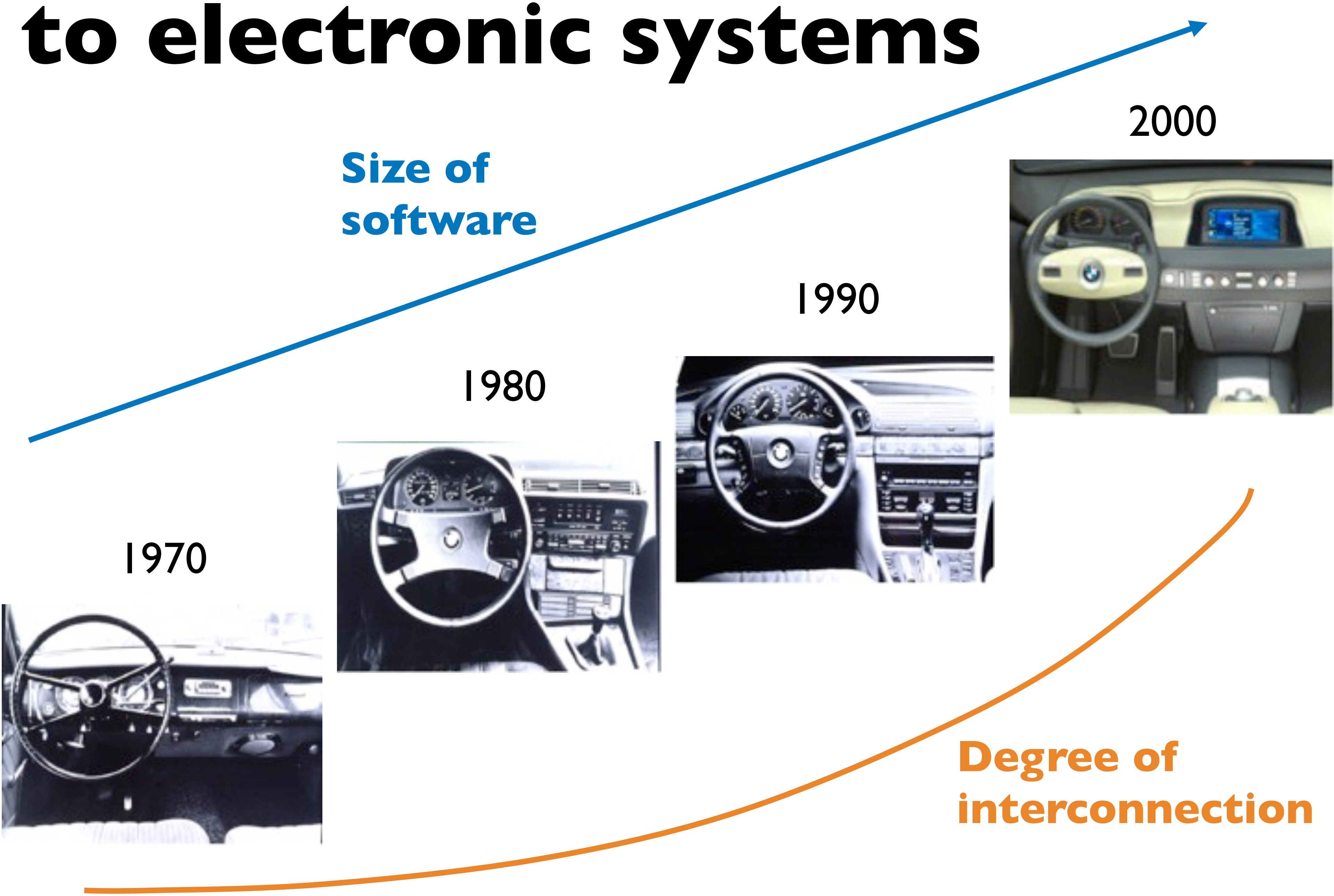
Combining **STAMP/STPA** and **Assurance Cases**

Stefan Wagner
Institute of Software Technology

STAMP Workshop 2012
Cambridge, MA
19 April 2012

Automobiles evolve...

