Management Data Access

Alex Kirn

akirn@eos.hitc.com

16 April 1996

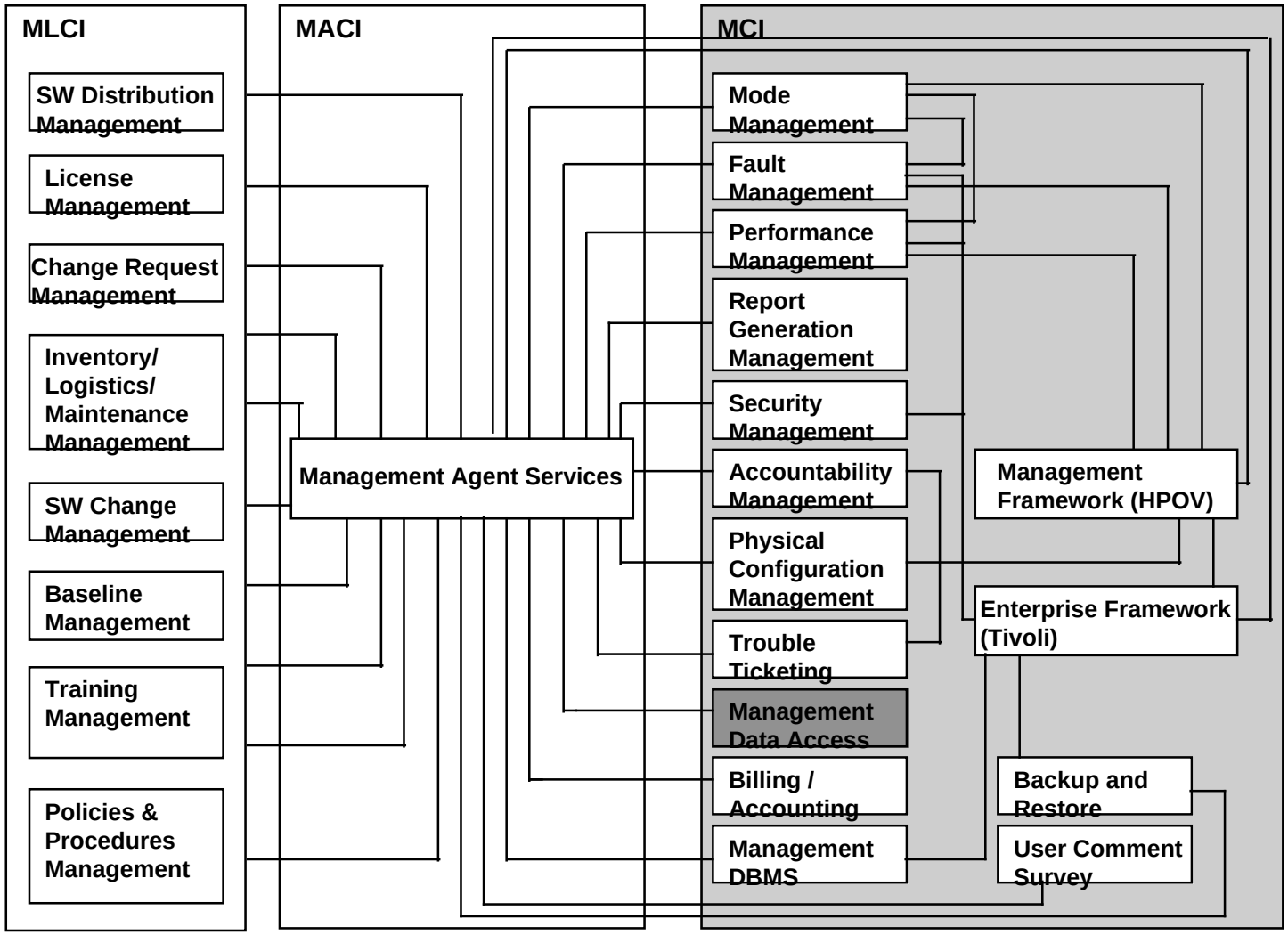
Management Data Access Outline

- Driving Requirements
- Software Design
- Physical Design
- User Interfaces
- Data Model
- Object Model
- Event Trace Diagrams

Driving Requirements

- **MDA shall provide the capability to chain management events to their ancestor event**
- **MDA shall be able to distinguish log records based on mode**
- **MDA shall support a separate management database for each active mode**

