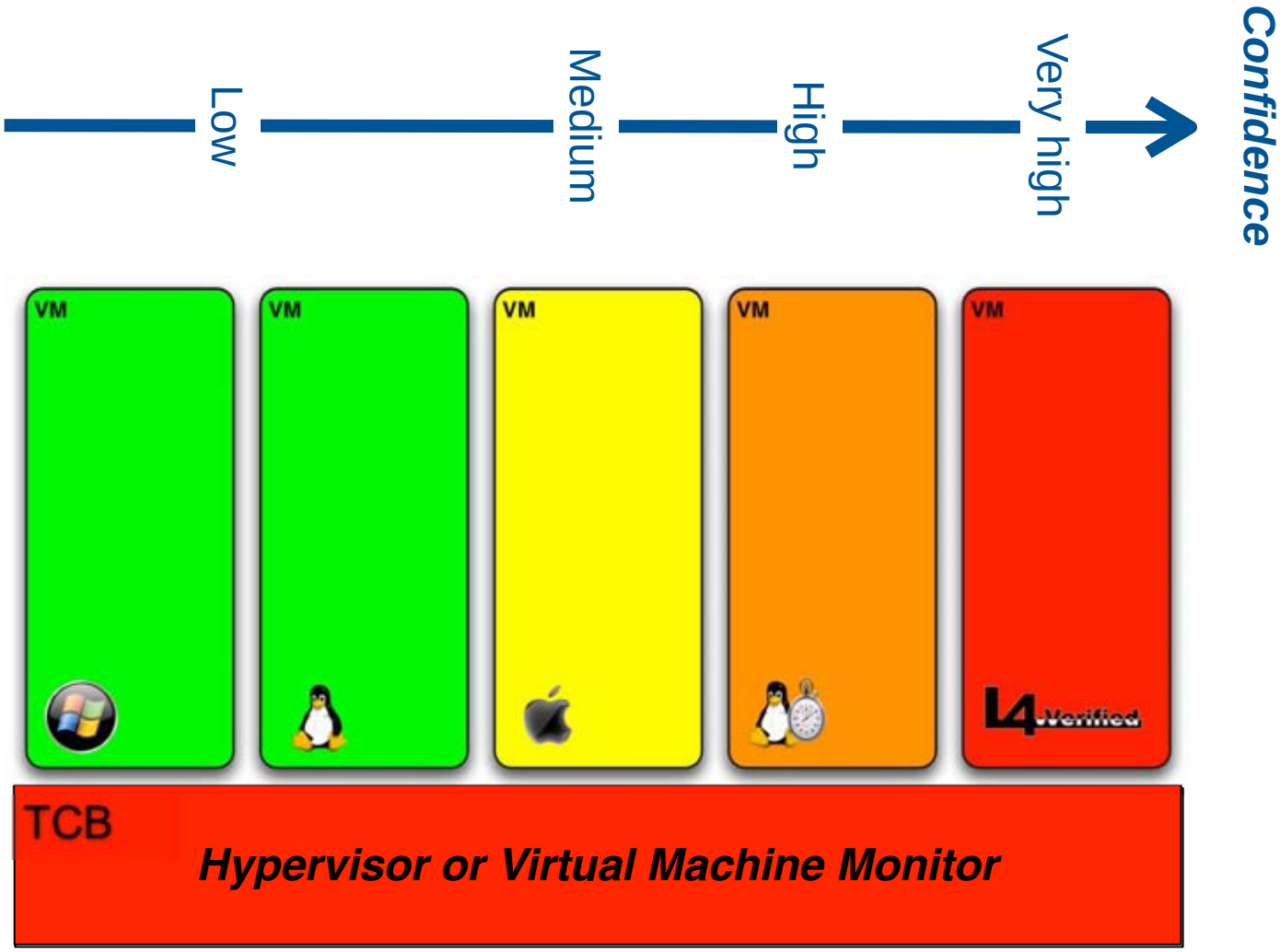
TCB implementation

- **Totel prototypes (1998)**
  - CORBA-compliant middleware
  - Micro-kernel
- **Current work**
  - Hypervisor

