

Architectural Trade-offs: Dependability

FDIR approach to system and software

- The **verification** of the system FDIR is difficult and requires tuning experimentally a lot of parameters in the software FDIR component
→ cost and delay in integration
- The system FDIR concept and the software FDIR component claim to have a “general logic” (e.g. reconfiguration levels), but happen to be a **toolbox** to monitor and reconfigure more or less everything.
→ over design
- For each mission, the “general logic” is twisted to fit the numerous **particular cases** that are discovered when running scenarios.
→ uncontrolled design
- FDIR “**emerge**” from the engineering process by necessity rather than by conscious intention.
→ no dedicated process, no support tools, difficult verification

- **Consistent** and **timely** FDIR conception, development, V&V
- **Fit-for-purpose** FDIR
- **Coherent, repeatable** Process and Methodology
 - Applicable from early Software and System architectural design
 - Coherent with System development lifecycle
 - Milestones with measurable FDIR maturity
 - Oriented towards Mission and System RAMS requirements
- Advanced **modelling** and analysis techniques
 - Specification of nominal, erroneous, FDIR behavior
 - Automated FTA, FMECA, Failure Propagation and FDIR Analyses
- Reference FDIR **architecture**
- Underpinnings for **Failure and Anomaly Management Engineering**

Main R&D result: FAME (1/3)

