

November 15-18, 2010



Santa Clara Marriott
Santa Clara, CA

Platform Alert Messages: A Technical Overview

Hemal V. Shah

Associate Technical Director, Broadcom Corporation

Platform Management Sub-committee Chair

DMTF



Disclaimer

- The information in this presentation represents a snapshot of work in progress within the DMTF.
- This information is subject to change. The Standard Specifications remain the normative reference for all information.
- For additional information, see the Distributed Management Task Force (DMTF) Web site.





Presentation Outline

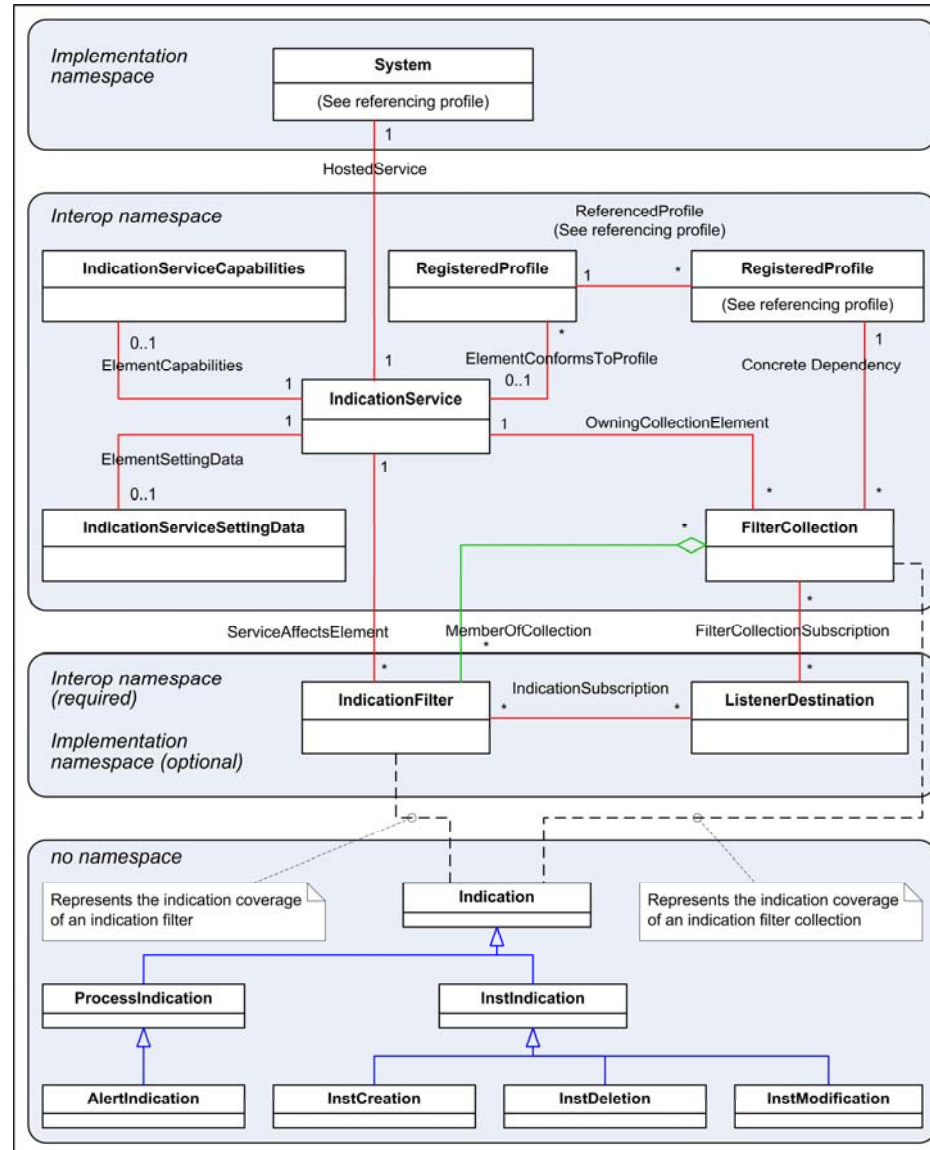
- Indications
 - Lifecycle vs. Alert Indications
 - DSP1054
- Platform Alert Message Registries
 - And how they affect alert indication payload
 - DSP0228, DSP8007
- Mapping messages from ASF & IPMI
 - DSP0244
- Record Log data model for Standard Messages
 - DSP1010 2.0
- Review



