Xception™ - Enhanced Automated Fault-Injection Environment

Ricardo Maia, Luis Henriques, Diamantino Costa
Critical Software SA
3030 Coimbra
Portugal

Henrique Madeira
DEI-FCTUC, University of Coimbra
3030 Coimbra
Portugal

1. Introduction

The increasing demand for mission and business critical applications in aerospace, industry, defense, telecommunications or business in general, is posing new challenges to the software industry in terms of high availability, reliability, and safety requirements. Failures in these applications may have a tremendous financial impact and may even threaten human lives. Before deploying such applications and systems, they must be subject of intensive testing in order to guarantee that the system and, particularly, built-in fault-tolerance mechanisms are working as expected. Assuring that the system responds appropriately to unusual or exceptional events requires more than traditional testing.

Fault Injection based testing is the practical answer to these new demands. It provides the ability to test those systems in exceptional situations, that may however occur in the field, validate recovery mechanisms in place, experiment worst failure scenarios, and spot weak-points in the system, proving feedback for correction or redesign.

2. Xception – The tool

Xception is an automated fault injection environment that enables accurate and flexible V&V (verification & validation) and evaluation of mission and business critical computer systems using fault injection. Xception is designed to accommodate a variety of fault injection techniques (according to a wide range of configurations of the tool) and emulate in this way different classes of faults, with particular emphasis to hardware and software faults.

One key aspect of Xception is the high degree of automation provided by the fault injection environment, which enables the users to plan and perform fault injection experiments in a straightforward way. A friendly graphical user interface (GUI) provides the means for fault definition, experiment execution and result analysis. Xception uses a standard SQL database to store all the data required to manage and fully automate the experiments.

Xception can affect both user and kernel code and is able to fine tune emulation of faults in specific threads and/or processes, in different processors or even in different nodes of a distributed system.

Xception has been used in various domains, ranging from Online Transaction Processing Systems (OLTP) [3] to Space Payload Systems[4]. Supported target systems include PowerPC, Intel Pentium and SPARC based platforms running LynxOS, SMX, WindowsNT/2000 and Linux operating systems. For more information on Xception refer to [1].

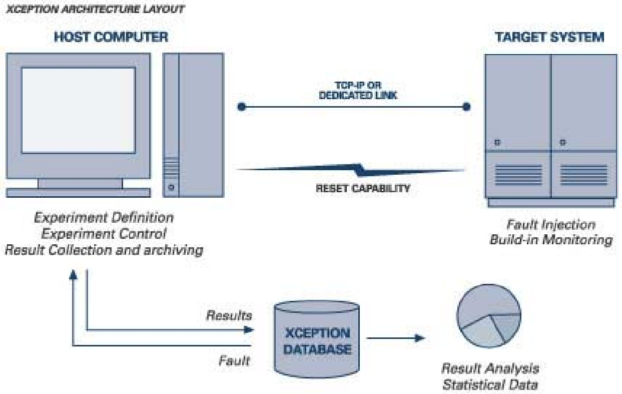

Figure 1 – Xception Testbed Layout

3. What's new in Xception 2.0

The new 2.0 version of Xception includes a set of new fault injection techniques such as Boundary Scan based Fault Injection (BSFI), Pin-Level Fault Injection (PLFI), and software mutation. The traditional Software-Implemented Fault Injection (SWIFI) used in previous versions of with Xception 2.0 has also been improved in order to reduce intrusiveness as much as possible. All these FI technologies on a single toolset environment allow to configure Xception 2.0 for multiple scenarios according to different requirements in terms of intrusiveness, data to be collected on the target system, availability of source code, etc.

4. References

[1] http://www.xception.org

[2] Xception White Paper , http://www.xception.org

[3] D. Costa, T. Rilho, and H. Madeira, "Joint Evaluation of Performance and Robustness of a COTS DBMS Through Fault-Injection", IEEE/IFIP Dependable Systems and Networks Conference – DSN, New York, USA, 25-28 June, 2000, pp. 251-260.

[4] H. Madeira, R. Some, F.Moreira, D. Costa, D. Rennels, "Experimental evaluation of a COTS system for space applications" to appear in the IEEE/IFIP Dependable Systems and Networks Conference – DSN 2002, USA, 2002.

0-7695-1597-5/02 $17.00 © 2002 IEEE