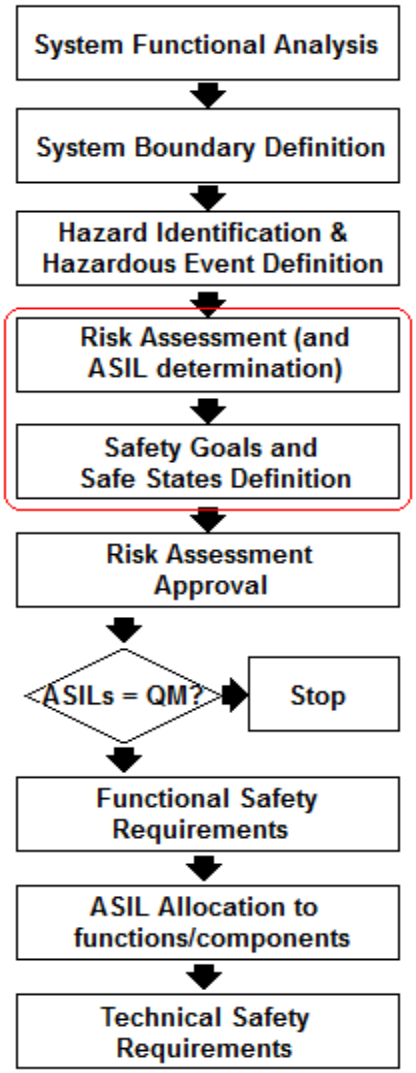
Automatic ASIL Decomposition

Background

- **ISO 26262 is the new automotive safety standard**
- **It uses ASILs – *Automotive Safety Integrity Levels* – to represent required levels of safety in a system**
- **ASILs can be decomposed over a system**
  - A high ASIL can be met by multiple redundant components working together, each with a lower ASIL
  - Many possible ways to decompose ASILs
- **Hence the importance of automating ASIL decomposition**
  - Allows the best strategies to be found more quickly

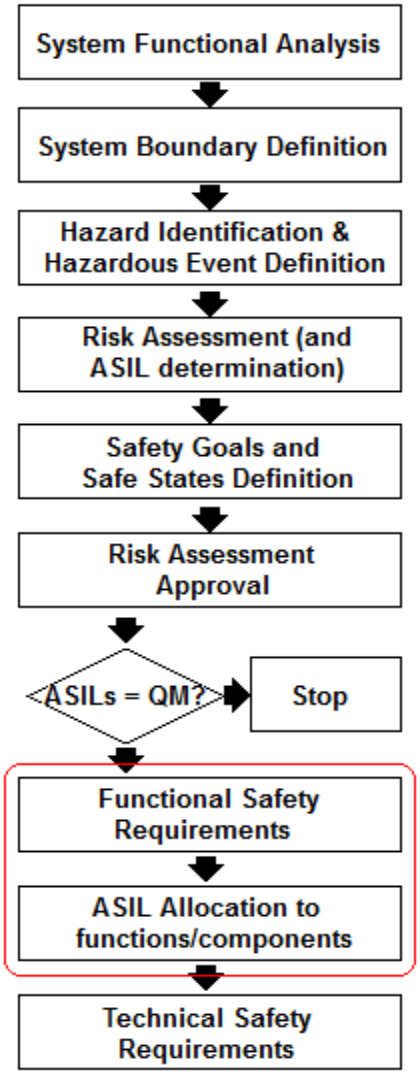
ASILs in ISO 26262

- **Definition of ASILs**
  - Carried out during Risk Assessment
  - Each Hazardous Event is assigned an ASIL (from A-D, or QM)
  - ASIL D is the highest, ASIL A the lowest
  - QM means no special safety requirement
  - Choice of ASIL is based on controllability, severity, and exposure time
  - Requires prior hazard analysis of system
- **Corresponding safety goals & safe states should also be defined**



