



Universal Event Subsystem

Event Definitions

Universal Products

Version 3.2.0

Universal Event Subsystem

Event Definitions

Universal Products 3.2.0

Document Name	Universal Event Subsystem 3.2.0 Event Definitions				
Document ID	ues-eventdef-3202				
Products	z/OS	UNIX	Windows	OS/400	HP NonStop
Universal Event Subsystem	√		√		

Stonebranch Documentation Policy

This document contains proprietary information that is protected by copyright. All rights reserved. No part of this publication may be reproduced, transmitted or translated in any form or language or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission, in writing, from the publisher. Requests for permission to make copies of any part of this publication should be mailed to:

Stonebranch, Inc.
950 North Point Parkway, Suite 200
Alpharetta, GA 30005 USA
Tel: (678) 366-7887
Fax: (678) 366-7717

Stonebranch, Inc.[®] makes no warranty, express or implied, of any kind whatsoever, including any warranty of merchantability or fitness for a particular purpose or use.

The information in this documentation is subject to change without notice.

Stonebranch shall not be liable for any errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this document.

All products mentioned herein are or may be trademarks of their respective owners.

© 2007-2010 by Stonebranch, Inc.

All rights reserved.



Summary of Changes

Changes for Universal Event Subsystem 3.2.0 Event Definitions (ues-eventdef-3202) September 8, 2009

Universal Enterprise Controller 3.2.0.4

- Modified Section [2.4.15 UBR_COMP_STOP_REQUEST \(307\)](#):
 - Added a condition under which the event is generated.
 - Deleted two fields from the event record body.

Changes for Universal Event Subsystem 3.2.0 Event Definitions (ues-eventdef-3201) September 5, 2008

- Added toll-free telephone number for North America in [Appendix B Customer Support](#).
- Added numeric values for component communication states in Section [2.4.12 UBR_COMP_COMM_STATE \(304\)](#).
- Added ARC format for UES exported data in Section [1.4 Event Data Export Formats](#).

Changes for Universal Event Subsystem 3.2.0 Event Definitions (ues-eventdef-320) May 16, 2008

Universal Enterprise Controller 3.2.0.0

- This is the first version of the Universal Event Subsystem Event Definitions document.

Contents

Summary of Changes	5
Contents	6
List of Tables	9
Preface	10
Document Structure	10
Conventions	10
Vendor References	11
Document Organization	12
Chapter 1 Overview	13
1.1 Introduction	13
1.2 Universal Products Architecture	14
1.2.1 Components and Associations	15
1.3 Event Structure	17
1.3.1 Event Types	17
1.3.2 Event Data	18
Data Formats	18
Data Characteristics	18
1.4 Event Data Export Formats	19
1.4.1 Exported Fields	19
1.4.2 XML Format	19
1.4.3 CSV Format	20
1.4.4 ARC Format	20

Chapter 2 Event Definitions	21
2.1 Overview	21
2.1.1 Event Components	21
2.2 Event Header	22
2.3 Event Types Summary	25
2.3.1 Event Type Ranges	25
2.3.2 Universal Products	26
2.3.3 Universal Command	27
2.3.4 Universal Broker	28
2.3.5 Universal Data Mover	29
UDM Sessions	30
2.3.6 Universal Control	31
2.4 Event Definition Details	32
2.4.1 USER_AUTHENTICATION (100)	33
2.4.2 UACL_AUTHORIZATION (101)	34
2.4.3 RESOURCE_AUTHORIZATION (102)	36
2.4.4 UCMSRV_JOB_START (200)	38
2.4.5 UCMSRV_JOB_STOP (201)	40
2.4.6 UCM_WORK (204)	42
2.4.7 UCMSRV_WORK (205)	44
2.4.8 UBR_START (300)	47
2.4.9 UBR_STOP (301)	48
2.4.10 UBR_SRV_COMP_START (302)	49
2.4.11 UBR_SRV_COMP_STOP (303)	51
2.4.12 UBR_COMP_COMM_STATE (304)	53
2.4.13 UBR_MGR_COMP_START (305)	55
2.4.14 UBR_MGR_COMP_STOP (306)	56
2.4.15 UBR_COMP_STOP_REQUEST (307)	58
2.4.16 UBR_MGR_RESTART (308)	60
2.4.17 UDM_OPEN_COMMAND_START (400)	61
2.4.18 UDM_COPY_COMMAND_START (401)	62
2.4.19 UDM_COPYDIR_COMMAND_START (402)	63
2.4.20 UDM_DELETE_COMMAND_START (403)	64
2.4.21 UDM_RENAME_COMMAND_START (404)	65
2.4.22 UDM_OPEN_COMMAND_STOP (405)	66
2.4.23 UDM_COPY_COMMAND_STOP (406)	67
2.4.24 UDM_COPYDIR_COMMAND_STOP (407)	68
2.4.25 UDM_DELETE_COMMAND_STOP (408)	69
2.4.26 UDM_RENAME_COMMAND_STOP (409)	70
2.4.27 UDM_FILE_DELETE (450)	71
2.4.28 UDM_FILE_RENAME (451)	72
2.4.29 UDM_FILE_XFER_START (452)	73

2.4.30 UDM_FILE_XFER_STOP (453)	74
2.4.31 UDM_EXEC_OPERATION_START (470)	75
2.4.32 UDM_EXEC_OPERATION_END (471)	76
2.4.33 UDM_EXECSAP_OPERATION_START (472)	77
2.4.34 UDM_EXECSAP_OPERATION_END (473)	78
2.4.35 UCTSRV_CMD_START (800)	79
2.4.36 UCTSRV_CMD_STOP (801)	81
2.4.37 UCT_CMD_REQUEST (802)	83
Appendix A Universal Products Error Codes	84
Appendix B Customer Support	86

List of Tables

Chapter 1 Overview	13
Table 1.1 Event Record Fields - Data Formats	18
Table 1.2 Event Record Fields - Data Characteristics	18
Chapter 2 Event Definitions	21
Table 2.1 Event Header Fields	24
Table 2.2 Universal Products - Event Types	25
Table 2.3 Universal Products - Event Types	26
Table 2.4 Universal Command - Event Types	27
Table 2.5 Universal Broker - Event Types	28
Table 2.6 Universal Data Mover - Event Types	29
Table 2.7 Universal Control - Event Types	31
Table 2.8 Component Communication States	54
Appendix A Universal Products Error Codes	84
Table A.1 Universal Products Error Codes	85

Preface

Document Structure

This document is written using specific conventions for text formatting and according to a specific document structure in order to make it as useful as possible for the largest audience. The following sections describe the document formatting conventions and organization.

Conventions

Specific text formatting conventions are used within this document to represent different information. The following conventions are used.

Typeface and Fonts

This Font identifies specific names of different types of information, such as file names or directories (for example, `\abc\123\help.txt`).

Operating System-Specific Text

Most of this document describes the product in the context of all supported operating systems. At times, it is necessary to refer to operating system-specific information. This information is introduced with a special header, which is followed by the operating system-specific text in a different font size from the normal text.

z/OS

This text pertains specifically to the z/OS line of operating systems.

This text resumes the information pertaining to all operating systems.

Vendor References

References are made throughout this document to a variety of vendor operating systems. We attempt to use the most current product names when referencing vendor software.

The following names are used within this document:

- **z/OS** is synonymous with IBM z/OS and IBM OS/390 line of operating systems.
- **Windows** is synonymous with Microsoft's Windows 2000 / 2003 / 2008, Windows XP, Windows Vista, and Windows 7 lines of operating systems. Any differences between the different systems will be noted.
- **UNIX** is synonymous with operating systems based on AT&T and BSD origins and the Linux operating system.

Note: These names do not imply software support in any manner. For a detailed list of supported operating systems, see the Universal Products 3.2.0 Installation Guide.

Document Organization

The document is organized into the following sections:

- [Overview](#) (Chapter 1)
General architectural and functional overview of the product
- [Event Definitions](#) (Chapter 2)
Definitions for all event messages generated by Universal Products components.
- [Universal Products Error Codes](#) (Appendix A)
List of Universal Products error codes.
- [Customer Support](#) (Appendix B)
Customer support contact information for Universal Event Subsystem Event Definitions.

Chapter 1

Overview

1.1 Introduction

Within Universal Products, an event is the occurrence of some action or condition at a particular location in the computer network and at a particular time at that location. There are a number of different types of events, such as the start of a Universal Products component, a user authentication failure, or a file transfers completing.

The Universal Event Subsystem (UES) provides the means by which Universal Products components generate data about those events and, in a single repository, have those events recorded. This collection of recorded events (that is, the event records) is maintained in the UES database and archived to external storage. It represent the work and activity of all distributed workload managed by Universal Product components.

Universal Products consist of a set of components distributed across a computer network. The components work together to perform some unit of work. The components that are working together have an association that must be maintained in the event data. For that reason, UES event records not only include information about the event, but also information about associations between the components reporting the events.

The Universal Enterprise Controller (UEC) maintains a central UES database for all event data within its domain of responsibility. The UES database contains all UES event records collected by UEC from Universal Broker components that are defined to it. The UES database provides medium-term persistent storage for the UES events. Periodically, the UES database events must be exported to long-term storage in order to maintain a historical record of events. If the export is not performed periodically, the UES database will continue to grow and eventually exhaust all disk space available to it.

1.2 Universal Products Architecture

Universal Products consist of many product components distributed throughout a computer network working together to perform a unit of work. Some components remain executing as long as the operating system on which they are installed remains executing. Other components are being started and stopped, as necessary, when new work is requested.

Universal Event Subsystem (UES) event data provides the information necessary to associate the distributed Universal Product components working together in performing a unit of work.

Examples of components and their associations are:

- Universal Command Manager is associated with a remote Universal Command Server, and the Universal Command Server is associated with the job process it has started on behalf of the Universal Command Manager.
- Universal Data Mover Manager is associated with a remote Universal Data Mover Server, and the Universal Data Mover Server is associated with a file being transferred on behalf of the Universal Data Mover Manager.

