Implementation of Fault–Tolerant Software in Distributed System

Ko Iseki

School of Information Science,
Japan Advanced Institute of Science and Technology

February 14,1997

Keywords: fault tolerance,software fault tolerance,FTAG,CML.

The increased use of computers and our increased reliance on them have led to a need for highly reliable computer systems. Once failure occured in computer system, we need the system that to guard the failure influence the whole of system. Even if system has faluire in it ,system is giving the service. Then this property is called fault tolerant. The goal of fault tolerance is to avoid system failure,even if faults are present. There are two fault tolerance approach: hardware fault tolerance and software fault tolerance. In this paper, we focus on software fault tolerance. Software fault tolerance is a structure that constructs to continue providing service despite the existence of software faults and/or operational faults.

Building fault tolerance software by functional approach, there is many advantage compared with imperative approach. FTAG(Fault Tolerant Attribute Grammers) model is a model that functional model for writing fault tolerant software that is based attribute grammers. As described there,this model can facilitate the writing of programs that use various mechanisms for dealing with software faults and/or faults in the underlying computation platform. These include recovery blocks,N–version programing and process replication. With this approach , a program is written as a series of module docompositions,with provisions for redoing and replicating modules used to implement fault tolerance requirements.

This FTAG model ,however, is not yet implemented completely computer systems. Therfore, it is necessary to implement FTAG model to computer system, and make some experience on it.

The goal of this paper is to show avarage of fault tolerance based on functional model and how to construct a fault tolerant software by implementation on loosely coupled multi processor system.

In FTAG model,every computation consists of a collection of pure mathematical functions called modules. Computing order of modules is determined by attributes' dependence. And, modules that no dependency of attributes is possible that running in concurrently. Accordingly, it is to be desired that the language to implement FTAG have architecture of concurrency. In this paper, we will use CML(concurrent ML).

CML is a language for concurrent programing. It is an extension of Standard ML (SML), and is implemented on top of Standerd ML of New Jersey(SML/NJ). A CML program consists of a number of threads, which use message passing on typed channels to communicate. Threads and channels can create dynamically, this proprety is useful to simulate attributes flow between the threads.

The conversion method for FTAG programing list is convert to CML are as follows.
(1). FTAG program is converted CML program that have a management of fault tolerant software. This program is actual apply of application software.And this program detect failure and raise redoing.
(2). Workspace manager and node manager is created that manage attributes stored object base is required by fault tolerant structure redoing and replication.
(3). Composing these two managers and CML program which converted before, generate the program which run on CML interpreter. Then, this program is fault tolerant software in CML.

We call three component that describe above, node manager, work space manager, and application applier. These components are important component in fault tolerant software in CML. Each components are communicate with other component by channels. Channel is message passing way between thread. In this system, channel is used by nodes of comptational tree to carry attribute values.

Node manager controls the tree that created by application applier to apply functions. Node manager stored shape of comptation tree. Then redoing occured by detecting failure ,make tree to no contradiction in system, redoing operation is execute. Workspace manager is store and manage attribute values required when redoing and replication happens. Application applier is that converted program FTAG to CML. Actually,this component execute the program of CML.

Messages from the node manager to the workspace manager are required that read attritute value and require to delete workspace. Messages from work spaces to the node manager are reported that required attribute values to the workspaces. Messages between other composes are required and reported as avobe.

In decomposition modules to threads with channels, each node can work concurrently. By this property,this model is possible to implement to distributed system. Based the architecture we describe above,we show this implementation FTAG model is applicable to destributed system. And through description of name sarver system, in confirm the efficiency of FTAG model, and its implementation of on distributed systems.