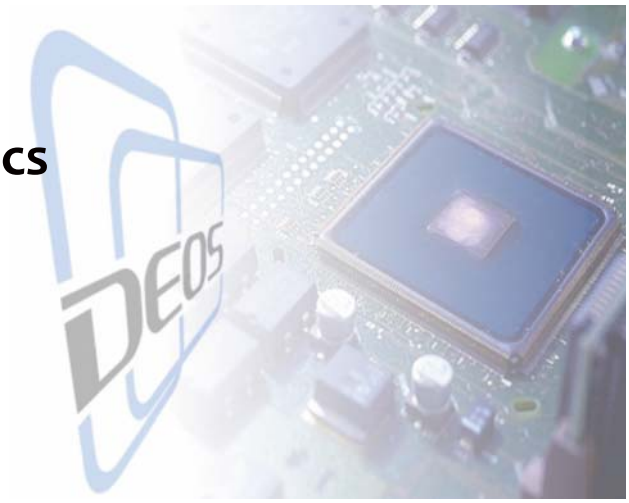
Dependability Case and Metrics for Open Systems Lifecycle



Yutaka Matsuno
Information Technology Center
The University of Tokyo
matsu@cc.u-tokyo.ac.jp

December 16, 2010

科学技術振興機構
Japan Science and Technology Agency

Today's Contents

- D-Case: Dependability modeling language
- D-Case writing example on Reception Robot
- D-Case Tools
- Integration with D-fops: DEOS framework architecture
- D-Case's challenges to Open Systems Dependability

D-Case/Metrics Team Members

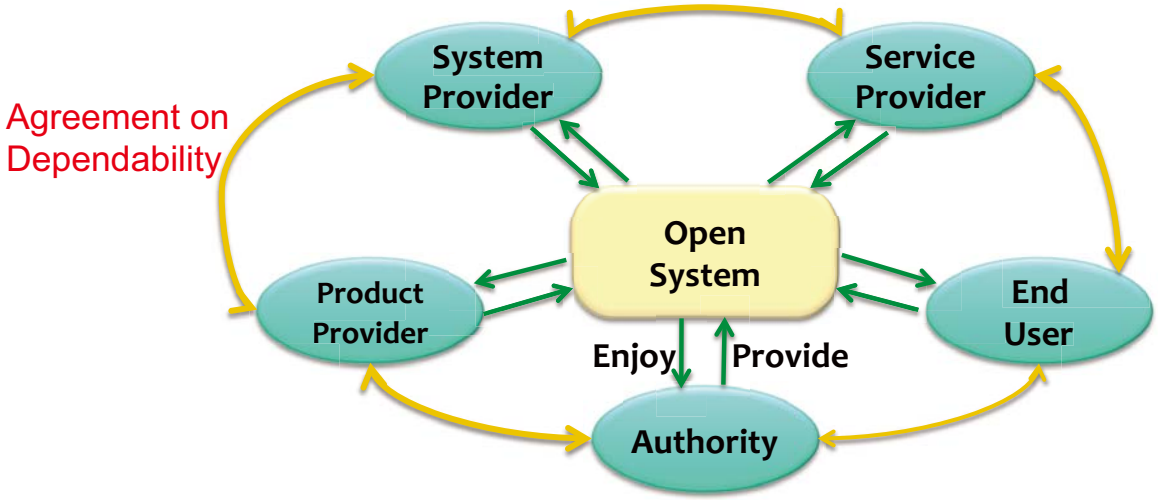
- University of Tokyo
  - Yutaka Matsuno
- Keio University
  - Jin Nakazawa
- AIST
  - Makoto Takeyama
  - Toshinori Takai
  - Takeo Matsuzaki
  - Kenji Taguchi
- Fuji Xerox
  - Atsushi Ito
  - Hajime Ueno
- DEOS Center
  - Hiroki Takamura
- Thanks to D-fops team and DEOS Center

Open Systems Dependability (1/3)

- Dependability is the system's property for sustaining services even in the existence of risks
  - Discussed as composite of Availability, Safety, Reliability, … (Jean-Claude Laprie et al)
- Open Systems: systems whose functions, structure, boundaries are changing by time to time
- DEOS project is discussing what is dependability from its root, and aims to establish ``Open Systems Dependability''

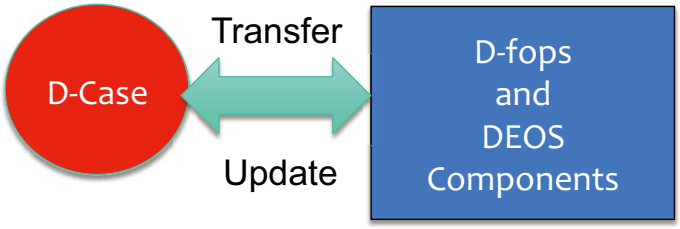
Open Systems Dependability (2/3)

- In Open System, everything is uncertain. All stakeholders have only limited information
  - All stakeholders must communicate each other and agree on dependability of the system
  - System must provide evidence for the agreement



Open Systems Dependability (3/3)

- D-Case: Dependability Modeling Language for mutual dependability agreement among stakeholders
- D-fops and DEOS components: provide evidence for the agreement



D-Case: Dependability Modeling Language

Current D-Case documents are tree structured and mostly the same as *Goal Structuring Notation (GSN)* for ***Safety Cases***

