HSII-to-MC3 Data Interface

(ReplicationTriggers)

MIS Executive Overview

William Hanna, Project Leader

Prepared by: Jeff Szuhay
KHP Services, Inc.

April 25, 1995

Introduction

This is intended to provide a *brief* summary of the HSII-to-MC3 Data Interface, often referred to as "replication triggers." Please note that other data pathways between MC3 and HSII data repositories are not discussed.

This document assumes general familiarity with KHP Service's MC3 application, HSII application, as well as certain data communications concepts. Other documents exist to provide more detail; please see the "For Further Reading" section at the end of this overview.

HSII-to-MC3 Data Interface

The HSII-to-MC3 data interface focuses primarily on replicating and synchronizing "triggered" transactions initiated on the HSII "back-end" application with the MC3/Oracle "front-end" system. "The HSII back-end runs in a MUMPS environment on Unix. The MC3 front-end is built with PowerBuilder.

It should be noted that replication triggers are but one type of transaction among many transaction types that are handled by the general system transaction processor. All of the other types of transactions are used internally within the HSII application modules.

Currently, the replication triggers are built into all of the filing operations of the HSII application (all programs) and are the only mechanism to get real-time (non-batched) updates from the HSII/MUMPS system to any external system.

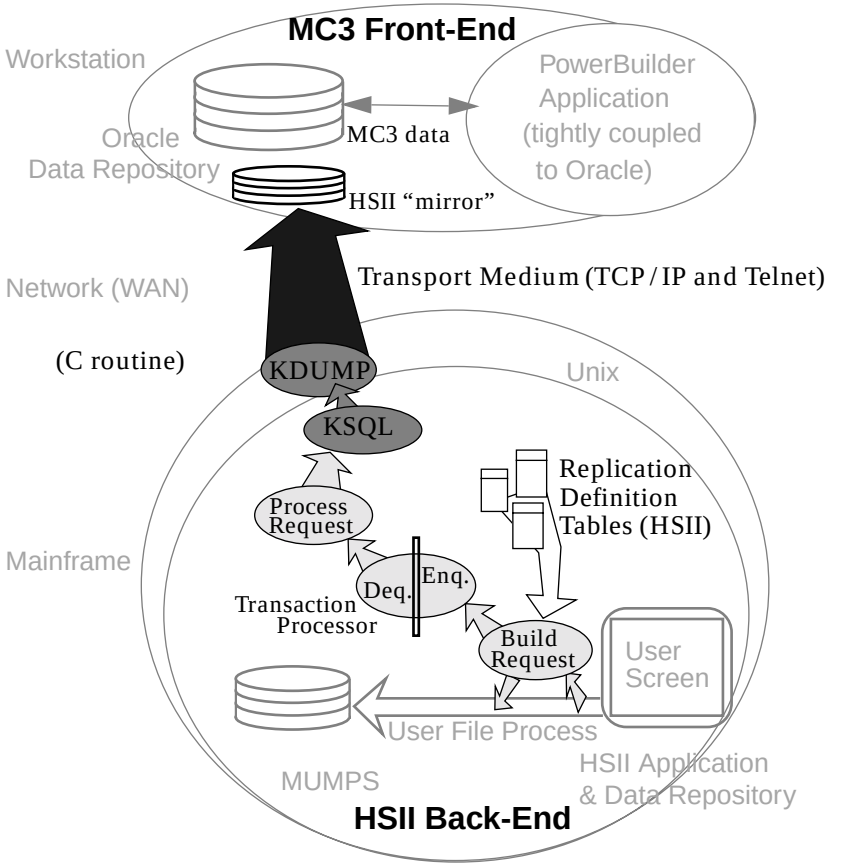
This mechanism does work. However, it was not originally designed to handle the enormous volume expected to be generated by the MC3 applications. In this regard, HSII replication trigger mechanism should be considered a work in process and is currently undergoing massive re-implementation and streamlining. Its general design is not expected to change dramatically from the current design.

This data interface consists of 5 major parts:

1. A set of "mapping" tables which serve to define the mapping of field names/table names from the HSII files to the target SQL relational database schema.
2. the "core" triggering mechanism, which itself consists of building and enqueuing the replication request; processing/handling transaction queue, dequeueing and processing the replication trigger request. The main function here is to get the replication data out of the HSII system in the manner specified by the database administrator. The output of this segment is one or more SQL statements.
3. intermediate transport routines which transmit the SQL statement to the MC3/Oracle repository. One piece of this resides in M and the other is a Unix C program with embedded procedural calls to Oracle.
4. a transport medium from the HSII mainframe over a wide-area network to the MC3/Oracle repository. This consists of TCP/IP utilizing Telnet sessions. It is compatible with Novell, Vines, and LAN Man-

ager; however, none of these is required for its operation.

5. A set of "mirrored" Oracle tables which exist separately from the MC3-specific data tables.



Notes on Each of the 5 Parts

The Replication Definition Tables consist of a set of MUMPS files and supporting screens to map table names and field names from one HSII file to one SQL table. Pre- and post-triggers may be defined for insert, update and delete operations upon a records. These tables are used to build the replication trigger request.

The "core" triggering mechanism consists of a group of related programs to build the transaction request, enqueue the request to the general trans-

action processor(s), dequeue and in a single-threaded manner generate SQL statements to be processed by site-specific custom code (transport routine). The building and enqueuing of the transaction occurs at such a low level that it is "hidden" from the user; user processing continues as if triggers are not present. This family of routines has been developed and is maintained by HSII. Site-specific modification of these routines is strongly discouraged.

The intermediate transport processes transmit the built SQL statements from from HSII/MUMPS file format to Unix text files and from there, a Unix C routine is invoked to interact with Oracle. These routines are relatively simple-minded and straightforward. The bulk of the transport mechanism is handled by embedded SQL*NET library routines in the C routine. These routines, being site-specific, are under our direct control for modification; they are our current, immediate focus for the optimization of this process.

The transport medium is intended to be "network neutral" and currently employs TCP/IP (which requires no additional file or communication server or NOS). It also relies upon a Telnet session with the Oracle host, providing Unix-level security and ease of administration.

The mirrored tables in Oracle provide for rapid updates while promoting synchronization integrity across the network media. Because of the enormous differences in data structures between MC3 and HSII, the existence of these tables provides straightforward maintenance and "local" synchronization entirely on the Oracle data repository.

For Further Reading

Please contact Bill Hanna (760-9059) for the most current edition of each of the documents listed below.

HSII Data Pathways Overview by Jerry Sandridge

HSII Data Replication Triggers by Jeff Szuhay

April 22, 1995, Fax to HSII Regarding Technical Concerns by Jeff Szuhay