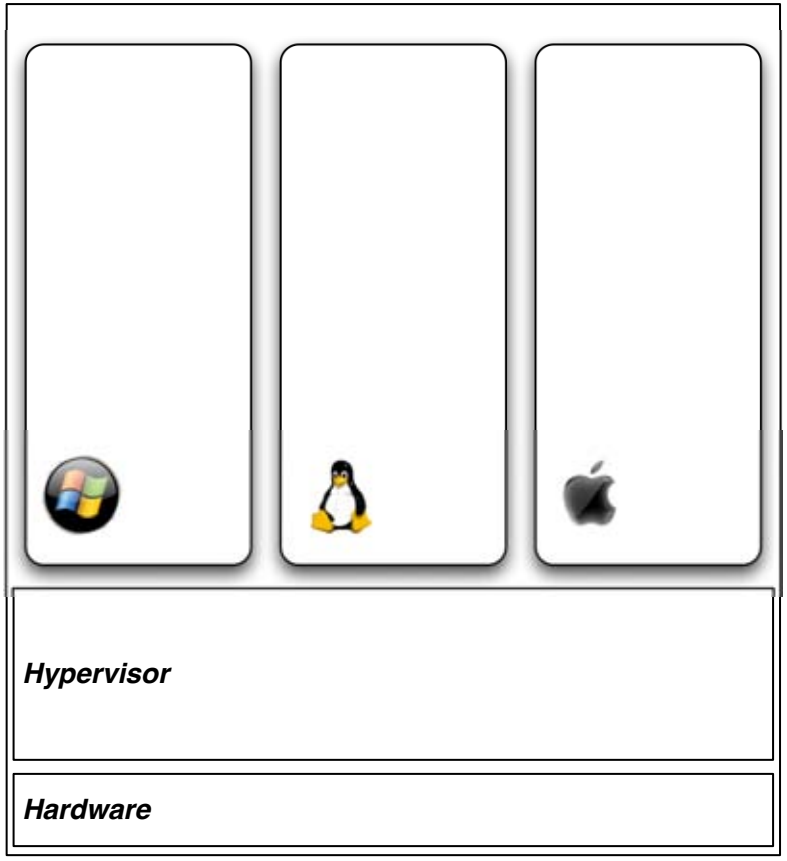
Outline

- Introduction
- Levels of confidence
- Multi-level confidence models
- **Platform virtualization**
- Laptop prototype
- Conclusion

Platform virtualization

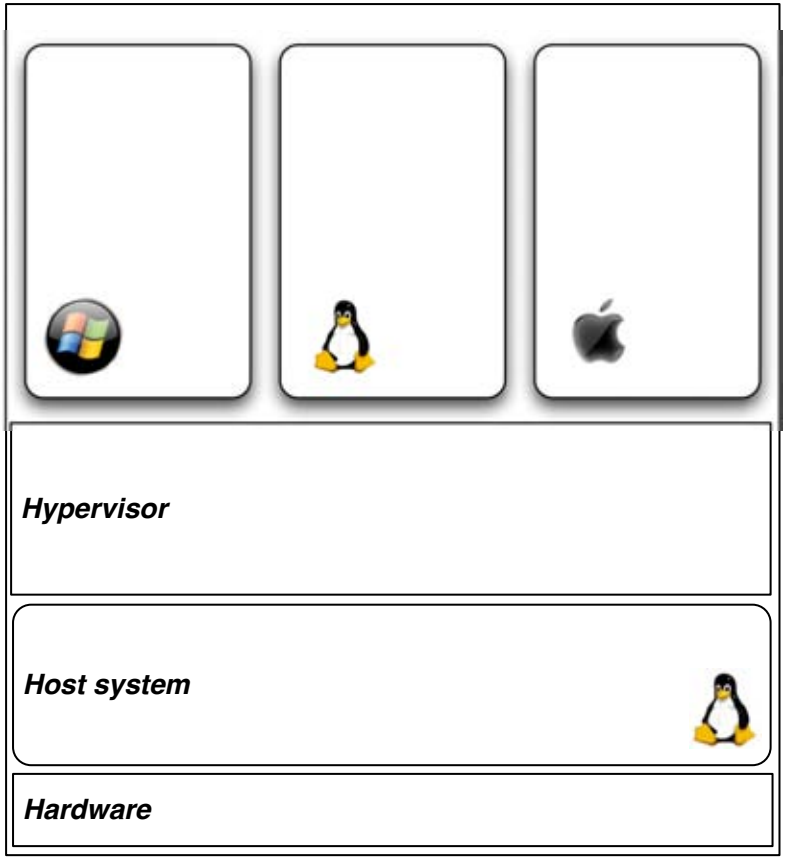


Virtualization techniques



Type 1

e.g., Xen



Type 2

e.g., VMware

Some certified hypervisors

- **Polyxène**
  - Bertin Technologie
  - CC EAL 5 certification
- **LynxSecure**
  - LynuxWorks
  - "Designed to CC EAL-7 and DO-178B level A"
- **INTEGRITY Secure Virtualization**
  - Green Hills Software, Inc.
  - "Built on the world's only CC EAL6+ High-Robustness-certified OS technology"
    - (INTEGRITY-178B separation kernel certified to CC EAL-6+)

Outline

- Introduction
- Levels of confidence
- Multi-level confidence models
- Platform virtualization
- **Laptop prototype**
- Conclusion