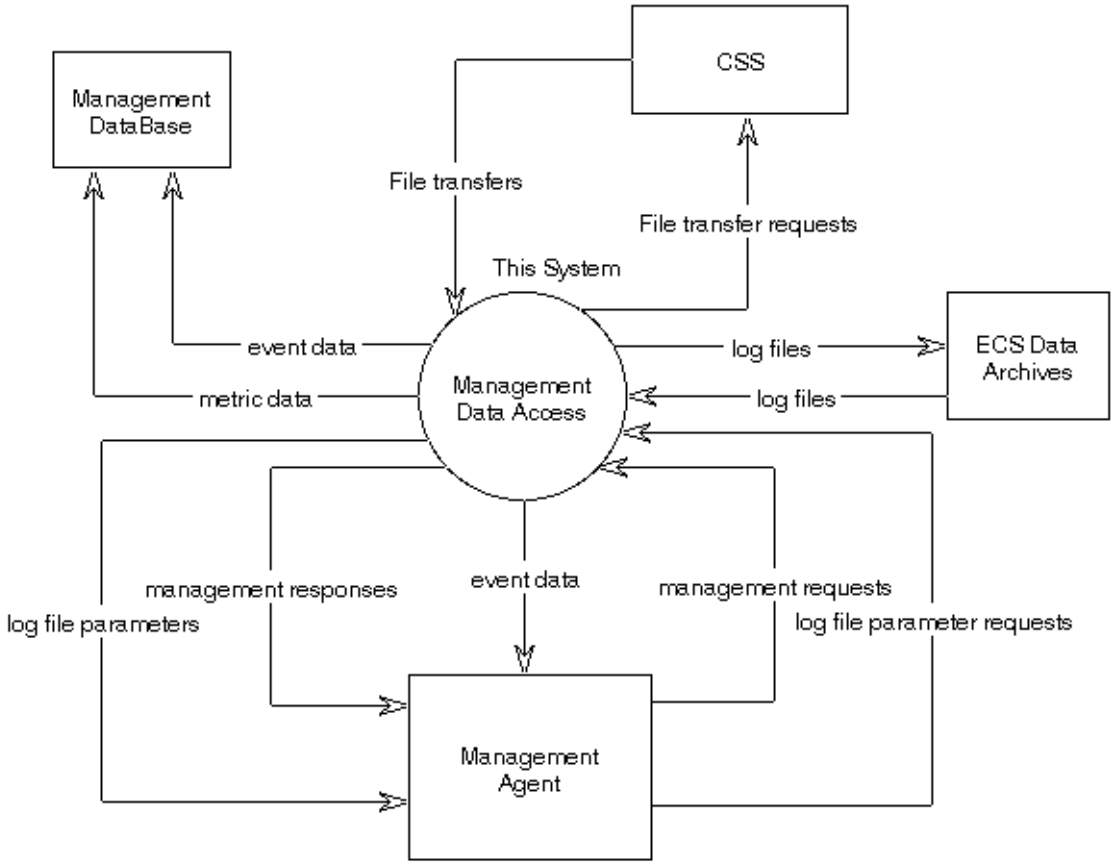
MSS Software Architecture Overview



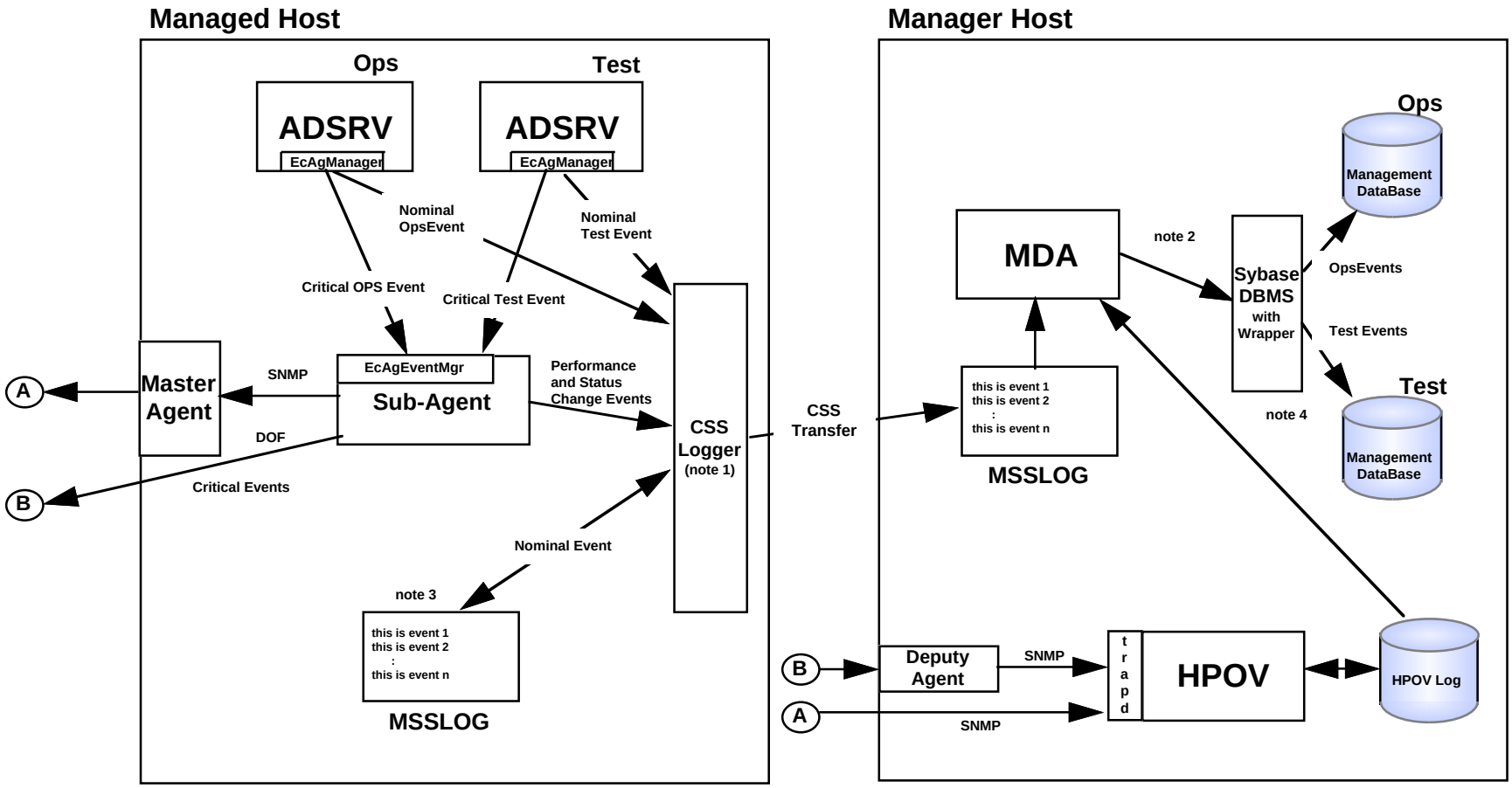
Release B Enhancements

- Event Chaining
  - Done via the log file browser
  - EcAgEvent contains two attributes, rsTransactionID and rsParentTransactionID, which provide a mechanism for linking an event to its parent event
- Mode Management
  - Transfers mode specific events to mode specific management database
  - Transfers non-mode specific events to all mode specific management databases

MDA Software Architecture



MDA Physical Design



Notes:

1. **The CSS Logger is instantiated within each application's address space.**
2. **All non-mode specific events will be logged to each MSS management database.**
3. **Events will be logged with their mode, or "shared" if non-mode specific.**
4. **Database location and file naming convention: /ecs/$SITE/$MODE/adv/<filename>**

User Interface

- **MDA Log Browser**
  - **Allows selective viewing of remote log files and/or local log files**
  - **Provides Event Chaining interface**
- **MDA configuration Interface**
  - **For Scheduling transfers per host**
    - » **Absolute time**
    - » **Time interval**
    - » **Operator initiated**
  - **Specify Processing intervals**

Data Model

- **Details contained in the 311 document.**

Object Model

- **The following object model will be reviewed**

Model Name	Diagram Name	Document Reference
Mgmt Data Access	MDA_Object_Model	305 Vol29, Section 6.10.3

Dynamic Model

- **The following event traces will be reviewed**

Model Name	Diagram Name	Document Reference
Event Chaining	MDA_Event_Chaining	305 Vol29, Section 6.10.5
Log Processing	MDA_Log_Processing	305 Vol29, Section 6.10.6

Event Trace Description

- **Event Chaining**
  - **From the MDA user interface, a user requests to browse log file data. This request specifies a time period, host name and filter.**
  - **The MsMdManager examines the request criteria to calculate the location of the data. In this scenario, the time period specified is assumed to be large enough to require that data be extracted from: the ECS Data Archives, MSS server, and a managed host. It should be noted that this situation is unlikely, and is demonstrated in this scenario purely to illustrate the method for retrieving data from all sources.**