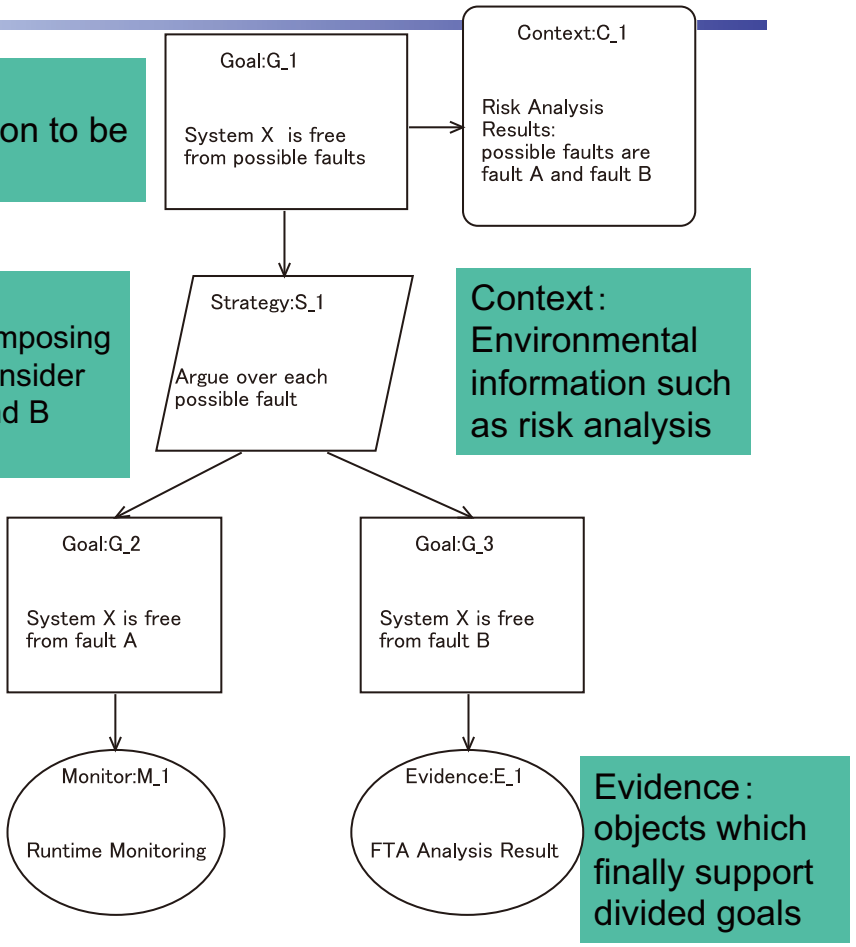
Goal: Proposition to be Argued

Strategy: Rationale for decomposing a goal. Here we consider cases for fault A and B separately

Context: Environmental information such as risk analysis

Monitoring: Evidence by monitoring in operational phase

Evidence: objects which finally support divided goals

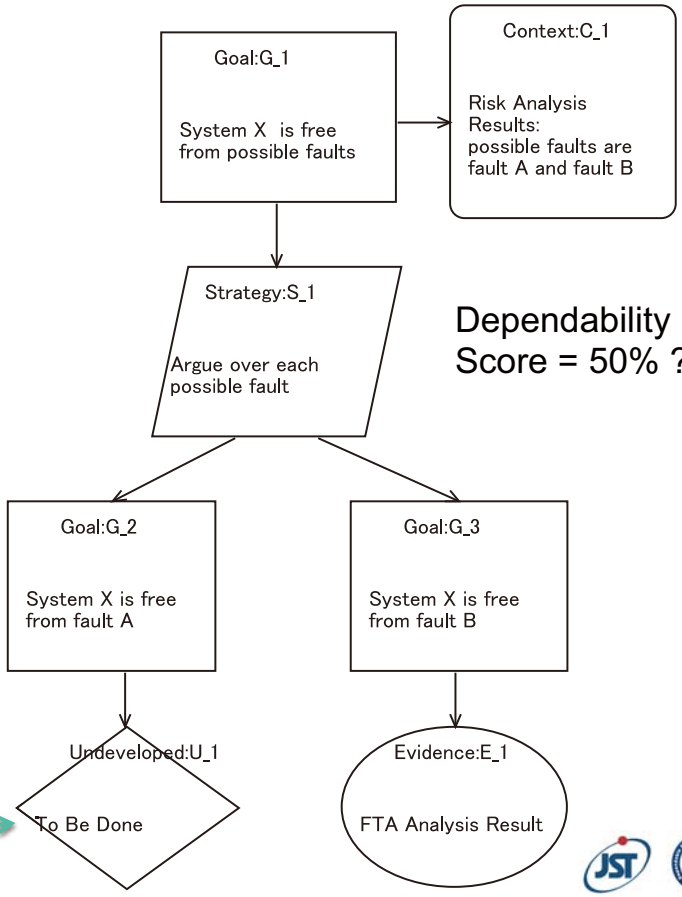


Dependability Metrics

- **Dependability should be evaluated by how well stakeholders argue dependability of the system**
- **D-Case can be used to show Dependability coverage**
- **Weighting Goals by e.g., Risk Analysis is future work**

Dependability Score = 50% ?