The components and their associations partly define the Universal Products architecture. This section provides the necessary understanding of the Universal Product architecture as presented by the UES event data.

1.2.1 Components and Associations

UES event data has many references to components. In the context of UES, a component is a Universal Products program, a job or command, a file, or any object that is being managed by Universal Products.

Components are highly likely to be associated with other components, since Universal Products consists of components distributed across a computer network working together to accomplish a unit of work. UES event data represents component associations in order to provide a complete understanding of a distributed unit of work.

Components are uniquely identified using a Universally Unique Identifier (UUID). The UUID format is based upon a version 4 UUID as defined in ISO/IEC 11578:1996. The value of each UUID is unique regardless of the time or location in which it was generated. The UUID allows for each instance of a component execution to be uniquely identified.

There are four UUID values maintained in all event records:

- Object UUID
- Parent UUID
- Unit UUID
- Group UUID

The object UUID uniquely identifies the component, or object, from all other component instances. No two components will have the same object UUID value.

The parent UUID defines the association between a child object and its parent object. The parent object either starts or is responsible for the child object. For example, a Universal Command Manager starts a Universal Command Server. In this case, the Server component is the child object and the Manager component is the parent object.

The unit UUID identifies a set of components working together to perform a service. The set of components are distributed on different host systems and communicating with each other over the network to accomplish the requested service. The set of components is considered a work unit and identified by the unit UUID. An example would be a Universal Command Manager job is comprised of a Manager, Server and Job component. The three components comprise the work unit.

The group UUID identifies a set of work units that are working together to perform a service. Some services require multiple work units to accomplish the service. A work group may consist of one to many work units. An example would be a Universal Data Mover (UDM) Manager that executes a Universal Command Manager (UCM) to perform a service. There are two work units: UDM and UCM.

There is one component that is the head of a work unit and work group. A component with an object UUID that matches a unit UUID is the head of the work unit. Similarly, a component with an object UUID that matches the group UUID is the head of the work group.

As an example, [Figure 1.1](#), below, illustrates several different components and their associations with each other.

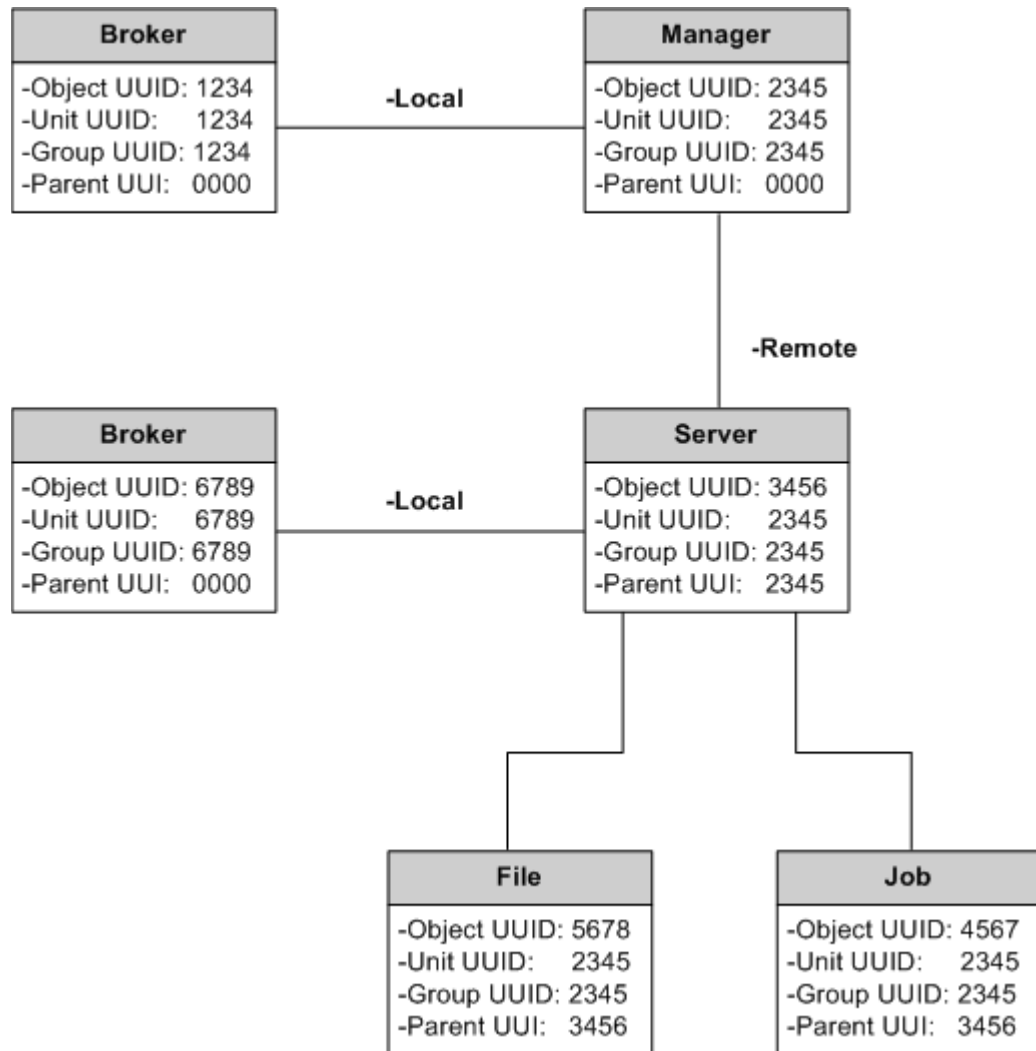


Figure 1.1 Universal Event Subsystem - General Architecture

The following points should be noted about this illustration:

- Each component has an object, a parent, a unit, and a group UUID. The object UUID uniquely identifies each component.
- The Manager component is the unit and group leader, as its object UUID is equal to its unit and group UUID.
- The Manager, Server, Job, and File objects all have the same unit and group UUID.
- The Server's parent UUID identifies the Manager as its parent, while the job's parent UUID identifies the Server as its parent.
- Broker components report on Manager and Server components, but there are no associations made between a Broker and components started on some other entity's behalf.

1.3 Event Structure

Events are comprised of an event header and an event body:

- Event header is comprised of a fixed number of fields, regardless of the event type.
- Event body content varies based on the event type.

Event data generated by a Universal Products component is related either to the component itself or to another component or object. The event header distinguishes between these two cases with the source and reporter fields.

- Source fields identify the component or object that is the topic of the event.
- Reporter fields identify the component reporting the event.

If the source and reporter are the same entity, the reporter fields are empty.

1.3.1 Event Types

There are a number of different types of events (that is, event records) created by UES. New types of events may be added with new releases or product maintenance.

Each type of event is identified by the `eventType` field in the event header (see [Table 2.1 Event Header Fields](#)).

1.3.2 Event Data

Each event consists of a multiple fields. The data in the fields has different formats and characteristics.

Data Formats

Field Format	Description
String	Text value consisting of any printable character. Depending on the value there may or may not be any restrictions on its length.
Signed integer	Signed integer value ranging from -2,147,483,646 to 2,147,483,647.
Unsigned integer	Unsigned integer value ranging from 0 to 4,294,967,295.
Boolean	True or false (or yes or no) value, where: <ul style="list-style-type: none"> • 1 represents true or yes. • 0 represents false or no.
UUID	Universally Unique Identifier that is a 128-bit value that is represented externally as a string value according to ISO/IEC 11578:1996.
Date and time	All date and time values are represented as Coordinated Universal Time (UTC).

Table 1.1 Event Record Fields - Data Formats

Data Characteristics

Field Format	Description
Case dependent	String value that may be case sensitive depending on the originating O/S or file system.
Optional	Some fields may not contain any value if not applicable or not provided by the source of the event.

Table 1.2 Event Record Fields - Data Characteristics

1.4 Event Data Export Formats

UES event data is exported from Universal Enterprise Controller's UES database using the UEC Load utility. (For information on UEC Load, see the Universal Enterprise Controller 3.2.0 User Guide.)

There are three formats for UES exported data:

1. XML Event data is exported as an XML (extensible markup language) document. XML is often used for exchanging data between two systems.
2. CSV Event data is exported as a CSV (comma separated value) text file. The CSV file can be used by most spreadsheet software, such as Microsoft Excel.
3. ARC Event data is exported in ARC (archival) format for long-term storage. The exported ARC format data (written in UTF-8) is portable between operating systems with differing code pages.

1.4.1 Exported Fields

Exported event data includes all the event fields defined by the event record including the event header and event body.

Additionally, two other fields are exported that are maintained by Universal Enterprise Controller:

- **agentName** is the name of the agent assigned to the agent definition in UEC.
- **agentUuid** is the UUID that identifies the Universal Product installation. The UUID is provided by the Universal Broker and remains the same for the life of the Broker installation. Note that if the product is uninstalled and reinstalled, or the Broker databases are deleted, a new agent UUID is generated.

1.4.2 XML Format

The XML (extensible markup language) format consists of an exported document enclosed in an export tag. Each event record is enclosed in a record tag. The event fields are enclosed in tags named after the field name.

Date and time formats are specified as:

yyyy-mm-ddThh:mm:ss

The date and time are separated by the character T.

1.4.3 CSV Format

The CSV (comma separated value) format consists of all event records exported as a row and all event fields as a column in the CSV file. The first row is a column heading row.

Since all event records are exported in the CSV file, there is a column for each event field for all the event types. This produces a large number of columns of which only a subset pertain to any particular row (event record). The column names are based upon the event field name.

Due to limitations in CSV date and time formats, two columns are produced for all date and time fields. The column names are appended with `_date` and `_time` for the date portion and time portion, respectively.

1.4.4 ARC Format

The ARC (archival) format data is used internal to the product only. It can be read by UECLoad and exported into either XML or CSV formats.

