NICTA

An Architectural Approach for Cost Effective Trustworthy Systems

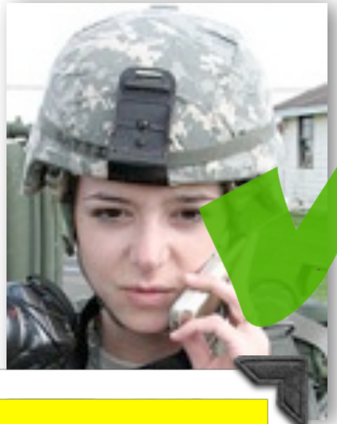
Ihor Kuz, Liming Zhu, Len Bass, Mark Staples, Xiwei Xu

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

Trustworthy Systems

Availability



Security

Deserving of Trust

Reliability

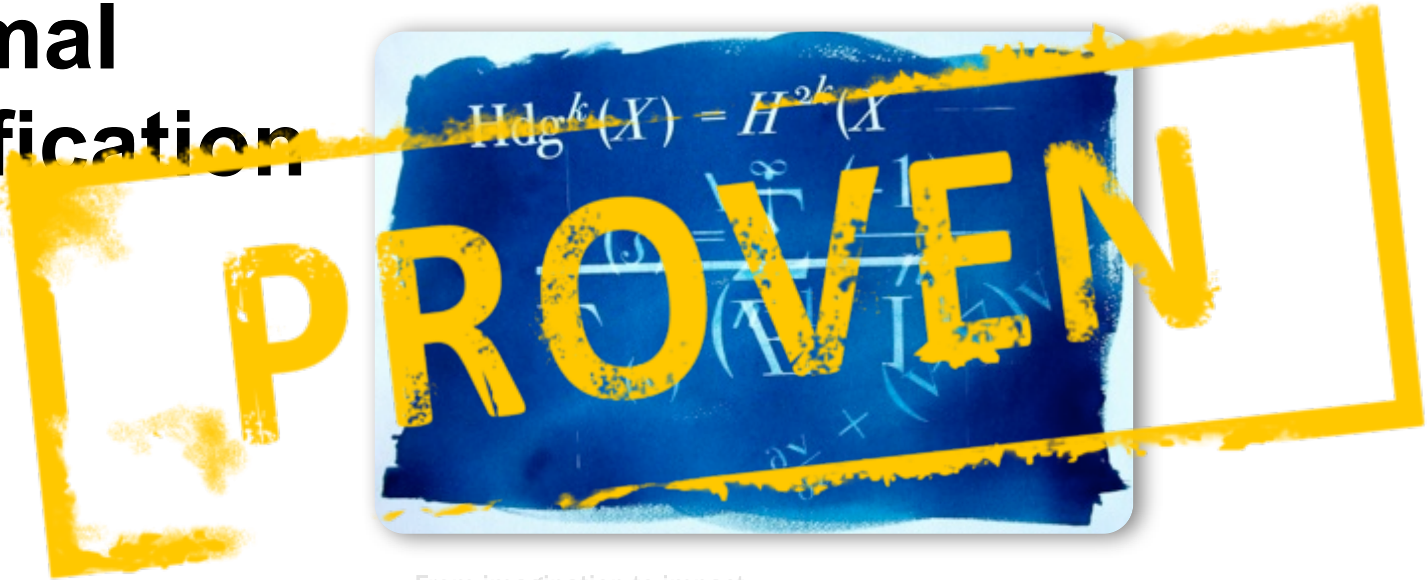