Background on Indications

- An **Event** is an occurrence of the phenomenon of interest
- An **Indication** is an observation of characteristics of an event
 - The occurrence of an event may only be derivable from Indications
 - Clients subscribe, specifying indication destination (a listener)
 - Servers produce indications & deliver them to listeners
- DSP 1054 Indication Profile defines
 - **Indication Service**, it's settings and capabilities
 - **Indication Filter**
 - An element that controls the selection of generated Indications for delivery to listeners
 - **Filter Collection**
 - A collection of Indication Filters that can be subscribed to
 - **Listener Destination**
 - Location & method of delivering indications to a client
 - **Alert Indications** and **Lifecycle Indications**

DSP1054 Indication Profile Class Diagram



Platform Alert Message

- Declaration of an Alert in standard message format
- Contains static (text) and dynamic elements
- The context for an instance of the message is provided by the dynamic elements
- Uniquely identified by MessageID & OwningEntity
- Each message describes the possible content for an instance of CIM_AlertIndication
 - A message does not define new classes or new class properties
 - A message does not change the semantics of existing classes or properties



DSP0228 - Message Registry Schema

- XML Schema for DMTF message registries
- Latest published version is 1.1
- DSP0228 Version 1.1 Can be retrieved from
 - http://schemas.dmtf.org/wbem/messageregistry/1/dsp0228_1.1.xsd
- Key elements defined in the schema include
 - OWNING_ENTITY
 - MESSAGE
 - MESSAGE_ID
 - MESSAGE_DESCRIPTION
 - MESSAGE_COMPONENTS
 - DYNAMIC_ELEMENT, STATIC_ELEMENT, REPEAT_DYNAMIC_ELEMENT
 - FIXED_MESSAGE_INSTANCE_VALUES
 - ALERTING_MANAGED_ELEMENT, ALERT_TYPE, PERCEIVED_SEVERITY.... (for Alerts)
 -

Message Registry

- A message registry is a published set of standard messages
- A message registry ensures that the same message is not defined more than once
- DMTF message registries are compliant to
 - DSP0228 Message Registry Schema
- The CIM data model defines standard message properties and their meaning
- The message registry extends this definition by narrowly defining specific content of properties for each message



DSP8007 Platform Alert Message Registry

- A set of platform alert messages corresponding to physical platform events
- Defines messages from various sources
 - PET related messages
 - ASF/IPMI PET assertions and de-assertions related messages
 - CIM related
 - Messages corresponding to DASH and SMASH CIM Profiles
 - Other messages
 - Vendors have been free to add messages as well
- DSP8007 defines the format and semantics of the contents of the following properties of CIM_AlertIndication

```
string Description;  
string AlertingManagedElement;  
uint16 AlertType;  
uint16 PerceivedSeverity;
```

```
string Message;  
string MessageArguments[];  
string MessageID;  
string OwningEntity;
```