  - **The locations of the data holdings are maintained in the MsMdAggregateLogFileList.**
  - **MsMdManager opens the applicable Management Database based on the mode within the event being chained and retrieves the associated events.**
  - **MsMdManager calls TransferLogFile() to transfer the log file from the remote host to the MSS server.**
  - **MsMdManager calls the MsMdProcessEvent's RetrieveArchivedLogFile to extract the associated events from the ECS Data Archives. When the data has been returned, the required filter is applied to the data**

Event Trace Description

- **Event Chaining (cont.)**
  - **Next the data from the managed host is transferred to the MSS server. This data is appended to the MsMdAggregateLogFile for the particular host. The union of this data is returned to the MDA user interface**
  - **At this point the interface allows the user to chain events by inputting an event TransactionID. The event chain sequence is computed and returned to the user interface**

Event Trace Description

- **Log Processing**
  - **An MSS logfile is scheduled to be processed and transferred to the management database**
  - **The MsMdManager utilizes the MsMdAggregateLogfileList to determine location of the data.**
  - **The MsMdManager then initiates the processing by retrieving the logfile data from the CsFtFTPRelA class.**
  - **Once the data has been retrieved, MsMdProcessEvent determines the active modes of the managed host from which the processing is occuring by reading that host's active mode-configuration file.**
  - **This information is then used by MsMdProcessEvent to open the appropriate management databases (a management database will exist for each active mode)**
  - **MsMdProcessEvent will process, then transfer the logfile records to their appropriate management database according to the event's mode via the LogMgmtData call. When all of the logfile records have been transferred, the open management databases will be closed.**
  - **At this point, the remote logfile has been transferred to the MDA, logfile records processed, and the records written to the management database.**