Request Routing using the Data Integrity Extensions

Martin K. Petersen <martin.petersen@oracle.com>

December 4th 2007

Introduction

This document tries to explain the Data Integrity Extension operation codes in detail.

As described in *I/O Controller Requirements for DIF Aware Operating Systems*, each command bound for a controller will be tagged with an operation code which tells the driver/firmware how to handle the I/O.

There are two distinct protection envelopes to consider:

- Transfer of protection information between Operating System and I/O Controller
- Transfer of protection information between I/O Controller (initiator) and Disk (target)

Whether to protect the path between OS and controller is up to the application, the operating system or system administrator preference. Thus there is no guarantee that an I/O request bound for a controller supporting the Data Integrity Extensions will provide a scatter-gather list for integrity metadata. It is a per-I/O property.

Similarly, whether the path between controller and disk should be protected with DIF is controlled by the `RDPROTECT/WRPROTECT` field in the CDB.

The two protection envelopes are completely *orthogonal*. And any combination can be expected on hardware that supports it.

Example: A write request could include an integrity metadata scatter-gather list despite being bound for a storage device that is not formatted using DIF. In that case the controller must read and verify the protection information and then use standard 512 byte sectors when communicating with the target. The operating in this case is called `WRITE_STRIP` because it is a *write* request and the protection data must be *stripped* off of the I/O after verification.

The following charts will illustrate the various combinations.

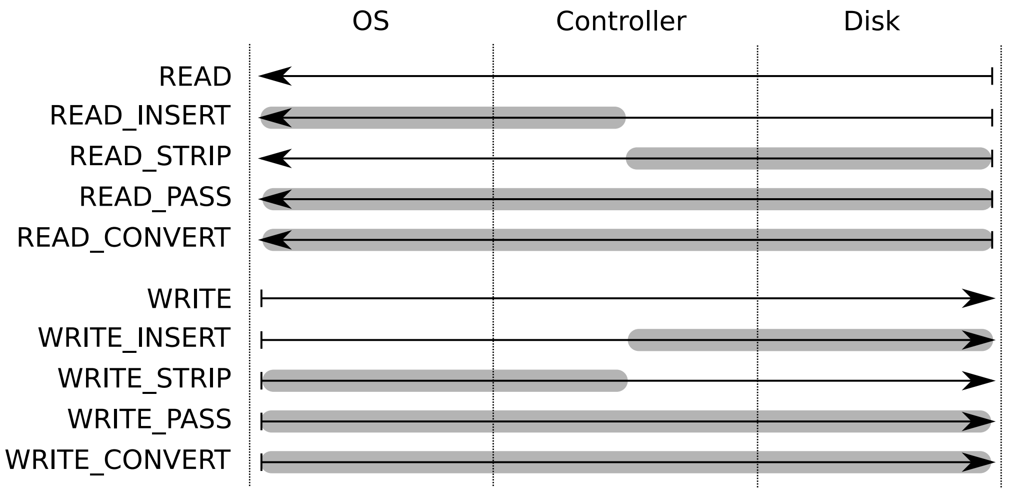

Figure 1 Data Integrity Extensions: Operating Codes. The grey area indicates the protected path.

