# Coral: a tool for Compositional Reliability and Availability analysis†

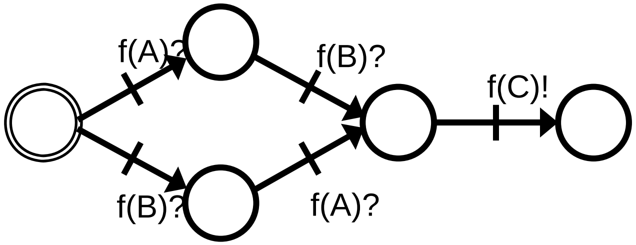
**Hichem Boudali[1], Pepijn Crouzen[2], and Mariëlle Stoelinga[1].**

**[1]Formal Methods and Tools group**
**CS, University of Twente, NL.**
**[2]Dependable Systems and Software group,**
**CS, Saarland University, Germany**

†This research has been partially funded by
the EU under grant IST-004527 (ARTIST2)

# Introduction

Science

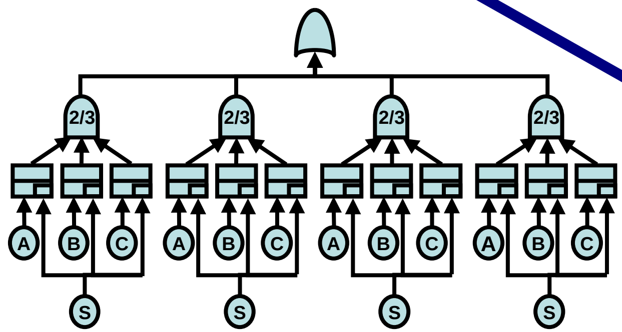



**Input/Output Interactive Markov Chains**

- *Precise*
- *Efficient*

- *Graphical*
- *Easy to use*

Dynamic Fault Trees




Engineering

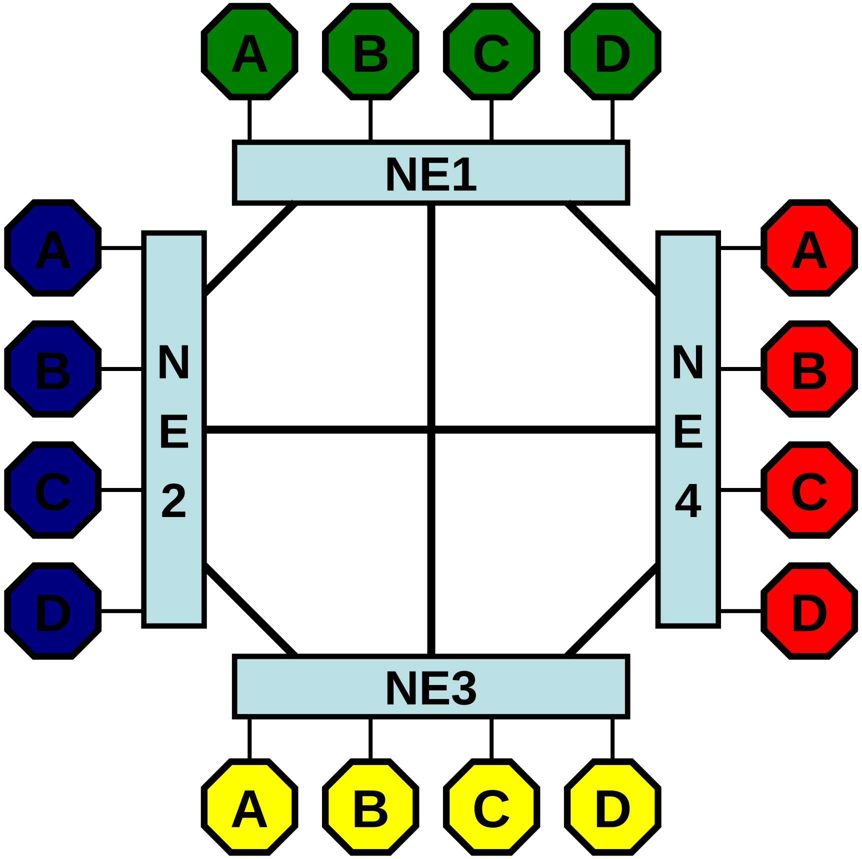
# But… DFT Drawbacks

- State-space explosion.
- Ambiguous syntax and semantics.
- Lack of modularity:
  - Dynamic modules can not be reused.
  - Restrictions on spares and dependencies.
- Existing analysis technique is hard to extend or modify.