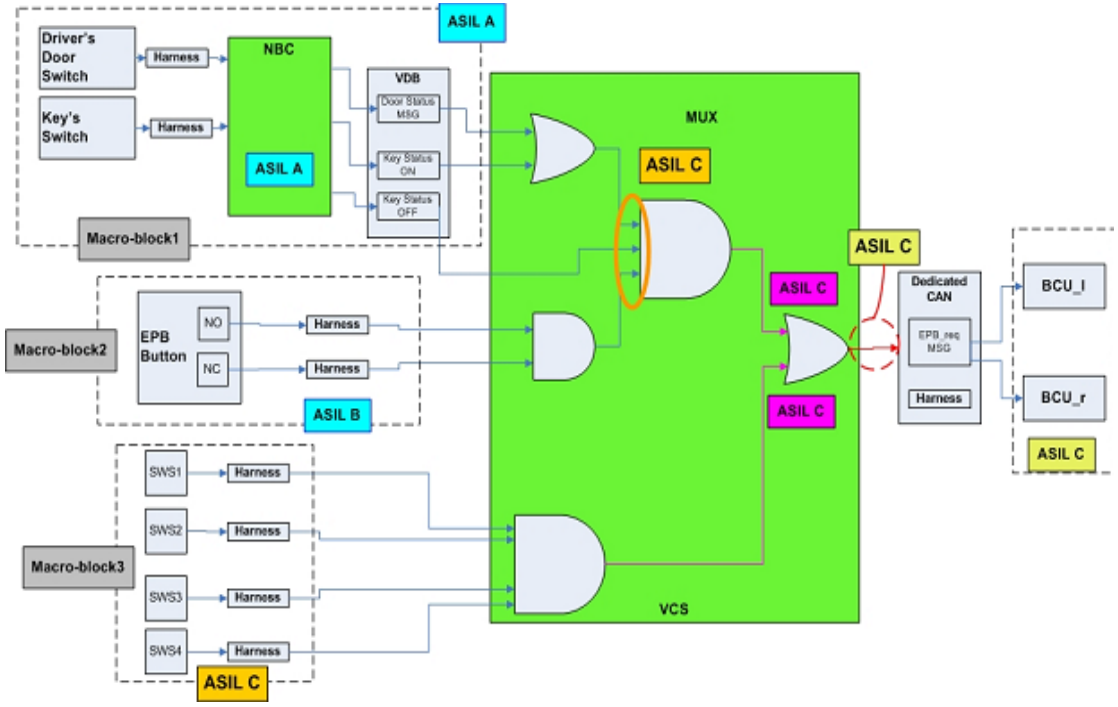
ASILs in ISO 26262

- **FTA is carried out as part of Functional Safety Requirements definition**
- **ASILs decomposed and allocated to system functions/components**
  - Decomposition is determined by system failure logic (i.e. AND vs OR)
  - The ASIL assigned is determined by an ASIL algebra (e.g. ASIL C = B + A)
- **Can be many possible ASIL assignments**
  - This makes it difficult to perform manually