Safety

Assurance

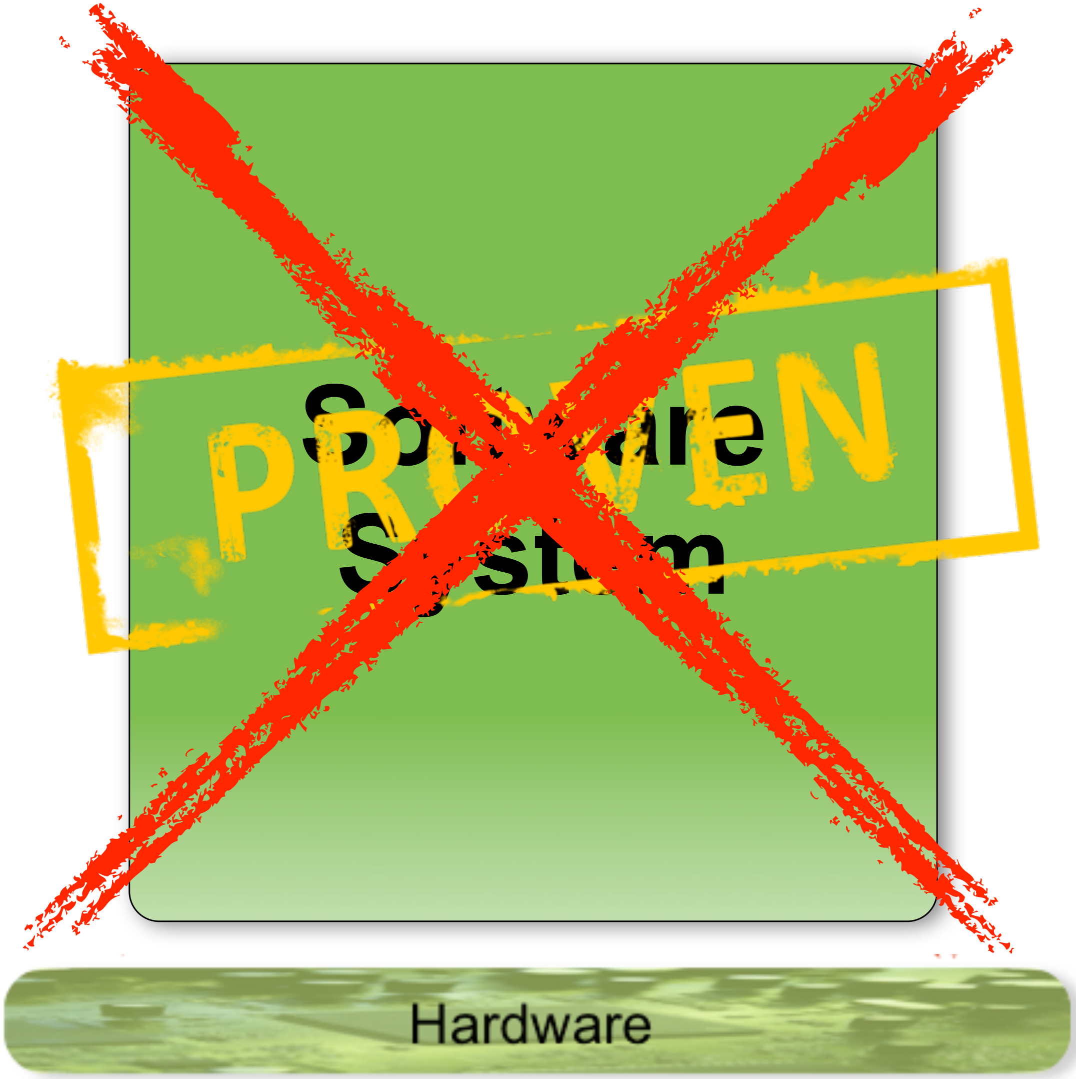
Testing

Process Certification

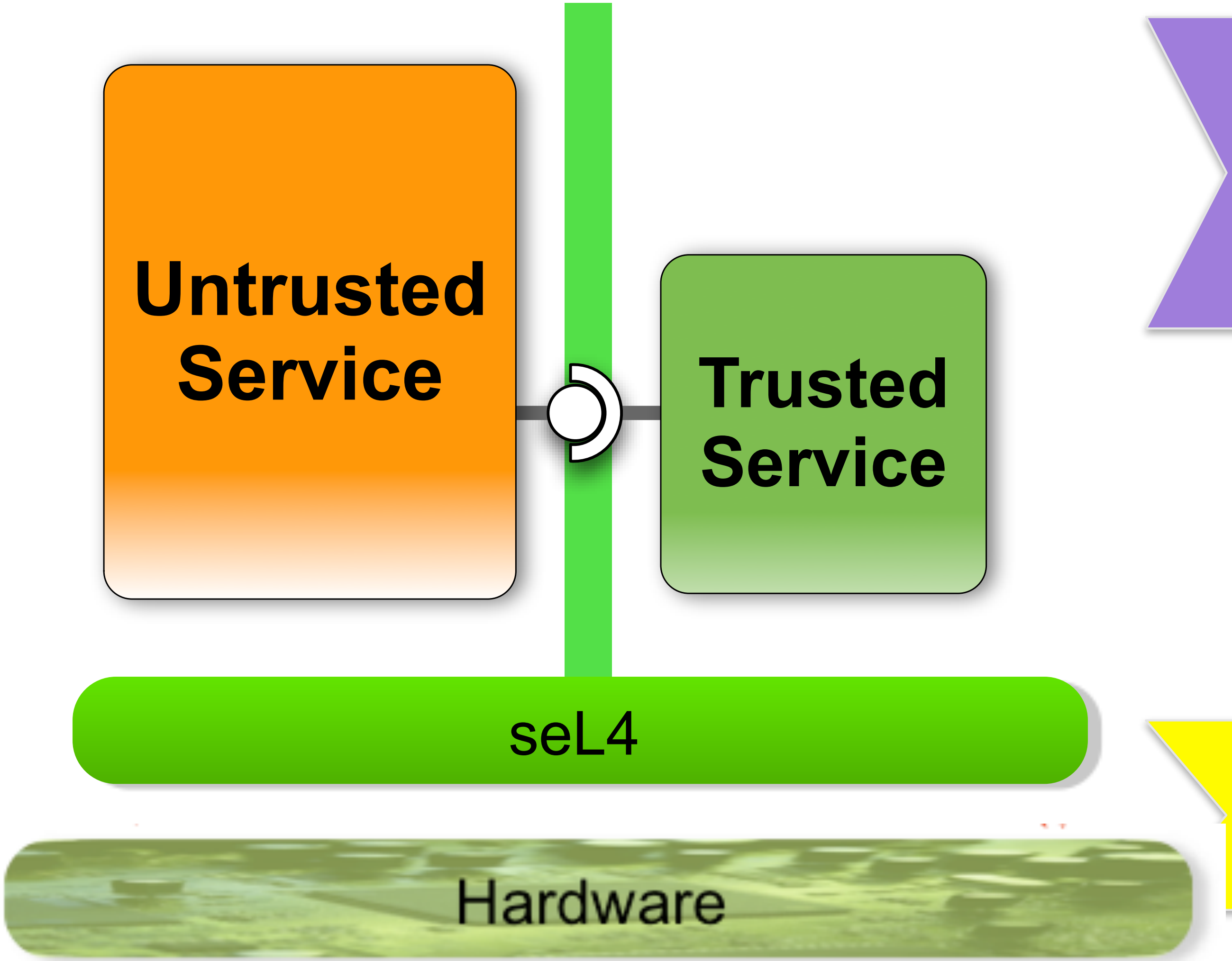
Formal Verification



Building Trustworthy Systems



Building Trustworthy Systems

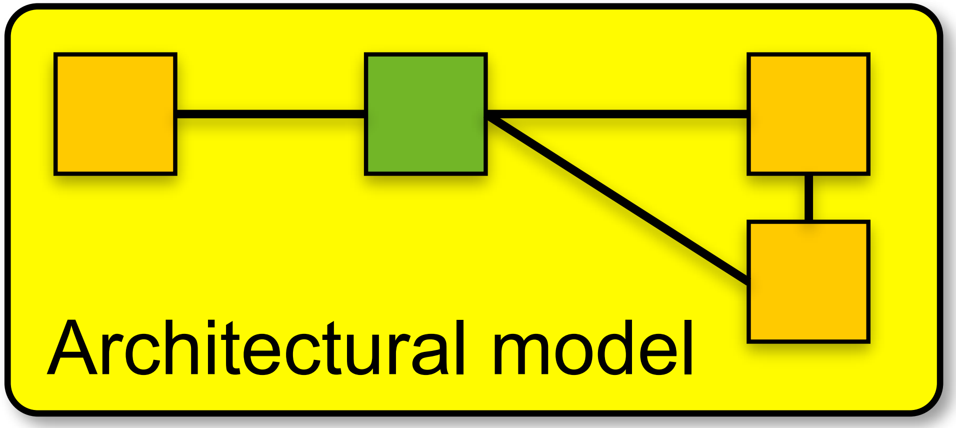


Reason about a whole system without having to reason about the behaviour of every component