# Outline

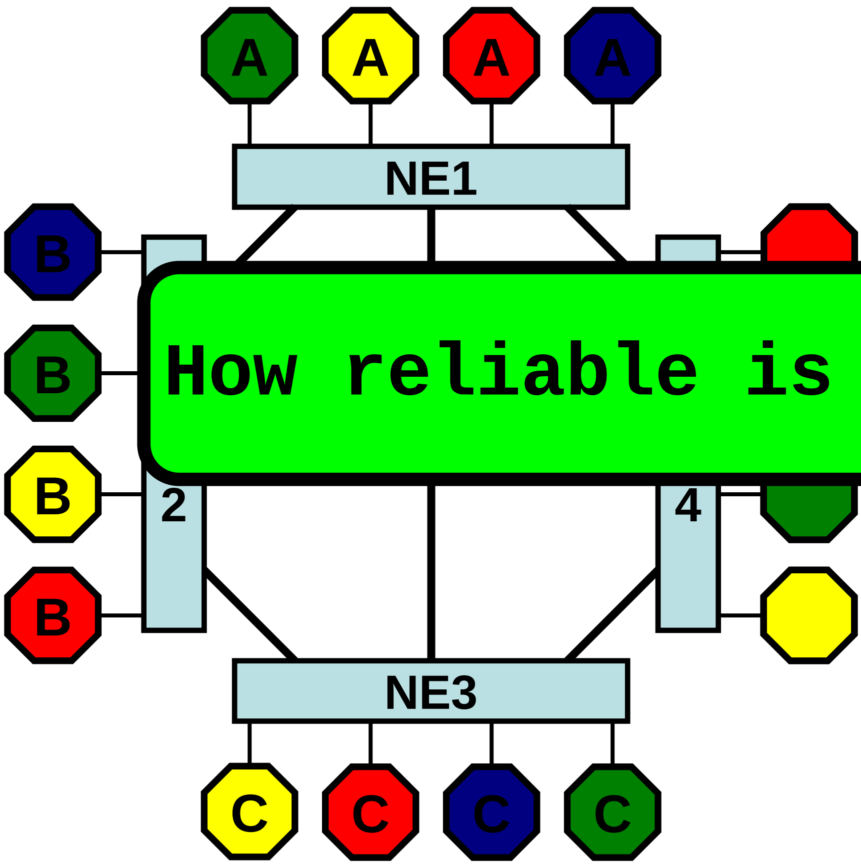
- Case study: FTPP system.
- Dynamic fault trees (DFT).
- DFT semantics in terms of I/O-IMCs.
- Deep compositionality.
- Prototype tool chain.
- Conclusion.

# Case study: FTPP




- 16 processors divided into 4 groups
- 4 network elements connect the processors
- Per group 2 processors must be operational
- Different configurations are possible

# Case study: FTPP

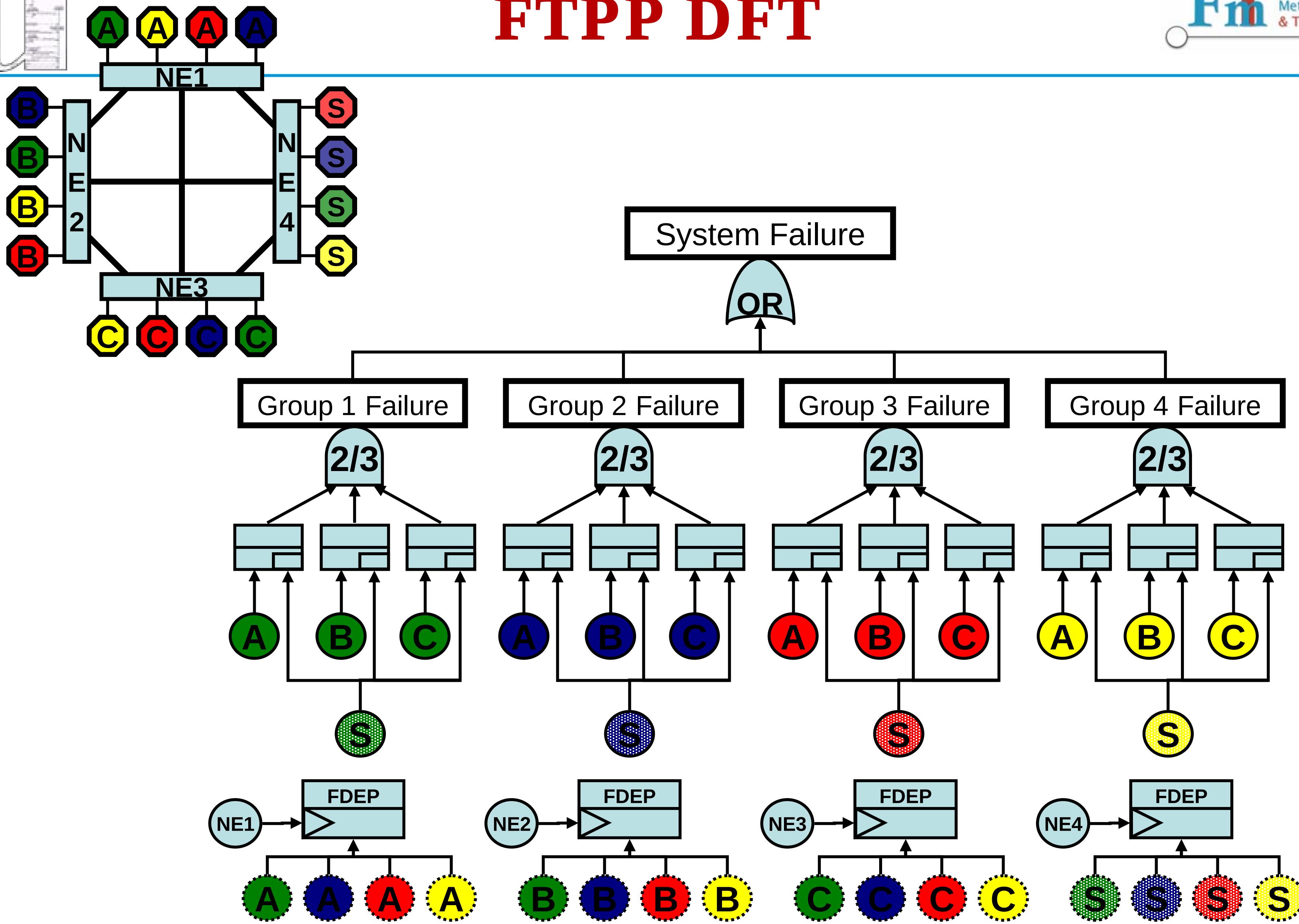



- 16 processors divided into 4 groups
- 4 network elements connect the processors

