Software and Application Control Management Object

Candidate Version 1.0 – 19 Jul 2011

Open Mobile Alliance

OMA-TS-SACMO-V1_0-20110719-C

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Use of this document is subject to all of the terms and conditions of the Use Agreement located at http://www.openmobilealliance.org/UseAgreement.html.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the "OMA IPR Declarations" list at http://www.openmobilealliance.org/ipr.html. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE "OMA IPR DECLARATIONS" LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. Under the terms set forth above.

Contents

Figures

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Tables

No table of figures entries found.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

1. Scope

This document defines the technical specification for Software and Application Control Management Object, which is based on OMA DM v1.2 specifications and makes use of the functionality's provided by OMA DM v1.2 specifications to define special capabilities to manage software and applications in the client device.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

2. References

2.1 Normative References

[DMPRO] "OMA Device Management Protocol, Version 1.2". Open Mobile Alliance™. OMA-TS-DM-Protocol-V1_2,
URL:http://www.openmobilealliance.org/

[DMStdObj] "OMA Device Management Standardized Objects, Version 1.2". Open Mobile Alliance™. OMA-TS-DM-StdObj-V1_2,
URL:http://www.openmobilealliance.org/

[DMTND] "OMA Device Management Tree and Description, Version 1.2". Open Mobile Alliance™. OMA-TS-DM-TND-V1_2,
URL:http://www.openmobilealliance.org/

[META] "SyncML Meta Information Specification, version 1.2. Open Mobile Alliance™.
OMA-TS-SyncML_MetaInfo-V1_2.
URL: http://www.openmobilealliance.org/

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997,
URL:http://www.ietf.org/rfc/rfc2119.txt

[RFC4234] "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. October 2005,
URL:http://www.ietf.org/rfc/rfc4234.txt

[SACMO-RD] SACMO Requirements, Open Mobile Alliance™, OMA-RD-SACMO- V1_0,
URL:http://www.openmobilealliance.org/

[SCRRULES] "SCR Rules and Procedures", Open Mobile Alliance™, OMA-ORG-SCR_Rules_and_Procedures,
URL:http://www.openmobilealliance.org/

2.2 Informative References

[OMADICT] "Dictionary for OMA Specifications", Version 2.6, Open Mobile Alliance™,
OMA-ORG-Dictionary-V2_6,
URL:http://www.openmobilealliance.org/

[SSL 3.0] "The SSL 3.0 Protocol", A. Frier, P. Karlton, and P. Kocher, Nov 1996

[TLS 1.0] "The TLS Protocol Version 1.0", T. Dierks, C. Allen, January 1999.
URL:http://www.ietf.org/rfc/rfc2246.txt

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

Process	A Process is a basic unit for a specific operation execution.
Step	A Step is a basic unit of the Workflow which consists of a Process and information for next Step(s).
Transaction	A Transaction is an instance of a Workflow execution. SACMO Client executes the Workflow which in turn creates a Transaction.
Workflow	A Workflow is a sequence of Step(s) which is conditionally executed.

3.3 Abbreviations

DM	Device Management
DMS	Device Management Server
OMA	Open Mobile Alliance
SACMO	Software and Application Control Management Object

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

4. Introduction

Software and Application Control Management aims to enable remote operations for software and application control in the Device. Software and Application Control Management specifications will provide the capability to process management workflows that enable on-device management of software and applications utilising existing management objects or applications that could be a native executable application or a script either in the Device or the remote server.

A management tree object [DMTND] defined for Software and Application Control Management (SACMO) will be used for setting up parameters and operational functionality necessary for managing a workflow object.

The objective of this document is to provide the technical specification for managing Software and Applications.

4.1 Version 1.0

The SACMO 1.0 enabler release is expected to meet all the requirements defined in [SACMO-RD] and no additional phases are planned at this stage.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

5. Software and Application Control Management Object Framework

5.1 Components of SACMO

5.1.1 Process

A Process is a basic unit for a specific operation execution. A Process consists of an URI path that indicates the node of target MO or an application to execute. The application could be a native executable application or a script either in the Device or the remote file server. The Process is indicated by a unique ProcessID and it can be reused in Workflows.