Example of an entry

```
<MESSAGE NAME="Fan added">
  <MESSAGE_ID PREFIX="PLAT" SEQUENCE_NUMBER="0456"/>
  <MESSAGE_DESCRIPTION> This message is for the use case when an implementation has
    detected a fan was added. The ComputerSystemObjectPath element contains the CIM object
    path to the computer system.
  </MESSAGE_DESCRIPTION>
  <MESSAGE_COMPONENTS>
    <STATIC_ELEMENT>The Fan </STATIC_ELEMENT>
    <DYNAMIC_ELEMENT NAME="FanElementName" SOURCE_PROPERTY="CIM_Fan.ElementName"
      DATATYPE="string"/>
    <STATIC_ELEMENT> has been added.</STATIC_ELEMENT>
    <DYNAMIC_ELEMENT NAME="ComputerSystemObjectPath" DATATYPE="string"
      EXPECTED_IN_MESSAGE="false"/>
  </MESSAGE_COMPONENTS>
  <FIXED_MESSAGE_INSTANCE_VALUES TYPE="ALERT">
    <ALERTING_MANAGED_ELEMENT>
      <DESCRIPTION> Object Path to alerting CIM_Fan instance indicated in FanElementName
    </DESCRIPTION>
    </ALERTING_MANAGED_ELEMENT>
    <ALERT_TYPE> 5