Chapter 2

Event Definitions

2.1 Overview

This chapter provides detailed information about the event data generated by all Universal Product components and recorded by UES.

2.1.1 Event Components

Each event (that is, event record) is comprised of two parts:

- [Event Header](#)
- Event Body

2.2 Event Header

Table 2.1, below, describes the fields in an event header.

Field Name	Description
eventCategory	<p>Category of the event.</p> <p>Event categories are:</p> <ul style="list-style-type: none"> • Availability (10) Deals with situations reported from the component regarding its operational state and availability. • Configuration (20) Deals with components identifying configuration changes. • Stop (30) Deals with the shutdown process for a component. • Start (40) Deals with the startup process for a component. • Network (50) Deals with situations related to network errors between components. • Exception (60) Some exception has occurred within the resource not covered by any other category. • Security (70) Some security issue has been detected. • Heartbeat (80) A component has been configured to send a heartbeat event. • Status (90) Some change of status that does not affect availability or capability of the component. • Log (100) A component has been configured to report log entries. • Diagnostic (110) A component has been configured to generate diagnostic events. <p>This category list is ordered by priority, from highest to lowest.</p>
eventSeverity	<p>Severity of the event with respect to the reporting component.</p> <p>Severity values are:</p> <ul style="list-style-type: none"> • Fatal (10) A condition is unrecoverable and the service is no longer available. • Critical (20) A condition affecting the service has occurred. Immediate corrective action is required. • Major (30) A problem with relatively high severity has occurred. It is likely that normal use of the service is impeded. • Minor (40) A problem with relatively low severity has occurred. It is unlikely that normal use of the service is impeded. • Warning (50) A problem with the service may occur or may have occurred. Diagnostic and corrective action is recommended. • Informational (60) A message considered purely normal and expected. <p>This severity list is ordered by priority, from highest to lowest.</p>

Field Name	Description
eventTopic	Type of object to which the event pertains. Event topic values are: <ul style="list-style-type: none"> • Job A process created by a Universal Products component to perform work on behalf of a client request. • User A user identity associated with work performed by a Universal Products component. • UACL A Universal Access Control List entry. • Component A Universal Product component, which includes a manager or server component. • File A file resource in the O/S file system. • Event A Universal Event Monitor event resource. • Spool A Universal Products spool database resource. • Command An O/S command, an O/S API, or a component script command.
eventType	Uniquely identifies the event class. Events with the same type represent a specific class of events.
eventUuid	Universally Unique Identifier of each event message generated on any system at any time.
recordedTime	UTC date and time that the event was recorded by the local Broker.
recordedTimeUsec	Microsecond portion of the TIME value.
reporterGroupUuid	Universally unique identifier of the work group of which the reporter is a member. If the reporterGroupUuid and the reporterObjectUuid are the same, the component is the work group leader.
reporterIpAddr	Host name of the component that generated the event. This is the host name as reported by the operating system, typically using the gethostname () function.
reporterName	Name of the component that generated the event. Valid names include: <ul style="list-style-type: none"> • UBROKER • UCMD-MANAGER • UCMD-SERVER • UDM-MANAGER • UDM-SERVER • UEM-MANAGER • UEM-SERVER • UEC • USAP • UCTL-MANAGER • UCTL-SERVER
reporterObjectUuid	Universally unique identifier of the object identified as the reporter of the event.
reporterParentUuid	Universally unique identifier of the parent component of the process that reported the event.
reporterSysId	Installation system identifier of the component that generated the event. The system identifier identifies a specific Universal Products installation on a host system.
reporterUnitUuid	Universally unique identifier of the work unit of which the reporter is a member. If the reporterUnitUuid and the reporterObjectUuid are the same value, the component is the unit leader.

Field Name	Description
sourceGroupUuid	Universally unique identifier of the work group of which the source is a member. If the sourceGroupUuid and the sourceObjectUuid are the same, the component is the work group leader.
sourceIpAddr	Host name of the component that generated the event. This is the host name as reported by the operating system, typically using the <code>gethostname()</code> function.
sourceName	Name of the component that generated the event. Valid names include: <ul style="list-style-type: none"> • UBROKER • UCMD-MANAGER • UCMD-SERVER • UDM-MANAGER • UDM-SERVER • UEM-MANAGER • UEM-SERVER • UEC • USAP • UCTL-MANAGER • UCTL-SERVER
sourceObjectUuid	Universally unique Identifier of the object identified as the source of the event. The type of object is represented by the eventTopic .
sourceParentUuid	Universally unique identifier of the parent component of the process that generated the event.
sourceSysId	Installation system identifier of the component that generated the event. The system identifier identifies a specific Universal Products installation on a host system.
sourceUnitUuid	Universally unique identifier of the work unit of which the source is a member. If the sourceUnitUuid and the sourceObjectUuid are the same value, the component is the unit leader.
timeZone	Time zone in which the event was generated. The value is the number of seconds between UTC and local time.
version	Numeric value that is incremented whenever the header changes (that is, changes to the header format or header fields).

Table 2.1 Event Header Fields

2.3 Event Types Summary

Each Universal Products event is categorized by type. Each event type uniquely identifies a specific class of events.

This section provides a summary of the event types for each Universal Product.

2.3.1 Event Type Ranges

Each event type has a unique three-digit number. This number is identified in the event header (see the **eventType** field in [Table 2.1 Event Header Fields](#)).

Each Universal Product is assigned a range of numbers for its event types (see [Table 2.2](#), below).

Universal Product	Event Type Range
Universal Products (all)	100-199
Universal Command	200-299
Universal Broker	300-399
Universal Data Mover	400-499
Universal Control	800-899

Table 2.2 Universal Products - Event Types

2.3.2 Universal Products

Universal Products event types represent events that apply generically to all Universal Product components. The header source fields identify the component that generated the event.

Universal Product event types are in the range of 100-199.

Event Name	Type	Category	Description
USER_AUTHENTICATION	100	Security	Authentication of a user identifier failed.
UACL_AUTHORIZATION	101	Security	A client request was denied access to a Universal Access Control List protected resource.
RESOURCE_AUTHORIZATION	102	Security	Authorization was denied to a general resource. The resource is a non-UACL protected resource.

Table 2.3 Universal Products - Event Types

2.3.3 Universal Command

Universal Command event types represent events that are generated by Universal Command Managers and Servers.

Universal Command event types are in the range of 200-299.

Event Name	Type	Category	Description
UCMSRV_JOB_START	200	Start	A UCMD Server starts a job process at the request of a UCMD Manager.
UCMSRV_JOB_STOP	201	Stop	A UCMD Server detects the job process stopped.
UCM_WORK	204	Status	Universal Command Manager logs the work it has performed.
UCMSRV_WORK	205	Status	Universal Command Server logs the work it has performed.

Table 2.4 Universal Command - Event Types

2.3.4 Universal Broker

Universal Broker event types represent events that are generated by Universal Brokers. Universal Broker event types are in the range of 300-399.

Event Name	Type	Category	Description
UBR_START	300	Start	Universal Broker has started and completed its stage 1 initialization steps. Stage 1 initialization steps include all steps required to initialize the event subsystem.
UBR_STOP	301	Stop	Universal Broker has stopped.
UBR_SRV_COMP_START	302	Start	Universal Broker started a Server component at the request of a client.
UBR_SRV_COMP_STOP	303	Stop	Server component that the Universal Broker started has stopped.
UBR_COMP_COMM_STATE	304	Status	Component's communication state changed.
UBR_MGR_COMP_START	305	Start	Universal Broker detected the start of a Manager component.
UBR_MGR_COMP_STOP	306	Stop	Universal Broker detected the stop of a Manager component.
UBR_COMP_STOP_REQUEST	307	Log	Universal Broker receives and processes a request to stop a component.
UBR_MGR_RESTART	308	Start	Universal Broker has been notified by a Universal Command Manager that it has successfully restarted with a Universal Command Server.

Table 2.5 Universal Broker - Event Types

2.3.5 Universal Data Mover

Universal Data Mover event types represent events that are generated by Universal Data Mover Managers and Servers.

Universal Data Mover event types are in the range of 400-499.

Event Name	Type	Category	Description
Command-Related Events			
UDM_OPEN_COMMAND_START	400	Start	UDM opens a session.
UDM_COPY_COMMAND_START	401	Start	UDM starts a copy operation.
UDM_COPYDIR_COMMAND_START	402	Start	UDM starts a directory copy operation.
UDM_DELETE_COMMAND_START	403	Start	UDM starts a delete operation.
UDM_RENAME_COMMAND_START	404	Start	UDM starts a rename operation.
UDM_OPEN_COMMAND_STOP	405	Stop	UDM session has stopped.
UDM_COPY_COMMAND_STOP	406	Stop	UDM copy operation has stopped.
UDM_COPYDIR_COMMAND_STOP	407	Stop	UDM directory copy operation has stopped.
UDM_DELETE_COMMAND_STOP	408	Stop	UDM delete operation has stopped.
UDM_RENAME_COMMAND_STOP	409	Stop	UDM rename operation has stopped.
File-Related Events			
UDM_FILE_DELETE	450	Status	UDM file is deleted.
UDM_FILE_RENAME	451	Status	UDM file is renamed.
UDM_FILE_XFER_START	452	Start	UDM starts a transfer operation for an individual file or directory.
UDM_FILE_XFER_STOP	453	Stop	UDM transfer operation for an individual file or directory has stopped.
Execution Events			
UDM_EXEC_OPERATION_START	470	Start	UDM starts an EXEC operation.
UDM_EXEC_OPERATION_END	471	Stop	UDM EXEC operation ends.
UDM_EXECSAP_OPERATION_START	472	Start	UDM starts an EXECSAP operation.
UDM_EXECSAP_OPERATION_END	473	Stop	UDM EXECSAP operation ends.