**How reliable is each configuration?**

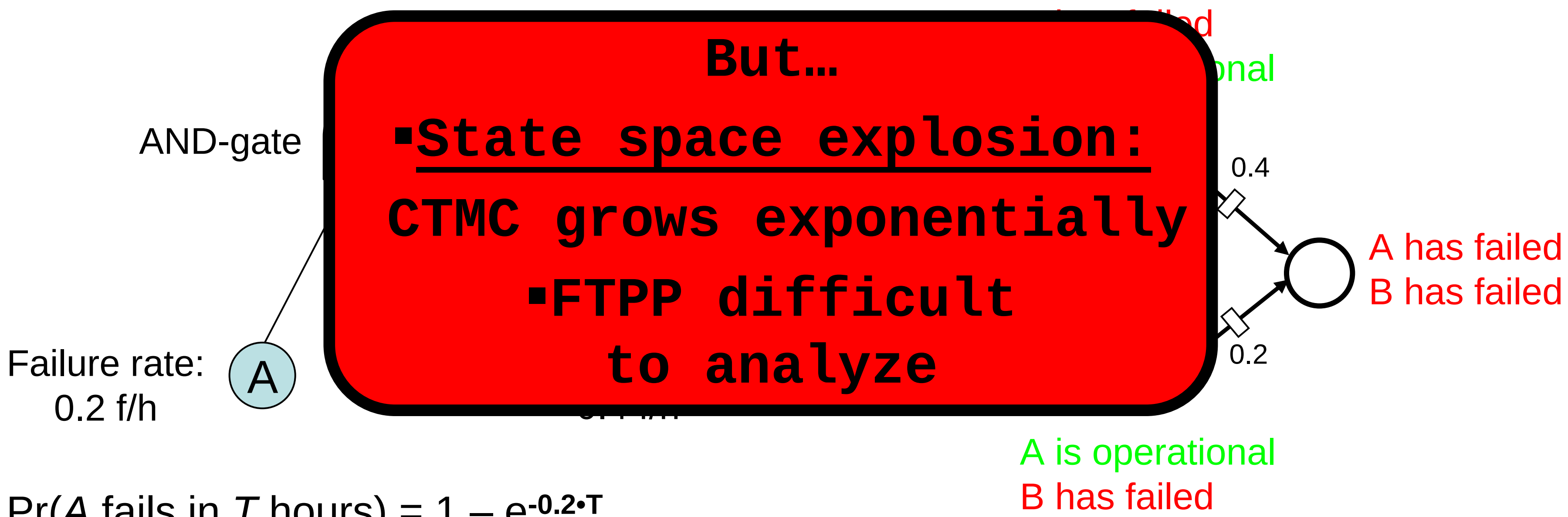
- Different configurations are possible
- Dynamic redundancy management is possible

# FTPP DFT


NE1
NE2
NE3
NE4
System Failure
OR
Group 1 Failure
Group 2 Failure
Group 3 Failure
Group 4 Failure
2/3
FDEP

# Monolithic DFT analysis
# [Dugan et al. 1992]

- Convert the DFT into a Continuous-time Markov chain.
- Analyze CTMC using standard solution techniques.
- In special cases binary decision diagrams can be used!



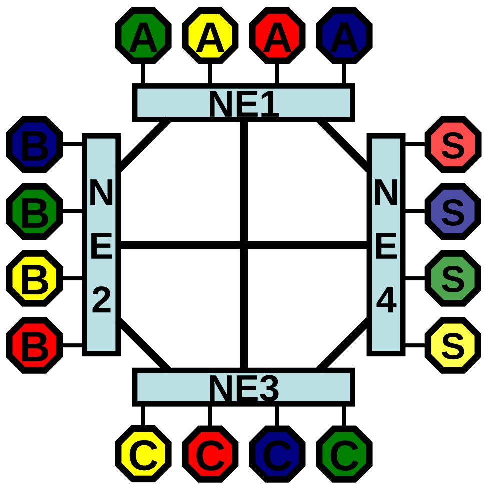

Pr($A$ fails in $T$ hours) = $1 - e^{-0.2 \cdot T}$

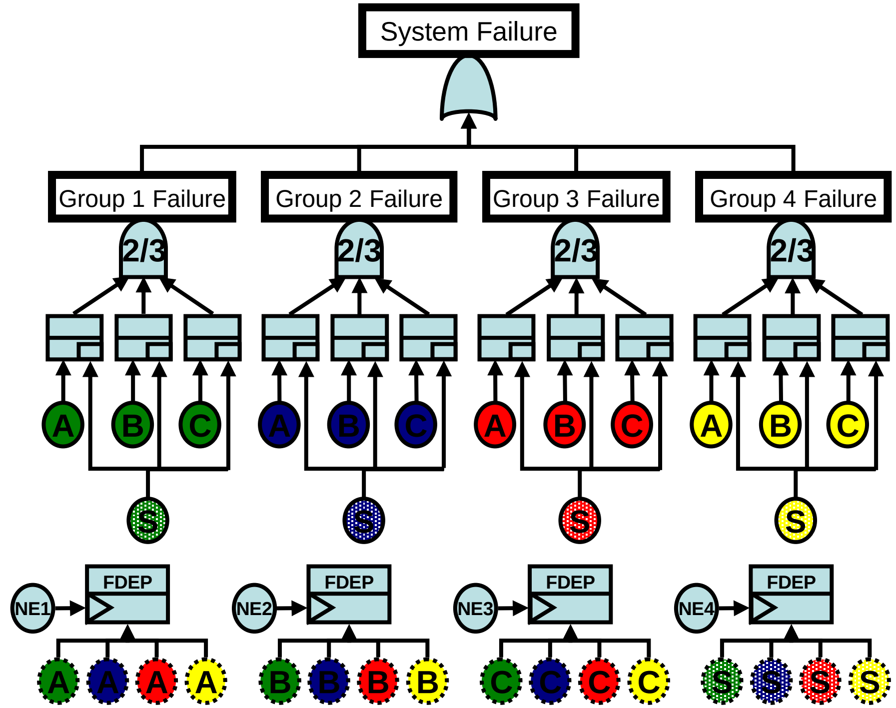
$A$'s Mean time to failure = 1/0.2 = 5 hours