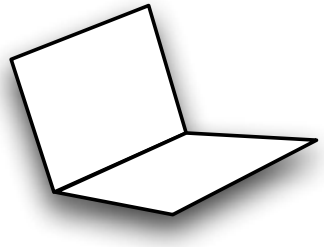
Connecting a laptop

Flight management

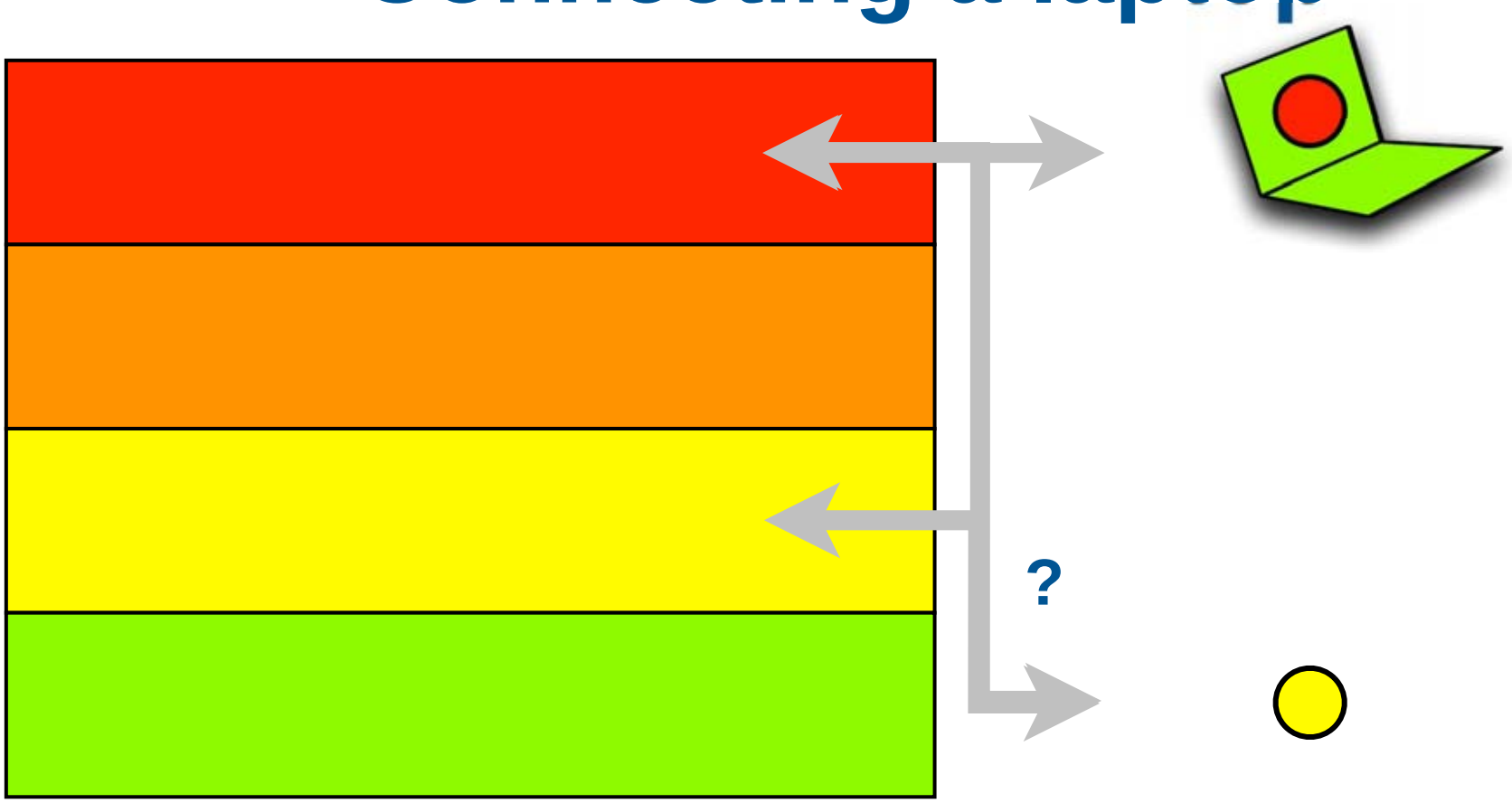
Aircraft management

Aircraft information system

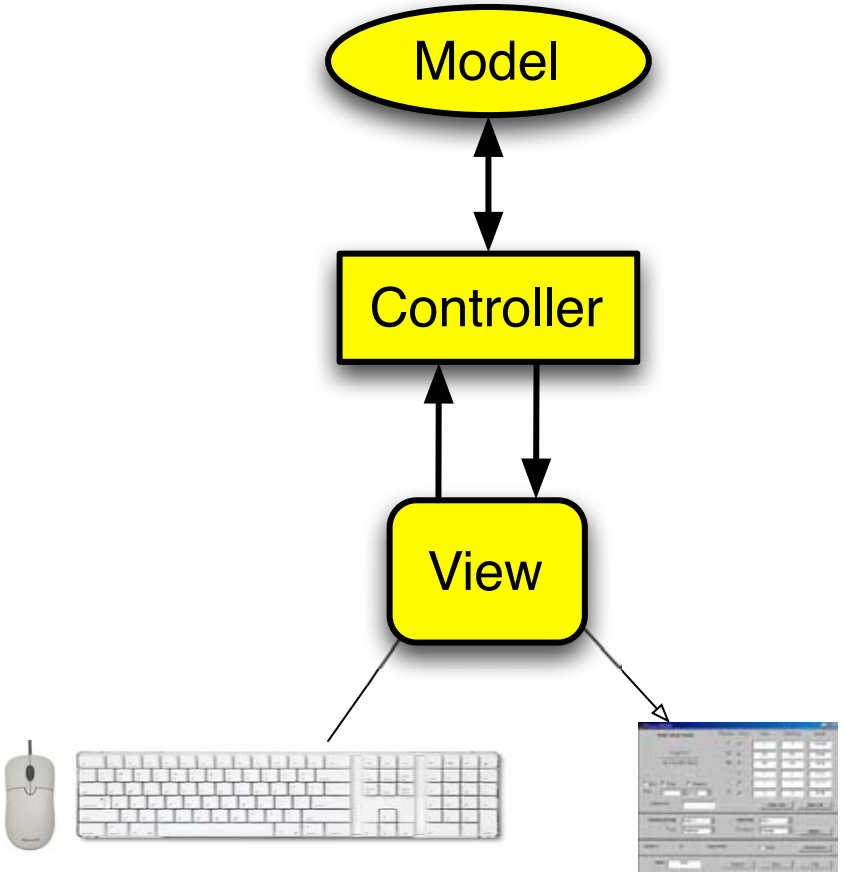
"Off-board"