      <DESCRIPTION>Device Alert</DESCRIPTION>
    </ALERT_TYPE>
    <PERCEIVED_SEVERITY> 2
      <DESCRIPTION>Information</DESCRIPTION>
    </PERCEIVED_SEVERITY>
  </FIXED_MESSAGE_INSTANCE_VALUES>
</MESSAGE>
```

- What new in DSP8007 1.2?
 - New messages (PET Event Data 2/3 related)
 - System firmware progress
 - System firmware error
 - Watchdog time expiration
 - System firmware/software version change
 - Other updates
 - Additional optional arguments to existing messages
 - Description fixes



DSP0244 IPMI PET to Platform Message Registry Mapping

- DSP8007 started with ASF/IPMI PET list
- Original spreadsheet map was turned into DSP0244 and published
- DSP0244 specifies the mapping from sensors/offsets to Platform Alert Messages
- DSP0244 tries to adhere to the following policy:
 - Even numbered messages are for Assertions
 - Odd numbered messages are reserved for de-assertions

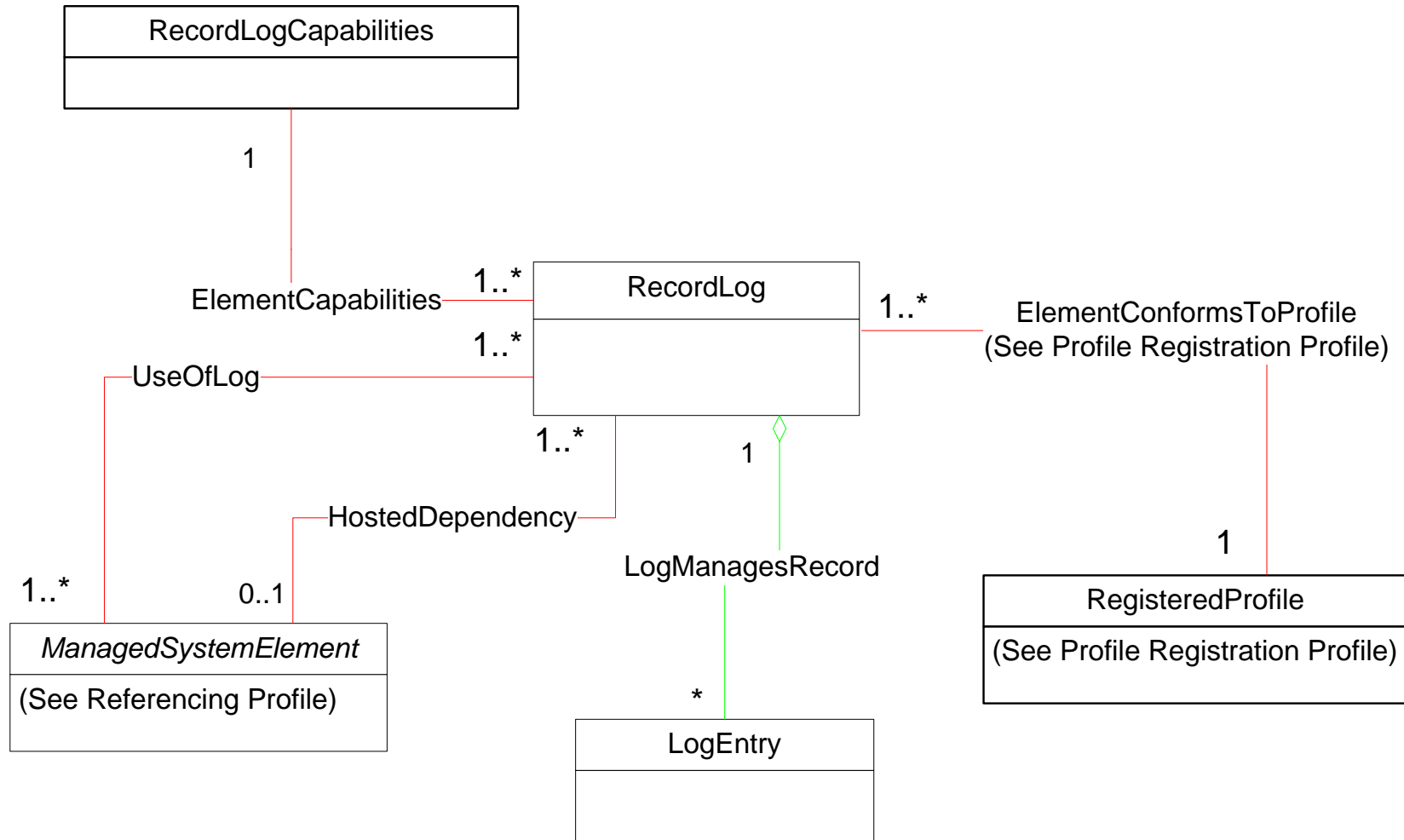
What's new in DSP0244 1.2?