to electronic systems



Size of software

2000

1990

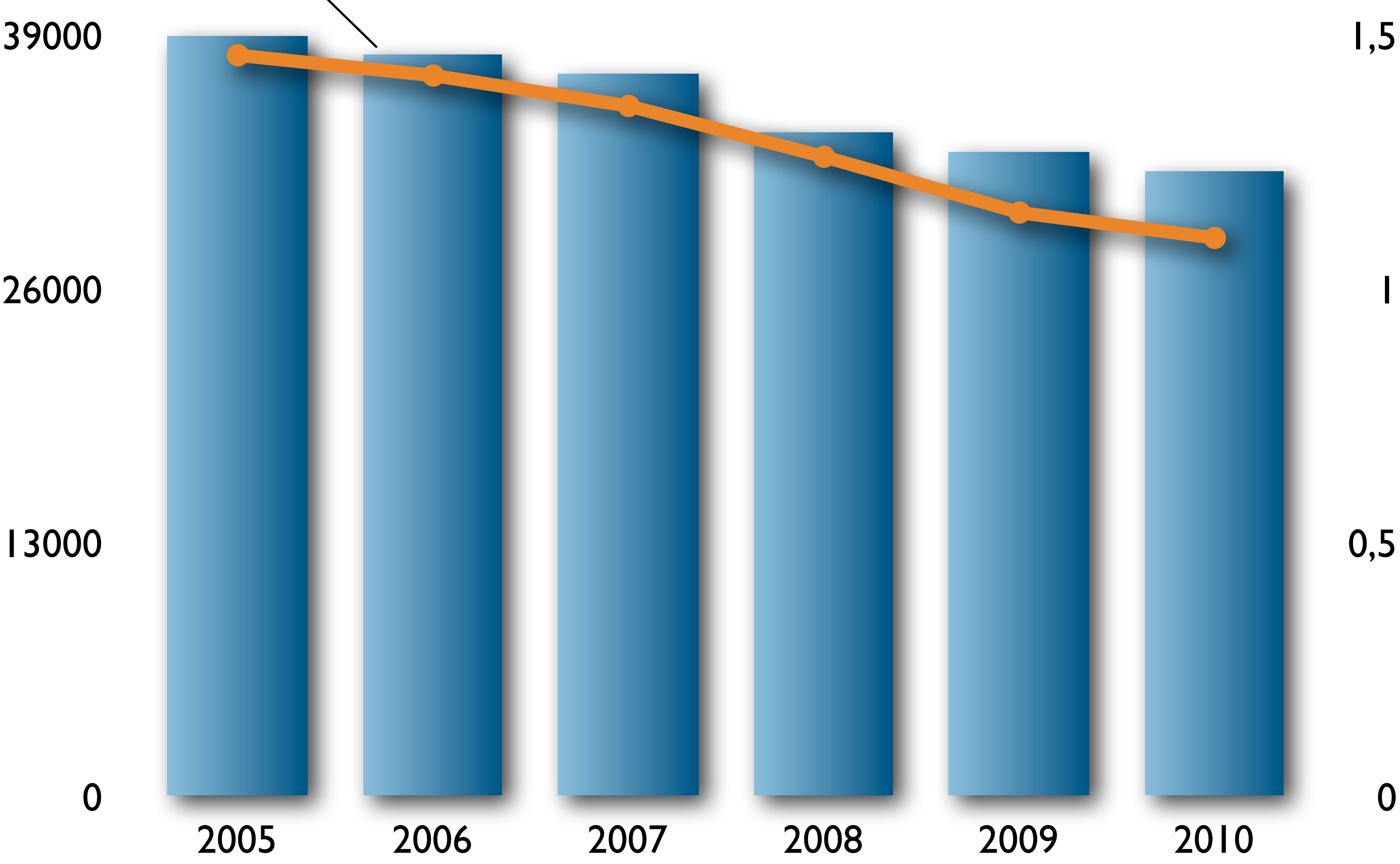
1980

1970

Degree of interconnection

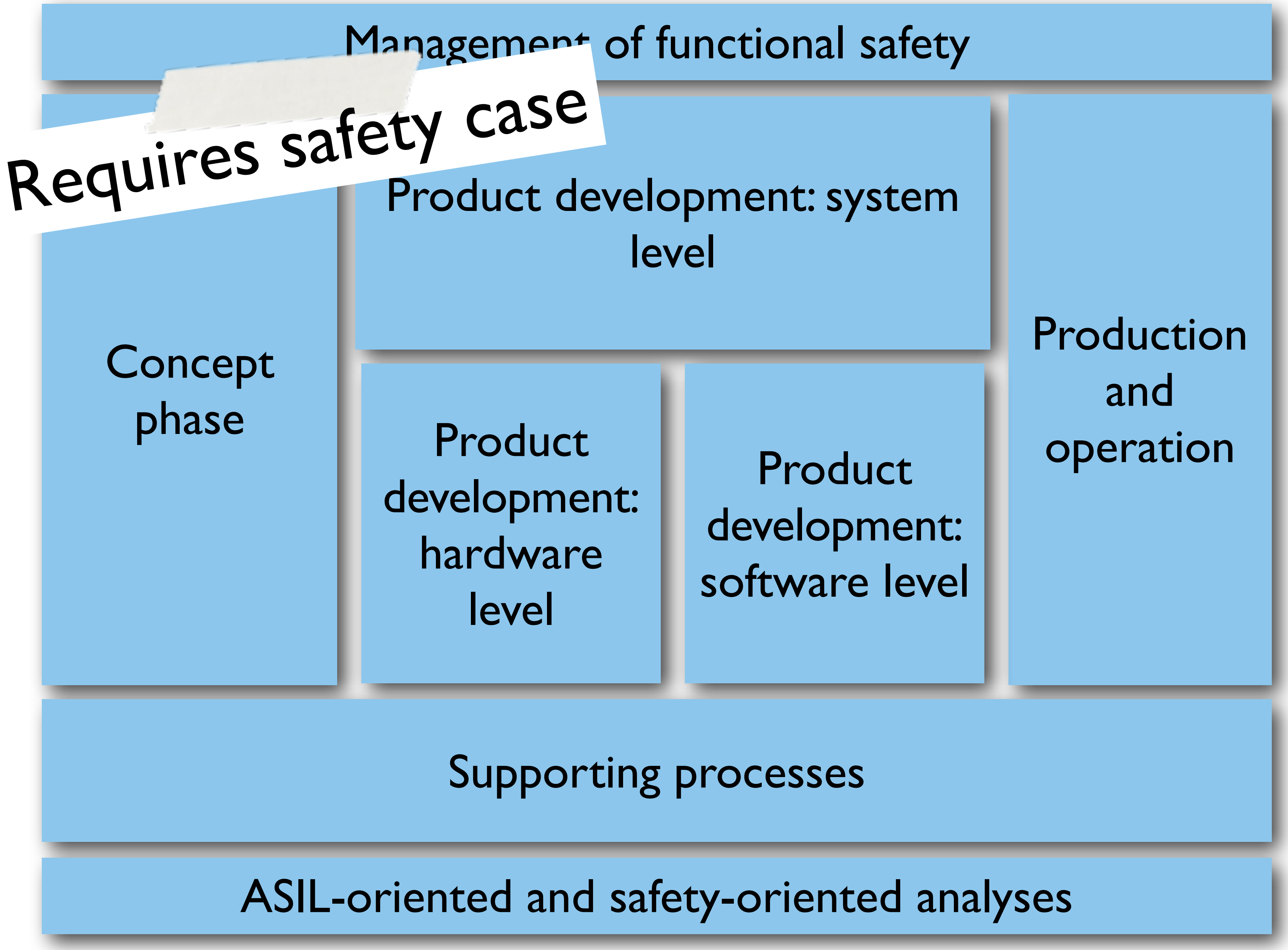
Crashs with fatalities

Fatalities per year in the USA



Fatalities per 100,000,000 miles

ISO 26262: Road vehicles – Functional safety



Requires safety case

Management of functional safety

Concept phase

Product development: system level

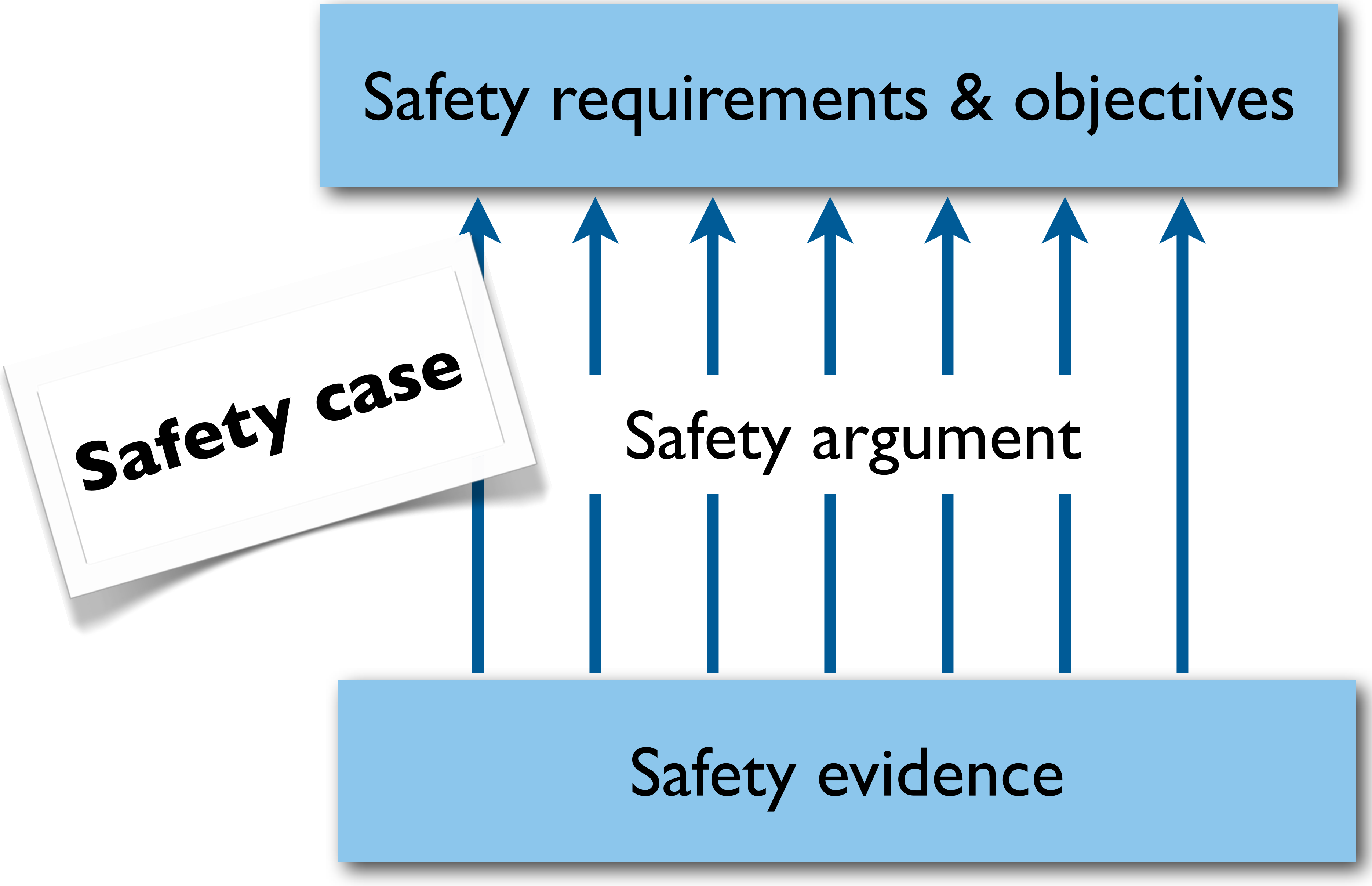
Product development: hardware level

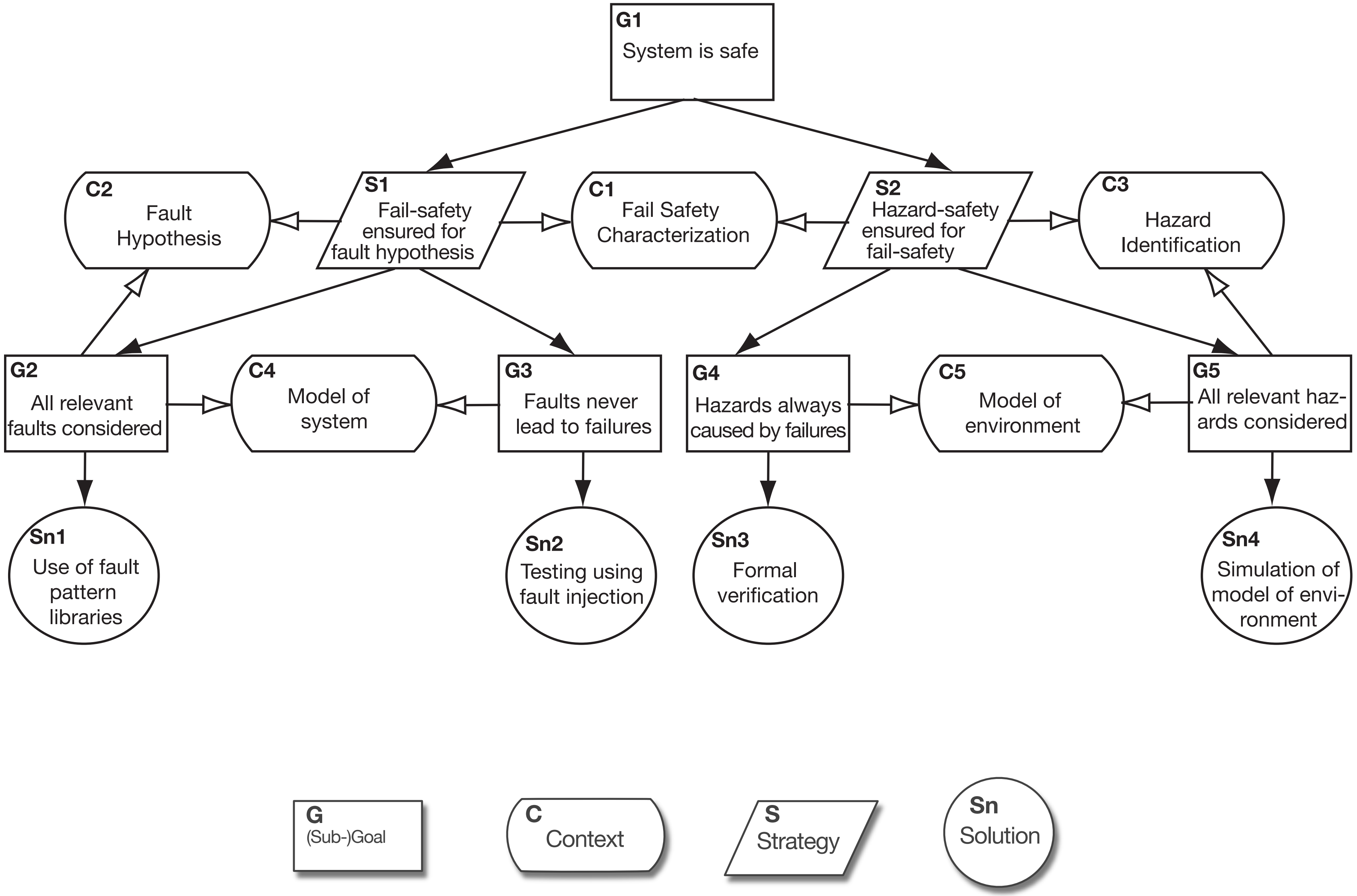
Product development: software level

Production and operation

Supporting processes