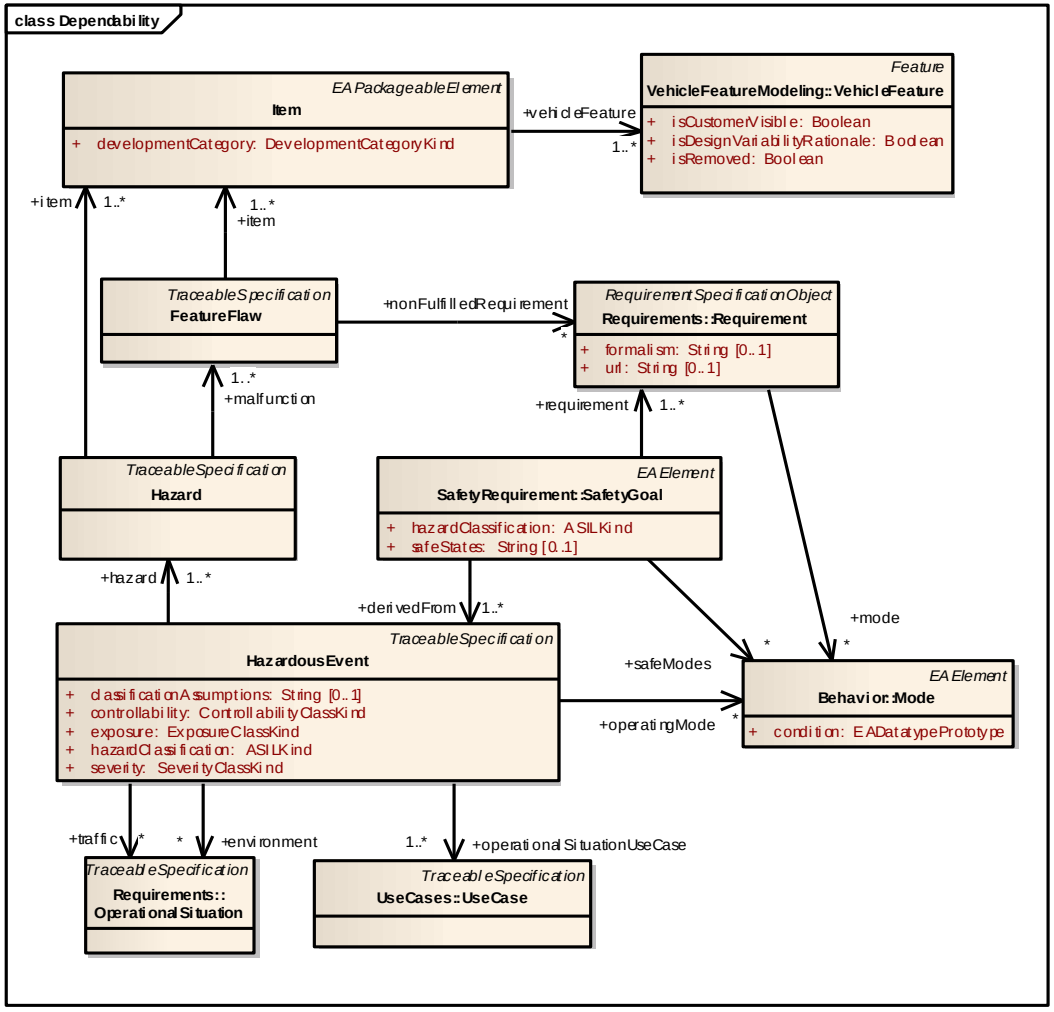
ASILs in ISO 26262

- **Decomposition allows greater granularity of safety requirements**
- **Not all parts of the system need to conform to the highest levels of safety**
- **Allows resources to be focused on the most critical elements**



ASILs in EAST-ADL

- **EAST-ADL provides support for hazard analysis and assignment of ASILs**
- **Hazards link to the error model, allowing them to be used in safety analysis**
- **Safety requirements are traceable across EAST-ADL layers**

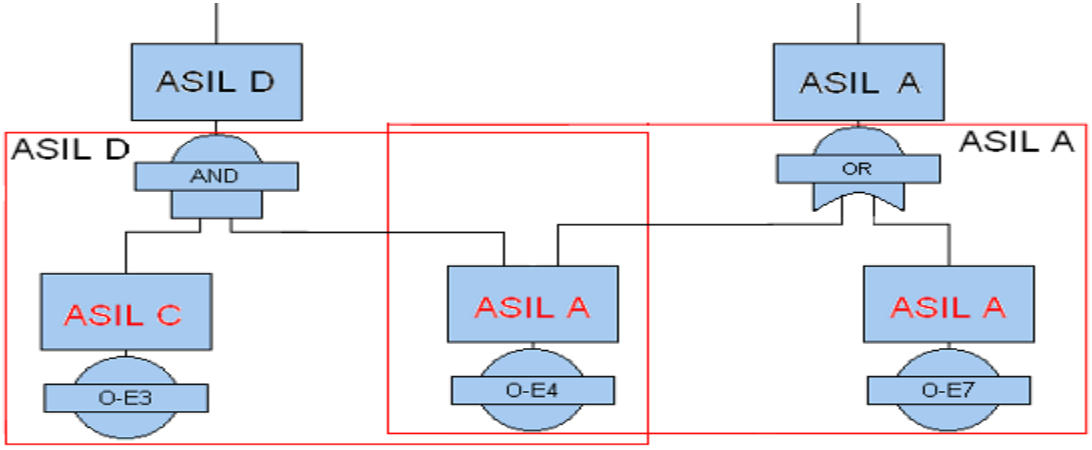


ASILs in EAST-ADL

- **Hazard Analysis performed on vehicle feature level**
  - ASILs are assigned to Hazardous Events
- **Initial safety analysis carried out on later levels**
  - FTA and/or FMEA can be applied on FAA/FDA models
  - Detailed information about failure modes is unnecessary
  - Propagation logic is what matters
  - ASIL decomposition & allocation can then take place
- **Can also make assumptions about ASILs for SEooC**
  - Safety Element out of Context – no context in which to perform hazard analysis

FTA and ASIL decomposition

- System failure logic is represented by fault trees
- Results of fault tree analysis (FTA) are cut sets
- Cut sets represent combinations of failures that can cause a hazard
- ASILs for that hazard can therefore be decomposed to the failures in the cut sets