Controlled separation

Cost Effective Trustworthy Systems

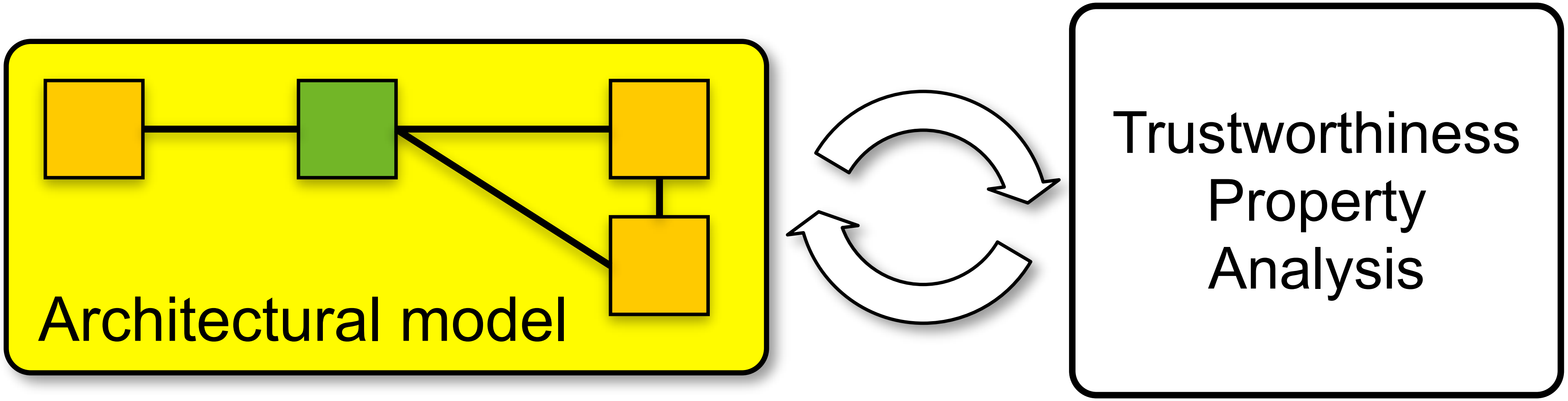
- Verification is expensive
  ➡ make sure it works the first time
- Architecture-driven approach



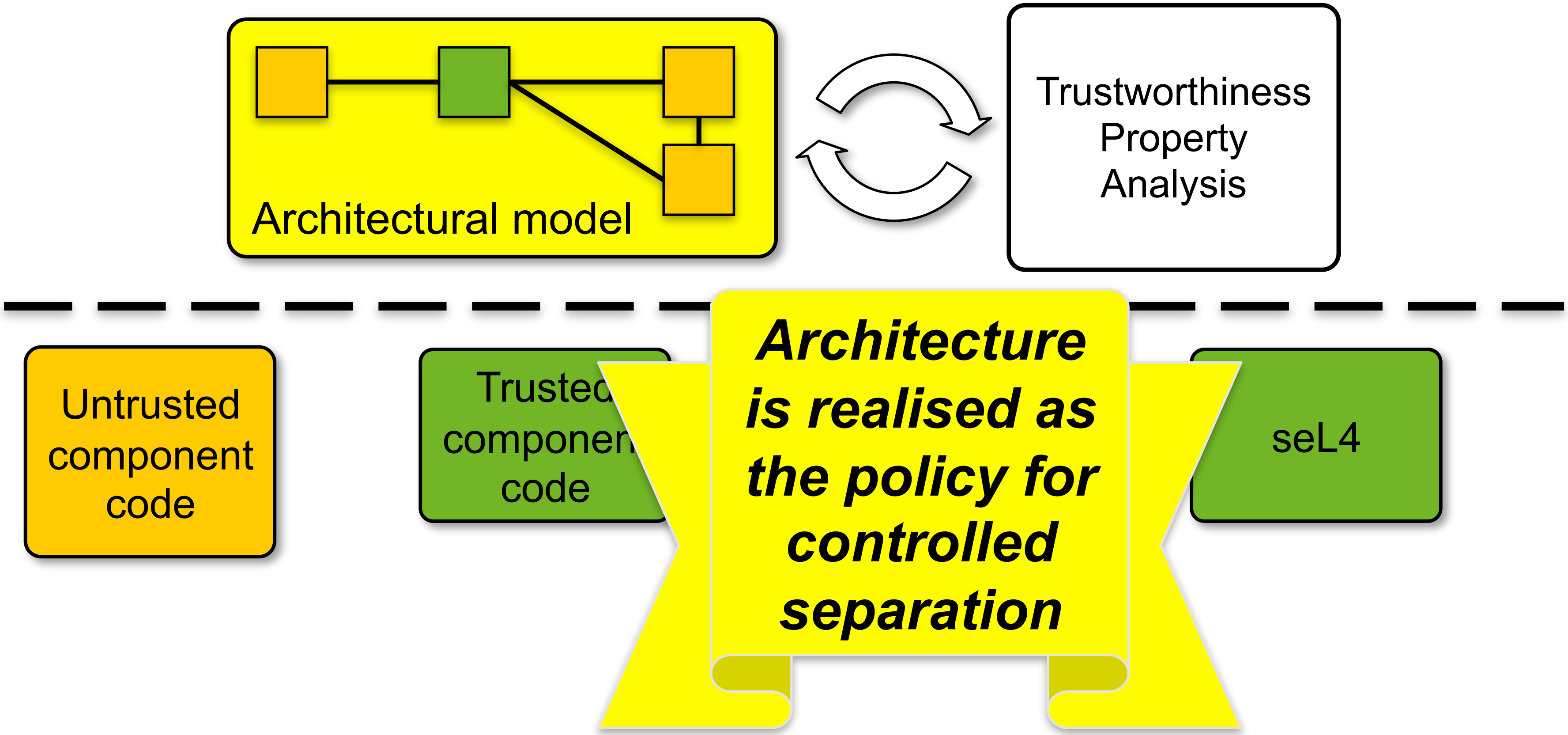
Trustworthiness Property

Cost Effective Trustworthy Systems

- Verification is expensive
  - ➡ make sure it works the first time
- Architecture-driven approach