- Mappings for PET Event Data 2/3
 - System firmware progress
 - System firmware error
 - Watchdog time expiration
 - System firmware/software version change
 - Event Logging
 - System Event

Record Log Profile

- Describes
 - Properties/methods of logs generated by managed system/component
 - Association between managed system and logs
 - How log entries are contained within a log
- Defines extrinsic methods for
 - Log state management
 - Clearing of a log

Record Log Profile Class Diagram





What's new in Record Log Profile 2.0?

- Added Record Log entry format definitions
- Two formats defined
 - Record Data format – free form format
 - Standard Message format
- A Record Log entry shall be compliant to one of these two formats
- A Record Log may contain entries with both types
- Additional formats can be defined in the future

CIM_LogEntry class

New properties

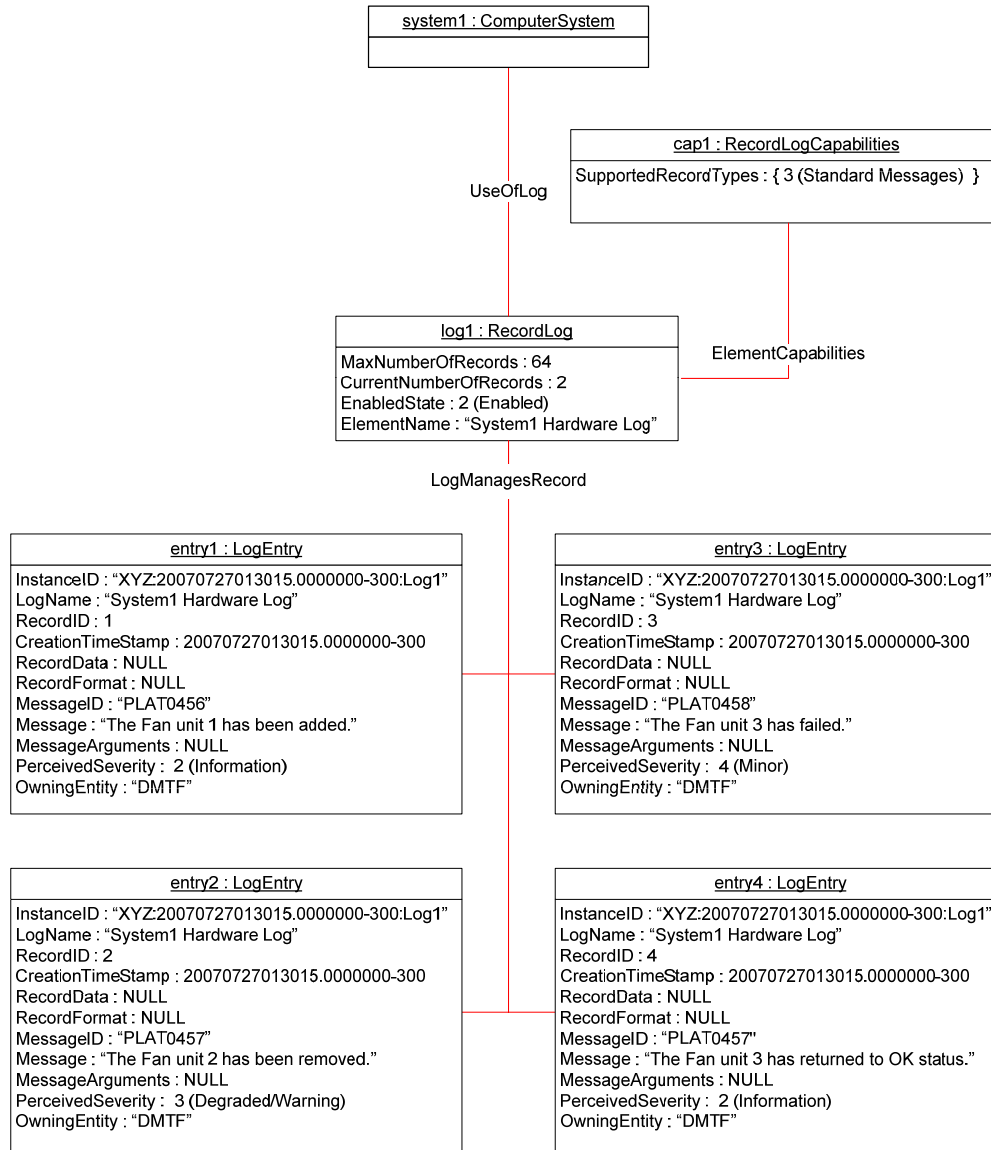
Elements	Requirement	Notes
InstanceID	Mandatory	Key
LogInstanceID	Optional	See 7.1.1.
LogName	Optional	See 7.1.2.
RecordID	Mandatory	None
CreationTimeStamp	Mandatory	None
RecordData	Conditional	See 7.1.3.1.1 and 7.2.1.
RecordFormat	Conditional	See 7.1.3.1.2 and 7.2.1.
ElementName	Mandatory	The property shall match pattern " *"
PerceivedSeverity	Conditional	See 7.1.3.3.4 and 7.2.1.
OwningEntity	Conditional	See 7.1.3.3.5 and 7.2.1.
MessageID	Conditional	See 7.1.3.3.1 and 7.2.1.
Message	Conditional	See 7.1.3.3.2 and 7.2.1.
MessageArguments	Conditional	See 7.1.3.3.2 and 7.2.1.