ASIL-oriented and safety-oriented analyses

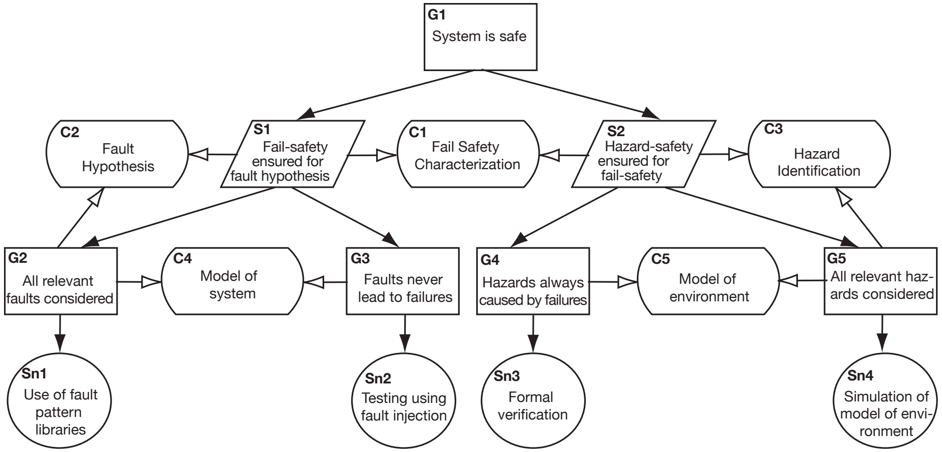



G1
System is safe
S1
Fail-safety ensured for fault hypothesis
C2
Fault Hypothesis
C1
Fail Safety Characterization
S2
Hazard-safety ensured for fail-safety
C3
Hazard Identification
G2
All relevant faults considered
C4
Model of system
G3
Faults never lead to failures
G4
Hazards always caused by failures
C5
Model of environment
G5
All relevant hazards considered
Sn1
Use of fault pattern libraries
Sn2
Testing using fault injection
Sn3
Formal verification
Sn4
Simulation of model of environment
G
(Sub-)Goal
C
Context
S
Strategy
Sn
Solution

