

High Performance, Environmentally Adaptive Fault Tolerant Computing (EAFTC)

John Samson and Jeremy Ramos, Honeywell Inc., Defense & Space Electronic Systems
Alan George, University of Florida,
Minesh Patel, Tandel Systems LLC
Rafi Some, NASA Jet Propulsion Laboratory

Abstract

Current and future space-based processing applications are requiring, and will require, increasing amount of onboard processing capability. One way to achieve a high level of processing capability is through the use of COTS high performance processors. While current COTS high performance processors are exhibiting adequate Total Integrated Dose (TID) performance to meet the requirements of the natural space radiation environment, Single Event Upsets) SEUs caused by heavy ions and solar flares are, and will remain, a problem. Traditional approaches to mitigate the SEU problem involve fixed redundancy schemes such as Self Checking Pairs (SCP) or Triple Modular Redundancy (TMR). While effective in mitigating the effects of SEUs, use of these techniques comes at a high price, 100% overhead for SCP, and 200% overhead for TMR, particularly when such a level of protection is not needed. In such cases, it would be beneficial to be able to convert that overhead into useful mission processing capability. The idea behind Environmentally Adaptive Fault Tolerant Computing (EAFTC) is to sense the environment and configure the processing system appropriately to maximum the processing capability available to the mission.

To satisfy this need, the Environmentally Adaptive Fault Tolerant Computing concept has been demonstrated and is currently being developed further as one of the flight experiments for NASA's New Millennium Program (NMP) ST-8 project. The objective of this NMP ST-8 effort is to combine high performance, fault tolerant, COTS-based cluster processing with replication services, and fault tolerant middleware in an architecture and software framework capable of supporting a wide variety of mission applications.

The software architecture for this EAFTC framework is depicted in Figure 1. Figure 1 shows two types of processing nodes: one type, a rad hard system controller, which can operate through any environment without upsetting, and a second type, a cluster of COTS high performance processing engines with FPGA accelerators for enhanced performance. A high level API (Application Interface) and a high level SAL (System Abstraction Layer) provide both application independence and platform independence, while allowing the particular mission applications and platforms to take advantage of fault tolerance services and reliable messaging offered by the fault tolerant middleware layer.