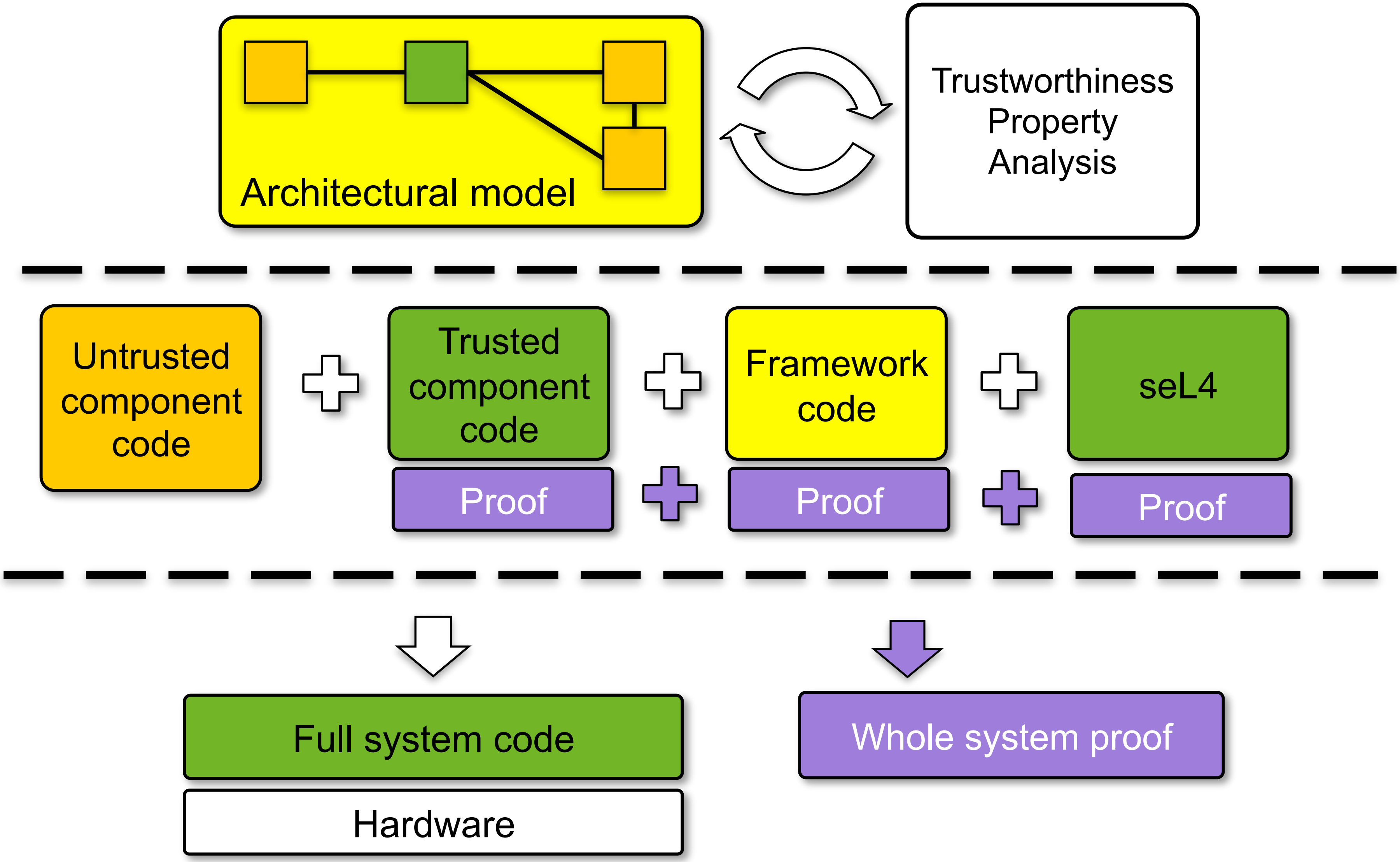


Cost Effective Trustworthy Systems



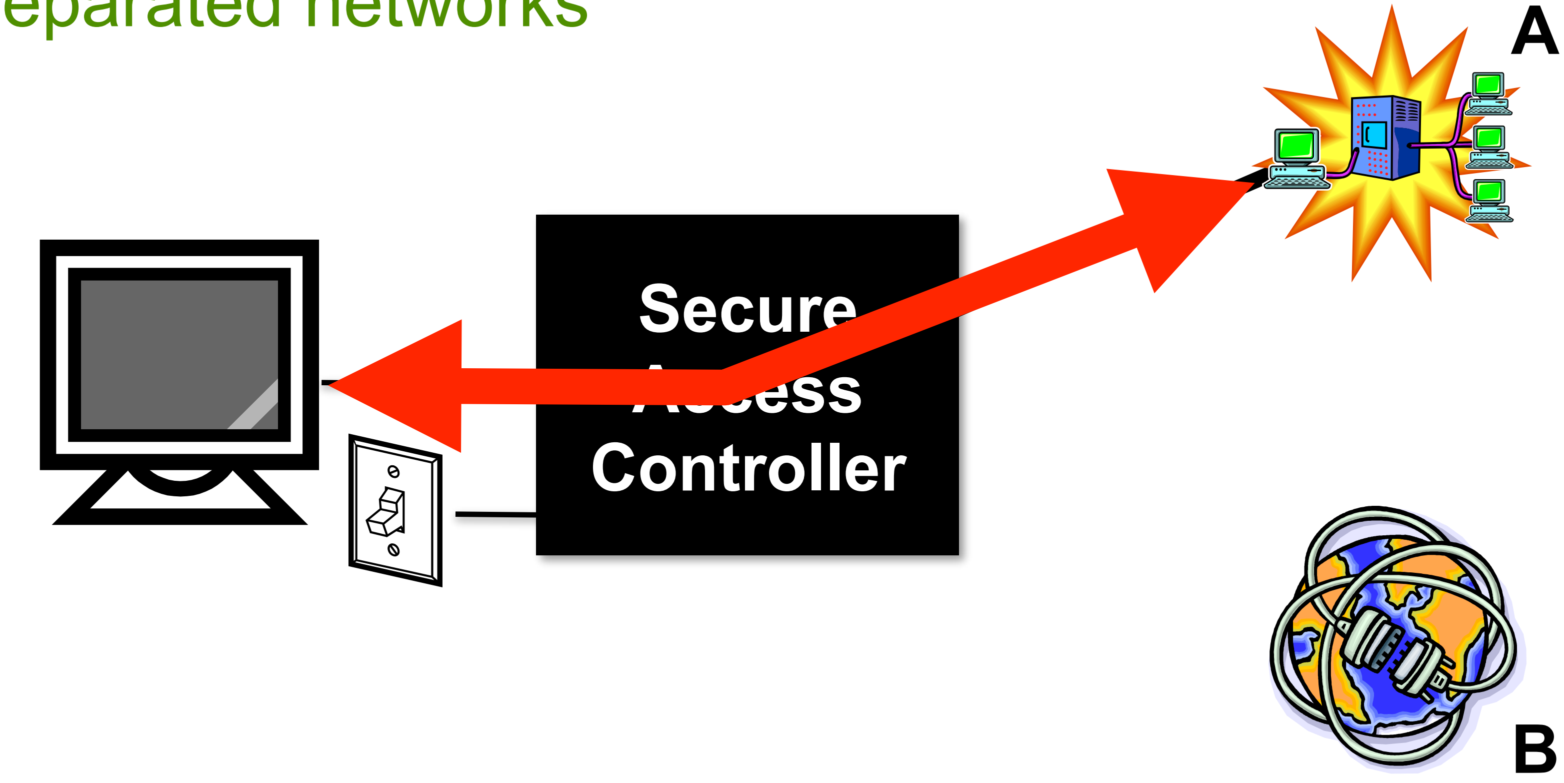
Architecture is realised as the policy for controlled separation