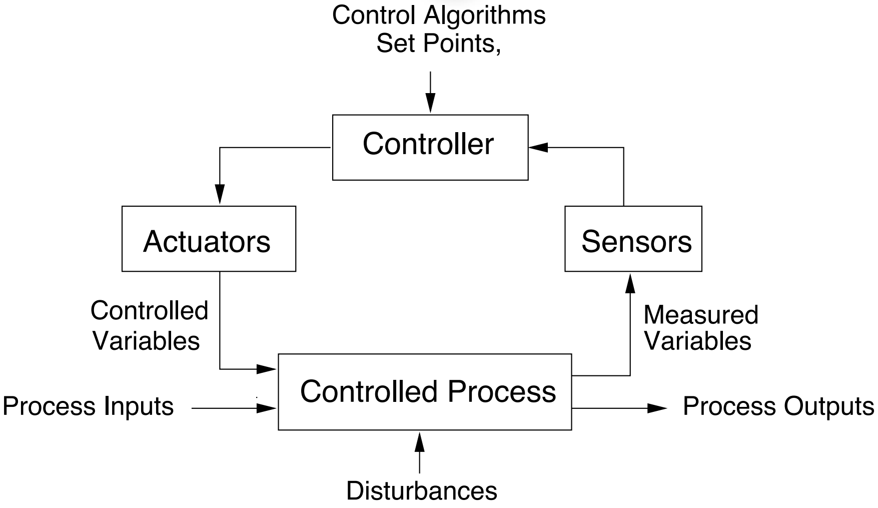
Cruise control case study with MAN

97X J89

Shift by wire case study with BMW


G1
System is safe
C2
Fault Hypothesis
S1
Fail-safety ensured for fault hypothesis
C1
Fail Safety Characterization
S2
Hazard-safety ensured for fail-safety
C3
Hazard Identification
G2
All relevant faults considered
C4
Model of system
G3
Faults never lead to failures
G4
Hazards always caused by failures
C5
Model of environment
G5
All relevant haz-ards considered
Sn1
Use of fault pattern libraries
Sn2
Testing using fault injection
Sn3
Formal verification
Sn4
Simulation of model of envi-ronment

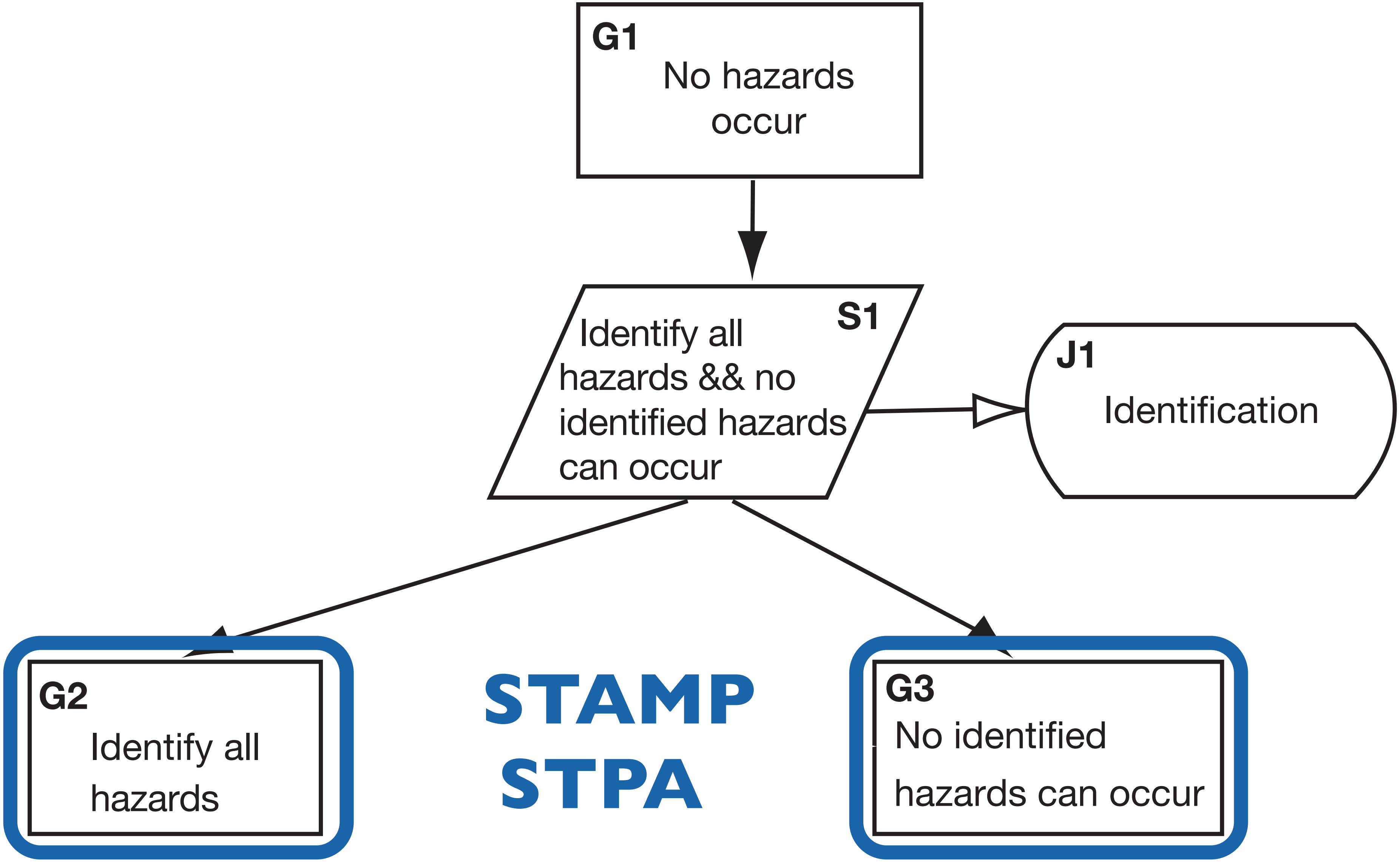
Safety cases are good for a **structured argumentation**