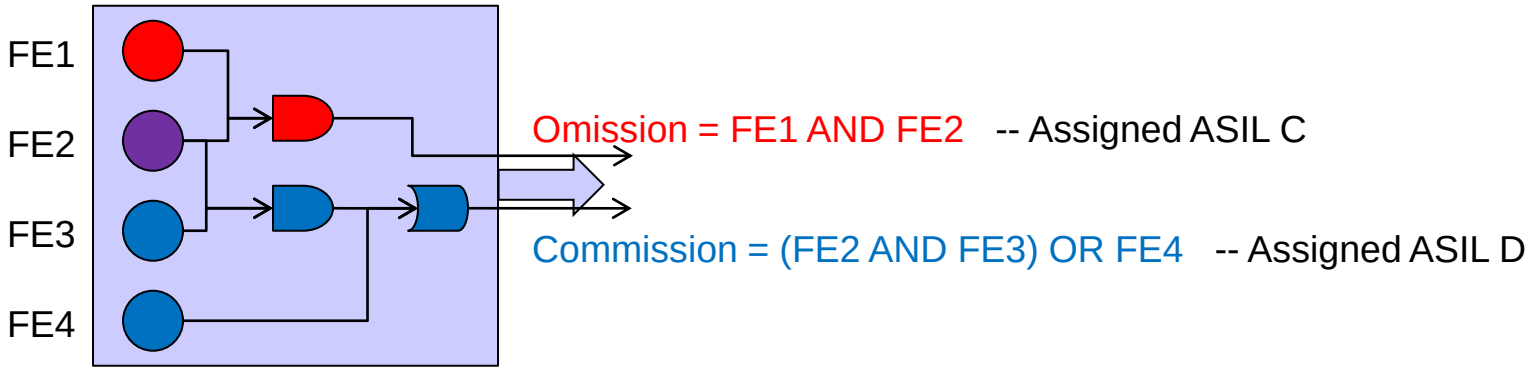


ASIL decomposition algorithm

- **ASILs from top level failures are decomposed across the cut sets that cause those failures**
  - Multiple failure events in a cut set mean all must occur to cause the system failure
- **Decomposition of ASILs is based on ASIL algebra:**
  - Each ASIL is worth one point (ASIL A = 1, ASIL B = 2 etc)
  - Sum of constituents should be >= overall ASIL
  - e.g. if A = 1 and C = 3, then A + A + A = C
- **For any given cut set, there are a maximum of $p = (m + 1)^n$ permutations**
  - p = assignments, m = max ASIL, n = number of events

ASIL decomposition algorithm

- Each cut set is covered by a decomposed ASIL
- Cut sets are iterated and for each one, all permutations of possible ASIL assignments are generated
- Assignments that meet requirements are kept, and those that do not are discarded
- Example: simple function with two system failures (omission + commission) and four basic events



ASIL decomposition algorithm

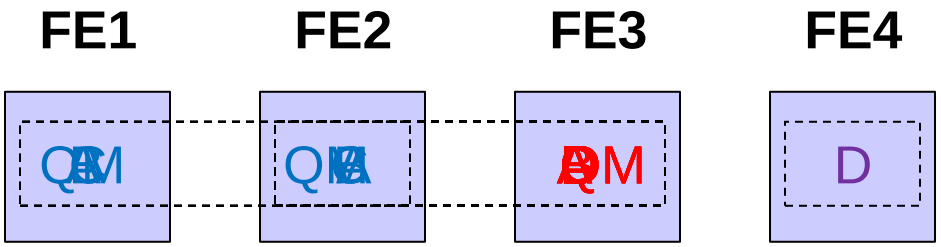
- All possible assignments for Omission (ASIL C)

FE1/FE2	FE1/FE2	FE1/FE2	FE1/FE2	FE1/FE2
QM/QM	A/QM	B/QM	C/QM	D/QM
QM/A	A/A	B/A	C/A	D/A
QM/B	A/B	B/B	C/B	D/B
QM/C	A/C	B/C	C/C	D/C
QM/D	A/D	B/D	C/D	D/D

- Blue ones do not meet requirements – discard
- Black ones are optimal – precisely meet requirements
- Red ones are potentially redundant (overly strict) but still need to be explored further