***Unreliability* =**

***Prob*[Reaching ⬤ in time T]**

# FTPP Results



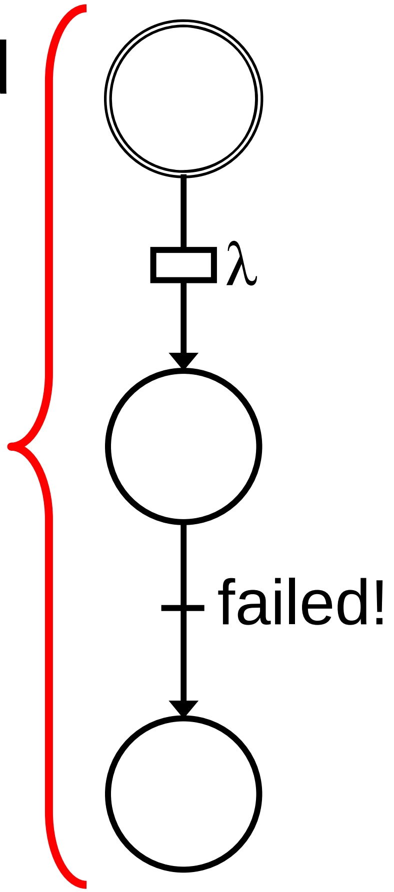




| Analysis method | Max number of states | Max number of transitions | Unreliability (T=10) |
|---|---|---|---|
| Monolithic | 32757 | 426826 | $2.55479 \cdot 10^{-8}$ |
| | | | |

# What's behind it?

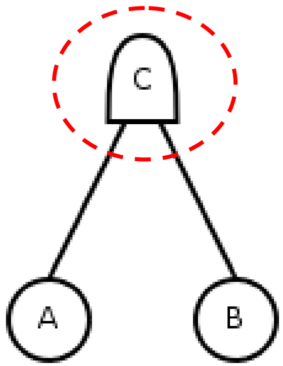
- Model local behavior
- Combination of **I/O automata** and **CTMC**; closely related to **IMCs**
- **Markovian** transitions (CTMC)
- **Interactive** tran
- Action signature
  - ? - Input actions
  - ! - Output actions
  - ; - Internal actions