Table 2.6 Universal Data Mover - Event Types

UDM Sessions

In most operations, UDM is session oriented. A UDM session is a connection between a UDM Manager and one or more UDM servers through which commands are executed. There is no limit to the number of commands that can be executed (sequentially) in a given session. Sessions are identified by a session UUID that is created when an **open** command is issued in UDM.

It is also important to keep in mind that a single UDM Manager instance may establish and close (sequentially) more than one session during its lifetime.

Events are not issued for every UDM command, but some commands (file operation and the open command) will have events that are issued by the UDM manager. The body of command events will have a session UUID (if applicable) and a transaction ID. The transaction ID is an identifier used by UDM that uniquely identifies a given command within the context the UDM Manager instance. This identifier cannot be assumed to be unique outside of the manager instance in which it was generated.

2.3.6 Universal Control

Universal Control event types represent events that are generated by Universal Control Managers and Servers.

Universal Control event types are in the range of 800-899.

Event Name	Type	Category	Description
UCTSRV_CMD_START	800	Start	Universal Control Server initiates a request for a control command to be executed by the local Universal Broker on behalf of a Universal Control Manager client.
UCTSRV_CMD_STOP	801	Stop	A control command initiated by a Universal Control Server has stopped.
UCT_CMD_REQUEST	802	Log	Universal Control Manager requests the execution of a control command by a Universal Control Server.

Table 2.7 Universal Control - Event Types

2.4 Event Definition Details

This section provides the following information for each event definition:

- Description
- Event header
- Event body

It lists the event definitions in numerical order by event type.

2.4.1 USER_AUTHENTICATION (100)

The event is generated when a Universal Products component was not able to successfully authenticate a user identifier and password.

USER_AUTHENTICATION Event Header

Field Name	Description
eventCategory	Security
eventSeverity	Major
eventTopic	User
eventType	100
(Source Fields)	Universal Products component that encountered the failed user authentication.
(Reporter Fields)	

USER_AUTHENTICATION Event Body

Field Name	Notes	Description
certId	1, 2	UACL certificate map identifier to which the client's digital certificate mapped.
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientId	1, 3, 4	User identifier of the client.
clientWorkId	1	Work identifier of the client.
userId	3	User identifier that failed authentication.

Notes

1. Client fields identify the client that requested user authentication. A client is not always the initiator of the authentication request however. Authentication may be performed by a Universal Products component on its own behalf as part of its normal processing.
2. Digital certificate information is only provided if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value always is considered case sensitive.

2.4.2 UACL_AUTHORIZATION (101)

The event is generated when a Universal Products component was denied access by a Universal Access Control List entry.

The conditions under which the event is generated include:

- A Broker or Server component finds that a client request matched a UACL entry that denies access to the protected resource.

UACL_AUTHORIZATION Event Header

Field Name	Description
eventCategory	Security
eventSeverity	Major
eventTopic	UACL
eventType	101
(Source Fields)	Universal Products component denying access based on the UACL entry.
(Reporter Fields)	

UACL_AUTHORIZATION Event Body

Field Name	Notes	Description
clientCertIssuer	1, 2, 5	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2, 5	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2, 5	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientUserId	1, 3, 4, 5	User identifier of the client.
clientWorkId	1, 5	Work identifier of the client.
uacEntryNumber		Entry number of the UACL entry that denied access to the resource.
uacEntryRule		Rule of the UACL entry that denied access to the resource.
uacEntryType		Type of UACL entry that denied access to the resource.
uacResource		UACL resource name to which access was denied.

Notes

1. Client fields identify the client that requested access to the UACL protected resource.
2. Digital certificate information is only provided if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.
5. The field is not provided when the `uac1EntryType` is `ubroker_access`.

2.4.3 RESOURCE_AUTHORIZATION (102)

The event is generated when a Universal Products component was denied access to a general resource protected by a Universal Products component.

Conditions under which the event is generated include:

- A UCMD Manager was restarted and reattached to a UCMD Server in a pending, disconnected or orphaned communication state and the user ID specified by the restarted Manager is different than the user ID specified by the originating UCMD Manager.
- A UCTL STOP command specified a user ID that is not authorized to STOP the requested Server component.

RESOURCE_AUTHORIZATION Event Header

Field Name	Description
eventCategory	Security
eventSeverity	Major
eventTopic	Specific to the resource.
eventType	102
(Source Fields)	Universal Products component denying access to the resource.
(Reporter Fields)	

RESOURCE_AUTHORIZATION Event Body

Field Name	Notes	Description
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientUserId	1, 3, 4	User identifier of the client.
clientWorkId	1	Work identifier of the client.
owner		User identifier that owns the resource.
securityResource		Name of the resource to which access was denied.
userId	3	User identifier that was denied access to the resource.

Notes

1. Client fields identify the client that requested access to the protected resource.
2. Digital certificate information is only provided if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.4 UCMSRV_JOB_START (200)

The event is generated when a job has been started by the Universal Command Server. The job executes the work as requested by a Universal Command Manager. The job may be a command or script.

UCMSRV_JOB_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Job
eventType	200
(Source Fields)	Job that started.
(Reporter Fields)	UCM Server that started the job.

UCMSRV_JOB_START Event Body

Field Name	Notes	Description
certId		Certificate map identifier that the client certificate was mapped to by the Universal Access Control List cert_map entries.
clientCertIssuer	1	Issuer field of the digital certificate provided by the UCMD Manager.
clientCertSerialNo	1	Serial number of the digital certificate provided by the UCMD Manager.
clientCertSubject	1	Subject field of the digital certificate provided by the UCMD Manager.
clientIpAddr		IP address from which the UCMD Manager's socket connection was established.
clientId	2, 3	User identifier of the UCMD Manager.
clientIdCaseSensitive		Indication of whether or not clientId is case sensitive: <ul style="list-style-type: none"> 0 = false (clientId is not case sensitive). 1 = true (clientId is case sensitive).
clientWorkId		Work identifier of the UCMD Manager.
jobCommand		Command line used to start the job. The field is relevant only if the jobFormat value is CMD.

Field Name	Notes	Description
jobFormat		Format of job requested by the UCMD Manager. Valid values are: <ul style="list-style-type: none"> CMD (1) SCRIPT (2)
jobLoginEnvironment		Whether or nor a login environment was established for the job.
jobType		Type of job requested by the UCMD Manager. Valid values are: <ul style="list-style-type: none"> shell Job is executed by a shell environment, such as Windows CMD.EXE, UNIX Bourne or Korn shell, or z/OS USS shell. stc Job is a z/OS started task. cmdref Job is a UCM Server command reference.
jobUserId	2	User identifier with which the job executed.
jobUserIdCaseSensitive		Indication of whether or not jobUserId is case sensitive: <ul style="list-style-type: none"> 0 = false (jobUserId is not case sensitive). 1 = true (jobUserId is case sensitive).
jobWorkId		Work identifier of the job started by UCMD Server.
managerFaultTolerant		Whether or nor the UCMD Server ran with the Manager Fault Tolerant protocol.
networkFaultTolerant		Whether or not the UCMD Server ran with the Network Fault Tolerant protocol.
startDateTime		UTC date and time the job started.

Notes

- Digital certificate information is only provided if the client provided a digital certificate.
- A case-dependent value.
- A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.5 UCMSRV_JOB_STOP (201)

The event is generated when a job has stopped. The job process executes the work requested by the Universal Command Manager.

UCMSRV_JOB_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Job
eventType	201
(Source Fields)	Job that stopped.
(Reporter Fields)	UCM Server that started the job.

UCMSRV_JOB_STOP Event Body

Field Name	Notes	Description
certId		Certificate map identifier that the client certificate was mapped to by the Universal Access Control List <code>cert_map</code> entries.
clientCertIssuer	1	Issuer field of the digital certificate provided by the UCMD Manager.
clientCertSerialNo	1	Serial number of the digital certificate provided by the UCMD Manager.
clientCertSubject	1	Subject field of the digital certificate provided by the UCMD Manager.
clientIpAddr		IP address from which the UCMD Manager's socket connection was established.
clientUserId	2, 3	User identifier of the UCMD Manager.
clientUserIdCaseSensitive		Indication of whether or not <code>clientUserId</code> is case sensitive: <ul style="list-style-type: none"> 0 = false (<code>clientUserId</code> is not case sensitive). 1 = true (<code>clientUserId</code> is case sensitive).
clientWorkId		Work identifier of the UCMD Manager.
elapsedTime		Number of seconds in which the job executed. The value may be zero if the job ran less than a second, resulting in the start and end time not changing at the 1 second resolution.
exitCode		Final exit code of the job as reported by operating system.

Field Name	Notes	Description
exitStatus		Final exit status of the job as reported by the operating system. Valid values are: <ul style="list-style-type: none"> • Normal (1) • Abnormal (2) An abnormal end indicates the job was interrupted by the operating system before it completed processing. Examples of abnormal exits are, for example, UNIX signals, Windows exceptions, and z/OS ABENDs.
jobCommand		Command line used to start the job. The field is relevant only if the jobFormat value is CMD.
jobFormat		Format of job requested by the UCMD Manager. Valid values are: <ul style="list-style-type: none"> • CMD (1) • SCRIPT (2)
jobLoginEnvironment		Whether or not a login environment was established for the job.
jobType		Type of job requested by the UCMD Manager. Valid values are: <ul style="list-style-type: none"> • shell Job is executed by a shell environment, such as Windows CMD.EXE, UNIX Bourne or Korn shell, or z/OS USS shell. • stc Job is a z/OS started task. • cmdref Job is a UCM Server command reference.
jobUserId	2	User identifier with which the job executed.
jobUserIdCaseSensitive		Indication of whether or not jobUserId is case sensitive: <ul style="list-style-type: none"> • 0 = false (jobUserId is not case sensitive). • 1 = true (jobUserId is case sensitive).
jobWorkId		Work identifier of the job started by UCMD Server.
managerFaultTolerant		Whether or not the UCMD Server ran with the Manager Fault Tolerant protocol.
networkFaultTolerant		Whether or not the UCMD Server ran with the Network Fault Tolerant protocol.
startDateTime		UTC date and time the job started.
stopDateTime		UTC date and time the job stopped.