Diamond indicates we can not argue further (no evidence) at that time



Safety Case

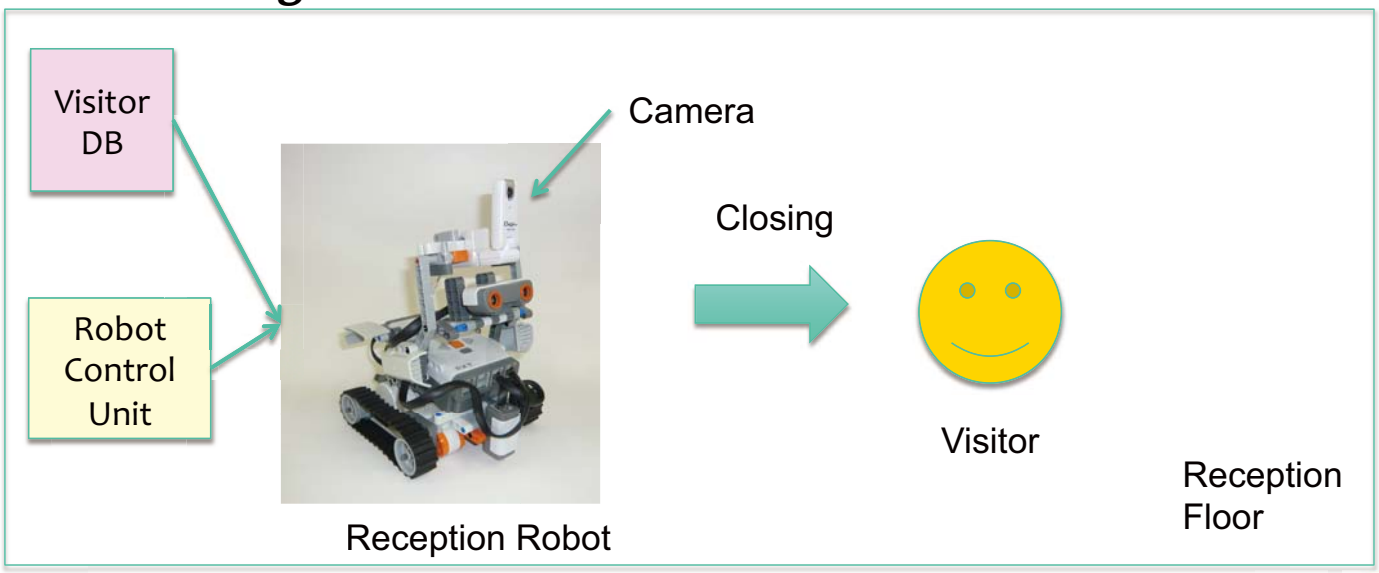
- "Case" is one of words in courts
- Recognized after serious incidents in UK
  - E.g., Piper Alpha North Sea Oil （167 dead, 1988)
- Not only following a procedure, but arguing why the procedure makes the system safe, based on evidence
- Widely required for regulation in UK, and now worldwide
  - ISO 26262: Functional Safety for Automobile
- We participate ISO and OMG system assurance meeting and visit York, Newcastle, … , and City University London

D-Case

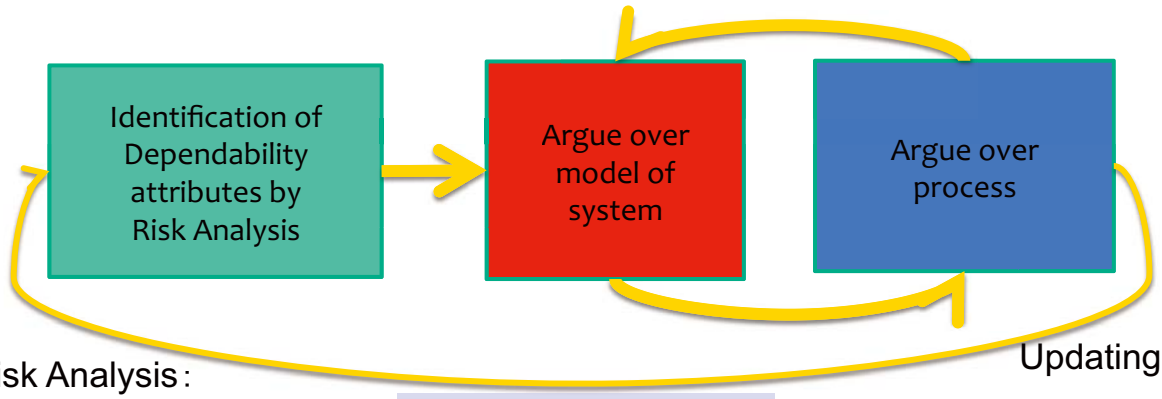
- We base our study on safety cases. We found goal oriented, evidence-based arguments are essential in open environment
- We aim D-Case to be a dependability modeling language, co-developed with D-fops and DEOS components
- By D-Case, we aim to describe the strategy to decide how to branch from a node, and the construction, or the condition of the branch, which are described as DEOS process

Writing Example: Reception Robot

- Let's consider Reception Robot in a company
  - Robot has Camera
  - Recognize visitor, approach and identify visitor by face recognition function



D-Case Process



Risk Analysis:
Unexpected approaching to visitor → Safety
Visitor must wait. 10 seconds waiting may cause some loss; maximum 1 minute wait → Availability
(we consider Availability)