Connecting a laptop



Connecting a laptop



MVC design pattern for HMI

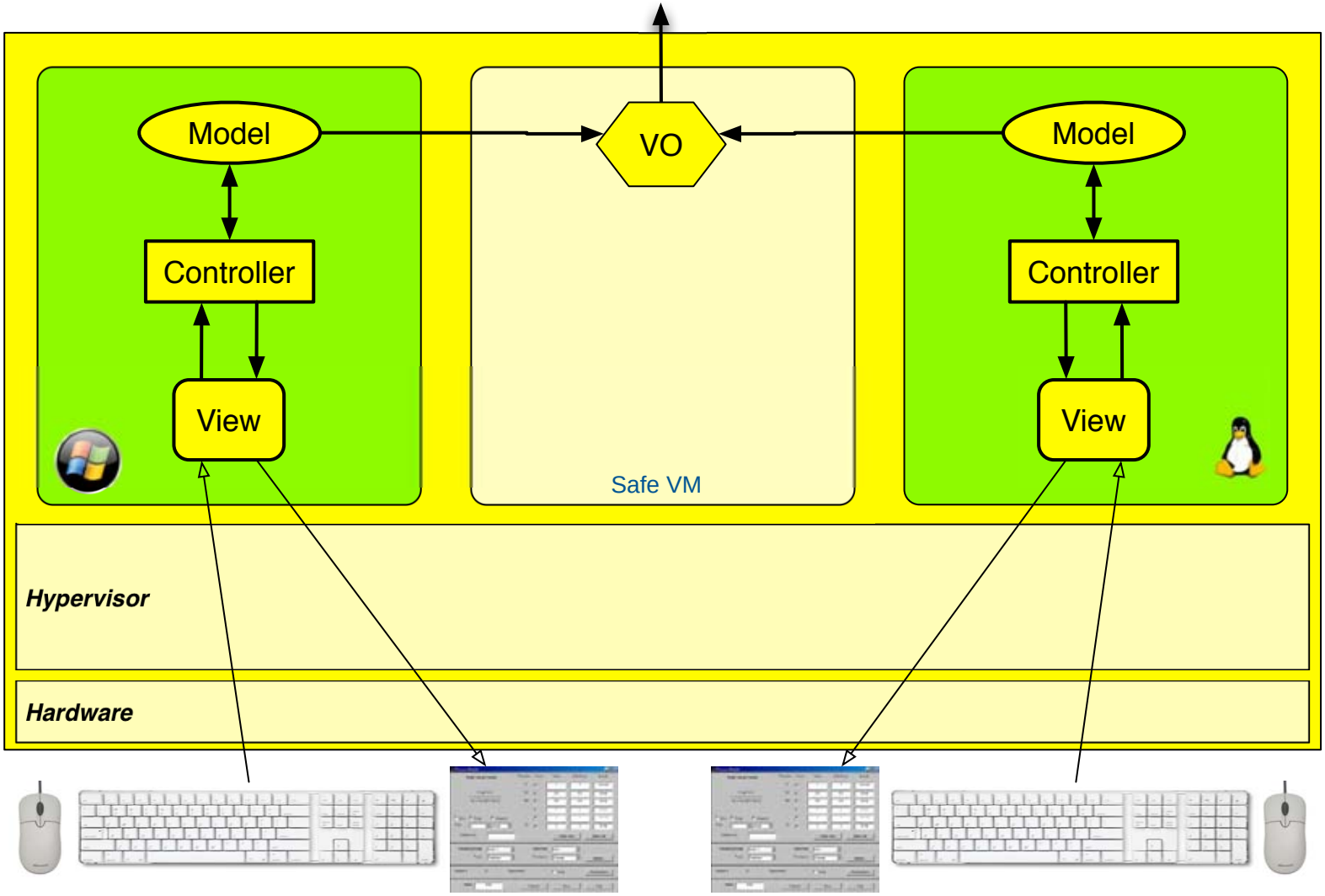
- **View**

  Visual presentation
- **Controller**

  Logic (responses to user events)
- **Model**

  Back-end database

Diverse OS's with virtualization



Solution 1 : custom-bred operator...