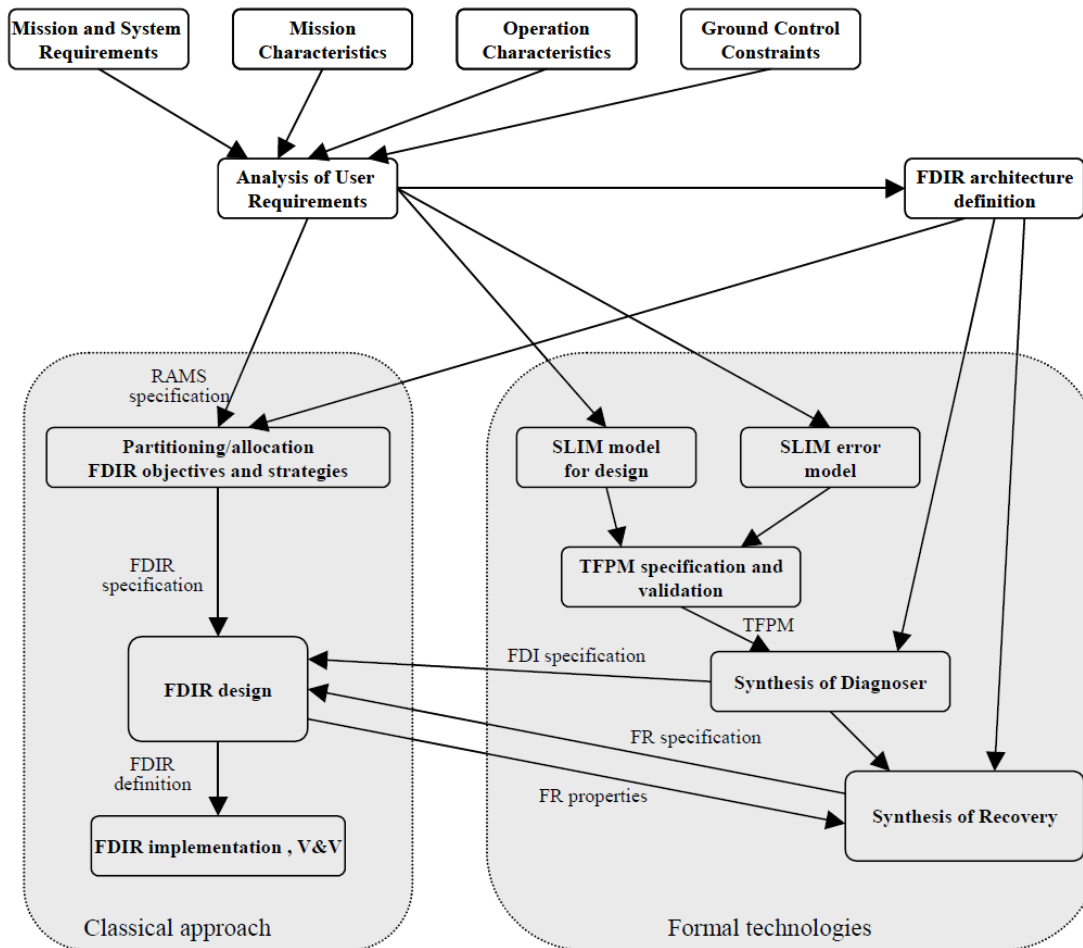
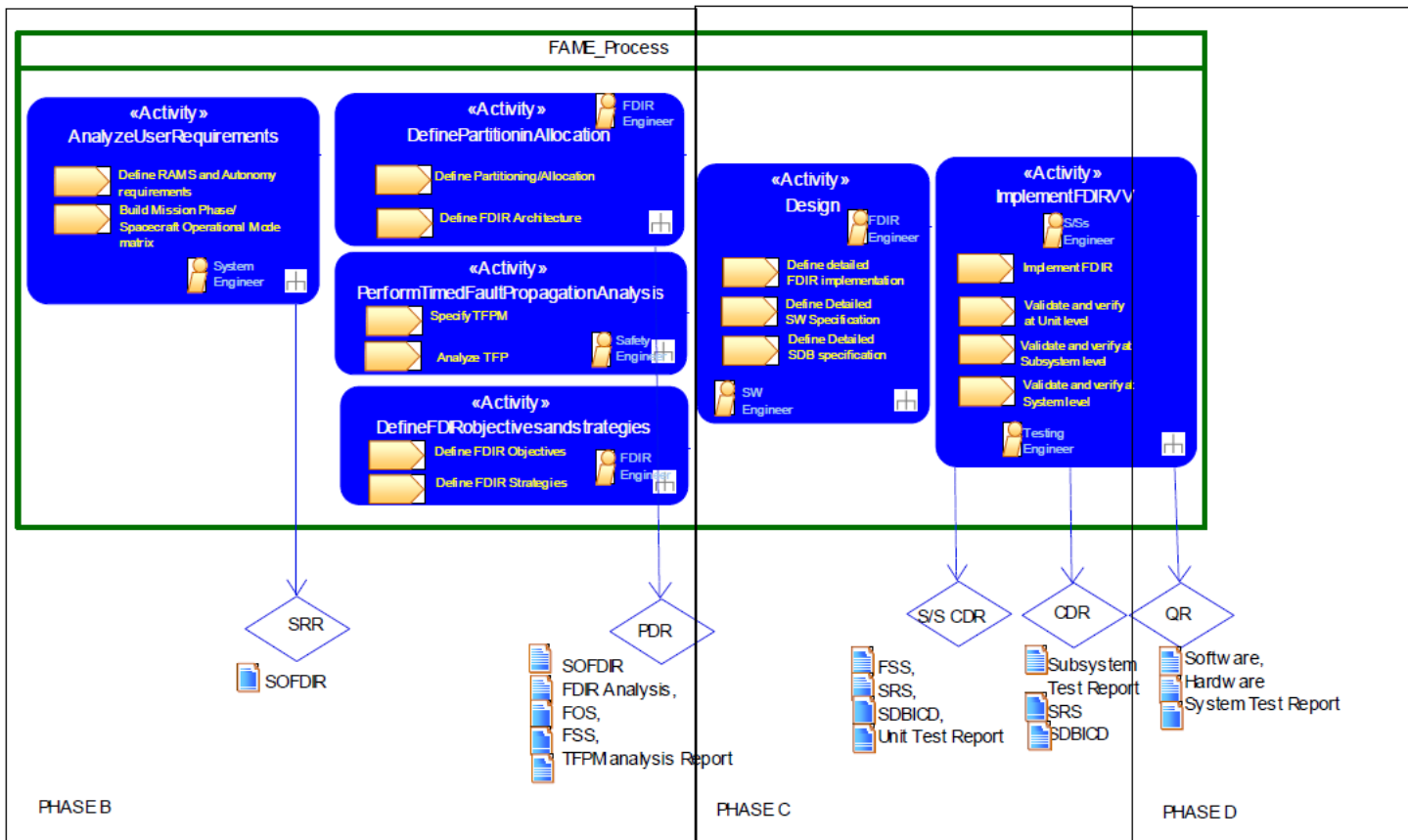