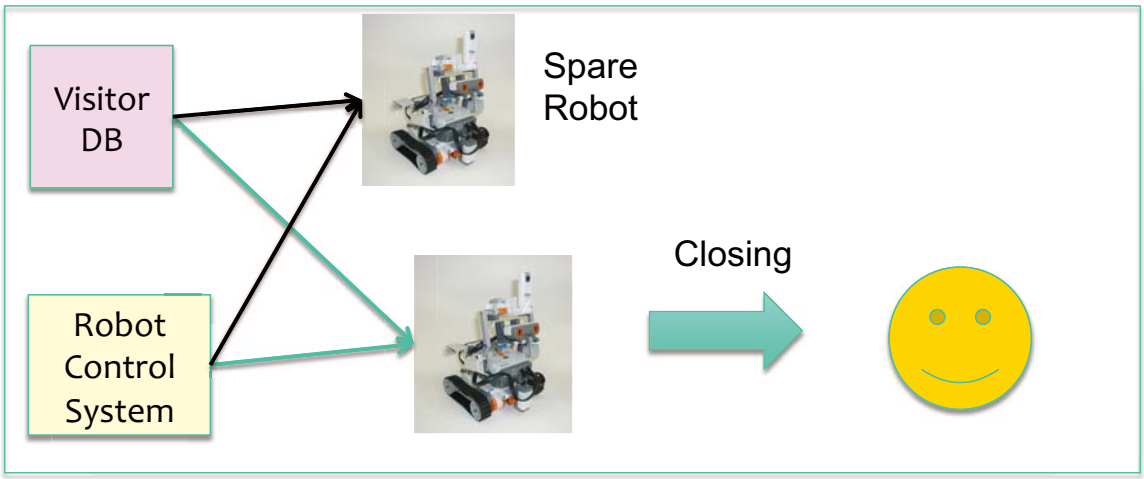
Consider a model (robot, camera, control system) and argue over each sub-components