Notes

1. Digital certificate information is only provided if the client provided a digital certificate.
2. A case-dependent value.
3. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.6 UCM_WORK (204)

The event is generated by Universal Command Manager to log the work it has performed.

UCM_WORK Event Header

Field Name	Description
eventCategory	Log
eventSeverity	Informational
eventTopic	Component
eventType	204
(Source Fields)	UCM Manager that performed the work.
(Reporter Fields)	

UCM_WORK Event Body

Field Name	Notes	Description
brokerHost		IP address of the Universal Broker that UCM Manager connected to request the execution of its work.
brokerPort		TCP port number of the Universal Broker that UCM Manager connected to request the execution of its work.
certIssuer	1	Issuer name of the digital certificate with which UCM Manager is executing.
certSerialNo	1	Serial number of the digital certificate with which UCM Manager is executing.
certSubject	1	Subject name of the digital certificate with which UCM Manager is executing.
commandId		Command identifier with which UCM Manager executed.
comment		Comment with which UCM Manager executed.
controlCipher		SSL cipher suite used for the control session between the UCM Manager and Server.
forceComplete		Specification for whether or not the UCM Manager executed with the FORCE_COMPLETE option set to a value of YES.
jobExitCode		Exit code of the job that was executed at the request of UCM Manager.
jobExitStatus		Exit status of the job that was executed at the request of UCM Manager.
jobUserId		User identifier with which the UCM Manager requested the job to execute.
licenseCustomer		Customer name of the UCM Manager license.

Field Name	Notes	Description
licenseExpiration		Expiration date of the UCM Manager license.
managerComponentId		Component identifier of the UCM Manager.
managerFaultTolerant		Specification for whether or not the UCM Manager executed with the <code>MANAGER_FAULT_TOLERANT</code> option set to a value of <code>YES</code> .
messageLevel		Message level with which UCM Manager is executing.
networkFaultTolerant		Specification for whether or not the UCM Manager executed with the <code>NETWORK_FAULT_TOLERANT</code> option set to a value of <code>YES</code> .
restarted		Specification for whether or not the UCM Manager was restarted: <ul style="list-style-type: none"> Value of <code>TRUE</code> indicates it was restarted. Value of <code>FALSE</code> indicates it was not restarted. If the Manager <code>RESTART</code> option was set to <code>AUTO</code> , this field is set to <code>TRUE</code> only if the Manager reestablished a session with a pending, orphaned, or disconnected UCM Server.
serverComponentId		Component identifier of the UCM Server with which the Manager is communicating.
stderrCipher		SSL cipher suite used for the standard error session between the UCM Manager and Server.
stderrCompression		Compression method used for the standard error session between the UCM Manager and Server.
stderrMode		Data translation mode used for the standard error session between the UCM Manager and Server.
stdinCipher		SSL cipher suite used for the standard input session between the UCM Manager and Server.
stdinCompression		Compression method used for the standard input session between the UCM Manager and Server.
stdinMode		Data translation mode used for the standard input session between the UCM Manager and Server.
stdoutCipher		SSL cipher suite used for the standard output session between the UCM Manager and Server.
stdoutCompression		Compression method used for the standard output session between the UCM Manager and Server.
stdoutMode		Data translation mode used for the standard output session between the UCM Manager and Server.
userId	2	User identifier with which UCM Manager is executing.
version		Version of the UCM Manager program.
workId		Work identifier with which UCM Manager is executing.

Notes

- Digital certificate information is provided only if the Manager is configured to use digital certificates.
- A case-dependent value.

2.4.7 UCMSRV_WORK (205)

The event is generated by Universal Command (UCM) Server to log the work it has performed.

UCMSRV_WORK Event Header

Field Name	Description
eventCategory	Log
eventSeverity	Informational
eventTopic	Component
eventType	205
(Source Fields)	UCM Server that performed the work.
(Reporter Fields)	

UCMSRV_WORK Event Body

Field Name	Notes	Description
clientCertIssuer	1	Issuer field of the digital certificate provided by the UCM Manager.
clientCertSerialNo	1	Serial number of the digital certificate provided by the UCM Manager.
clientCertSubject	1	Subject field of the digital certificate provided by the UCM Manager.
clientIpAddr		IP address from which the UCM Manager's socket connection was established.
clientId	2, 3	User identifier of the UCM Manager.
clientIdCaseSensitive		Indication of whether or not clientId is case sensitive: <ul style="list-style-type: none"> 0 = false (clientId is not case sensitive). 1 = true (clientId is case sensitive).
clientWorkId		Work identifier of the UCM Manager.
controlCipher		SSL cipher suite used for the control session between the UCM Manager and Server.
forceComplete		Specification for whether or not the UCM Server performed a force completion request.
jobCommand		Command used to start the job. The field is relevant only if the jobFormat value is CMD.
jobElapsedTime		Number of seconds that the job executed. The value may be zero if the job ran less than a second.

Field Name	Notes	Description
jobExitCode		Exit code of the job as reported by the operating system.
jobExitStatus		Final exit status of the job as reported by the operating system. Valid values are: <ul style="list-style-type: none"> • Normal (1) • Abnormal (2) An abnormal end indicates the job was interrupted by the operating system before it completed processing. Examples of abnormal exits include UNIX signals, Windows exceptions, and z/OS ABENDs.
jobFormat		Format of the job request. Valid values are: <ul style="list-style-type: none"> • CMD (1) • SCRIPT (2)
jobLoginEnvironment		Specification for whether or not a login environment was established in which to execute the job.
jobStartDateTime		UTC time the job started.
jobStopDateime		UTC time the job stopped.
jobType		Type of job requested. Valid values are: <ul style="list-style-type: none"> • shell Job is executed by a shell environment, such as Windows CMD.EXE, UNIX Bourne or Korn shell, or z/OS USS shell. • stc Job is a z/OS started task. • cmdref Job is a UCM Server command reference.
jobUserId		User identifier with which the job executed.
jobWorkId		Work identifier of the job.
managerFaultTolerant		Specification for whether or not the UCM Server is in Manager Fault Tolerant mode.
messageLevel		Message level with which UCM Server is executing.
networkFaultTolerant		Specification for whether or not the UCM Server is in Network Fault Tolerant mode.
restarted		Specification for whether or not the UCM Server was restarted.
serverComponentId		Component identifier of the UCM Server with which the manager is communicating.
stderrCipher		SSL cipher suite used for the standard error session between the UCM Manager and Server.
stderrCompression		Compression method used for the standard error session between the UCM Manager and Server.
stderrMode		Data translation mode used for the standard error session between the UCM Manager and Server.
stdinCipher		SSL cipher suite used for the standard input session between the UCM Manager and Server.
stdinCompression		Compression method used for the standard input session between the UCM Manager and Server.

Field Name	Notes	Description
stdinMode		Data translation mode used for the standard input session between the UCM Manager and Server.
stdoutCipher		SSL cipher suite used for the standard output session between the UCM Manager and Server.
stdoutCompression		Compression method used for the standard output session between the UCM Manager and Server.
stdoutMode		Data translation mode used for the standard output session between the UCM Manager and Server.
version		Version of the UCM Server program.
workId		Work identifier with which UCM Server is executing.

Notes

1. Digital certificate information is provided only if the Manager provided a digital certificate.
2. A case-dependent value.
3. A case-dependent value, but for pre-3.2.0 clients, the value always is considered case sensitive.

2.4.8 UBR_START (300)

The event is generated when the Universal Broker is started.

UBR_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Component
eventType	300
(Source Fields)	Universal Broker that started.
(Reporter Fields)	

UBR_START Event Body

Field Name	Notes	Description
CertIssuer	1	Issuer subject field of the digital certificate with which the Broker is executing.
CertSerialNo	1	Serial number of the digital certificate with which the Broker is executing.
CertSubject	1	Subject field of the digital certificate with which the Broker is executing.
servicePort		TCP port number on which the Broker is servicing network requests.
startDateTime		UTC date and time the Broker has started executing.
userId	2	User identifier with which the Broker is executing.

Notes

- Digital certificate information is only provided if the Broker is configured to use digital certificates.
- A case-dependent value.

2.4.9 UBR_STOP (301)

The event is generated when the Universal Broker is stops.

There are cases when a UBR_STOP event will not be generated. When the Broker is stopped with a method that does not permit the Broker program to continue executing, the event is not generated. Some methods that stop a Broker in this manner are:

- Windows termination with the Windows Task Manager.
- UNIX SIGKILL signal.
- z/OS CANCEL command or JES \$C command.
- OS/400 immediate end.

UBR_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Component
eventType	301
(Source Fields)	Universal Broker that stopped.
(Reporter Fields)	

UBR_STOP Event Body

Field Name	Notes	Description
elapsedTime		Amount of time (in seconds) that the Broker executed.
exitCode		Exit code with which the Broker stopped executing.
stopDateTime		UTC date and time the Broker stopped.

2.4.10 UBR_SRV_COMP_START (302)

The event is generated when the Broker starts a server component. A Broker server component is typically started at the request of a Universal Products Manager; however, there are times when the Broker starts a server component on its own behalf.

UBR_SRV_COMP_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Component
eventType	302
(Source Fields)	Server component that started.
(Reporter Fields)	Universal Broker that started the component.

UBR_SRV_COMP_START Event Body

Field Name	Notes	Description
applProperties		Application-specific properties used for event correlation by UEC.
certId		Certificate map identifier that the client certificate was mapped to by the Universal Access Control List cert_map entries.
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientPort	1	TCP port number from which the client's socket connection was established.
clientUserId	1, 3, 4	User identifier of the client.
clientWorkId	1	Work identifier of the client.
commandId		Command identifier, specified by the Manager, that is used to identify the unit of work.
comment		User-provided comment associated with the unit of work.
componentId		Component identifier the Broker assigns to the component.
compRestart		Whether or not the component is being restarted.