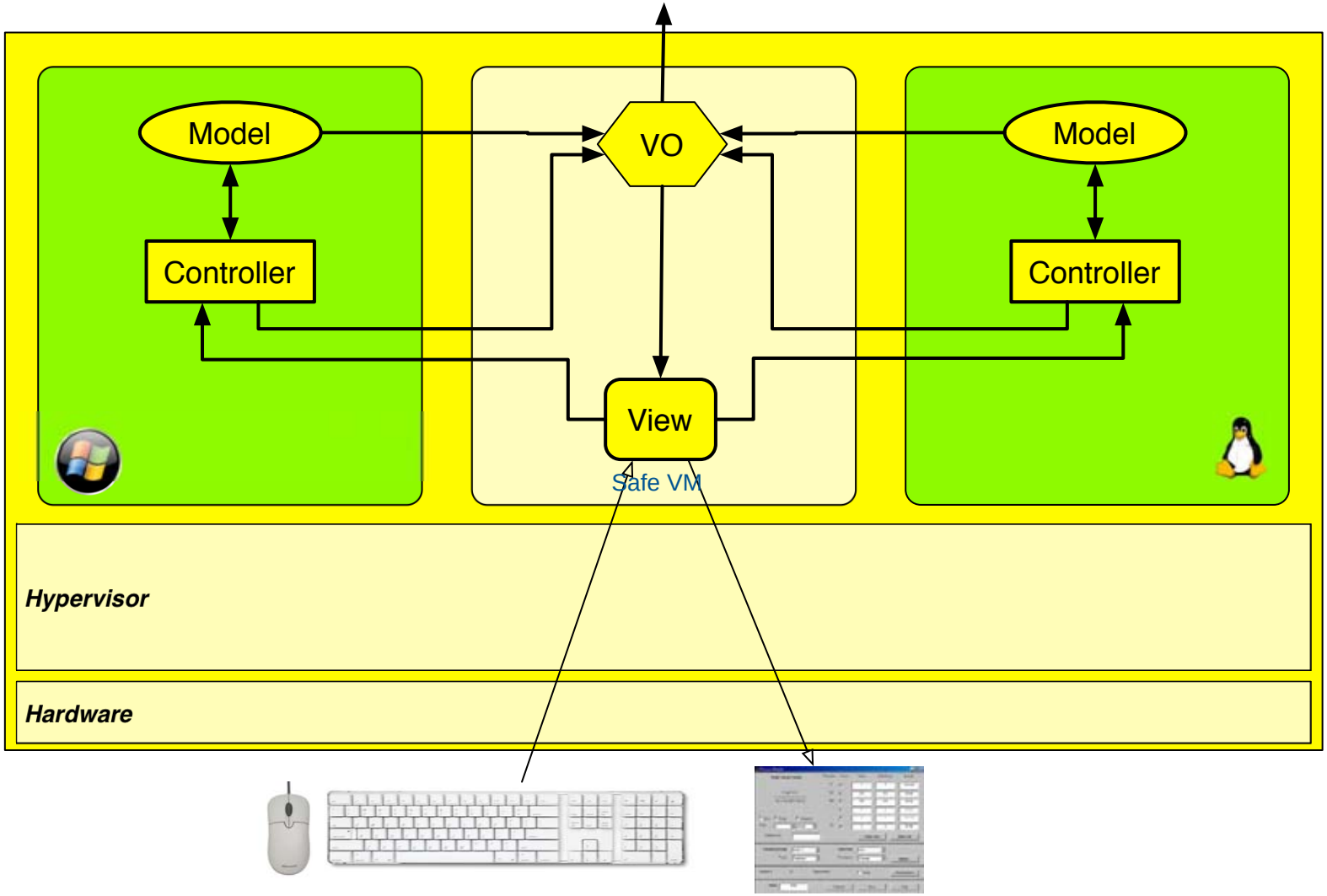
Vishnu

+

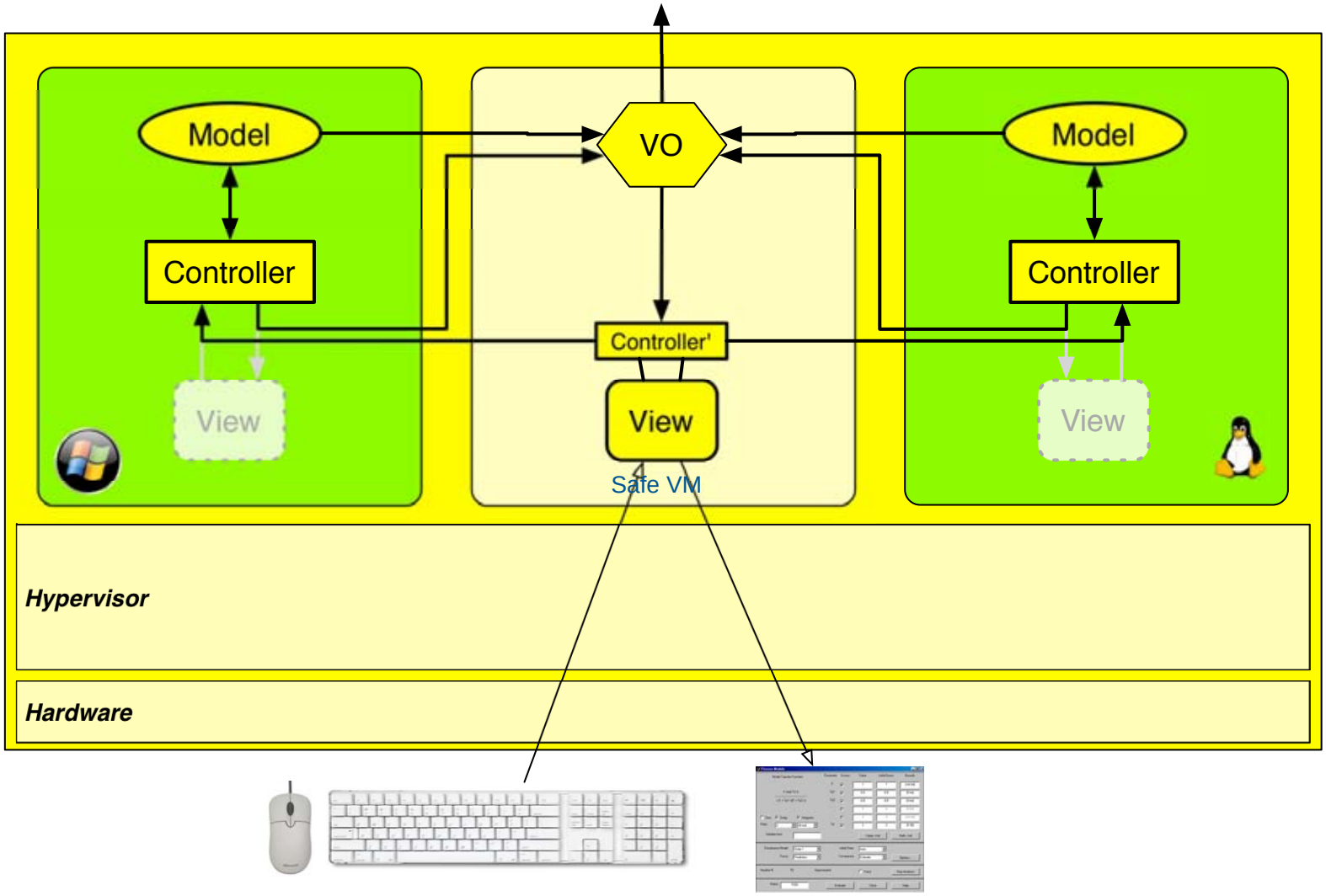
Janus

= ?