Argue over development, testing, operational,… phases of the robot system

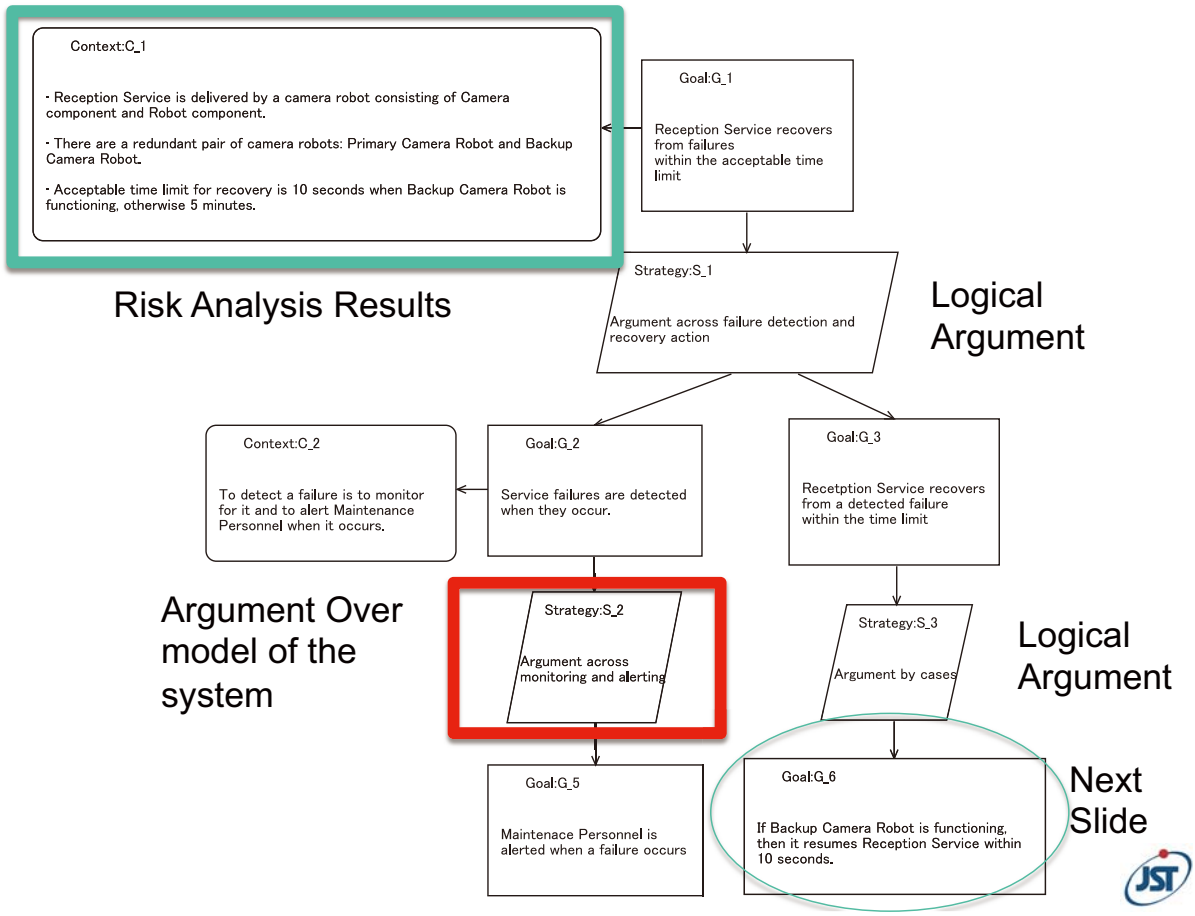
Argument should be logical, rational, …

Stakeholder agreement for Reception Robot

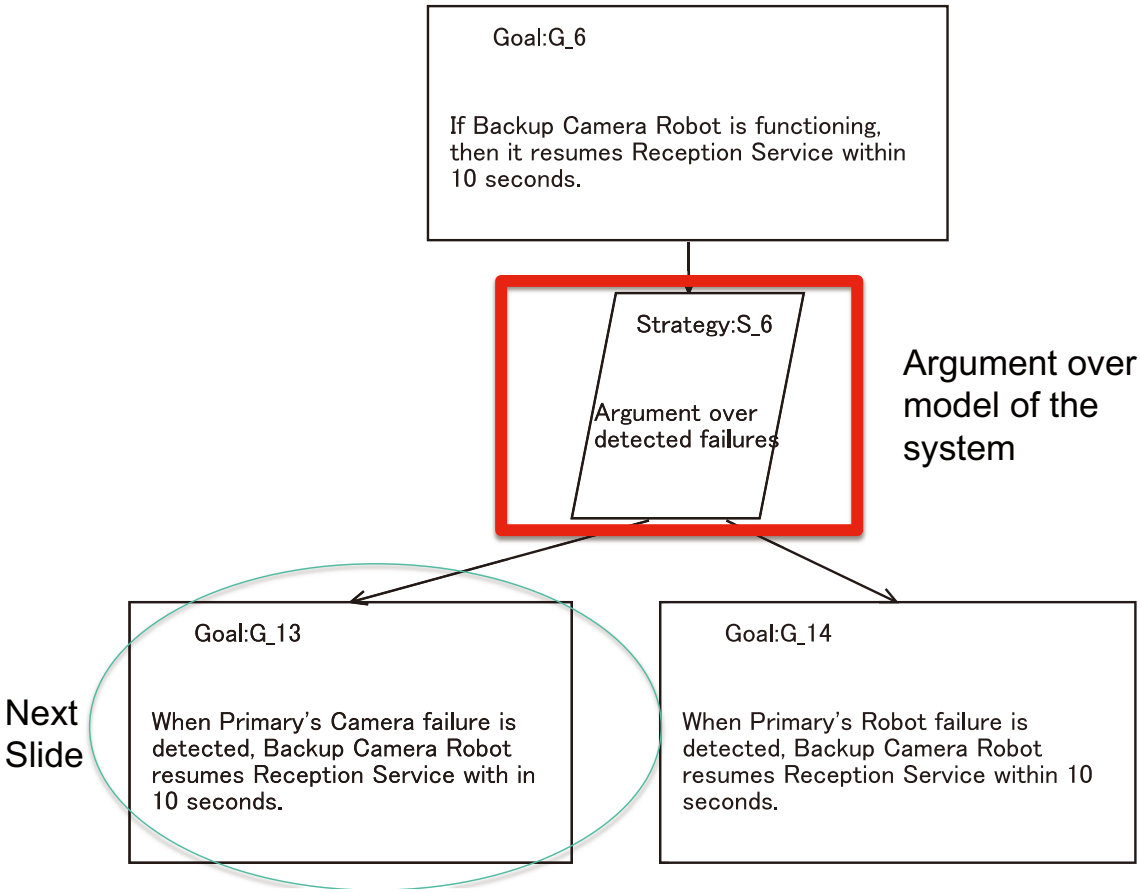
- **Prepare a spare robot, and implement Fail-Over mechanism. By Fail-Over, in most cases, visitor only needs to wait 10 seconds when a failure occur. In worst case (both robots are unavailable), visitor must wait 5 minute.**
- **The agreement is made by D-Case whose top goal is ``Robot recovers from failures within acceptable time"**