Cost Effective Trustworthy Systems



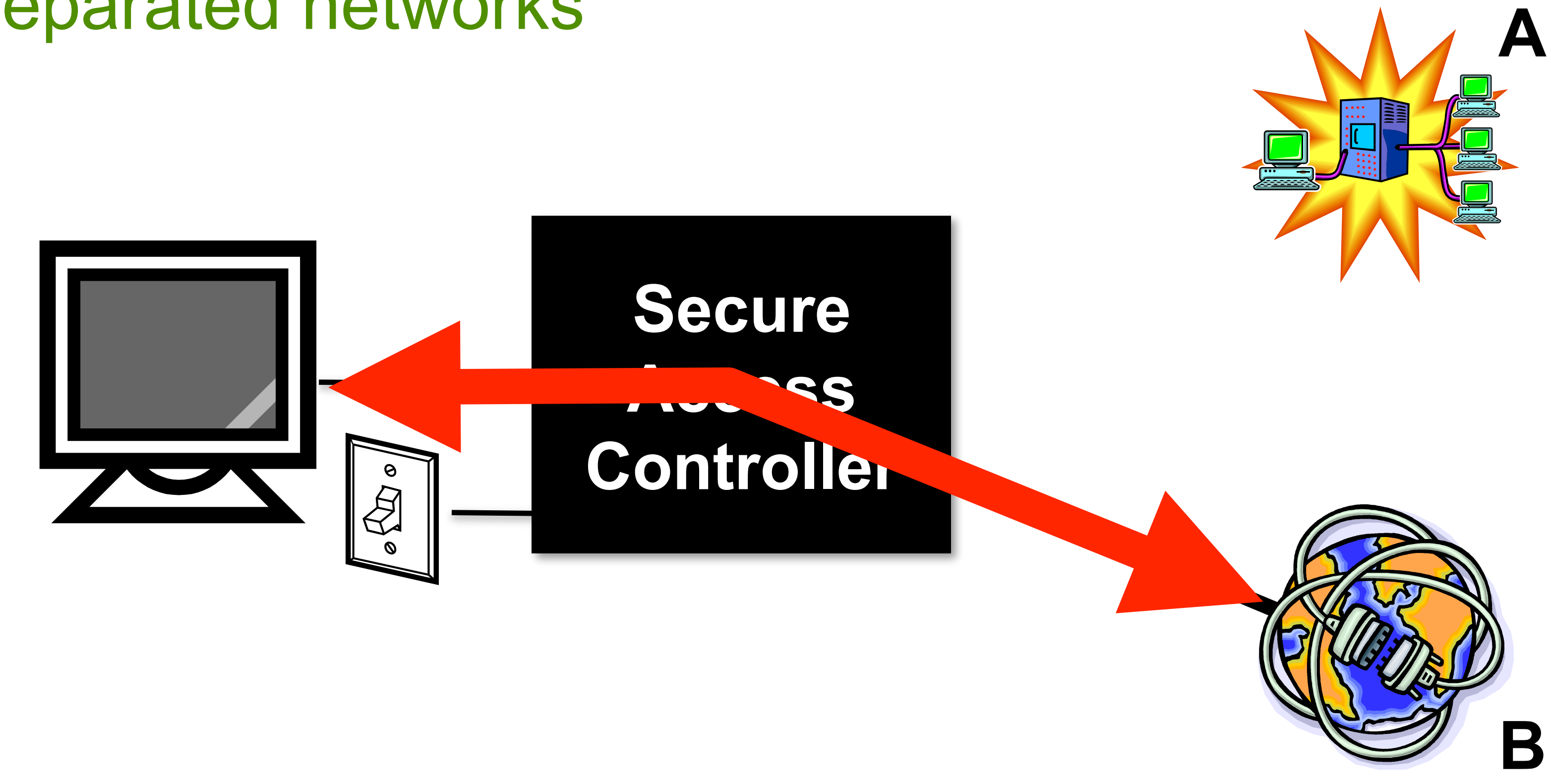
Working Example: SAC

- Secure Access Controller (SAC)
- Securely switch a terminal between two strictly separated networks