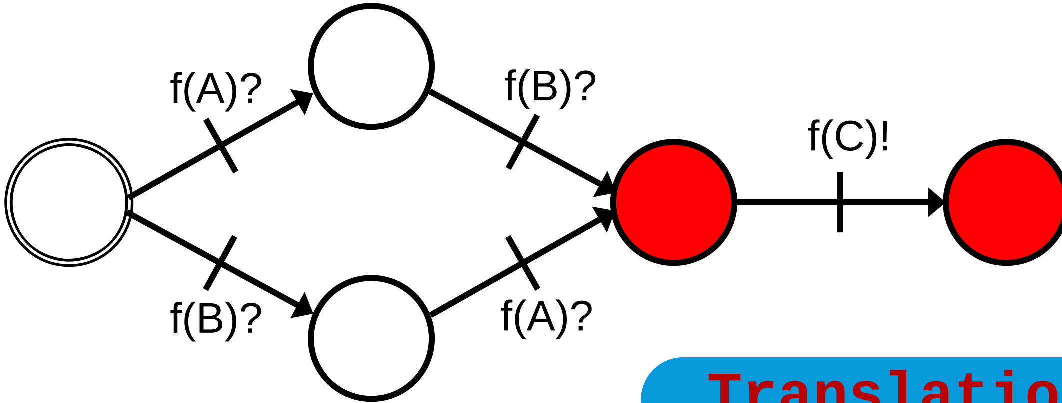
**I/O-IMC for Basic event**



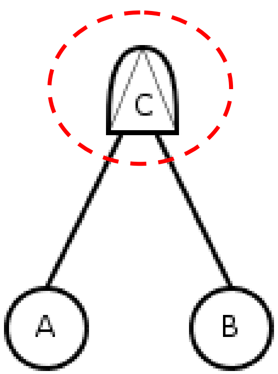

## Input/Output Interactive Markov Chains (I/O-IMC)

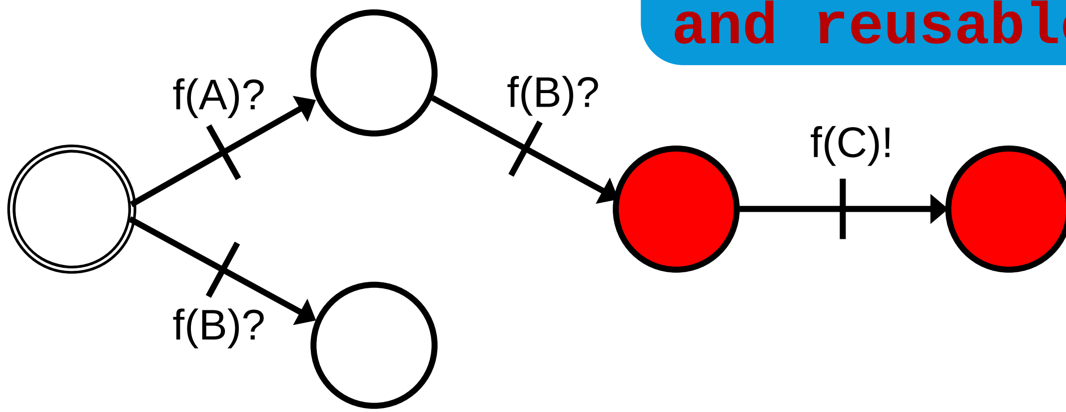
# DFT semantics
## DFT gate to I/O-IMC



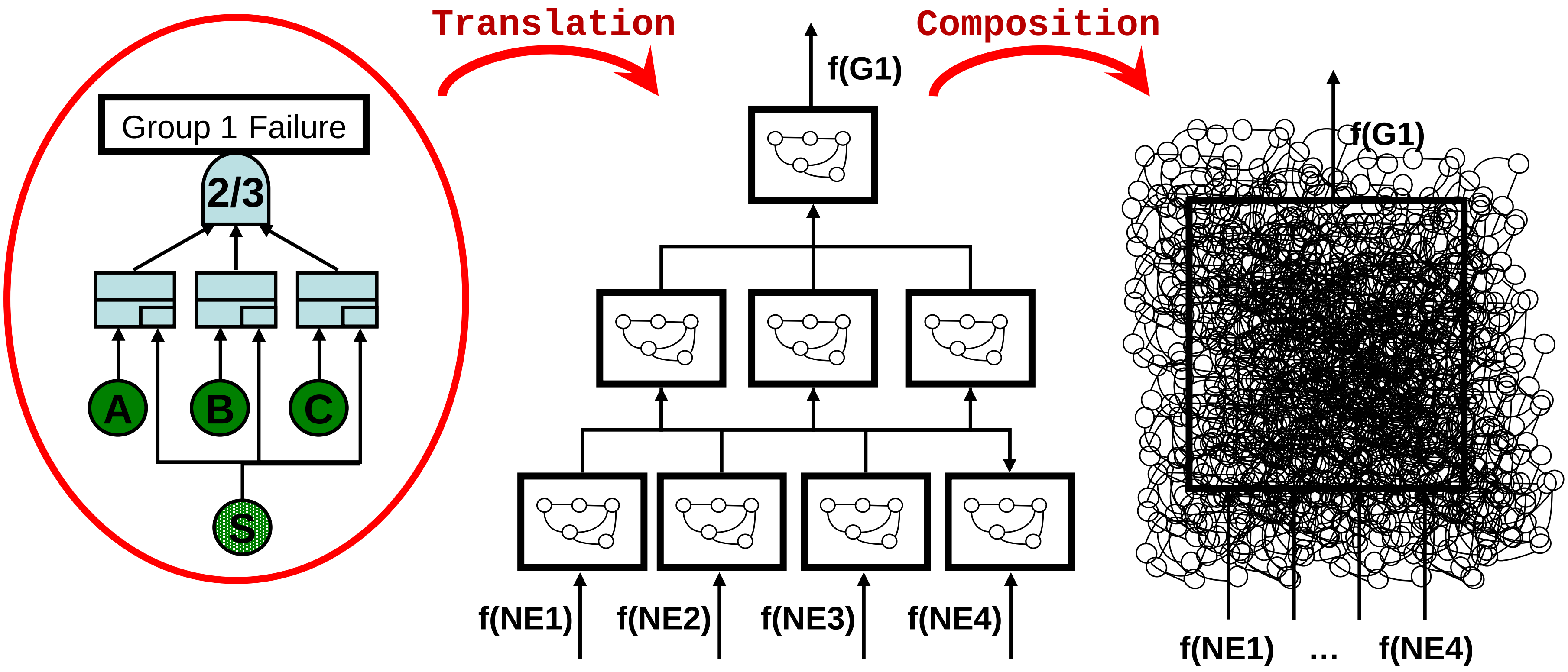

**Translation is scalable and reusable!**



# What is deep compositionality?

- Semantics of a DFT arises naturally as composition of the semantics of its building blocks




- But: This may lead to huge models.

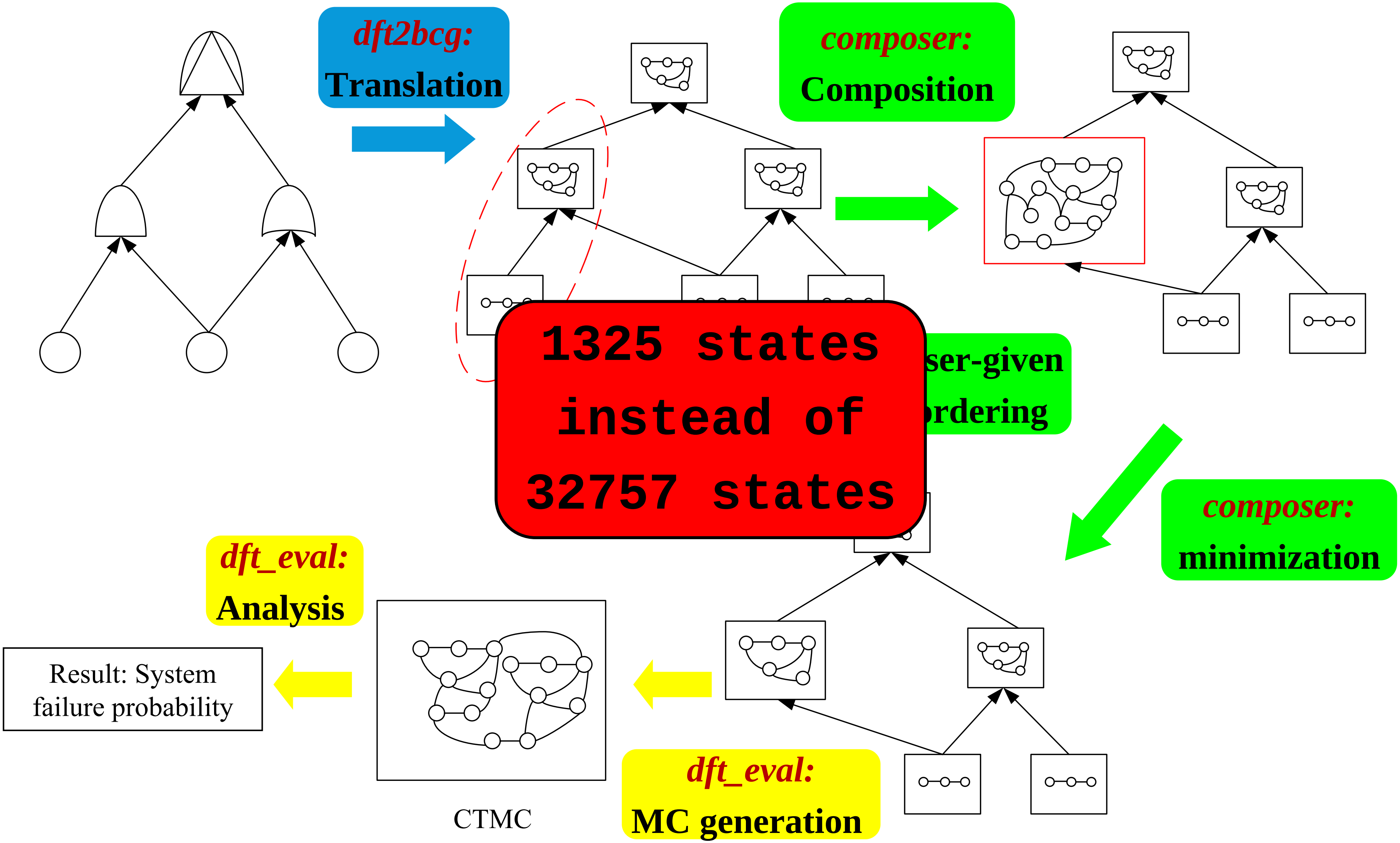
# Why use deep compositionality?