5.1.2 Step

A Step is a basic unit of a Workflow which consists of a Process and information for the next Step(s). A Step MUST have a ProcessID to indicate the Process to execute. If a Step is followed by another Step, a NextStep subtree is created. The NextStep subtree MAY contain multiple next Steps. Each next Step has a NextStepID to indicate the following Step and optionally a condition. The SACMO Client checks the condition, if the condition is passed, and then the next step will be executed.

5.1.3 Workflow

A Workflow is a sequence of Step(s) which is conditionally executed. The SACMO Server downloads a Workflow with a unique WorkflowID to the SACMO Client. InitStepID node indicates the first Step in a Workflow. Figure 1 shows the relationship of Process, Step and Workflow.

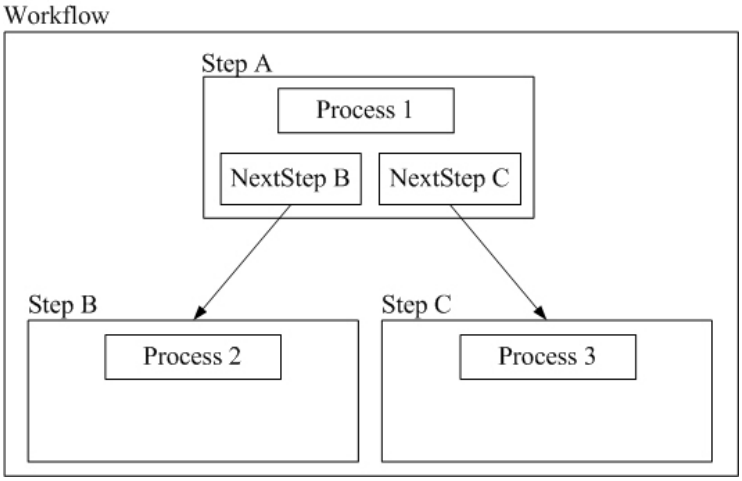


Figure 1: Relationship Diagram of SACMO Components

5.1.4 Transaction

A Transaction is an instance of a Workflow execution. SACMO Client executes the Workflow which in turn creates a Transaction SACMO Server delivers a Workflow to the SACMO Client. . SACMO Server MAY retrieve result of the Transaction execution from ./Transaction/Status.

For example in Figure 1, the transactions generated by this Workflow could be “Process1->Process2” or “Process1->Process3”.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

5.2 Transformation of Workflow

The section 7 defines a SACMO tree which allows SACMO Servers to transform a practical workflow to a SACMO tree. To map a practical workflow to a SACMO tree, each process and condition maps to a Process in the Process sub-tree and corresponding Step in the Step sub-tree. Each flow also maps to one sub-tree under ./<x>/Workflow/<x>/Step/<x>/NextStep/. An InitStepID MUST be specified in the Workflow sub-tree to identify the first Step to run the practical workflow.

Before a Transaction is executed to run a Workflow, the SACMO Client MUST check the existence of the WorkflowID related to this Transaction and ProcessID related to Step of the Workflow.

For example, in Figure 2, a SACMO Server decides whether to start an update process according to the percentage of charge remaining in the battery. In this figure, there are three processes and one condition in the practical workflow, thus four Process sub-tree(s) are created in SACMO tree. The SACMO Server also needs to create one Workflow sub-tree and four Step sub-tree(s) for this practical workflow. Each block represents a Step sub-tree. There are two NextStep sub-tree(s) are created under the "Check the percentage remaining in the battery" Step sub-tree. One has a Condition node: "The battery is low on charge" and the other has a Condition node: "The battery is not low on charge."