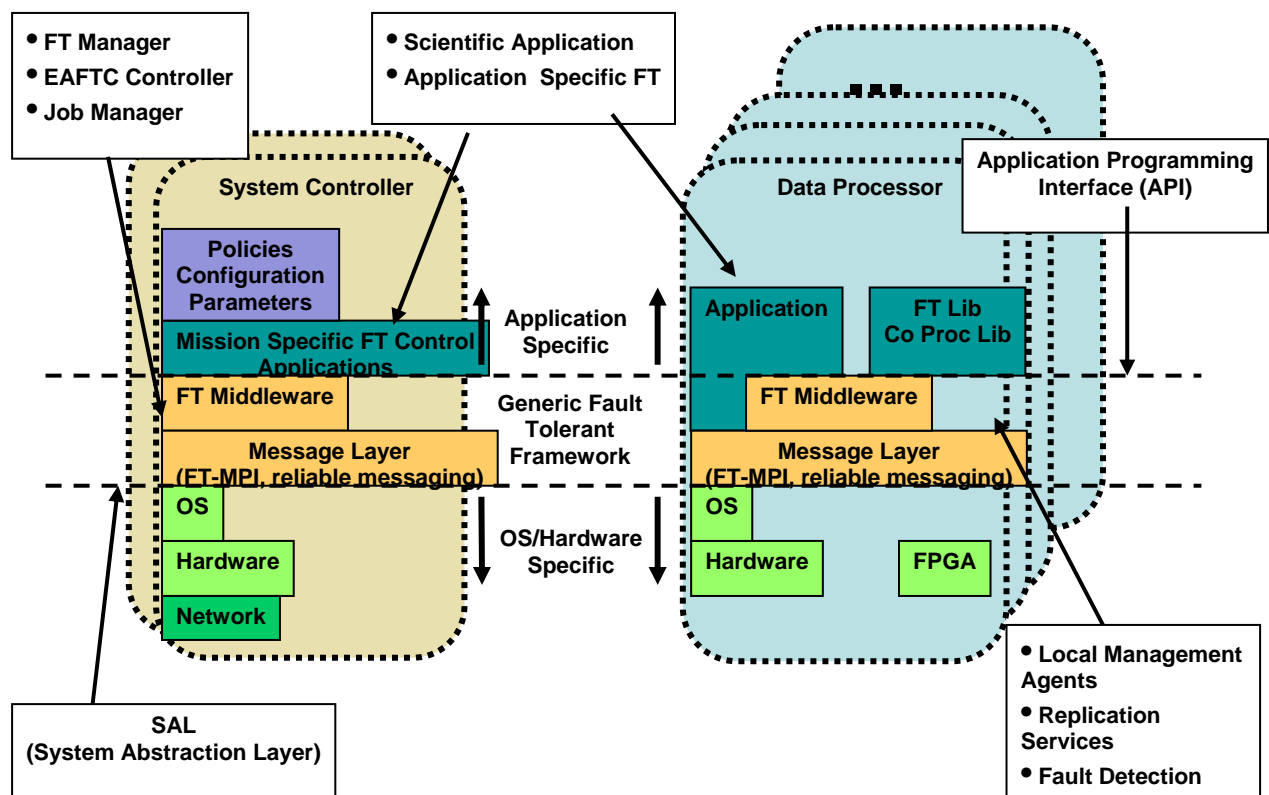


Figure 1 – EAFTC Software Architecture

A key element of the system is the EAFTC controller, an autonomous and adaptive controller for the fault tolerance configuration of the processor which is responsive to environmental conditions, application criticality and system mode. The EAFTC function includes environmental sensors, an environmental server, an alarm generator, a fault tolerance manager, and a task configuration controller which directs the onboard processing system to respond to the

sensed environmental conditions as illustrated in Figure 2. The paper will describes the experiments, demonstrations, and performance achieved to date.

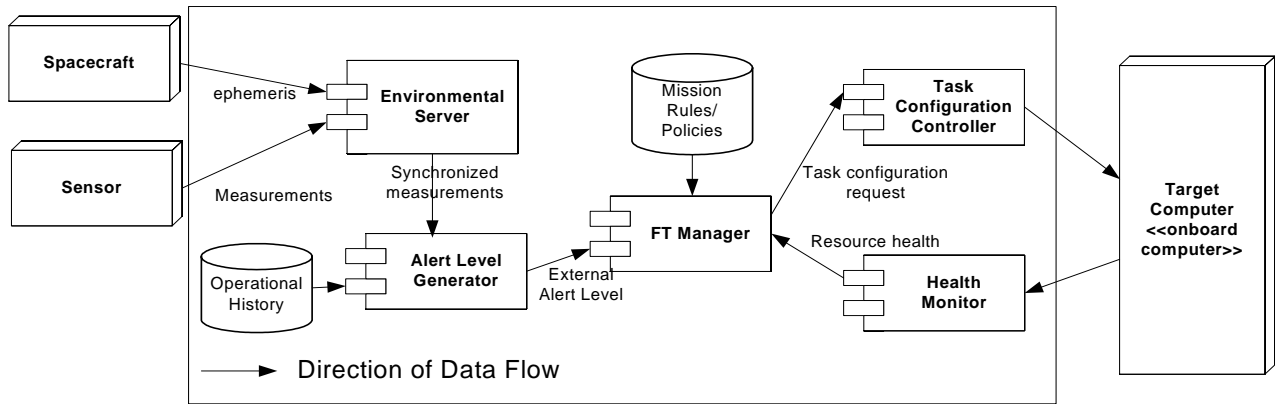


Figure 2 – EAFTC Controller Function