A Survey of Recent Trends in Testing Concurrent Software Systems

ABSTRACT

Concurrent systems are common in several application domains: many interactive applications exploit the multithreaded paradigm to decouple the input-output processing from the back-end computation; applications developed for single nodes often exploit multiple threads to enable parallel computations on multi-core architectures; Web applications implement the client-server communication paradigm with client-side and server-side computations; mobile applications often access data and interact with remote servers; peer-to-peer services coordinate a multitude of computing nodes.

EXISTING SYSTEM

In Existing System, Many modern software systems are composed of multiple execution flows that run simultaneously, spanning from applications designed to exploit the power of modern multi-core architectures to distributed systems consisting of multiple components deployed on different physical nodes. We collectively refer to such systems as concurrent systems. Concurrent systems are difficult to test, since the faults that derive from their concurrent nature depend on the inter leavings of the actions performed by the individual execution flows. Testing techniques that target these faults must take into account the concurrency aspects of the systems. The increasingly rapid spread of parallel and distributed architectures led to a deluge of concurrent software systems, and the explosion of testing techniques for such systems.

DIS ADVANTAGES

- Wrong inter leavings may lead to concurrency faults.
- Checking for the serializability of an interleaving can be very expensive.

PROPOSED SYSTEM

In Proposed System, we focused mainly on the problem of selecting inter leavings to be tested and has explored two classes of approaches: property based approaches, which target patterns of inter leavings that are more likely to lead to faults, and exploration approaches, which

explore the space of inter leavings exhaustively, based on coverage criteria or heuristically. The current research trends are towards predictive property based techniques and violations of expected order invariants rather than low level memory access conflicts such as data races.

ADVANTAGES

- Guarantees that all the execution flows in a concurrent system.
- It avoid potentially dangerous inter leavings.

SYSTEM REQUIREMENTS

H/W System Configuration:-

Processor	-	Pentium –III
RAM	-	256 MB (min)
Hard Disk	-	20 GB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

S/W System Configuration:-

Operating System	:	Windows95/98/2000/XP
Application Server	:	Tomcat5.0/6.X
Front End	:	HTML, Jsp
Scripts	:	JavaScript.
Server side Script	:	Java Server Pages.
Database	:	MySQL 5.0
Database Connectivity	:	JDBC