STAMP/STPA are good for a **systematic analysis**

I

Identification pattern



G1
No hazards occur

S1
Identify all hazards && no identified hazards can occur

J1
Identification

G2
Identify all hazards

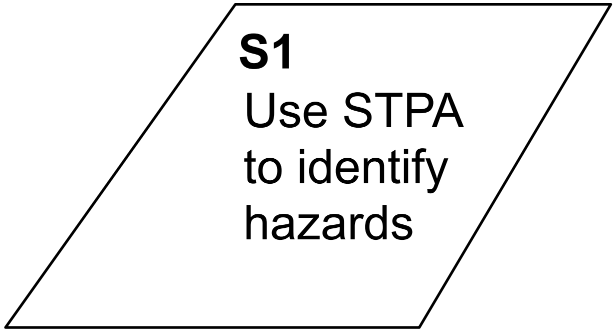
G3
No identified hazards can occur

STAMP
STPA

Example hazard identification

H-1: Gear for wrong direction

H-2: Shift to unsuitable gear for speed



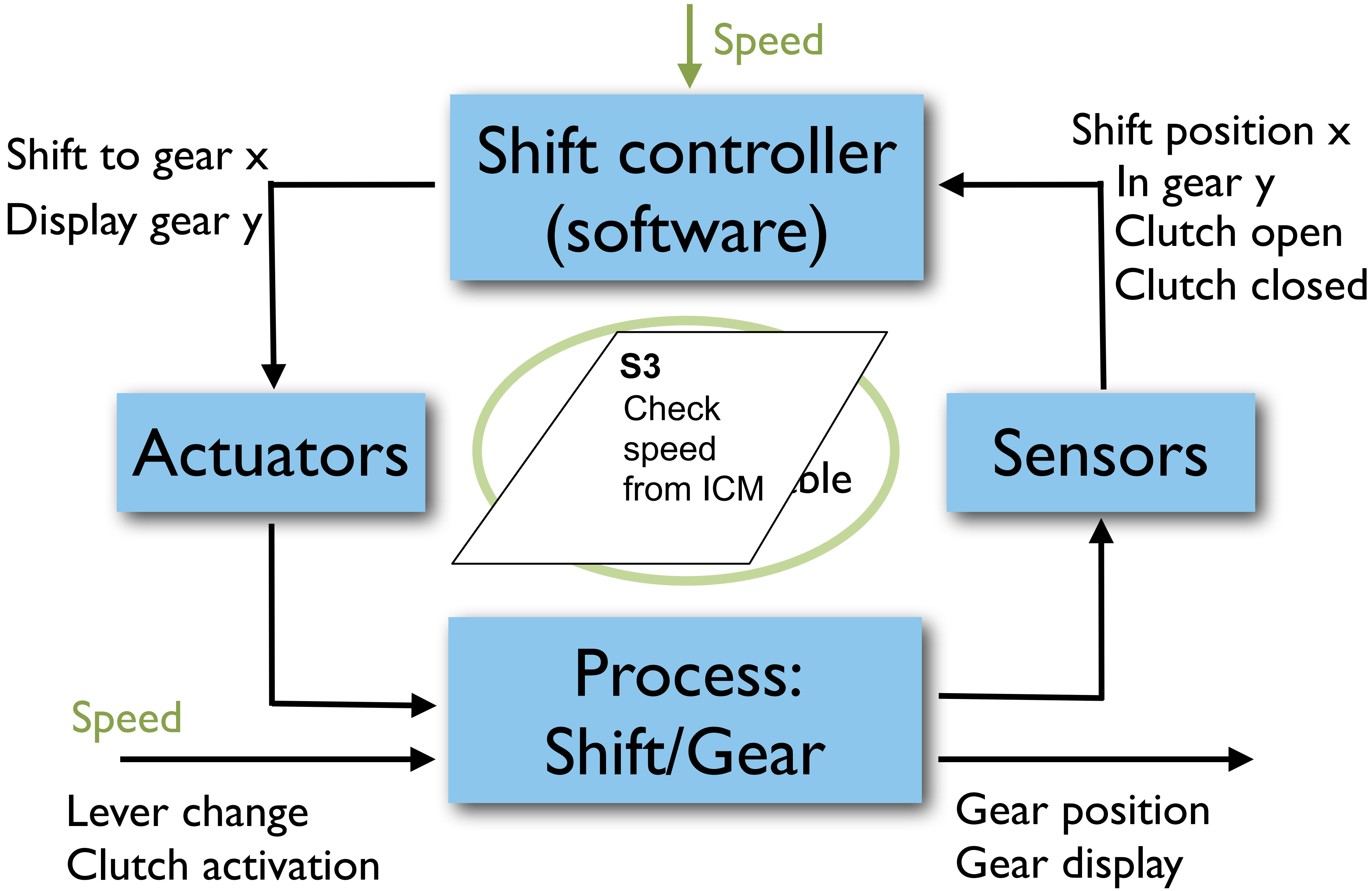
S1
Use STPA to identify hazards

Example hazard analysis

Control Action	Not Given or not Followed	Given Incorrectly	Wrong Timing or Order	Stopped Too Soon
Shift to gear	Controller does not shift gear to change direction	Controller shifts despite no lever change Shift despite no clutch Shift despite unsuitable speed	Shift too late so that driver opens clutch	–
Display gear	Controller does not send new direction to display	Sends wrong gear to display	Not hazardous	–

S2
Use STPA to identify causes for hazards

Example hazard avoidance



Speed

Shift controller (software)