Field Name	Notes	Description
managerFaultTolerant		Specifies whether or not the server component is manager fault tolerant.
startDateTime		UTC date and time the component started.

Notes

1. The client information identifies the entity that requested the Broker to start the component. The Broker may have started the component on its own behalf, in which case, the client information identifies the Broker.
2. Client digital certificate information is provided only if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.11 UBR_SRV_COMP_STOP (303)

The event is generated when the Broker detects that a server component has stopped.

UBR_SRV_COMP_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	<ul style="list-style-type: none"> Informational (if the component ended successfully) Warning (if the component ended unsuccessfully or with a PENDING communication state)
eventTopic	Component
eventType	303
(Source Fields)	Server component that stopped.
(Reporter Fields)	Universal Broker that started the component.

UBR_SRV_COMP_STOP Event Body

Field Name	Notes	Description
applProperties		Application-specific properties used for event correlation by UEC.
certId		Certificate map identifier that the client certificate was mapped to by the Universal Access Control List <code>cert_map</code> entries.
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientPort	1	TCP port number from which the client's socket connection was established.
clientUserId	1, 3, 4	User identifier of the client.
clientWorkId	1	Work identifier of the client.
commandId		Command identifier, specified by the manager, that is used to identify the unit of work.
comment		User-provided comment associated with the unit of work.
componentId		Component identifier the Broker assigns to the component.
elapsedTime		Number of seconds that the server component executed.

Field Name	Notes	Description
exitCode		Exit code with which the component ended as reported by the operating system.
exitStatus		Final exit status of the component as reported by the operating system. Valid values are: <ul style="list-style-type: none"> • Normal (1) • Abnormal (2) An abnormal end indicates the component was interrupted by the operating system before it completed processing. Examples of abnormal exits include UNIX signals, Windows exceptions, and z/OS ABENDs.
managerFaultTolerant		Specifies whether or not the server component is manager fault tolerant.
pending		Specifies whether or not the server component ended in a PENDING communication state. A PENDING state indicates that the server was running with the Manager Fault Tolerant feature and ended while no manager connection was established with the server.
startDateTime		UTC date and time that the server component started.
stopDateTime		UTC date and time that the component stopped.

Notes

1. The client information identifies the entity that requested the Broker to start the component. The Broker may have started the component on its own behalf, in which case, the client information identifies the Broker.
2. Client digital certificate information is provided only if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.12 UBR_COMP_COMM_STATE (304)

The event is generated when the Broker detects if a component's communications state has changed.

The conditions in which the event is generated include:

- State changing to DISCONNECTED, ORPHANED or PENDING.
- State changing from DISCONNECTED, ORPHANED or PENDING to ESTABLISHED.

UBR_COMP_COMM_STATE Event Header

Field Name	Description
eventCategory	Exception
eventSeverity	Warning
eventTopic	Component
eventType	304
(Source Fields)	Component whose communicate state changed.
(Reporter Fields)	Universal Broker that started the component.

UBR_COMP_COMM_STATE Event Body

Field Name	Notes	Description
changeDateTime		UTC date and time the component state changed.
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientPort	1	TCP port number from which the client's socket connection was established.
clientId	1, 3, 4	User identifier of the client.
clientWorkId	1	Work identifier of the client.
componentId		Component identifier the Broker assigns to the component.
newCommState	5	New communication state of the component.
prevCommState	5	Previous communication state of the component.

Notes

1. The client information identifies the entity that requested the Broker to start the component. The Broker may have started the component on its own behalf, in which case, the client information identifies the Broker.
2. Client digital certificate information is provided only if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.
5. The component communication state (new or previous) is represented by one of numeric values identified in [Table 2.8](#), below:

Value	Name	Description
1	STARTING	Started by Broker.
2	ESTABLISHED	Started and connected.
3	DISCONNECTED	No working connection to Manager.
4	ORPHANED	Manager has terminated.
5	RECONNECTING	Reconnect received and is in progress.
6	RESTARTING	Restart received and is in progress.
7	RESTART_ACCEPTED	Restart received by Broker.
8	PENDING	Manager has not received results.
9	COMPLETED	Manager has received results.

Table 2.8 Component Communication States

2.4.13 UBR_MGR_COMP_START (305)

The event is generated when a manager component registers with the Broker.

UBR_MGR_COMP_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Component
eventType	305
(Source Fields)	Manager component that started.
(Reporter Fields)	Universal Broker with which the manager component registered.

UBR_MGR_COMP_START Event Body

Field Name	Notes	Description
clientCertIssuer	1	Issuer field of the digital certificate identifying manager.
clientCertSerialNo	1	Serial number of the digital certificate identifying manager.
clientCertSubject	1	Subject field of the digital certificate identifying the manager.
clientUserId	2	User identifier of the manager.
clientUserIdCaseSensitive		Indication of whether or not clientUserId is case sensitive: <ul style="list-style-type: none"> 0 = false (clientUserId is not case sensitive). 1 = true (clientUserId is case sensitive).
clientWorkId		Work identifier of the manager.
componentId		Component identifier the Broker assigns to the component.
startDateTime		UTC date and time the component started.

Notes

1. Client digital certificate information is provided only if the client provided a digital certificate.
2. A case-dependent value.

2.4.14 UBR_MGR_COMP_STOP (306)

The event is generated when the Broker detects a manager component has stopped.

UBR_MGR_COMP_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	<ul style="list-style-type: none"> Informational (if the component ended successfully) Warning (if the component ended unsuccessfully)
eventTopic	Component
eventType	306
(Source Fields)	Manager component that stopped.
(Reporter Fields)	Universal Broker with which the manager component registered.

UBR_MGR_COMP_STOP Event Body

Field Name	Notes	Description
clientCertIssuer	1	Issuer field of the digital certificate identifying the manager.
clientCertSerialNo	1	Serial number of the digital certificate identifying the manager.
clientCertSubject	1	Subject field of the digital certificate identifying the manager.
clientId	2	User identifier of the manager.
clientIdCaseSensitive		Indication of whether or not clientId is case sensitive: <ul style="list-style-type: none"> 0 = false (clientId is not case sensitive). 1 = true (clientId is case sensitive).
clientWorkId		Work identifier of the manager.
componentId		Component identifier the Broker assigns to the component.
elapsedTime		Number of seconds in which the manager component ended.
exitCode		Exit code with which the component ended as reported by the operating system.
exitStatus		<p>Final exit status of the component as reported by the operating system.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> Normal (1) Abnormal (2) <p>An abnormal end indicates the component was interrupted by the operating system before it completed processing. Examples of abnormal exits include UNIX signals, Windows exceptions, and z/OS ABENDs.</p>
startDateTime		UTC date and time that the component started.
stopDateTime		UTC date and time that the component stopped.

Notes

1. Client digital certificate information is provided only if the client provided a digital certificate.
2. A case dependent value.

2.4.15 UBR_COMP_STOP_REQUEST (307)

The event is generated when a Universal Broker receives and processes a request to stop a component. The stop request is sent to the targeted component. The event does not indicate the components response to the stop request.

The conditions under which the event is generated are:

- Universal Broker requesting a component stop as the result of a manager termination matching a server stop condition value.
- Universal Control Server requesting a component stop on behalf of a Universal Control Manager.
- Universal Enterprise Controller requesting a component stop on behalf of a Universal Enterprise Controller client, such as the Universal Activity Monitor.

UBR_COMP_STOP_REQUEST Event Header

Field Name	Description
eventCategory	Log
eventSeverity	Informational
eventTopic	Command
eventType	307
(Source Fields)	Target component of the STOP command.
(Reporter Fields)	Universal Broker.

UBR_COMP_STOP_REQUEST Event Body

Field Name	Notes	Description
clientCertIssuer	1	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1	Serial number of the digital certificate provided by the client.
clientCertSubject	1	Subject field of the digital certificate provided by the client.
clientIpAddr		IP address from which the client's socket connection was established.
clientUserId	2, 3	User identifier of the client.
clientWorkId		Work identifier of the client.
componentId		Component identifier of the component whose is the target of the STOP request.
startDateTime		UTC time that the STOP request was sent to the target component.
userId	2	Local user identifier with which the STOP request executes.

Notes

1. Client digital certificate information is provided only if the client provided a digital certificate.
2. A case-dependent value.
3. A case dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.16 UBR_MGR_RESTART (308)

The event is generated by Universal Broker when it has been notified by a Universal Command Manager that it has successfully restarted with a Universal Command Server.

UBR_MGR_RESTART Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Component
eventType	308
(Reporter Fields)	Universal Broker
(Source Fields)	Universal Command Server

UBR_MGR_RESTART Event Body

Field Name	Notes	Description
certId	1	Certificate map identifier that the client certificate was mapped to by the Universal Access Control List cert_map entries.
clientCertIssuer	1	Issuer field of the digital certificate identifying the manager.
clientCertSerialNo	1	Serial number of the digital certificate identifying the manager.
clientCertSubject	1	Subject field of the digital certificate identifying the manager.
clientIpAddr		IP address from which the client is connecting.
clientPort		TCP port number from which the client is connecting.
clientId	2	User identifier of the manager.
clientIdCaseSensitive		Indication of whether or not clientId is case sensitive: <ul style="list-style-type: none"> • 0 = false (clientId is not case sensitive). • 1 = true (clientId is case sensitive).
clientWorkId		Work identifier of the manager.
componentId		Component identifier of the UCM server.

Notes

1. Digital certificate information is provided only if the Manager is configured to use digital certificates.
2. A case dependent value.

2.4.17 UDM_OPEN_COMMAND_START (400)

The event signals the beginning of an attempt to open a UDM session.