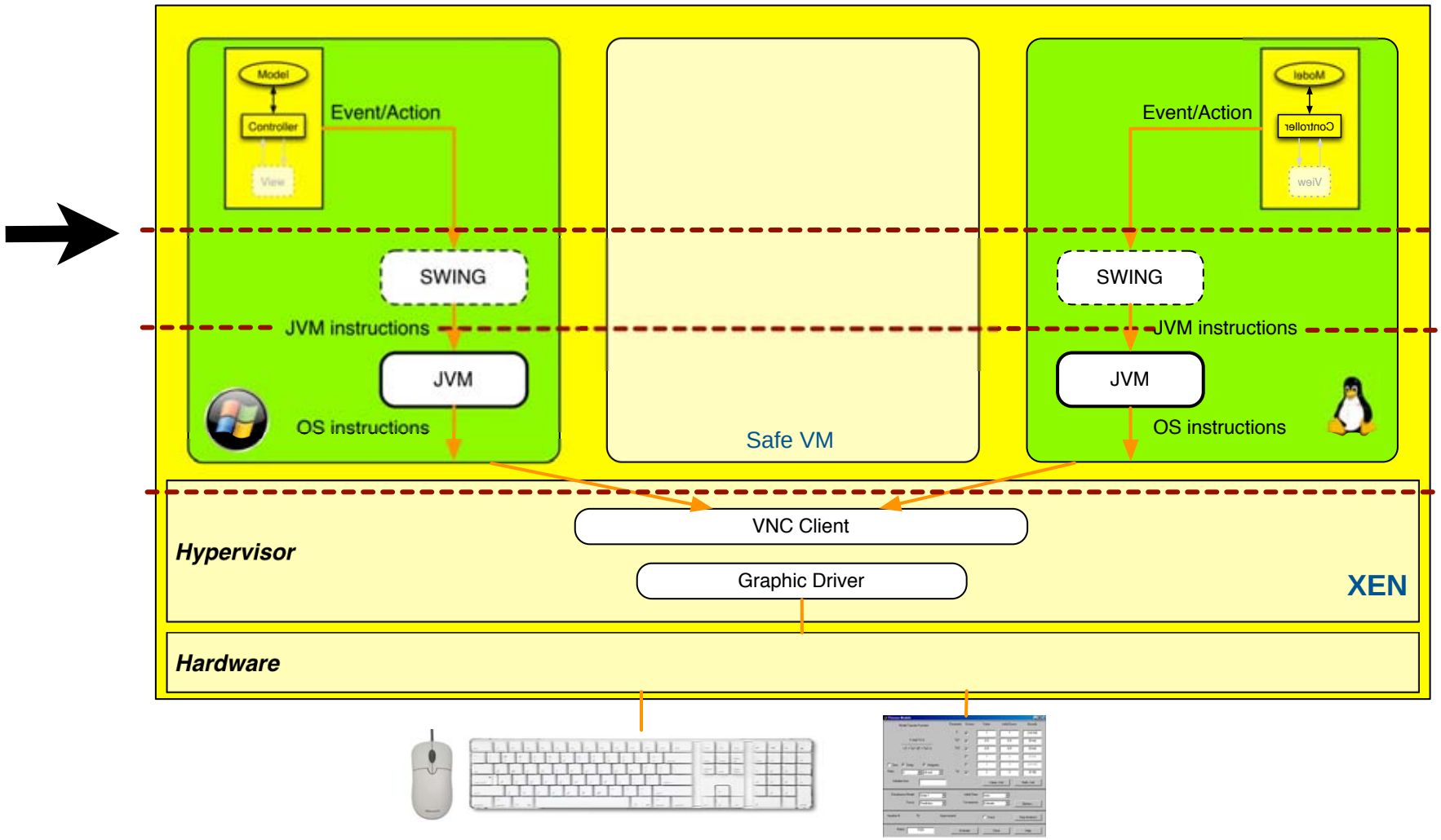
Solution 2 : custom-built software



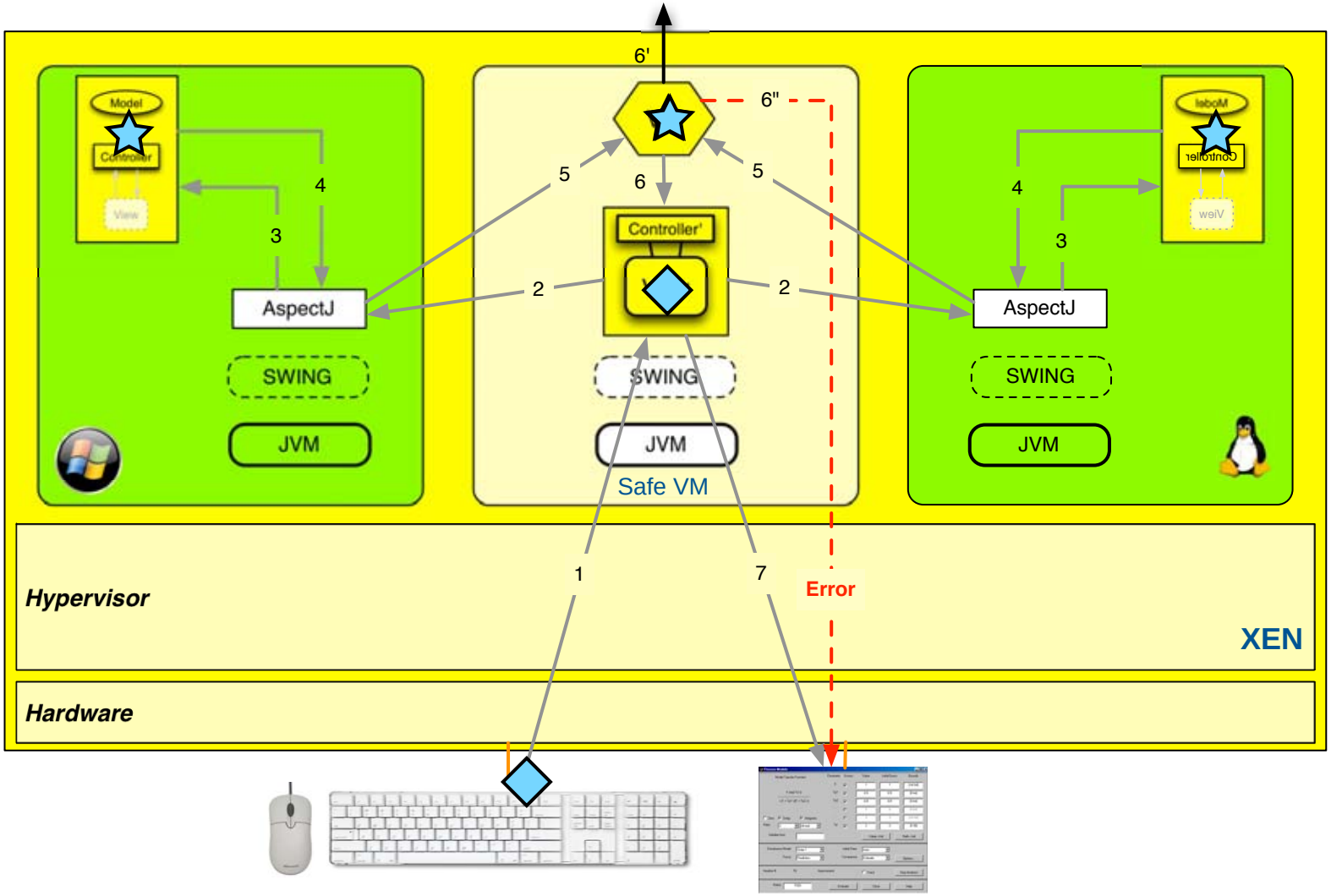
Solution 3 : I/O interception



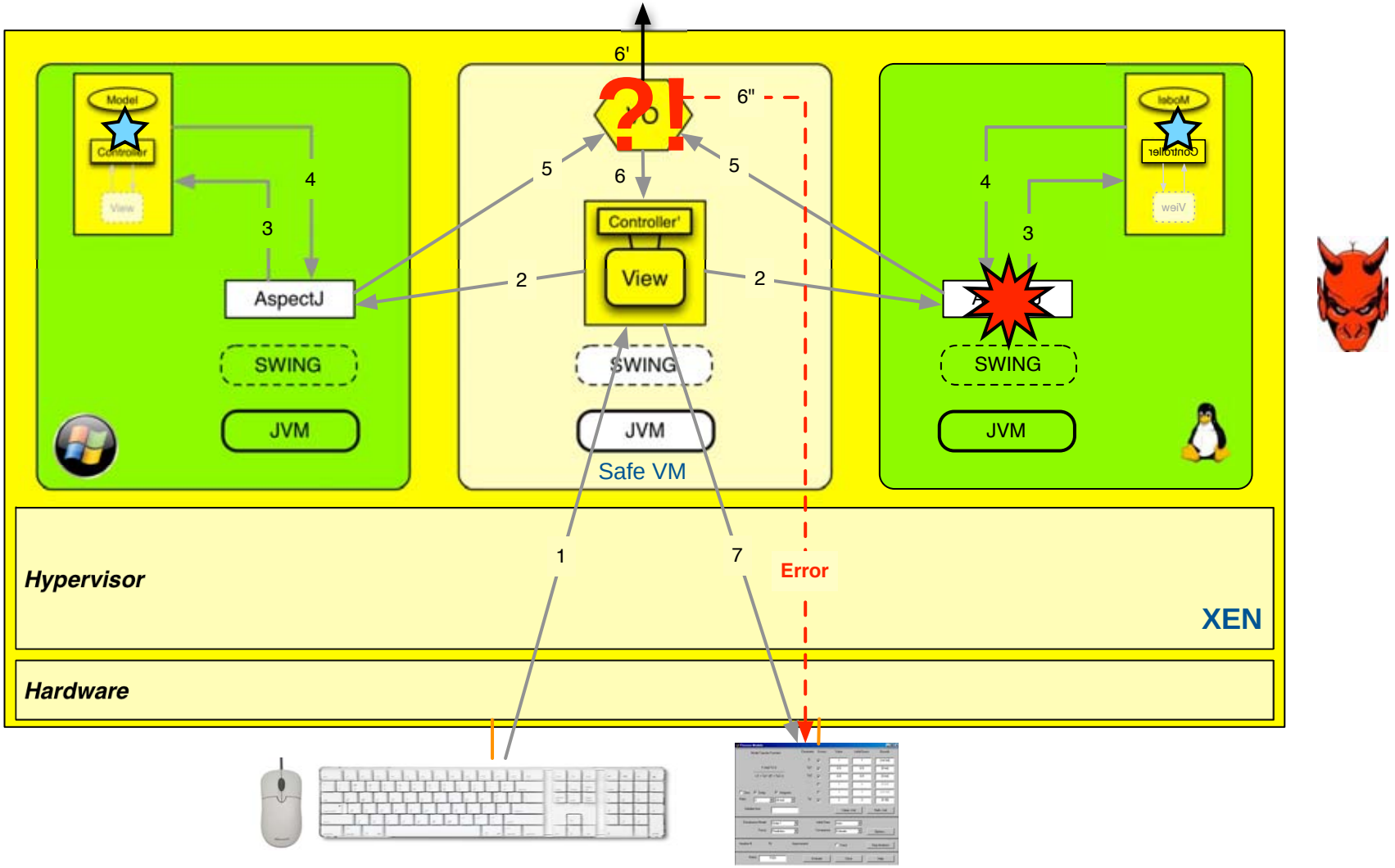
Interception options



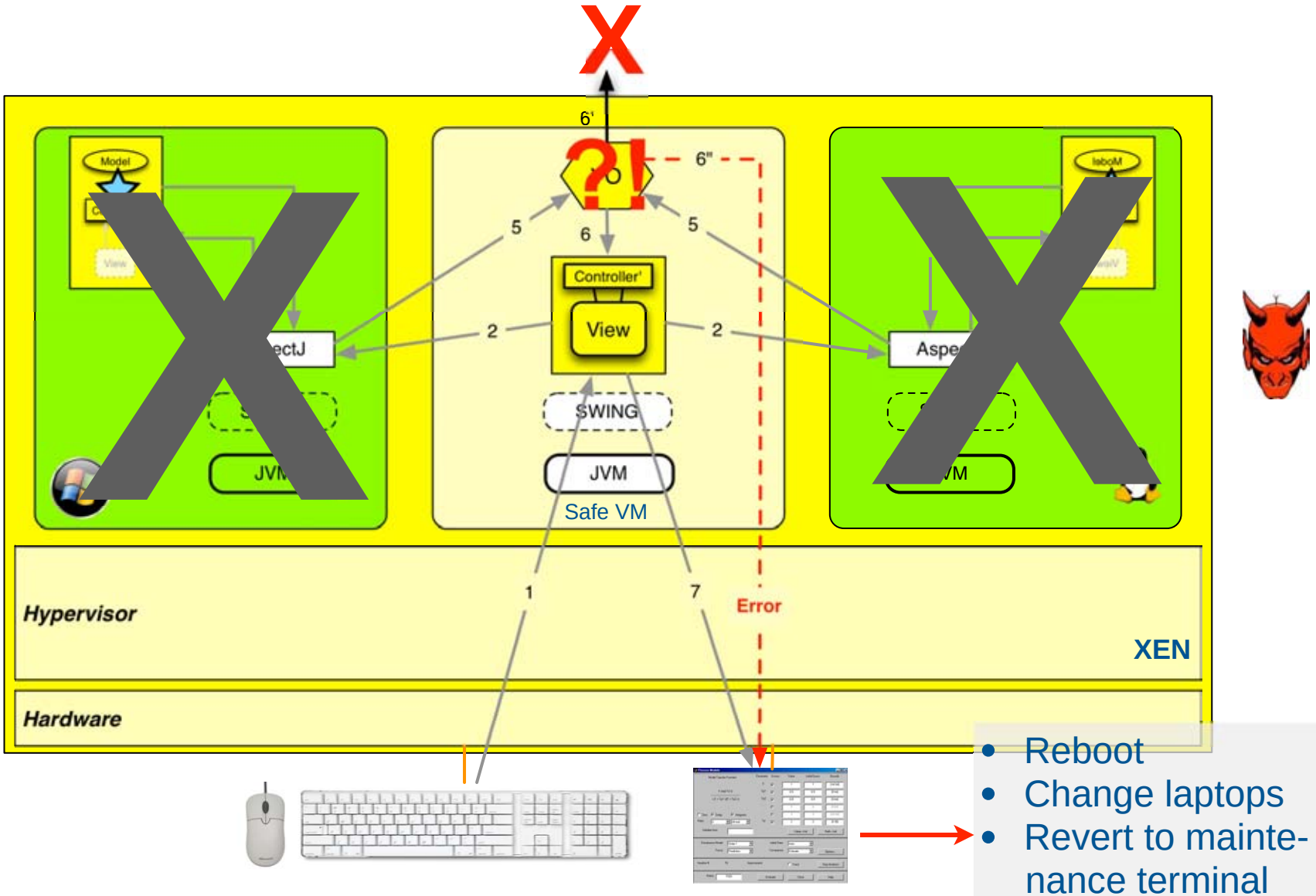
Implementation



Implementation



Implementation



- Reboot
- Change laptops
- Revert to maintenance terminal
- Go to the beach...

Replica Non-Determinism

Can cause false positives

- **Timing**
  - current solution :
    - over-dimensioned timeout on 2nd response ➔ 170 µs
- **Multi-threading**
  - current solution :
    - 3 threads are independent
    - outputs of each thread are identified and validated independently

Outline

- Introduction
- Levels of confidence
- Multi-level confidence models
- Platform virtualization
- Laptop prototype
- **Conclusion**

Conclusion

- **Virtualization**
  - attractive solution for implementing multiple levels of confidence on a single machine
- **Assumes**
  - hypervisor can be trusted at highest level of confidence
- **Proof-of-concept prototype**
  - maintenance laptop application
- **Future work**
  - relaxing constraints imposed to avoid false positives
  - dealing with non-determinism in a more general way
  - guarantee integrity of platform from boot to run-time