Shift to gear x

Display gear y

Shift position x

In gear y

Clutch open

Clutch closed

Actuators

S3

Check speed from ICM

ble

Sensors

Process: Shift/Gear

Speed

Lever change

Clutch activation

Gear position

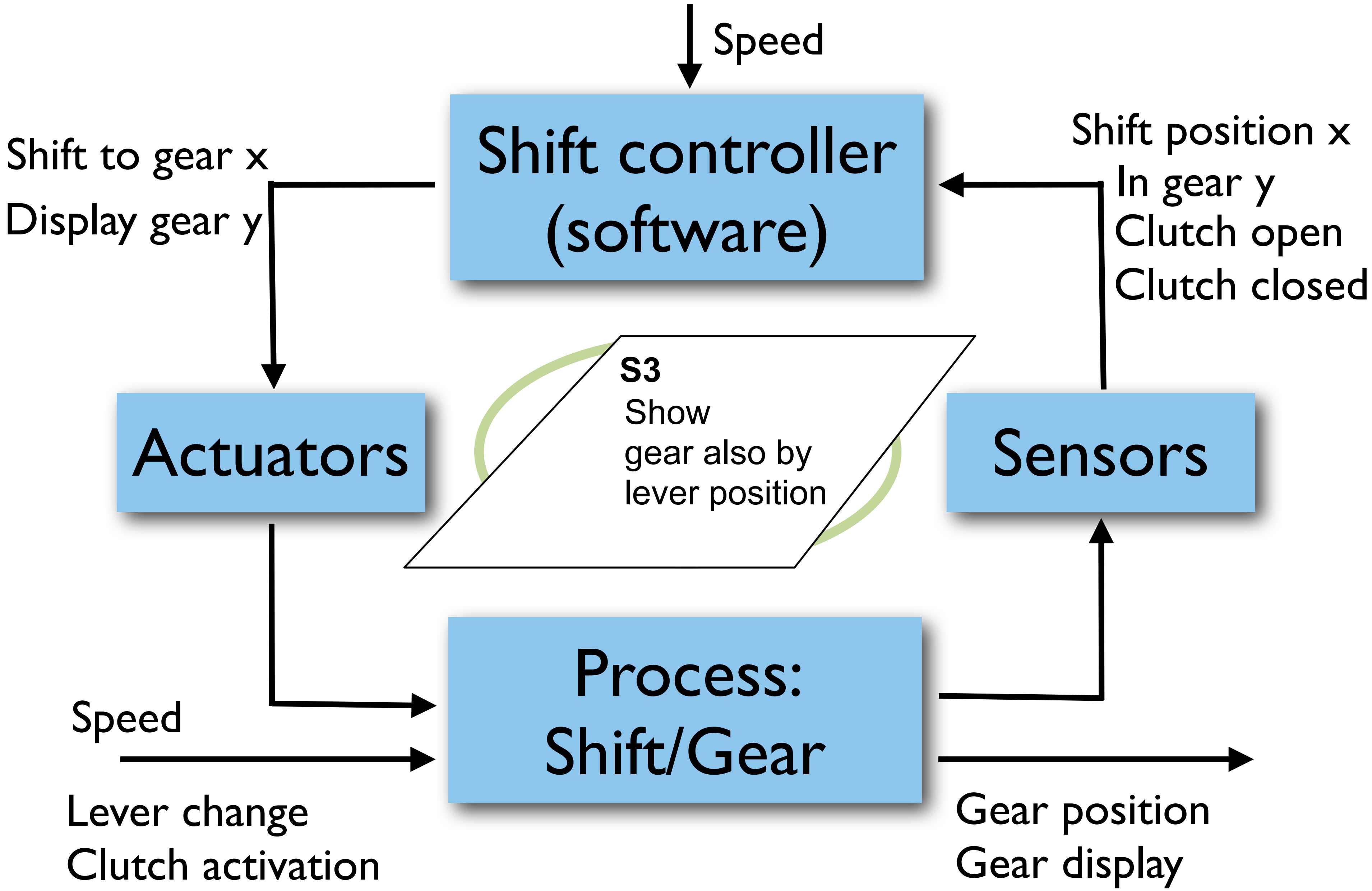
Gear display

2

A final step in STPA is to consider how the designed controls could degrade over time and to build in protection against it.

–Leveson (2011)

Example degradation protection



Speed

Shift controller (software)