To identify the UDM session, the manager creates a SESSION_UUID and places it in the body of the message.

UDM_OPEN_COMMAND_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Command
eventType	400
(Source Fields)	UDM Manager that issued the open command.
(Reporter Fields)	

UDM_OPEN_COMMAND_START Event Body

Field Name	Notes	Description
networkFaultTolerant		Specifies whether or not the session was created using network fault tolerance.
primSrvHost	1	Host or IP address of the primary server.
primSrvUserId	1	User ID of the primary server.
secSrvHost		Host or IP address of the secondary server.
secSrvUserId		User ID of the secondary server.
sessionUuid		Unique identifier for a UDM session.
startDateTime		UTC date and time the command was issued.

Notes

1. If the session is a two-party session, there is no primary server information as the manager plays the role of the primary server.

2.4.18 UDM_COPY_COMMAND_START (401)

The event signals the beginning of a UDM copy operation.

UDM_COPY_COMMAND_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Command
eventType	401
(Source Fields)	UDM Manager that issued the copy command.
(Reporter Fields)	

UDM_COPY_COMMAND_START Event Body

Field Name	Notes	Description
dstComponentUuid		UUID of the UDM component acting as the destination in the transfer.
dstFileMask	1	Destination file mask in the copy operation.
sessionUuid		Unique identifier for a UDM session.
srcComponentUuid		UUID of the UDM component acting as the source in the transfer.
srcFileMask		Source file mask in the copy operation.
startDateTime		UTC date and time the command was issued.
transactionId		Unique identifier for the command within a UDM manager instance.

Notes

1. There may not be a destination file mask.

2.4.19 UDM_COPYDIR_COMMAND_START (402)

The event signals the beginning of a UDM directory copy operation.

UDM_COPYDIR_COMMAND_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Command
eventType	402
(Source Fields)	UDM Manager that issued the <code>copydir</code> command.
(Reporter Fields)	

UDM_COPYDIR_COMMAND_START Event Body

Field Name	Notes	Description
dstComponentUuid		UUID of the UDM component acting as the destination in the transfer.
dstFileMask	1	Destination file mask in the copy operation.
sessionUuid		Unique identifier for a UDM session.
srcComponentUuid		UUID of the UDM component acting as the source in the transfer.
srcFileMask		Source file mask in the copy operation.
startDateTime		UTC date and time the command was issued.
transactionId		Unique identifier for the command within a UDM manager instance.

Notes

1. There may not be a destination file mask.

2.4.20 UDM_DELETE_COMMAND_START (403)

The event signals the beginning of a UDM delete operation.

UDM_DELETE_COMMAND_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Command
eventType	403
(Source Fields)	UDM Manager that issued the delete command.
(Reporter Fields)	

UDM_DELETE_COMMAND_START Event Body

Field Name	Notes	Description
componentUuid		UUID of the UDM component on which the deletion operation will take place.
fileMask		File mask for the delete operation.
sessionUuid		Unique identifier for a UDM session.
startDateTime		UTC date and time the command was issued.
transactionId		Unique identifier for the command within a UDM manager instance.

2.4.21 UDM_RENAME_COMMAND_START (404)

The event signals the beginning of a UDM rename operation.

UDM_RENAME_COMMAND_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Command
eventType	404
(Source Fields)	UDM Manager that issued the rename command.
(Reporter Fields)	

UDM_RENAME_COMMAND_START Event Body

Field Name	Notes	Description
componentUuid		UUID of the UDM component on which the rename operation will take place.
newFileName		New file name used in the rename operation.
oldFileName		Old file name used in the rename operation.
sessionUuid		Unique identifier for a UDM session.
startDateTime		UTC date and time the command was issued.
transactionId		Unique identifier for the command within a UDM manager instance.

2.4.22 UDM_OPEN_COMMAND_STOP (405)

The event indicates a UDM session created with the OPEN command has been closed.

UDM_OPEN_COMMAND_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Command
eventType	405
(Source Fields)	UDM Manager that issued the open command.
(Reporter Fields)	

UDM_OPEN_COMMAND_STOP Event Body

Field Name	Notes	Description
commandRc		Return value for the command once the session has terminated.
networkFaultTolerant		Whether or not the session was created using network fault tolerance.
primSrvHost	1	Host name or IP address of the primary server.
primSrvUserId	1	User ID of the primary server.
secSrvHost		Host name or IP address of the secondary server.
secSrvUserid		User ID of the secondary server.
sessionUuid		Unique identifier for a UDM session.
stopDateTime		UTC date and time the command finished executing.

Notes

1. If the session is a two-party session, there is no primary server information, as the manager plays the role of the primary server.

2.4.23 UDM_COPY_COMMAND_STOP (406)

The event indicates a UDM copy operation has ended.

UDM_COPY_COMMAND_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Command
eventType	406
(Source Fields)	UDM Manager that issued the copy command.
(Reporter Fields)	

UDM_COPY_COMMAND_STOP Event Body

Field Name	Notes	Description
commandRc		Return value for the command.
dstComponentUuid		UUID of the UDM component acting as the destination in the transfer.
dstFileMask	1	Destination file mask in the copy operation.
numFilesAttempted		Number of files in which a transfer was attempted.
numFilesTransferred		Number of files successfully transferred.
sessionUuid		Unique identifier for a UDM session.
srcComponentUuid		UUID of the UDM component acting as the source of the transfer.
srcFileMask		Source file mask in the copy operation.
stopDateTime		UTC date and time the command finished executing.
transactionId		Unique identifier for the command within a UDM manager instance.

Notes

1. There may not be a destination file mask.

2.4.24 UDM_COPYDIR_COMMAND_STOP (407)

The event indicates a UDM copy directory operation has ended.

UDM_COPYDIR_COMMAND_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Command
eventType	407
(Source Fields)	UDM Manager that issued the <code>copydir</code> command.
(Reporter Fields)	

UDM_COPYDIR_COMMAND_STOP Event Body

Field Name	Notes	Description
commandRc		Return value for the command.
dstComponentUuid		UUID of the UDM component acting as the destination in the transfer.
dstFileMask	1	Destination file mask in the copy operation.
numFilesAttempted		Number of files in which a transfer was attempted.
numFilesTransferred		Number of files successfully transferred.
sessionUuid		Unique identifier for a UDM session.
srcComponentUuid		UUID of the UDM component acting as the source of the transfer.
srcFileMask		Source file mask in the copy operation.
stopDateTime		UTC date and time the command finished executing.
transactionId		Unique identifier for the command within a UDM Manager instance.

Notes

1. There may not be a destination file mask.

2.4.25 UDM_DELETE_COMMAND_STOP (408)

The event indicates a UDM delete operation has ended.

UDM_DELETE_COMMAND_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Command
eventType	408
(Source Fields)	UDM Manager that issued the delete command.
(Reporter Fields)	

UDM_DELETE_COMMAND_STOP Event Body

Field Name	Notes	Description
commandRc		Return value for the command.
componentUuid		UUID of the component on which the operation will take place.
fileMask		File mask used in the delete operation.
sessionUuid		Unique identifier for a UDM session.
stopDateTime		UTC date and time the command finished executing.
transactionId		Unique identifier for the command within a UDM Manager instance.

2.4.26 UDM_RENAME_COMMAND_STOP (409)

The event indicates a UDM delete operation has ended.

UDM_RENAME_COMMAND_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Command
eventType	409
(Source Fields)	UDM Manager that issued the rename command.
(Reporter Fields)	

UDM_RENAME_COMMAND_STOP Event Body

Field Name	Notes	Description
commandRc		Return value for the command.
componentUuid		UUID of the component on which the operation will take place.
newFileName		New file name used in the rename operation.
oldFileName		Old file name used in the rename operation.
sessionUuid		Unique identifier for a UDM session.
stopDateTime		UTC date and time the command finished executing.
transactionId		Unique identifier for the command within a UDM Manager instance.

2.4.27 UDM_FILE_DELETE (450)

A message is posted when a file is deleted.

UDM_FILE_DELETE Event Header

Field Name	Description
eventCategory	Status
eventSeverity	Informational
eventTopic	File
eventType	450
(Source Fields)	File object that was deleted.
(Reporter Fields)	UDM Manager or Server that deleted the file.

UDM_FILE_DELETE Event Body

Field Name	Notes	Description
dateTime		UTC date and time of the operation.
filePath		Absolute path of the file being deleted.
fileResult		One of two values: success or failure.
fileType		Type of file being deleted, either a file or a directory.
sessionUuid		Unique identifier for a UDM session.
transactionId		Transaction identifier associated with the command invoking the copy.

2.4.28 UDM_FILE_RENAME (451)

A message is posted when a file is renamed.

UDM_FILE_RENAME Event Header

Field Name	Description
eventCategory	Status
eventSeverity	Informational
eventTopic	File
eventType	451
(Source Fields)	File object that was renamed.
(Reporter Fields)	UDM Manager or Server that renamed the file.

UDM_FILE_RENAME Event Body

Field Name	Notes	Description
dateTime		UTC date and time of the operation.
fileResult		One of two values: success or failure.
fileType		Type of file being transferred, either a file or a directory.
newFilePath		Absolute path of the new name for the file.
oldFilePath		Absolute path of the old name of the file.
sessionUuid		Unique identifier for a UDM session.
transactionId		Transaction identifier associated with the command invoking the copy.

2.4.29 UDM_FILE_XFER_START (452)

A message is posted to indicate the beginning of a transfer operation for an individual file or directory.

UDM_FILE_XFER_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	File
eventType	452
(Source Fields)	File object that is being transferred.
(Reporter Fields)	UDM Manager or Server that started the transfer.

UDM_FILE_XFER_START Event Body

Field Name	Notes	Description
fileMode		READ, WRITE, or APPEND
filePath		Absolute path of the file.
fileType		Type of file being transferred, either a file or a directory.
hostName		Host name of the machine that issued the event.
sessionUuid		Unique identifier for a UDM session.
startDateTime		UTC date and time the operation was begun.
transactionId		Transaction identifier associated with the command invoking the copy.
transferMode		Mode (text or binary) of the transfer.
transferUuid		Unique identifier for the transfer operation.