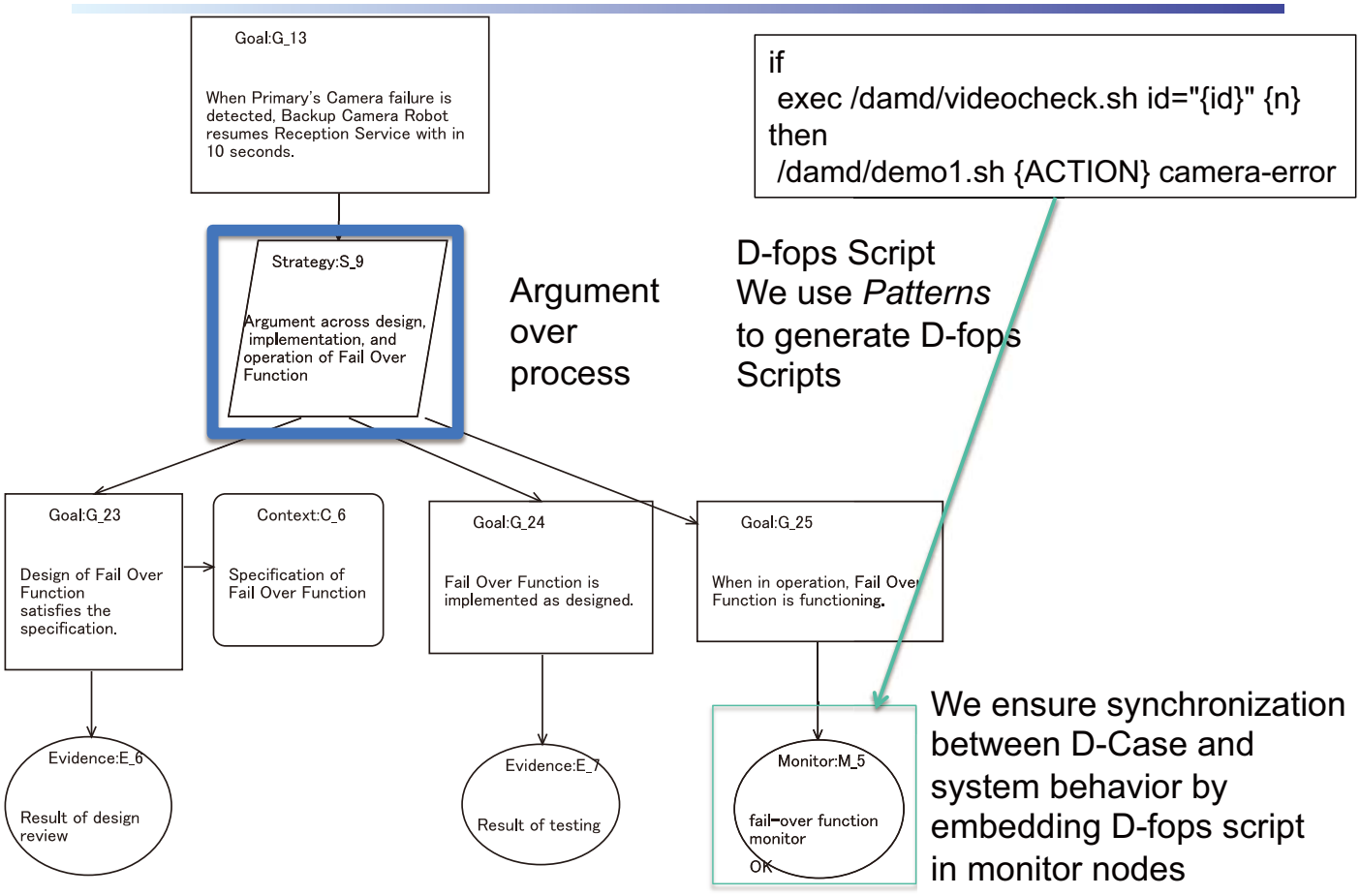
D-Case for Reception Robot: Top Level



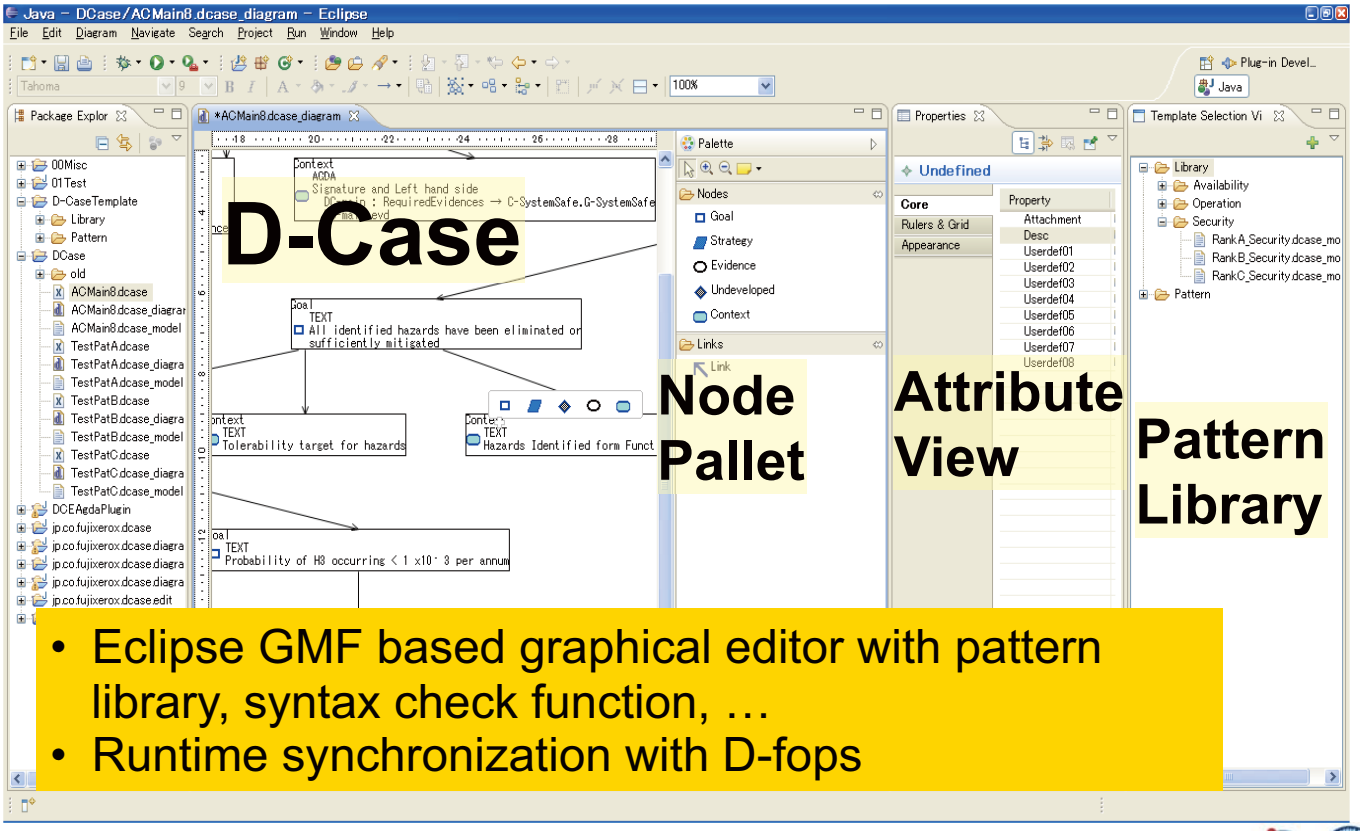
D-Case for Reception Robot: Fail Over Argument



