Abstract

This paper discusses an object-oriented approach based on design pattern and computational reflection concept to implement non-functional requirements of complex control systems. Firstly, we briefly describe software architecture design, followed by control-monitor safety pattern, Tri-Modular redundancy (TMR) pattern, reflective state pattern, and fault tolerance redundancy.
Design Patterns to Implement Safety and Fault Tolerance

patterns that are use for safety and fault management. Reflection state pattern is a refinement of the state design pattern based on reflection architectural pattern. With variation in reflective design pattern we can develop a well structured fault tolerant system. The main goal of this paper is to separate control and safety aspect from the application logic. It details its intent, motivation, participants, consequences and implementation of safety design pattern.

Reference


- Trivedi Kishor, “Probability and Statistics with Reliability, Queueing, and Computer Science Applications,”
- Magnus Penker and Hans-Erik Eriksson; “Business Modeling With UML: Business Patterns at Work”.
- Fuping Zeng; Aizhen Chen; Xin Tao;“Study on Software Reliability Design Criteria Based
Design Patterns to Implement Safety and Fault Tolerance

on Defect Patterns”, Reliability, Maintainability and Safety, 2009. ICRMS 2009. 8th International Conference on Digital Object Identifier: 10.1109/ICRMS.2009.5270095
- Luciane Lamour Ferreira and Cecília Mary Fischer Rubira, “Reflective Design Patterns to implement Fault Tolerance

Index Terms

Computer Science

Software Engineering

Key words

Reflective Design pattern

Fault tolerance

Safety tactics

Tri-modular redundancy

Digital distributed control system