2.4.30 UDM_FILE_XFER_STOP (453)

A message is posted to indicate the end of a transfer operation for an individual file or directory.

UDM_FILE_XFER_STOP Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	File
eventType	453
(Source Fields)	File object that is being transferred.
(Reporter Fields)	UDM Manager or Server that started the transfer.

UDM_FILE_XFER_STOP Event Body

Field Name	Notes	Description
byteCount		Number of bytes processed.
fileMode		READ, WRITE, or APPEND
filePath		Absolute path of the file.
fileResult		One of two values: success or failure.
fileType		Type of file being transferred, either a file or a directory.
hostName		Host name of the machine that issued the event.
sessionUuid		Unique identifier for a UDM session.
stopDateTime		UTC date and time the operation ended.
transactionId		Transaction identifier associated with the command invoking the copy.
transferMode		Mode (text or binary) of the transfer.
transferUuid		Unique identifier for the transfer operation.

2.4.31 UDM_EXEC_OPERATION_START (470)

A message is posted to indicate the beginning of an EXEC operation.

Since an EXEC operation involves UDM invoking the UCMD manager, the UCMD Manager's parent UUID will be that of the UDM instance that invoked it.

UDM_EXEC_OPERATION_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Component
eventType	470
(Source Fields)	UCM Manager started by UDM Manager.
(Reporter Fields)	UDM Manager that started the UCM Manager.

UDM_EXEC_OPERATION_START Event Body

Field Name	Notes	Description
sessionUuid		Identifier of the session if the exec command is issued in the context of a session.
startDateTime		UTC date and time the command was issued.
transactionId		Unique identifier for the command within a UDM Manager instance.

2.4.32 UDM_EXEC_OPERATION_END (471)

A message is posted to indicate the end of an EXEC operation.

UDM_EXEC_OPERATION_END Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Component
eventType	471
(Source Fields)	UCM Manager started by UDM Manager.
(Reporter Fields)	UDM Manager that started the UCM Manager.

UDM_EXEC_OPERATION_END Event Body

Field Name	Notes	Description
commandRc		Return value for the command.
execRc		Return value of the executed job.
sessionUuid		Identifier of the session if the exec command is issued in the context of a session.
stopDateTime		UTC date and time the command finished executing.
transactionId		Identifier for the command that is unique within a UDM Manager instance.

2.4.33 UDM_EXECSAP_OPERATION_START (472)

A message is posted to indicate the beginning of an EXECSAP operation. Since an EXEC operation involves UDM invoking the USAP Manager, the USAP Manager's parent UUID be will that of the UDM instance that invoked it.

UDM_EXECSAP_OPERATION_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Component
eventType	472
(Source Fields)	USAP component started by UDM Manager.
(Reporter Fields)	UDM Manager that started the USAP component.

UDM_EXECSAP_OPERATION_START Event Body

Field Name	Notes	Description
startDateTime		UTC date and time the command was issued.
transactionId		Unique identifier for the command within a UDM Manager instance.

Notes

The EXECSAP command can be issued outside of the context of a session, so there may not be a group UUID.

2.4.34 UDM_EXECSAP_OPERATION_END (473)

A message is posted to indicate the end of an EXECSAP operation.

UDM_EXECSAP_OPERATION_END Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	Informational
eventTopic	Job
eventType	473
(Source Fields)	USAP component started by UDM Manager.
(Reporter Fields)	UDM Manager that started the USAP component.

UDM_EXECSAP_OPERATION_END Event Body

Field Name	Notes	Description
commandRc		Return value for the command.
execRc		Return value of the executed job.
stopDateTime		UTC date and time the command finished executing.
transactionId		Identifier for the command that is unique within a UDM Manager instance.

2.4.35 UCTSRV_CMD_START (800)

The event is generated when a Universal Control Server initiates a request for a control command (for example: STOP, START, or REFRESH) to be executed by the local Universal Broker on behalf of a Universal Control Manager client.

UCTSRV_CMD_START Event Header

Field Name	Description
eventCategory	Start
eventSeverity	Informational
eventTopic	Command
eventType	800
(Source Fields)	UCT Server that made the START request to the Universal Broker.
(Reporter Fields)	

UCTSRV_CMD_START Event Body

Field Name	Notes	Description
clientWorkId	1	Work identifier of the client.
clientId	1, 3, 4	User identifier of the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
command		Universal Control command being executed on behalf of Universal Control Manager client.
commandId		Command identifier provided by the Universal Control Manager client.
commandUserId	3	User identifier with which the control command executes. The user identifier is provided by the Universal Control Manager client.
startDateTime		UTC time that the control command was sent to the local Universal Broker to be executed.

Notes

1. The client information identifies the Universal Control Manager that started the Universal Control Server.
2. Client digital certificate information is provided only if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.36 UCTSRV_CMD_STOP (801)

The event is generated when a control command initiated by a Universal Control Server has stopped. The control command (for example: STOP, START, or REFRESH) was sent to the local Universal Broker for execution on behalf of a Universal Control Manager client.

UCTSRV_CMD_STOP Event Header

Field Name	Description
eventCategory	Stop
eventSeverity	<ul style="list-style-type: none"> Informational (if the STOP request was accepted by the Universal Broker) Warning (if the Universal Broker reported an error)
eventTopic	Command
eventType	801
(Source Fields)	UCT Server that made the STOP request to the Universal Broker.
(Reporter Fields)	

UCTSRV_CMD_STOP Event Body

Field Name	Notes	Description
clientCertIssuer	1, 2	Issuer field of the digital certificate provided by the client.
clientCertSerialNo	1, 2	Serial number of the digital certificate provided by the client.
clientCertSubject	1, 2	Subject field of the digital certificate provided by the client.
clientIpAddr	1	IP address from which the client's socket connection was established.
clientUserId	1, 3, 4	User identifier of the client.
clientWorkId	1	Work identifier of the client.
command		Universal Control command being executed on behalf of Universal Control Manager client.
commandId		Command identifier provided by the Universal Control Manager client.
commandUserId	3	User identifier with which the control command executes. The user identifier is provided by the Universal Control Manager client.
elapsedTime		Number of seconds for the control command to be accepted by the Universal Broker. The time is calculated by stopDateTime - startDateTime .

Field Name	Notes	Description
exitCode		Universal Products error code returned by the Universal Broker in response to the control command request.
startDateTime		UTC time that the control command was sent to the local Universal Broker to be executed.
stopDateTime		UTC time that the Universal Broker indicated the control command was accepted. Note that the time does not necessarily indicate that the Universal Broker has completed processing the control command.

Notes

1. The client information identifies the Universal Control Manager that started the Universal Control Server.
2. Client digital certificate information is provided only if the client provided a digital certificate.
3. A case-dependent value.
4. A case-dependent value, but for pre-3.2 clients, the value is always considered case sensitive.

2.4.37 UCT_CMD_REQUEST (802)

The event is generated when a Universal Control Manager requests the execution of a control command by a Universal Control Server.

UCT_CMD_REQUEST Event Header

Field Name	Description
eventCategory	Log
eventSeverity	Informational
eventTopic	Component
eventType	802
(Source Fields)	UCT Manager that made the request.
(Reporter Fields)	

UCT_CMD_REQUEST Event Body

Field Name	Notes	Description
certIssuer	1	Issuer field of the digital certificate being used by the component.
certSerialNo	1	Serial number of the digital certificate being used by the component.
certSubject	1	Subject field of the digital certificate being used by the component.
command		Universal Control command requested for execution.
commandId		Command identifier associated with a component START command.
commandUserId	2	User identifier with which the control command executes.
componentUuid		UUID of the Universal Control Manager component reporting the event.
exitCode		Universal Products error code returned by the Universal Broker in response to the command request.

Notes

1. Digital certificate information is provided only if the component is using a digital certificate to identify itself.
2. A case-dependent value.

Appendix A

Universal Products Error Codes

Universal Product uses error codes to indicate error conditions that have occurred during processing. The codes are used to communicate the result of a requested command or service between Universal Product components.

Some event definitions include Universal Products error codes as part their event definition.

[Table A.1](#), below, provides a description of the error code values.

Error Code	Description
2	General application error.
3	Resource has exceeded its maximum size.
4	Requested resource was not found.
5	Bad or invalid value was provided.
6	Syntax error.
7	Resource failed to meet minimum size requirements.
8	Protocol error.
9	Negative acknowledgement.
10	Timed out.
11	Incompatible version.
12	Resource not available.
13	Maximum number of components.
14	Process is shutting down.
15	Prematurely closed.
16	Duplicate.
17	Authorization error.
18	Access denied.
19	Invalid operation attempted.

Error Code	Description
20	Empty resource.
21	Aborted operation.
22	Receive operation pending.
23	Send operation pending.
24	Disk space exhausted.
25	License violation.
26	Connect operation pending.
500	Operating system API or services error.
501	Memory error.
502	File open error.
503	File create error.
504	I/O error.
505	Network communication error.
506	SSL protocol error.
507	Database fatal error.
508	AS/400 exception.

Table A.1 Universal Products Error Codes

Appendix B

Customer Support

Stonebranch, Inc. provides customer support, via telephone and e-mail, for Universal Broker and all Universal Products.

TELEPHONE

Customer support via telephone is available 24 hours per day, 7 days per week.

North America

(+1) 678 366-7887, extension 6

(+1) 877 366-7887, extension 6 [toll-free]

Europe

+49 (0) 700 5566 7887

E-MAIL

All Locations

support@stonebranch.com

Customer support contact via e-mail also can be made via the Stonebranch website:

www.stonebranch.com



**950 North Point Parkway, Suite 200
Alpharetta, Georgia 30005
U.S.A.**