Figure 7-1 FAME Process overview

- FDIR comes from **system RAMS** requirements
- FDIR has **objectives, strategies** (different per op. mode and phase) and architecture
- FDIR must be verified
- FDIR is supported by a **model based approach**:
 - architectural model,
 - error model,
 - Timed Fault Propagation model.
- Diagnoser and Recovery controller may be **generated**

Main R&D result: FAME (2/3)

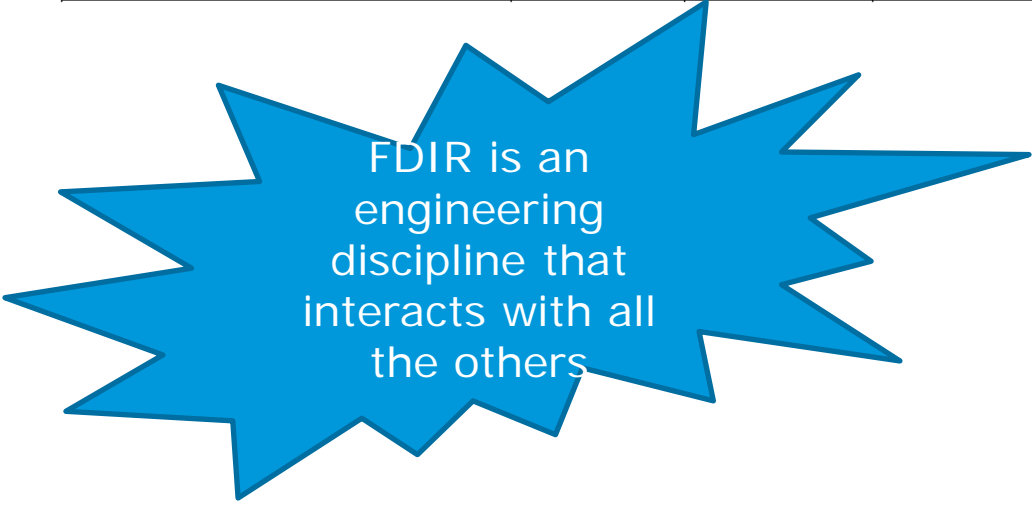


- FDIR activities are spread over the engineering process
- Output are reviewed at project milestones

Figure 7-2 FAME Process

Main R&D result: FAME (3/3)

Artifacts	SYS SRR	SYS PDR	S/S CDR	SYS CDR	SYS QR
SOFDIR	X	X			
FDIR Analysis		X			
FOS		X			
FSS		X	X	X	
TfPM Analysis Report		X			
SRS			X	X	
SDBICD			X	X	
Unit Test Report			X		
Subsystem Test Report				X	
System Test Report					X