Only applicable to standard message format

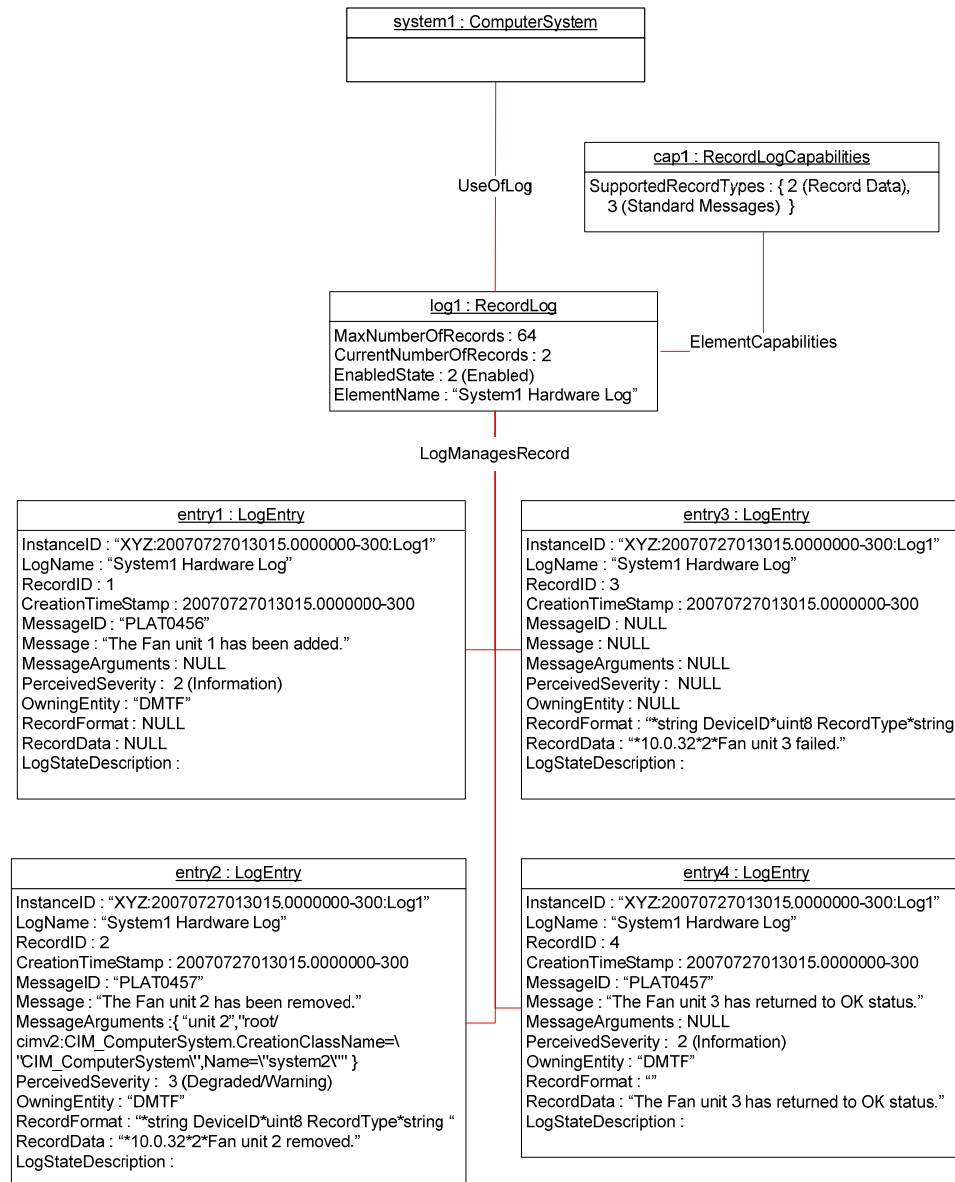
PerceivedSeverity, OwningEntity, MessageID properties and at least one of the properties Message and MessageArguments of the instance of CIM_LogEntry shall be non-NULL for standard message format.



Record Log Instance Diagram for Standard Messages



Record Log Instance Diagram for Both Record Types



Review

- Indication Profile (DSP1054) defines the indication mechanism
- Alert Indications content are defined by message registries
 - Registry Schema definition is in DSP0228
 - DSP8007 is Platform Alert Message Registry
- PET to Platform Alert Messages mapping can be found in DSP0244
- Record Log Profile 2.0 defines log data model for standard messages



Acknowledgements

Thanks to all the contributors and participants of the DMTF's Physical Platform Profiles WG!

Questions?

Thank you.