ASIL decomposition algorithm

- **For each valid assignment for Omission (ASIL C), we test possible assignments for Commission (ASIL D)**



Assignments Found:

QM : C : A : D
QM : C : B : D
QM : C : C : D
QM : C : D : D
A : B : B : D
A : B : C : D
A : B : D : D
B : A : C : D
B : A : D : D
C : QM : D : D

- FE4 can only be ASIL D (single cause)
- FE2 is set by Omission, so does not change with FE3
- To be accepted, FE2 + FE3 must be ASIL D

ASIL decomposition algorithm

- 65 results, of which all but 5 are redundant

QM:C:A:D	A:B:B:D	B:A:C:D	C:QM:D:D	D:QM:D:D
QM:C:B:D	A:B:C:D	B:A:D:D	C:A:C:D	D:A:C:D
QM:C:C:D	A:B:D:D	B:B:B:D	C:A:D:D	D:A:D:D
QM:C:D:D	A:C:A:D	B:B:C:D	C:B:B:D	D:B:B:D
QM:D:QM:D	A:C:B:D	B:B:D:D	C:B:C:D	D:B:C:D
QM:D:A:D	A:C:C:D	B:C:A:D	C:B:D:D	D:B:D:D
QM:D:B:D	A:C:D:D	B:C:B:D	C:C:A:D	D:C:A:D
QM:D:C:D	A:D:QM:D	B:C:C:D	C:C:B:D	D:C:B:D
QM:D:D:D	A:D:A:D	B:C:D:D	C:C:C:D	D:C:C:D
	A:D:B:D	B:D:QM:D	C:C:D:D	D:C:D:D
	A:D:C:D	B:D:A:D	C:D:QM:D	D:D:QM:D
	A:D:D:D	B:D:B:D	C:D:A:D	D:D:A:D
		B:D:C:D	C:D:B:D	D:D:B:D
		B:D:D:D	C:D:C:D	D:D:C:D
			C:D:D:D	D:D:D:D

ASIL decomposition algorithm

- **Can use heuristics to sort the remainder**

• Using sum of ASIL values (i.e. A=1, B=2, C=3, D=4):

QM:C:A:D	= 8
QM:D:QM:D	= 8
A:B:B:D	= 9
B:C:A:D	= 10
C:QM:D:D	= 11

• Using increasing points for ASIL values (A=1, B=10, C= 100 etc):

A:B:B:D	= 1021
QM:C:A:D	= 1101
B:C:A:D	= 1111
QM:D:QM:D	= 2000
C:QM:D:D	= 2100

- **This helps the analyst decide on the preferred option**
- **There may not be a single 'best' option**
- **More likely to be trade-offs between equivalent options**
  - Higher ASIL for one element means lower ASIL on another, and vice versa

ASIL assignment

- **Once an ASIL assignment for the failure modes has been chosen, ASILs can also be assigned to other parts of the model**
  - Can assign ASILs to input and output errors, to trace the propagation of failures
  - Can also assign ASILs to ports/interfaces of a component or function
  - ASILs can also be assigned to entire components/functions or subsystems
  - This also allows for ASILs to be assigned to process faults of functions

Optimisation of ASILs

- **Automatic ASIL decomposition is a good candidate for automatic optimisation algorithms**
- **When there are lots of possible assignments, exhaustive search becomes impractical**
  - Exhaustive search is subject to combinatorial explosion
  - For a single 4 event cut set, there are 625 permutations
- **In these cases, optimisation could be used**
  - Objective is to meet requirements at lowest cost (i.e. lowest total ASIL heuristic value)
  - Optimisation algorithms are more scalable as they are designed to explore large search spaces efficiently

Future tool integration

- **Plan for rest of MAENAD is to integrate this technology into tools**
  - HiP-HOPS provides the analysis capability
  - Papyrus provides modelling capability
  - Intention is to start ASIL decomposition directly from Papyrus as with other analysis plugins
- **Some challenges still remain**
  - Find a more scalable approach e.g. with optimisation
  - Could use a hybrid method:
    - Exhaustive algorithm for smaller models
    - Optimisation for larger models
  - How to present results and where to store them in the model

Summary

- EAST-ADL support for hazard analysis and ASIL decomposition is important to conform to ISO 26262
- But ASIL decomposition is difficult to perform manually
- Prototype version implemented in HiP-HOPS
- Allows more rapid determination of ASIL assignments
- Can use optimisation to increase efficiency