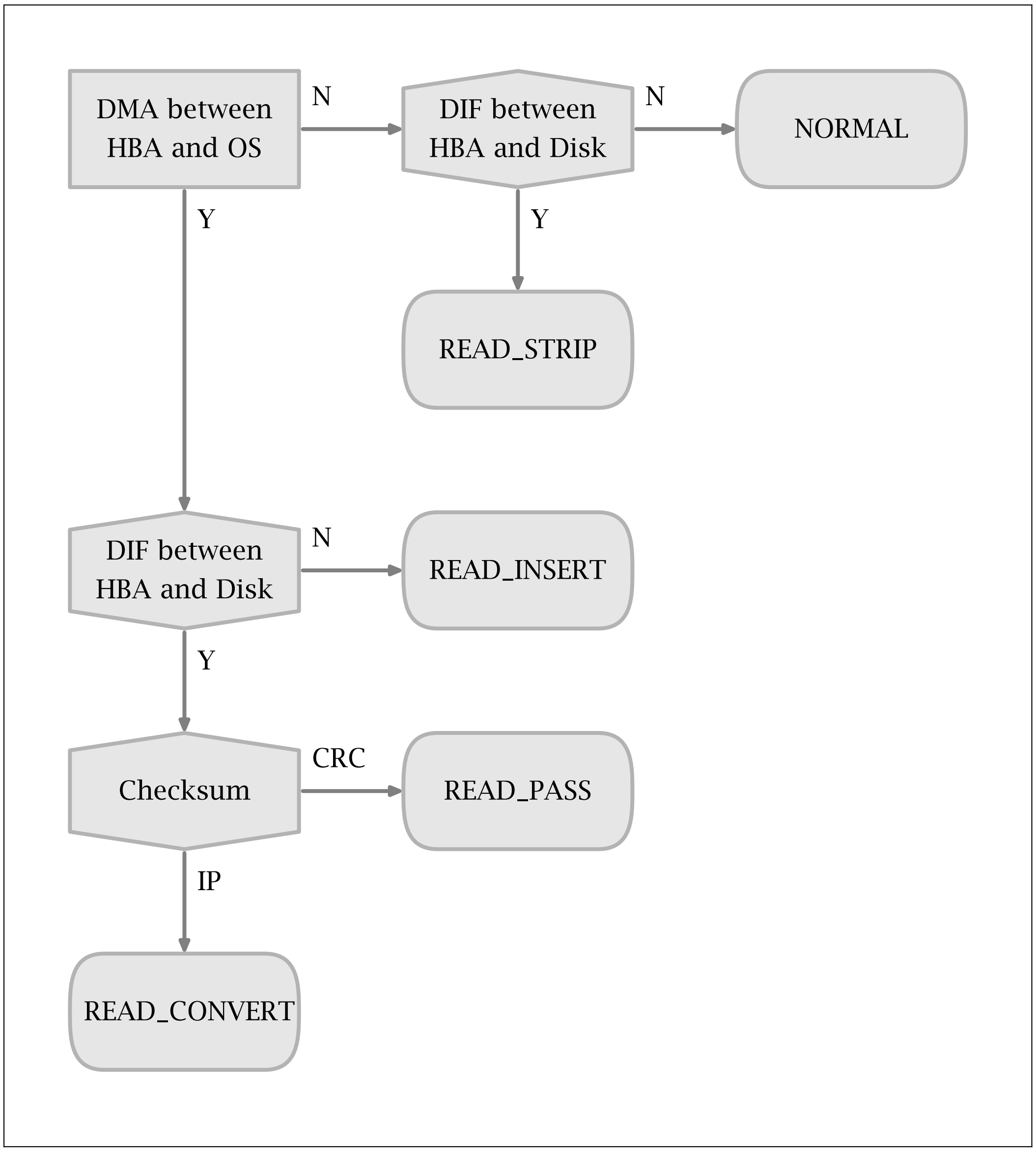

Figure 2 READ request

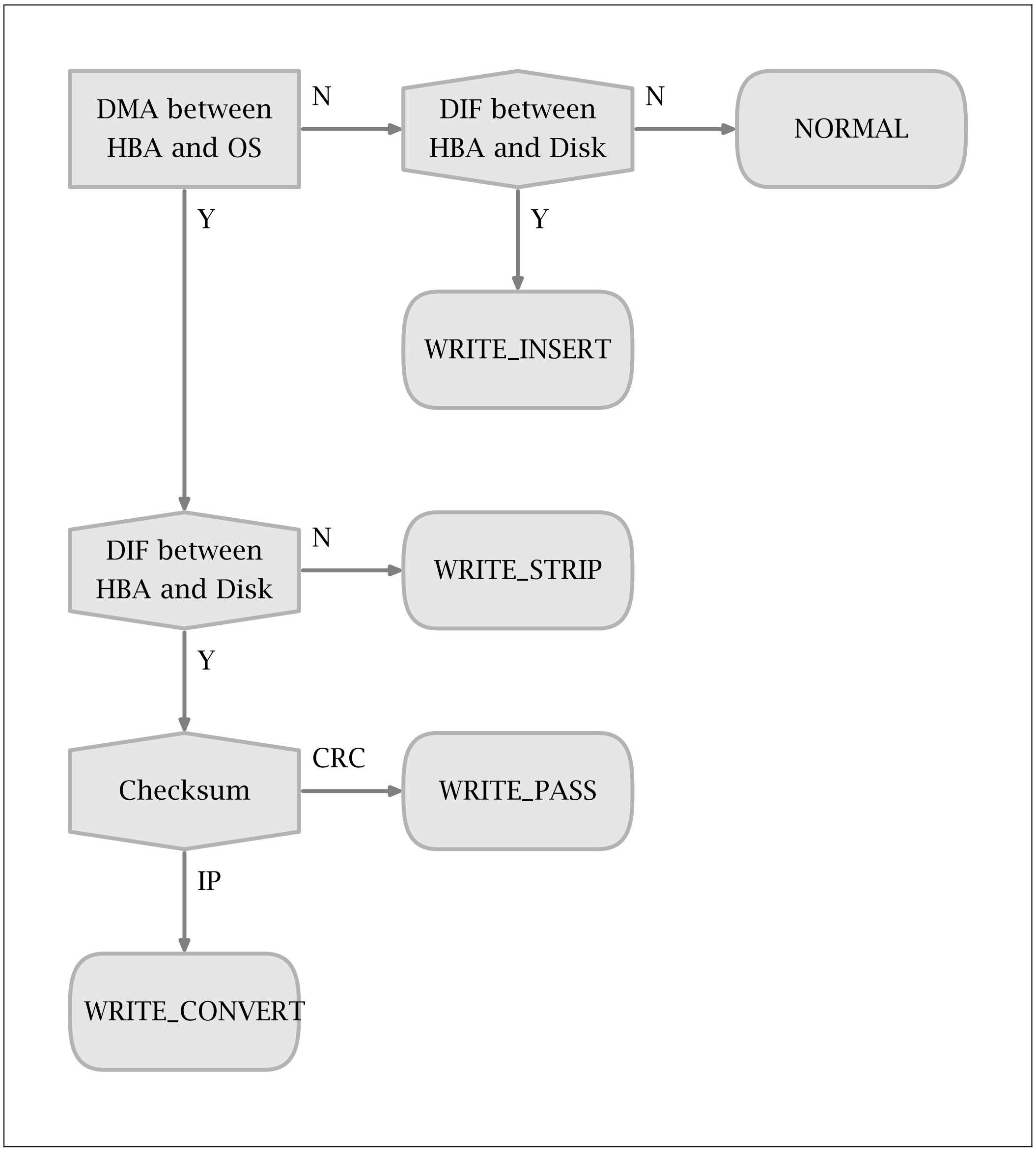

Figure 3 WRITE request