Shift to gear x

Display gear y

Shift position x

In gear y

Clutch open

Clutch closed

Actuators

S3
Show
gear also by
lever position

Sensors

Process: Shift/Gear

Speed

Lever change

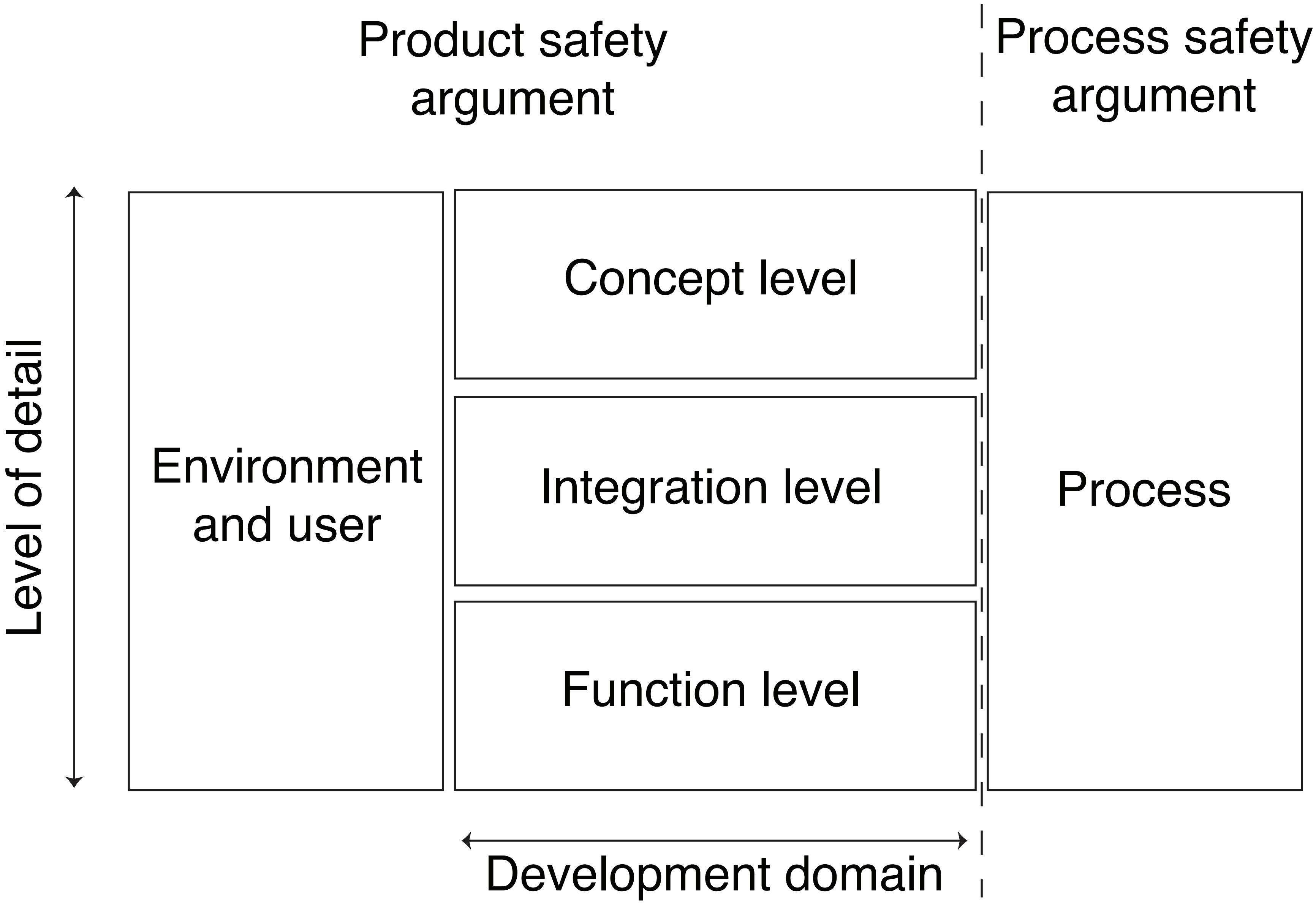
Clutch activation

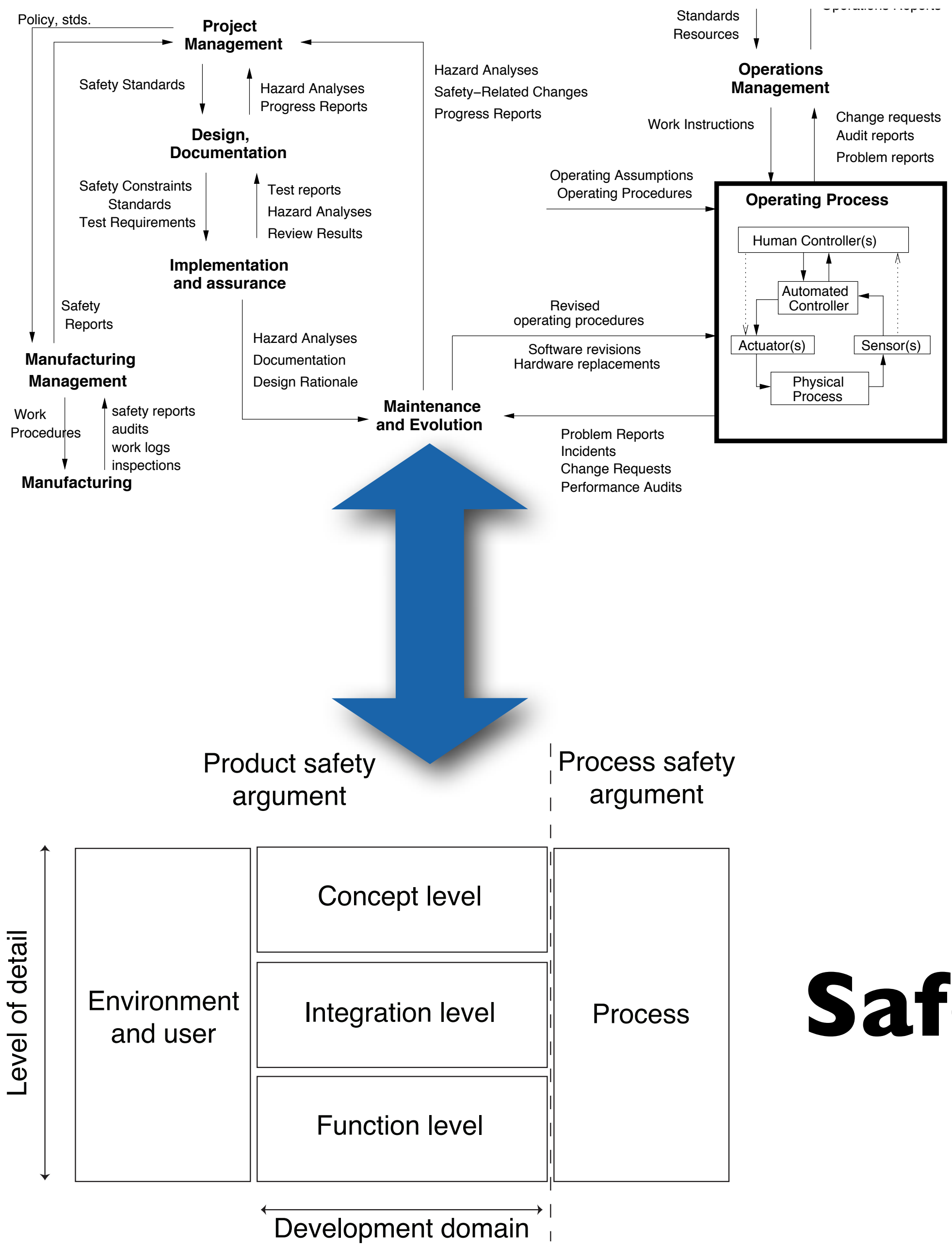
Gear position

Gear display

3

Safety case modules

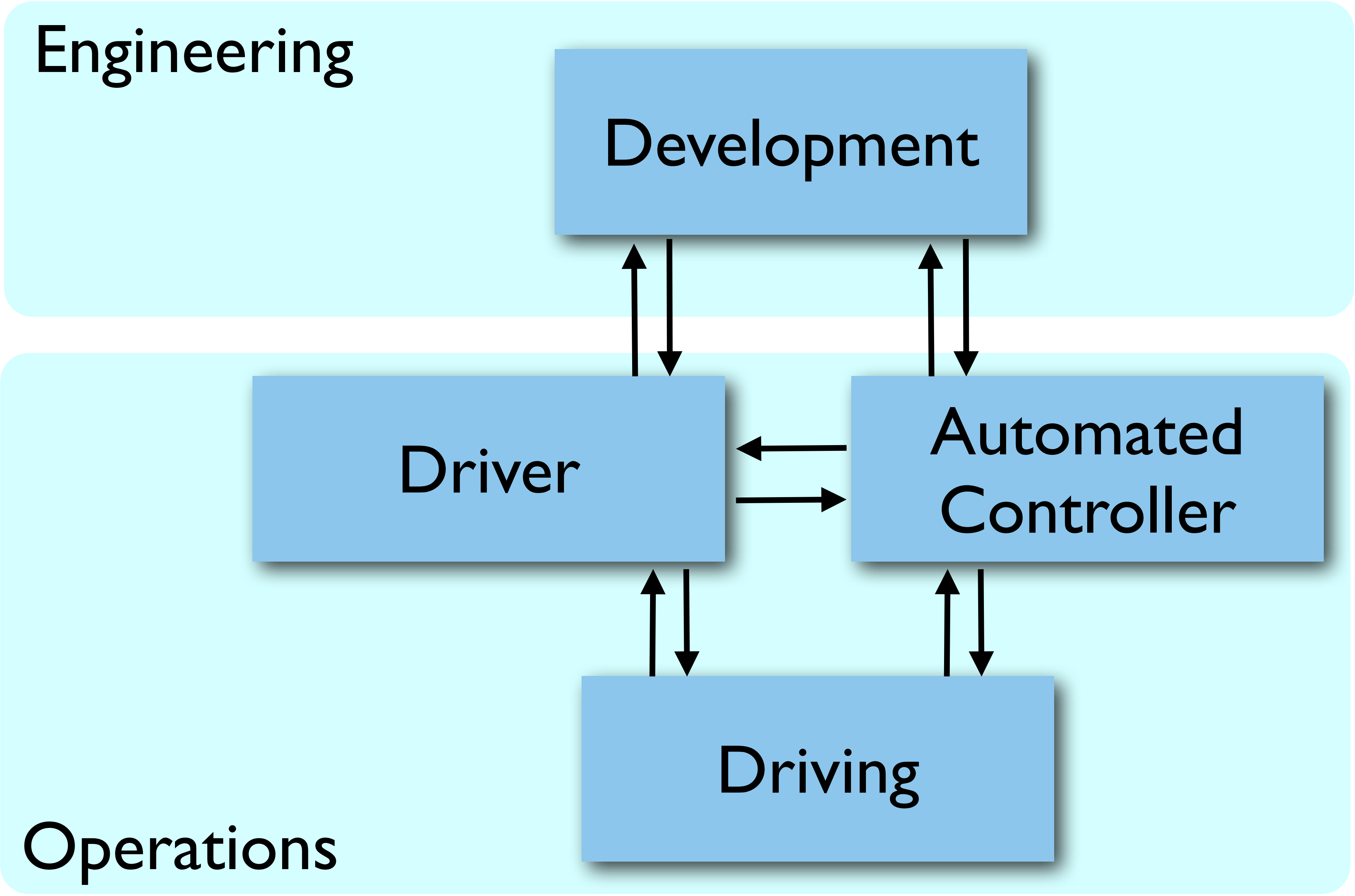




