Fail-Safe vs. Fail-Operational

A System relies on an Input for executing its Mission.

- **Fail-Safe**
  - After an Initial Error to the Input, the System fails, but assumes some Final Safe State, that will not cause further harm, but it can no longer perform its Mission.
  - A Secondary Error is not considered.

- **Fail-Operational**
  - After an Initial Error to the Input, the System has some Alternate Input enabling it to continue its Mission for a Limited Time.
  - After some Time or Secondary Error the System may
    - fail or
    - go into a Final Safe State.
  - A Ternary Error is (usually) not considered.
Redundancy Classes proposal

• **No Redundancy**: Fail safe – after loss immediate transition to a local safe state

• **Extended wear-out**: Ignore initial failure, second failure will lose system functionality

• **Fail gracefully**: Redundant data after initial failure used to mitigate transition to a system safe state within limited time to avoid secondary failure

• **Lip home**: Continue mission for extended period, maybe with reduced performance, but no reduction of safety level

A personal view, based on experience, not deemed normative nor complete (yet)!
The Problem of Availability

All elements are of the same “Quality”.

During Operation

B1 OR B2 must be operational

Subsystem A

Subsystem B1

Subsystem B2

Subsystem C

Reliability Block Diagrams
The Problem of Availability - Start-Up

During Operation
- Subsystem B
- Subsystem C

At Start-Up
- B1 OR B2 must be operational
- B1 AND B2 must be operational

All elements are of the same “Quality”.

Subsystem A
Subsystem B
Subsystem B’
Subsystem C
The Problem of Availability - Operation

During Operation and Start-Up

Subsystem A → Subsystem B' → Subsystem B → Comparator → Subsystem C

B1 AND B2 must be operational and agree

This is different from IT/Telecom!

Subsystem A → Subsystem B → Subsystem C

Still “better” than this!

All elements are of the same “Quality”.

IEEE contribution
THANK YOU

ETHERNOVIA  max.turner@ethernovia.com