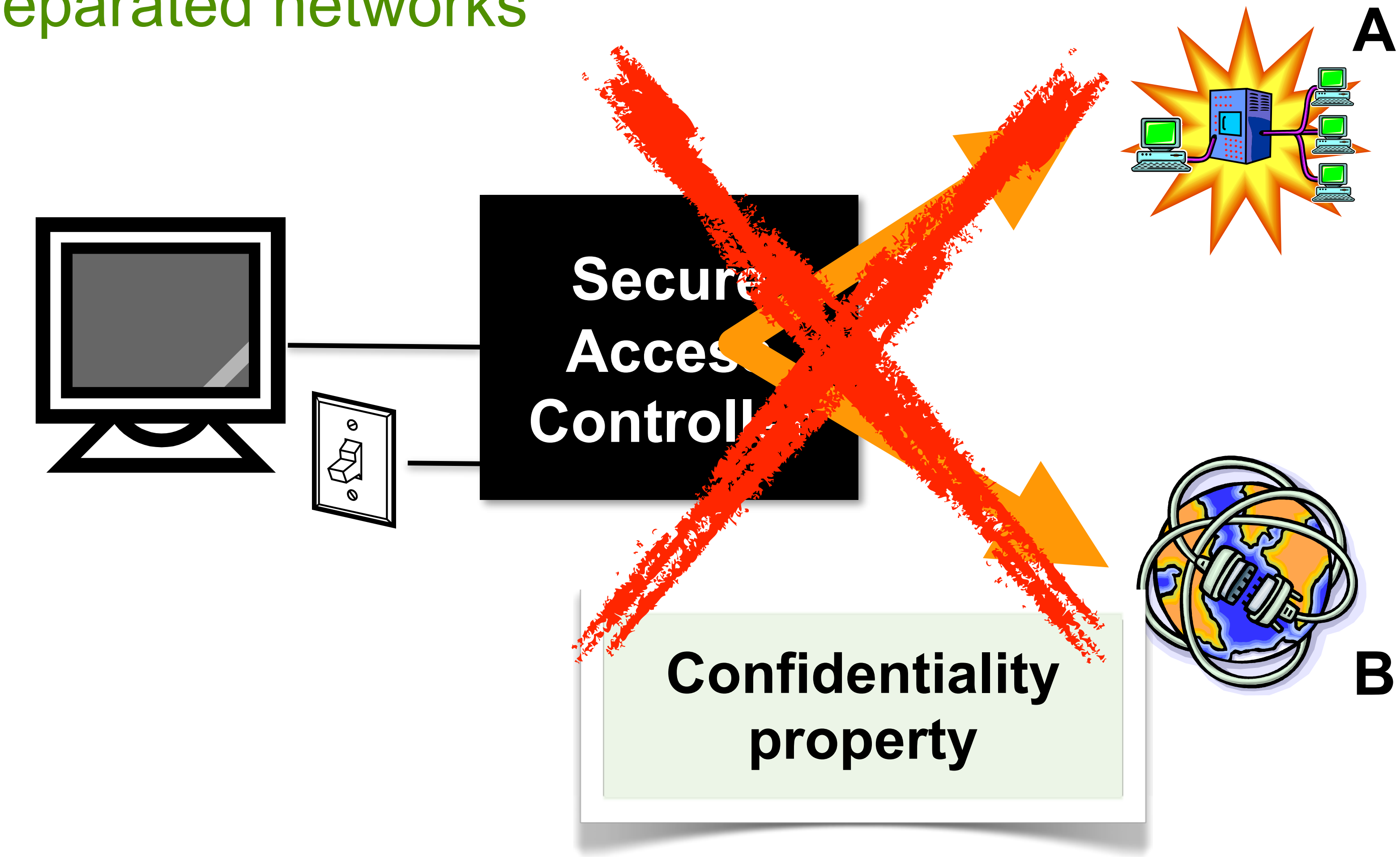
Working Example: SAC

- Secure Access Controller (SAC)
- Securely switch a terminal between two strictly separated networks