D-Case for Reception Robot: Fail Over Argument when Camera Fails

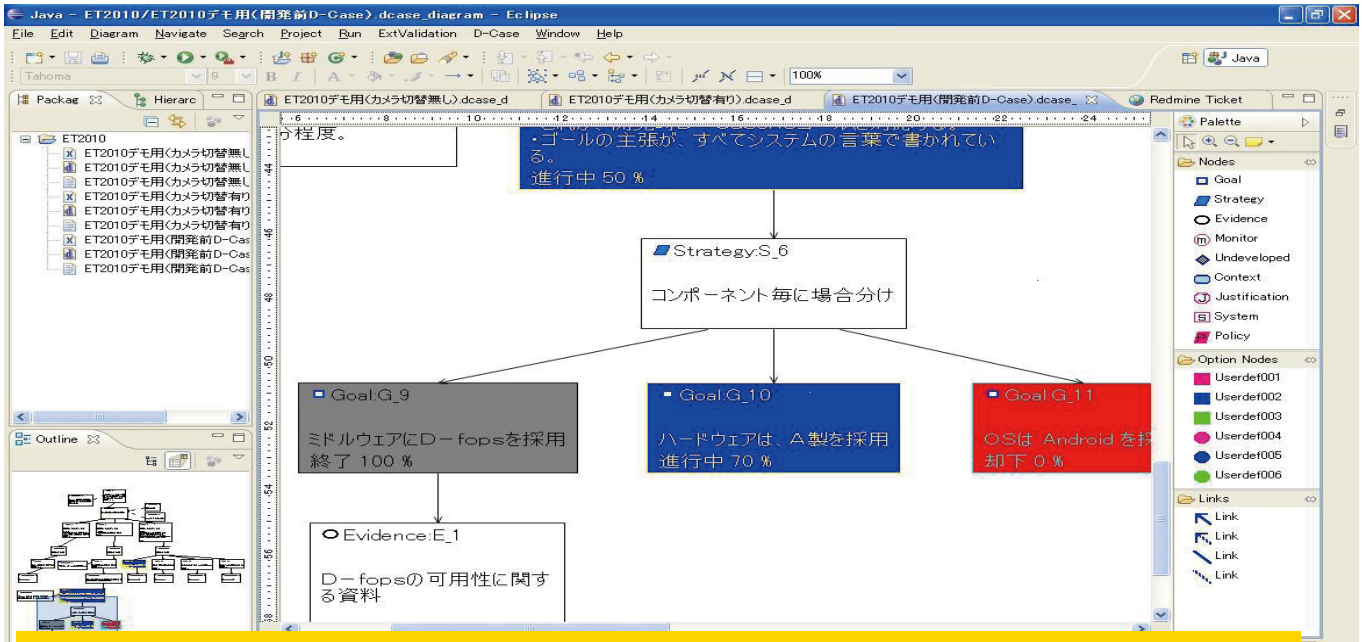


D-Case Tools: D-Case Editor



- Eclipse GMF based graphical editor with pattern library, syntax check function, …
- Runtime synchronization with D-fops