- Formally define semantics
- Many useful techniques
  - Combining models: **Composition**
  - Refining models: **Abstraction**
  - Minimizing models: **Aggregation**
  - Reusing models: **Renaming**

**Combat State-space explosion**

- Well supported by CADP toolset (VASY/INRIA)
  - Widely used in industry (e.g. Airbus)

# Prototype tool chain
## Coral – DFT analysis



# Conclusion:

## *How we tackled drawbacks*

- State-space explosion.
- Ambiguous syntax and semantics.
- Lack of modularity:
  - Dynamic modules can not be reused.
  - Restrictions on spares and dependencies.
- Existing analysis technique is hard to extend and/or modify.

**Compositional Aggregation**

**DAG** → **I/O-IMC**

**Formal translation**

**Renaming!**

**Lifted!**

**Extensions at the lowest level**

# Future work

- Fully automated tool
- More aggressive state reduction
  - Weaker equivalences
  - Interface constraints
  - Phase-type minimization
- Further extensions to DFT modeling capabilities
  - Extension to non-exponential distributions
  - New DFT building blocks
- Apply deep compositionality to other engineering formalisms!
  - E.g. Architectural description languages like AADL

# References

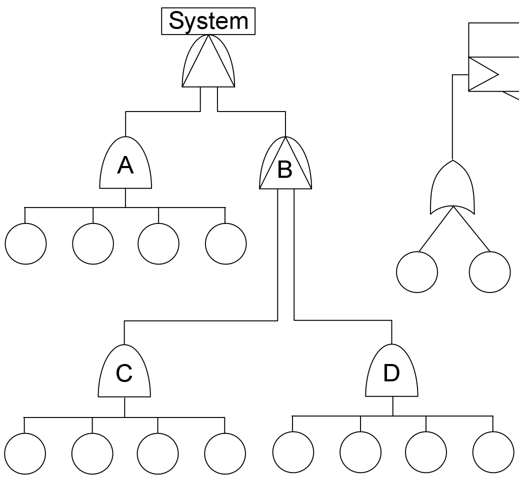

- H. Boudali, P. Crouzen, M. Stoelinga. “Dynamic Fault Tree analysis using Input/Output Interactive Markov Chains”, to appear, DSN 2007 proceedings.
- H. Boudali, P. Crouzen, M. Stoelinga. “A compositional semantics for Dynamic Fault Trees in terms of Interactive Markov Chains”, submitted to ATVA 2007.

- More info:

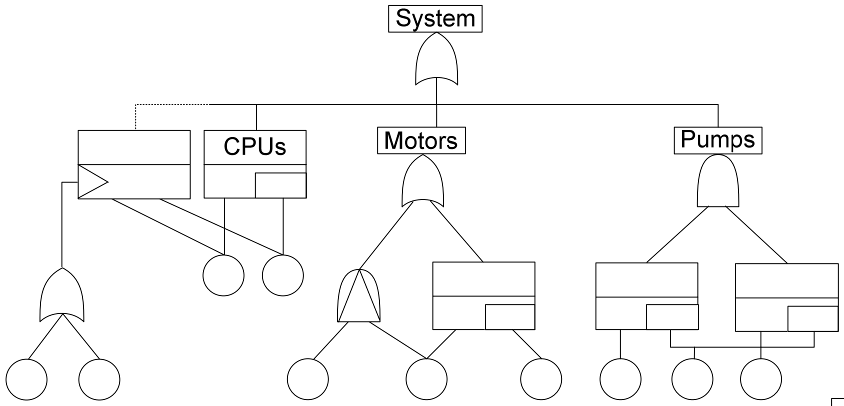
  - crouzen@alan.cs.uni-sb.de
  - hboudali@cs.utwente.nl
  - marielle@cs.utwente.nl