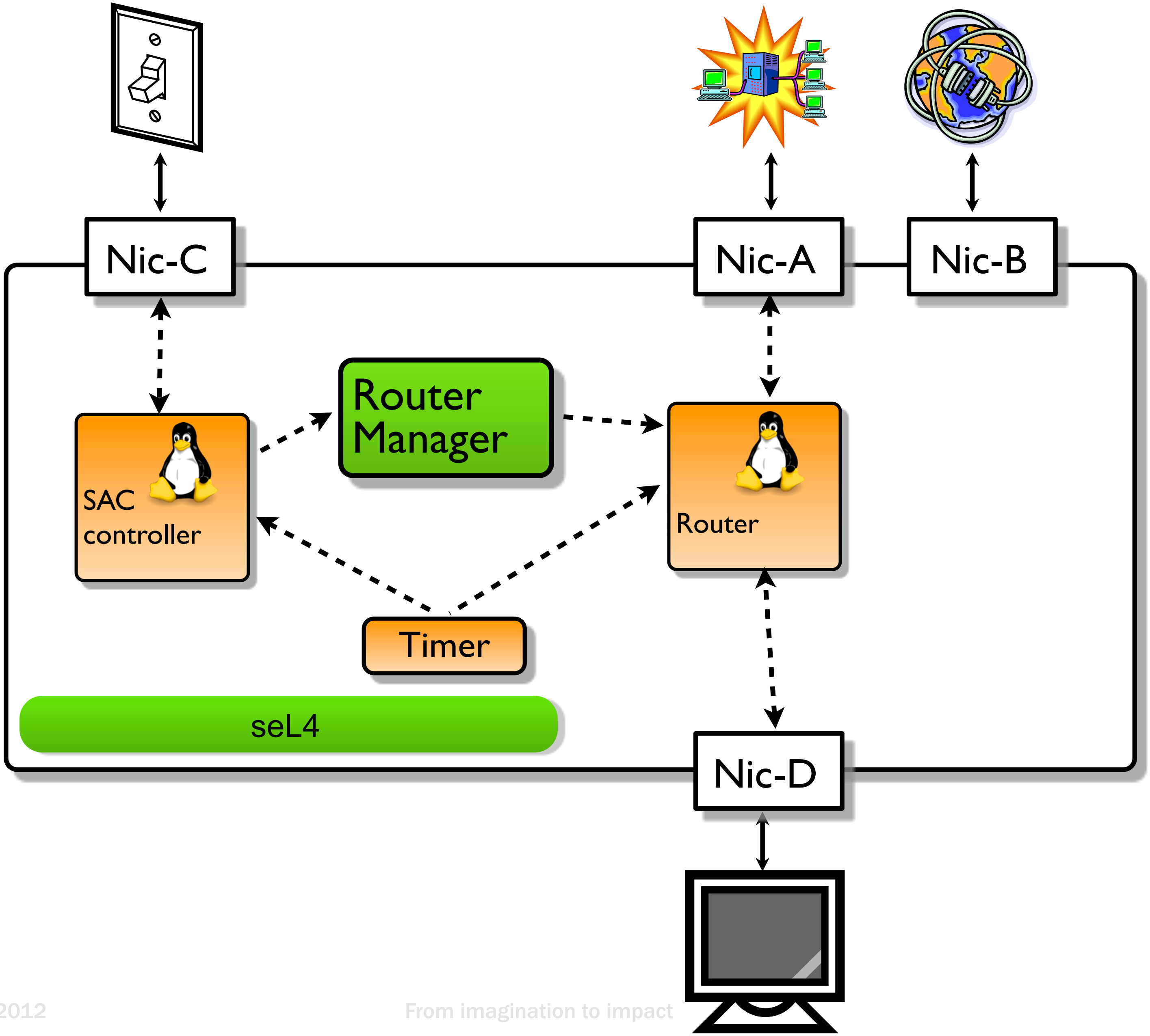


Working Example: SAC

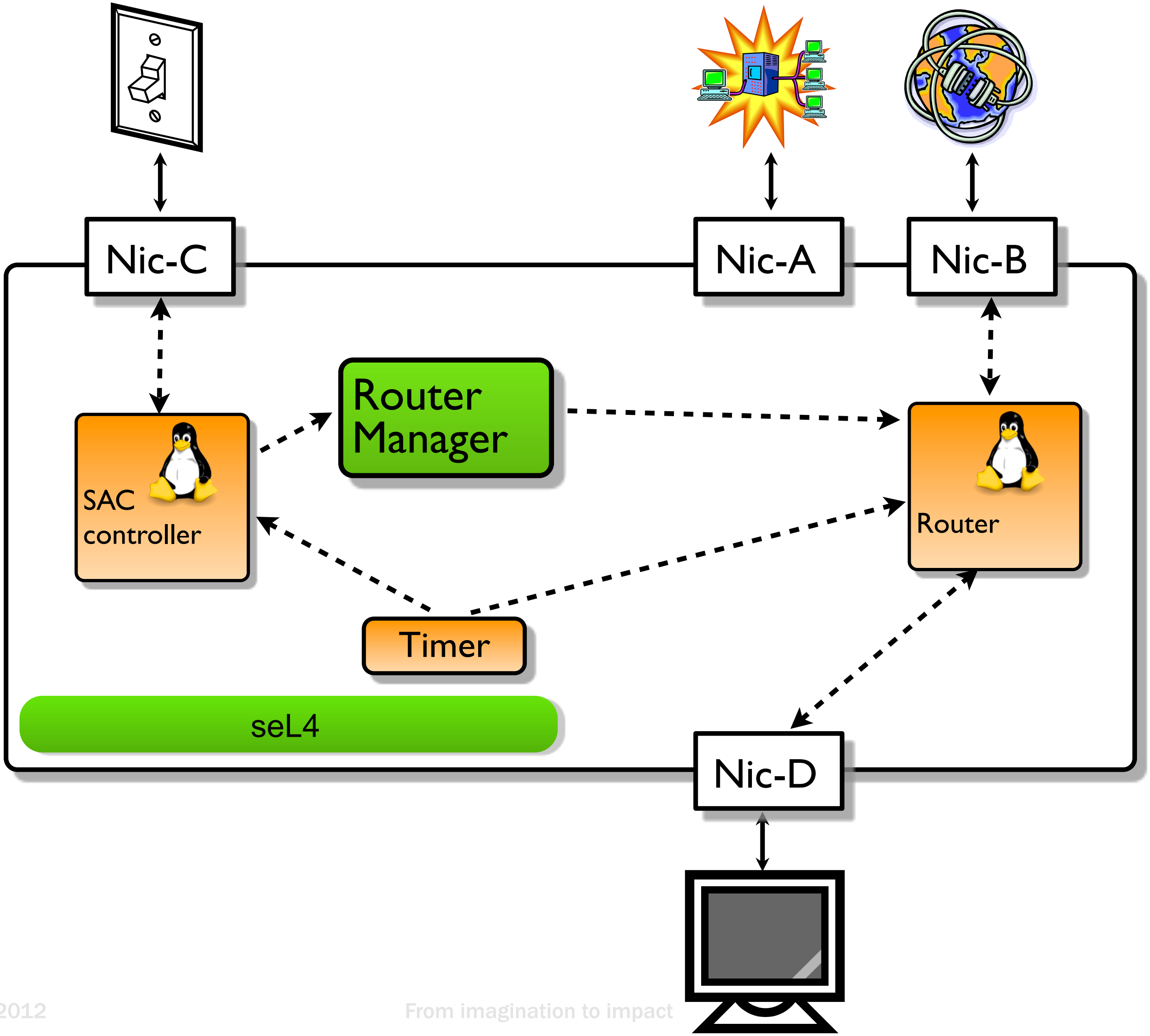
- Secure Access Controller (SAC)
- Securely switch a terminal between two strictly separated networks