STAMP hierarchical structure

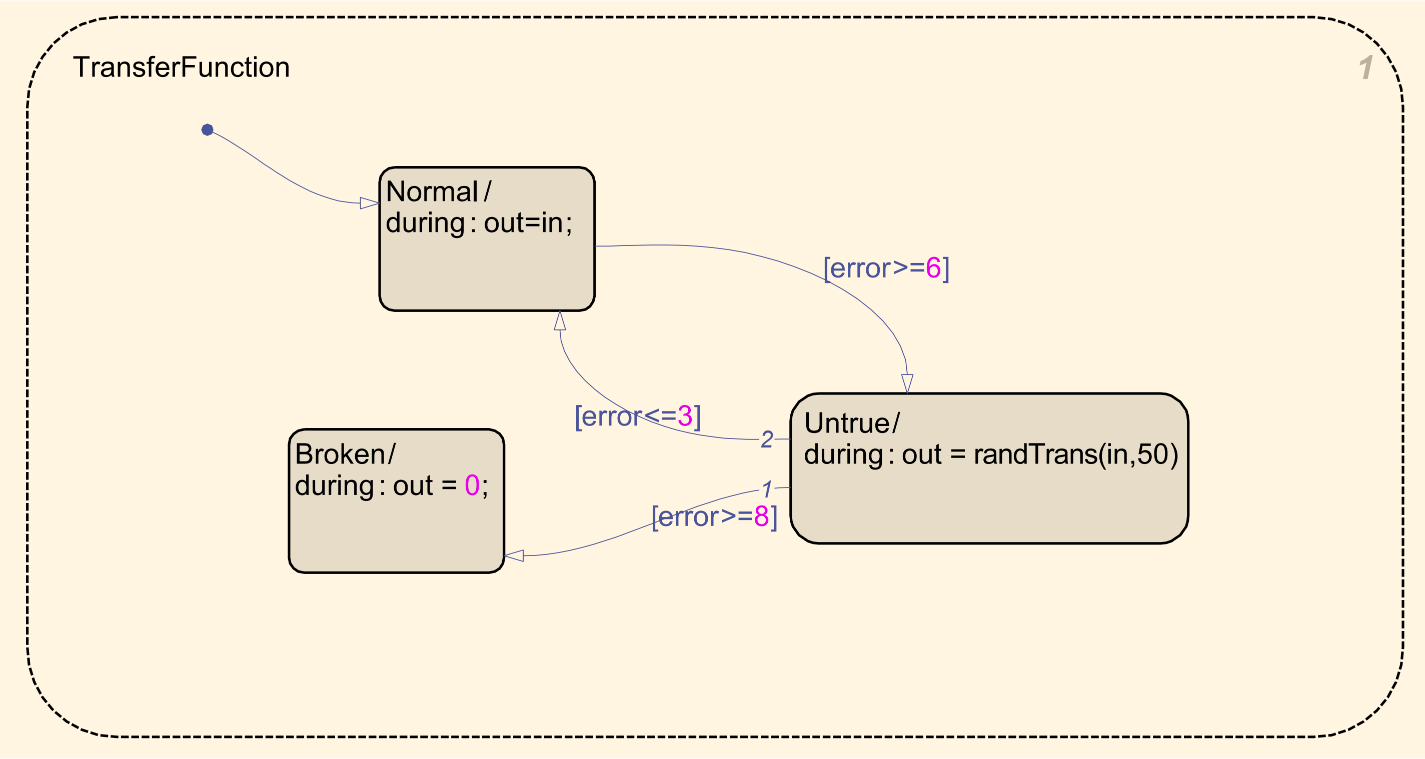
Safety case modules

Example structure mapping



4

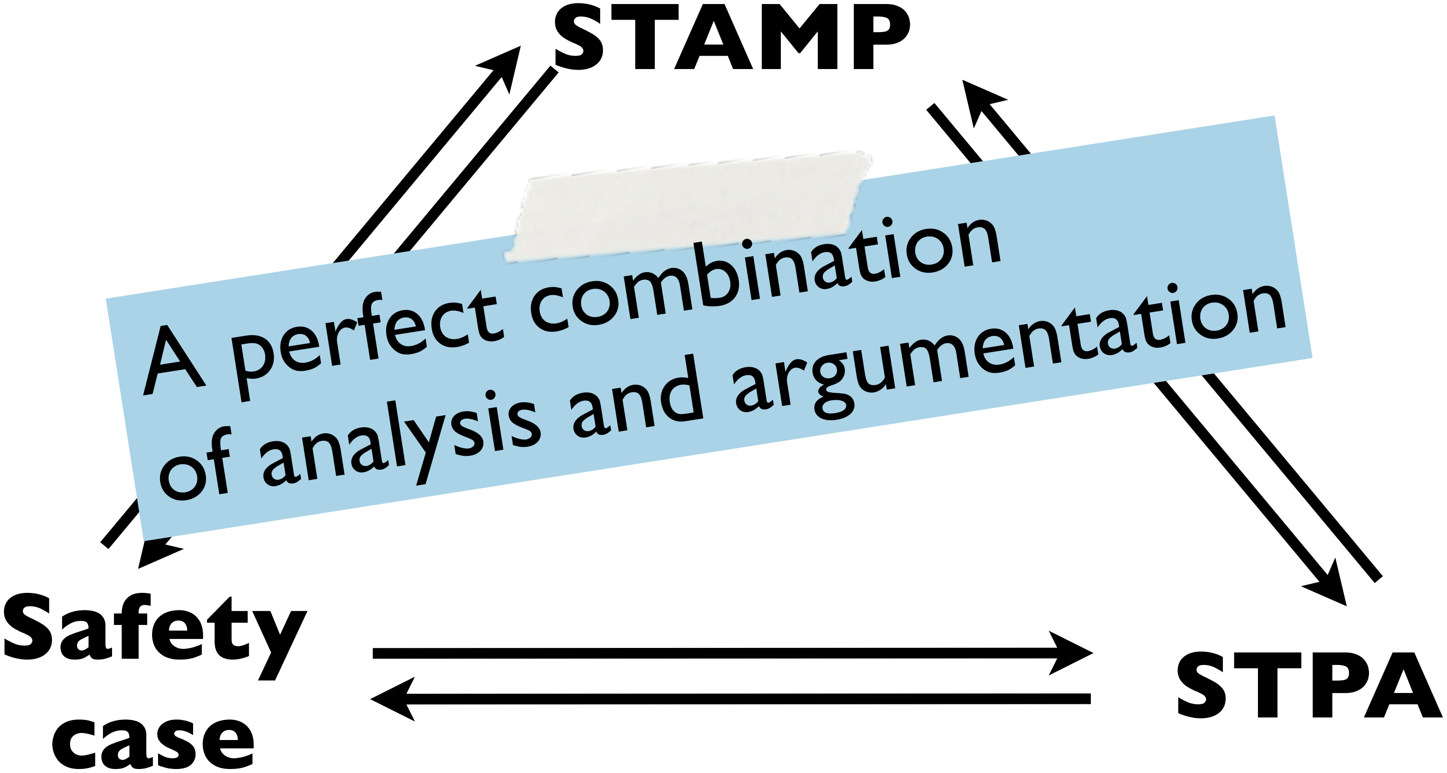
Process models



Sn1
Inspection in model

Sn2
Formal verification

1. **Hazard identification and avoidance**
2. **Degradation protection**
3. **Structure**
4. **Models**



STAMP

Safety case

STPA

A perfect combination of analysis and argumentation