# The END!

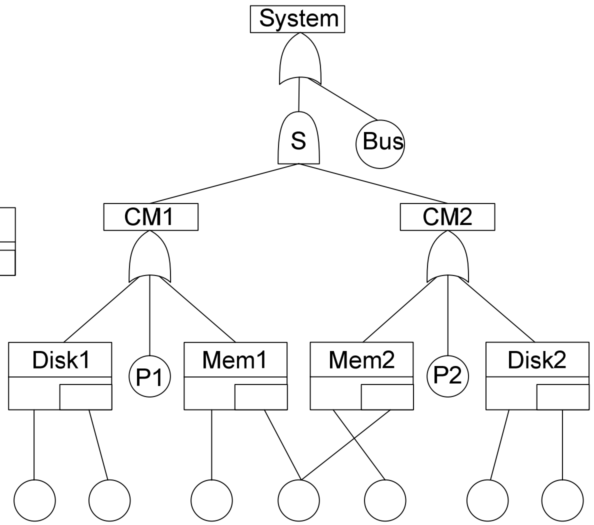
# Case studies




(a) The cascaded PAND system



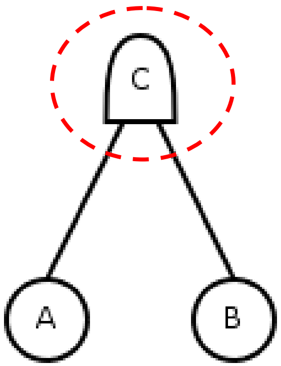

(b) The cardiac assist system

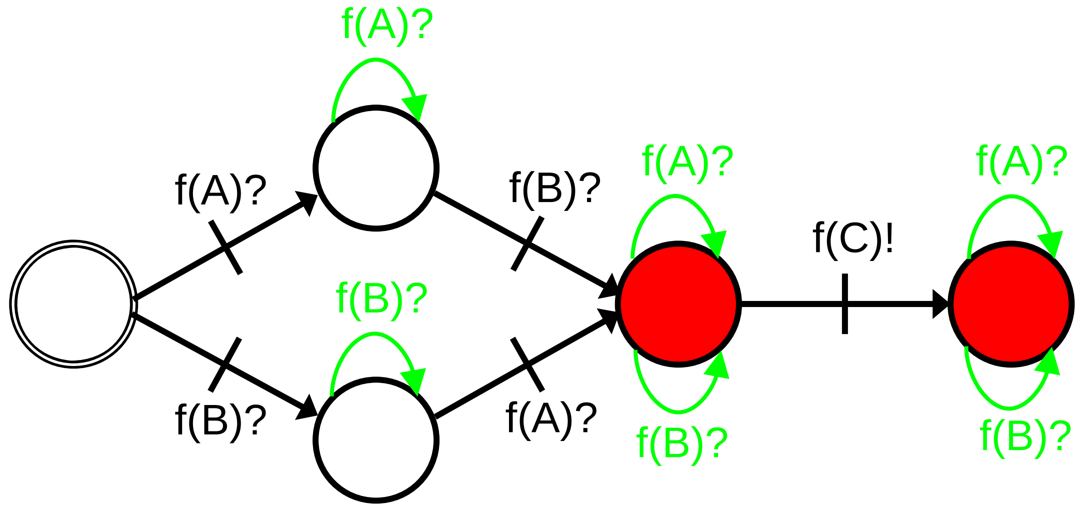


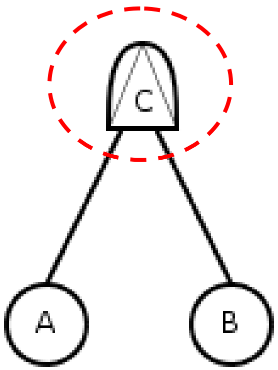

(c) A multi-processor distributed computing system

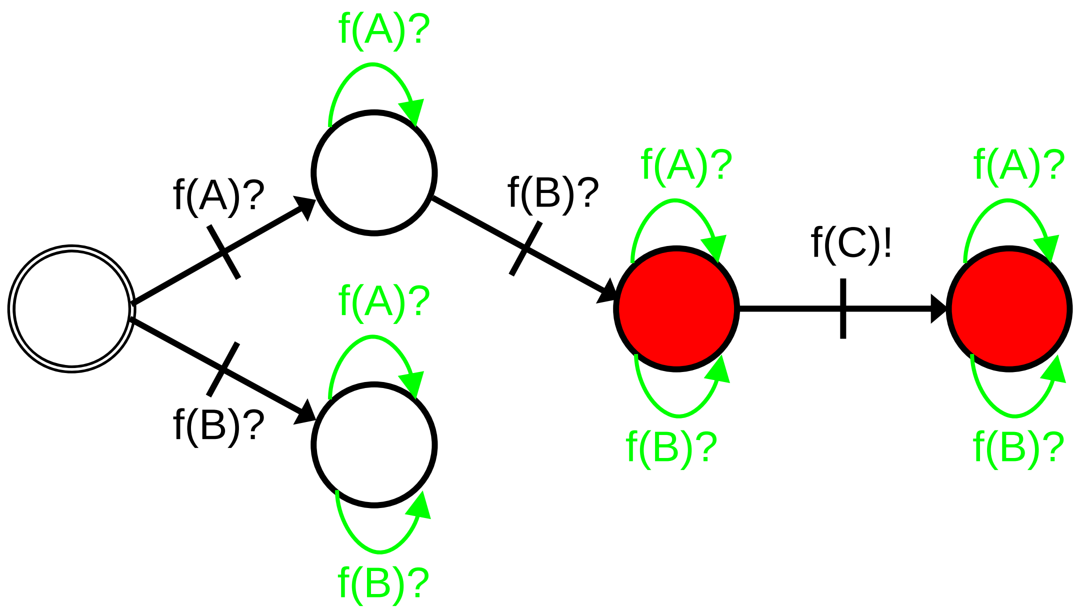
| Case study | Analysis method | Max number of states | Max number of transitions | Unreliability (T=1) |
|---|---|---|---|---|
| (a) | Monolithic | 4113 | 24608 | 0.00135668 |
| (a) | Compositional | 132 | 426 | 0.00135668 |
| (b) | Monolithic | 8 | 10 | 0.657900 |
| (b) | Compositional | 36 | 119 | 0.657900 |
| (c) | Monolithic | 253 | 1383 | $2.00025 \cdot 10^{-9}$ |
| (c) | Compositional | 157 | 756 | $2.00025 \cdot 10^{-9}$ |