D-Case Tools: D-Case Editor with Redmine

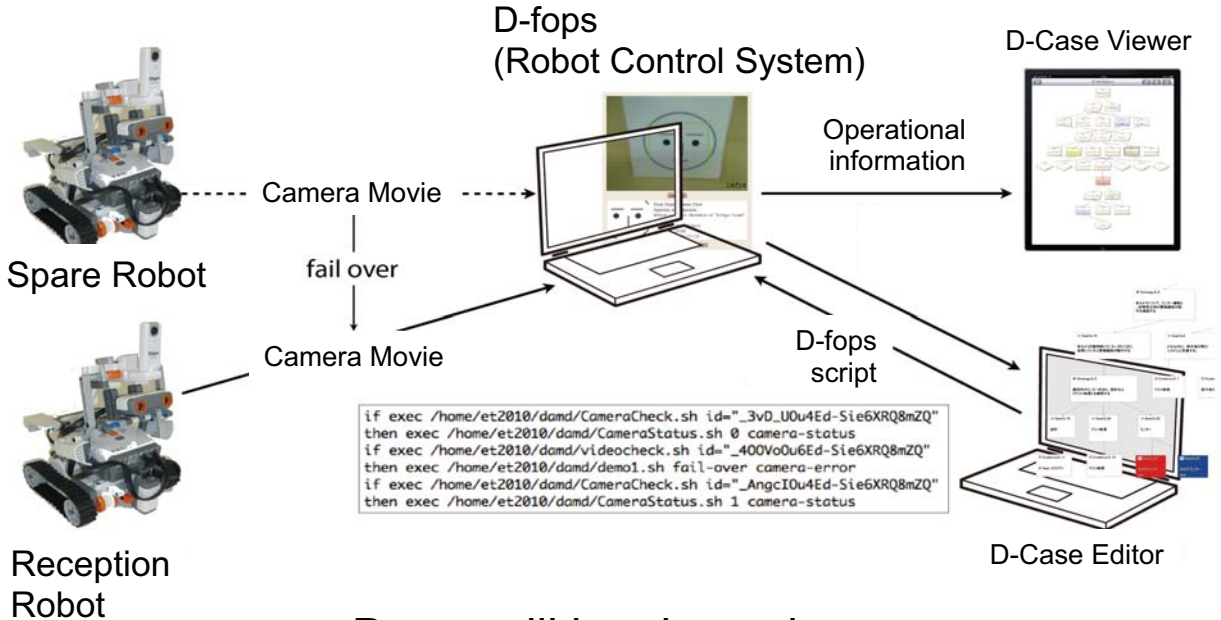


- Synchronization of D-Case goal construction and decomposition with Redmine's Tickets
- An example of embedding dependability management tool into development tool chains

D-Case Tools: D-Case Viewer

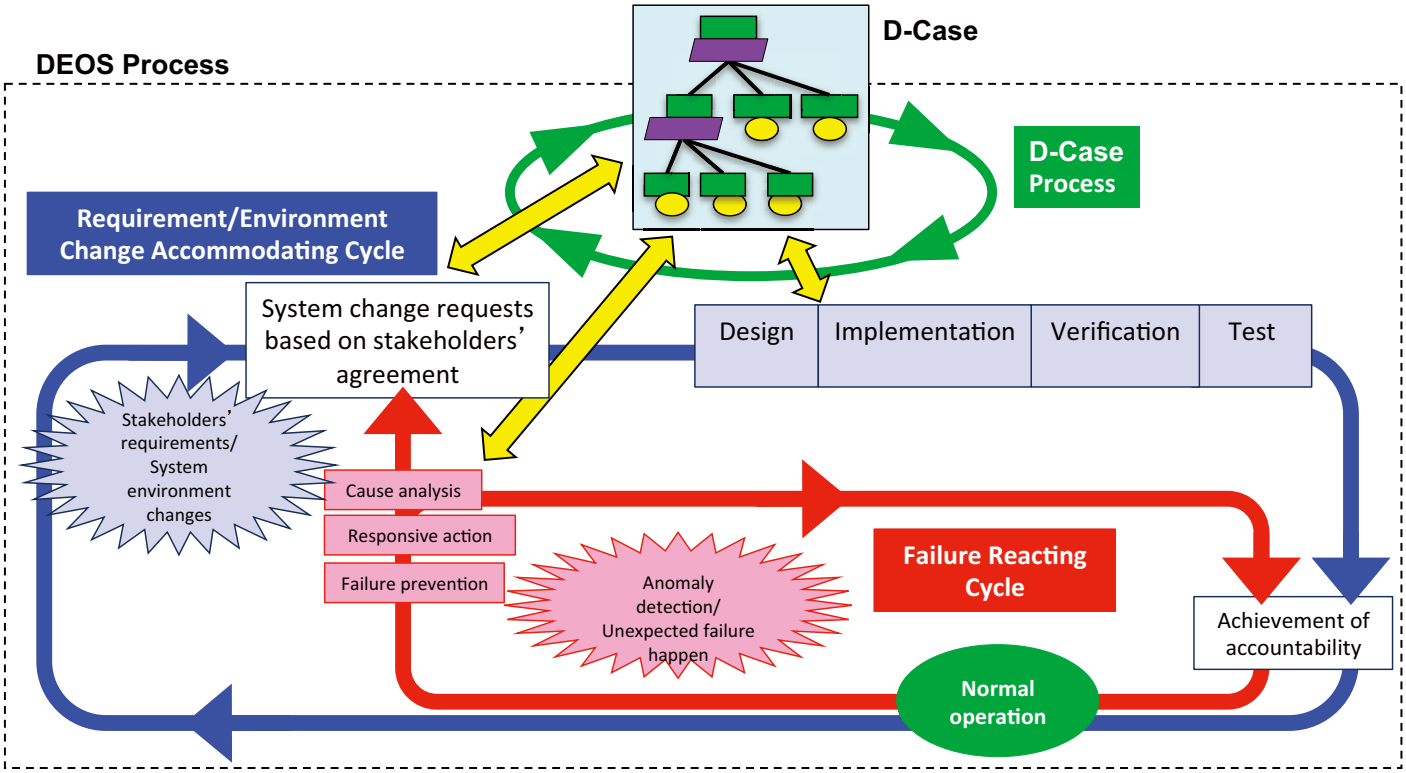
- Viewer on iPad
- Will be connected to D-Case DB
- To be used in operational phase for referring D-Case

Integration with D-fops: A Demo System of Reception Robot



Demo will be shown in tomorrow's workshop

DEOS Process



D-Case's Challenges to Open Systems Dependability

- Precisely describing combination of arguments over process and system
- D-Case maintaining and updating during whole Open Systems Lifecycle
- Modeling Agreement among stakeholders
- Modeling Dependability relation among open systems
- Evaluating Dependability: Dependability Metrics

Summary

- D-Case: Dependability modeling language
- D-Case writing example
- D-Case Tools
- Integration with D-fops (more detail in Yokote san's talk)
- D-Case's challenges to Open Systems Dependability