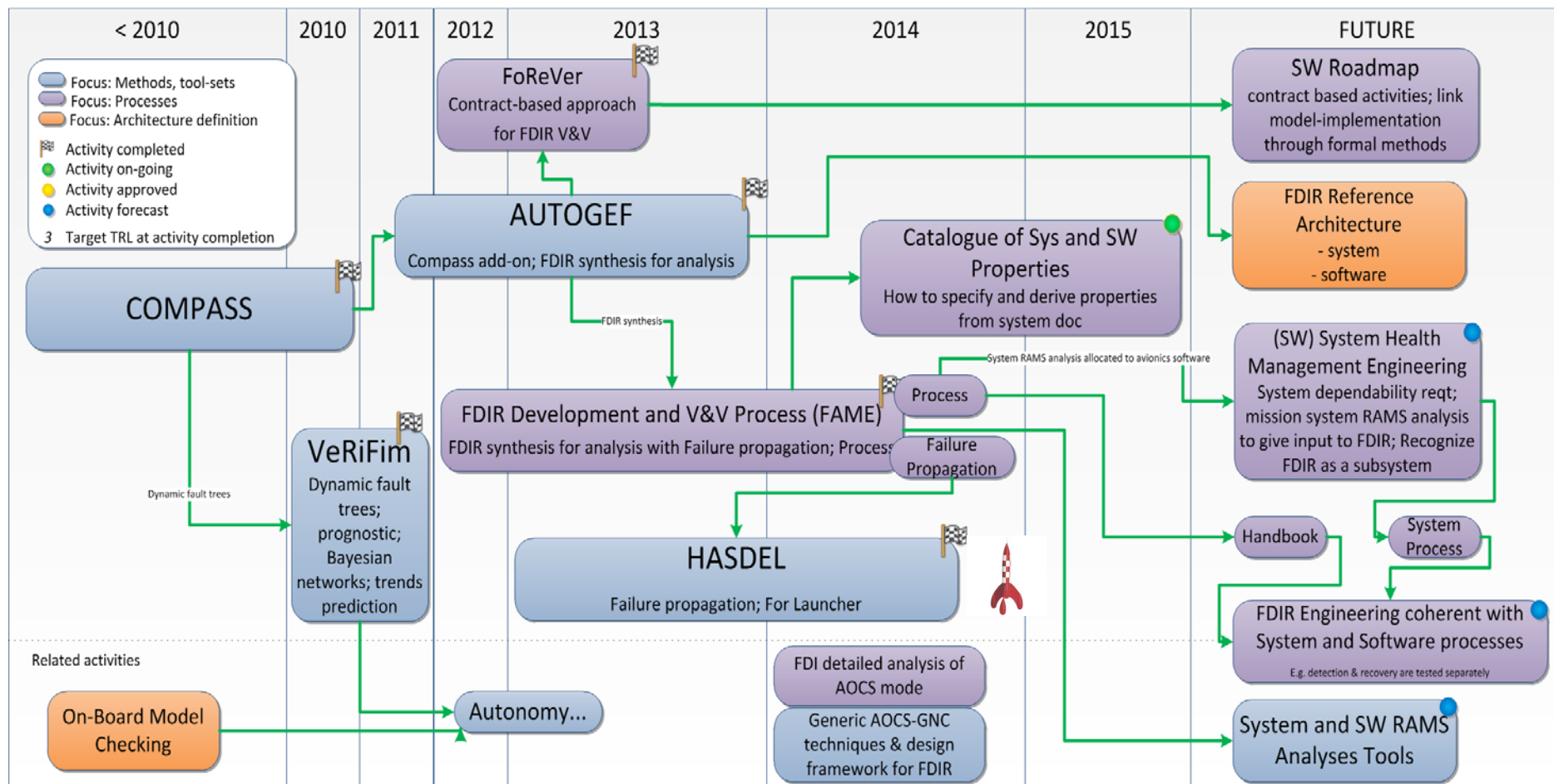


FDIR is an engineering discipline that interacts with all the others

The following roles have been identified for FAME process:

- System Engineer
- FDIR Engineer
- Safety Engineer
- SW Engineer
- SDB Engineer
- Subsystem Engineer
- Testing Engineer

Roadmap of activities



R&D Outlook: FDIR within system and software



- Novel approaches to System and Software level RAMS analyses and FDIR development enabling industrial deployment of the **Model-Based Dependability Engineering** and the required technologies
- Engineering models to support **early RAMS activities** and facilitate the development of FDIR elements allocated to Software
- System – Software Dependability and FDIR development from perspective of **System Health Management Engineering** discipline
- FDIR engineering approaches and techniques coherent with the System and Software level processes and activities. Technological gaps in achieving these objectives shall be investigated and missing technologies developed.
- Investigation and development of **FDIR Reference Architecture** suited for different levels of autonomy and Mission level RAMS requirements.

- ➔ Establish FDIR as an engineering discipline
- ➔ Create an FDIR community:
 - internal to ESA, (working group on Failure and Anomaly Management engineering domain)
 - in SAVOIR
 - in ECSS (FDIR handbook, FDIR reflection in other ECSS documents)
- ➔ Support FDIR process with a model based approach
 - Integrated in the system models and software models
 - e.g. COMPASS as a system tool, FDIR architecture model, state machines