# DFT semantics
## DFT gate to I/O-IMC



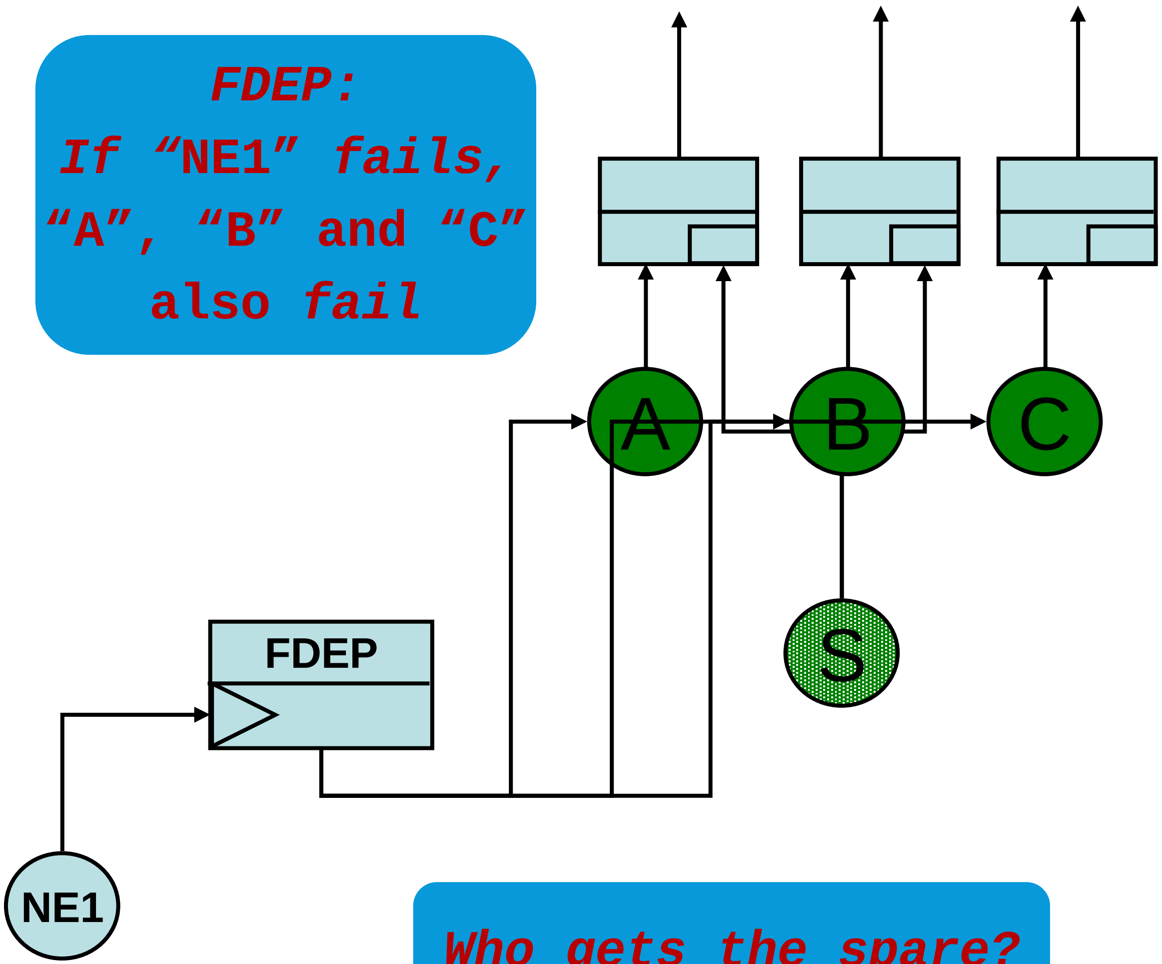



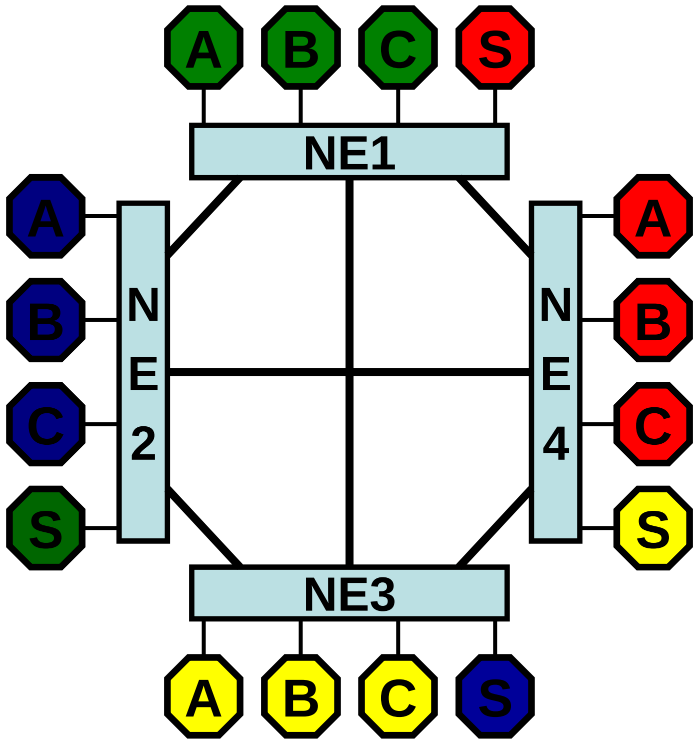



# Non-determinism!

***FDEP:***
***If "NE1" fails,***
***"A", "B" and "C"***
***also fail***




***Who gets the spare?***