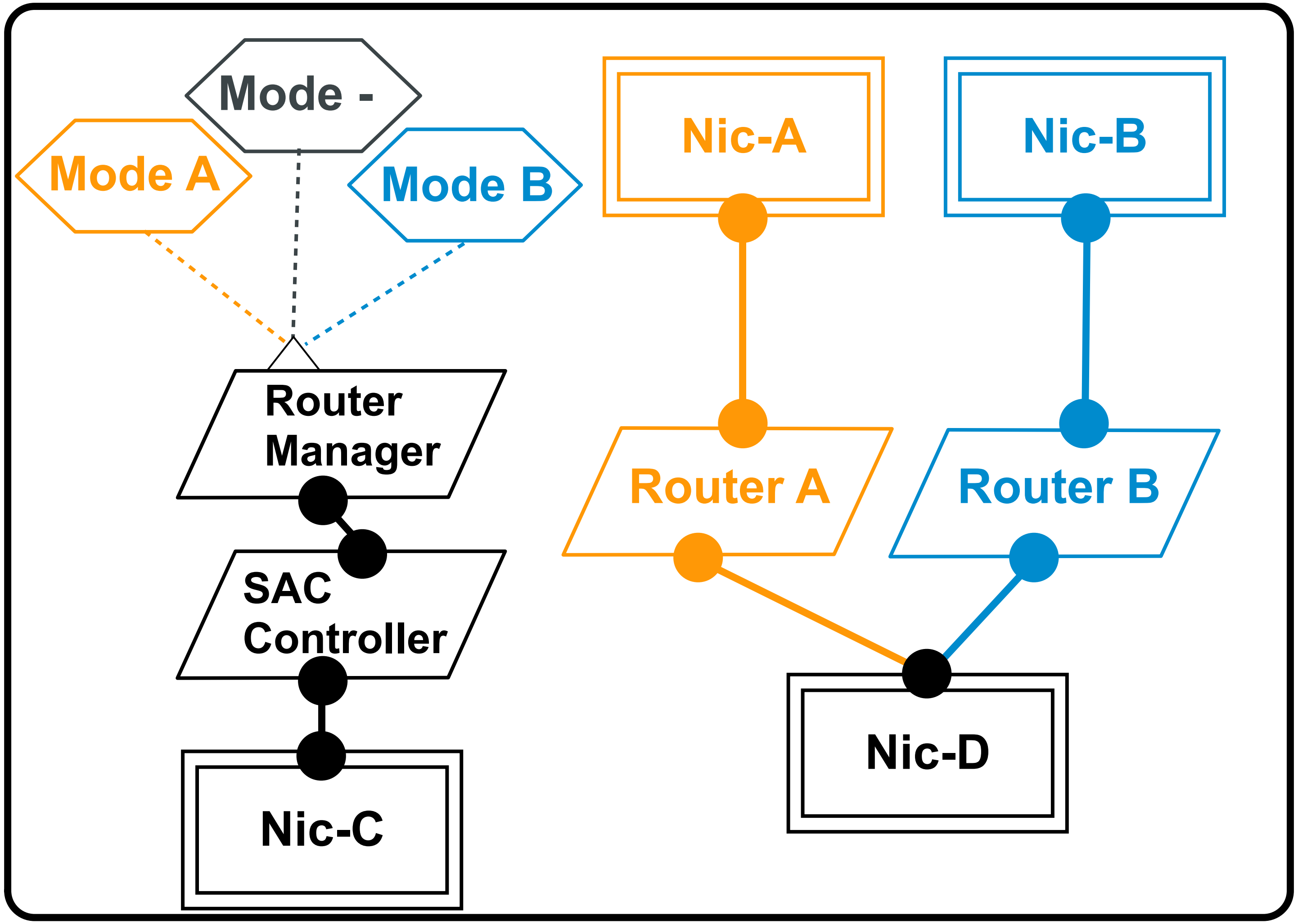
SAC Implementation



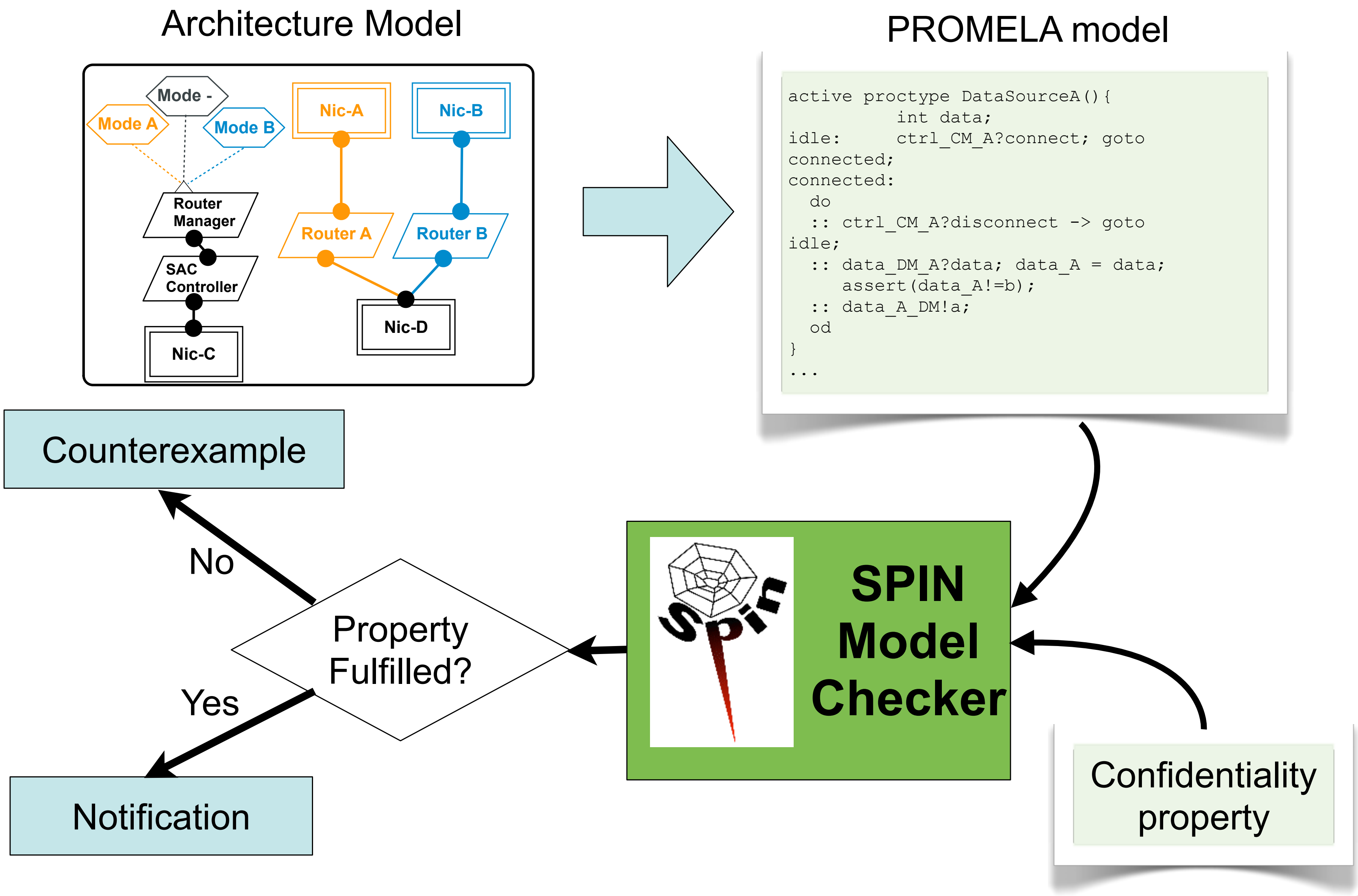
SAC Implementation



Architectural Reconstruction in AADL



Confidentiality Analysis with SPIN



Architecture Model

PROMELA model

```
active proctype DataSourceA(){
          int data;
idle:     ctrl_CM_A?connect; goto
connected;
connected:
  do
  :: ctrl_CM_A?disconnect -> goto
idle;
  :: data_DM_A?data; data_A = data;
     assert(data_A!=b);
  :: data_A_DM!a;
  od
}
...
```

SPIN Model Checker

Confidentiality property

Property Fulfilled?

No

Counterexample

Yes

Notification

Results

- Architecture analysis works
  - can reduce effort of whole system verification
- Helps spot problems early on
  - Terminal network card (NIC-D) can store data

    ➡ storage channel unless flushed explicitly
- AADL and SPIN sufficient for SAC
  - ***BUT:*** other systems need more dynamism
- Next steps
  - code generation: glue code and framework
  - architecture support for verification
  - trusted patterns