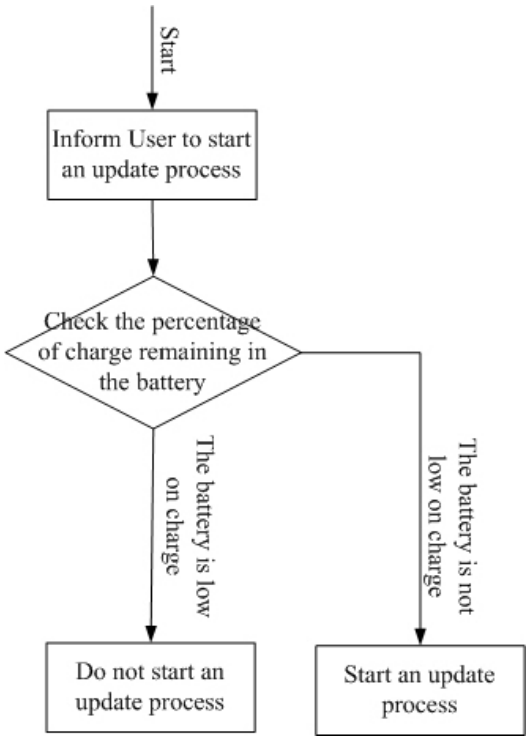


Figure 2: Practical Workflow

5.3 State Transition in SACMO

The Device internal operations and state transitions MUST appear to be atomic. If a state transition fails the Transaction remains in the previous state.

Each Transaction MUST have one of the four states as follows:

1. Inactive state
2. Active state

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

3. Suspend state
4. Rollback state

There are two kinds of state transition: Server-initiated transition and Client-initiated transition. The Server-initiated transition is triggered by a SACMO Exec operation which is sent from SACMO Server. The Client-initiated transition is triggered by Client when a transaction or a rollback is complete.

The following figure depicts the state transition:

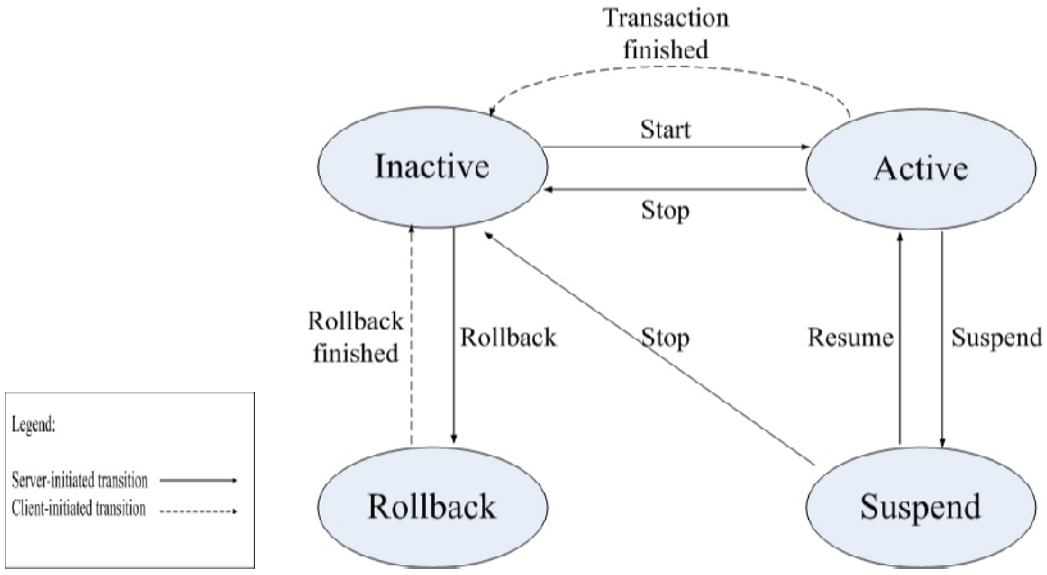


Figure 3: State Transition Diagram

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

6. Standardized Management Objects

6.1 Introduction to Management Objects

Management objects are the entities that can be manipulated by management actions carried over the OMA DM protocol. A Management Object can be as small as an integer or large and complex like a background picture, screen saver, or security certificate. The OMA DM protocol is neutral about the contents, or values, of the Management Objects and treats the node values as opaque data.

6.2 Definition and description of Management Objects

OMA DM Management Objects are defined using the OMA DM Device Description Framework [DMTND], or DDF. The use of this description framework produces detailed information about the device in question. However, due to the high level of detail in these descriptions, they are sometimes hard for humans to digest and it can be a time consuming task to get an overview of a particular object's structure.

In order to make it easier to quickly get an overview of how a Management Object is organized and its intended use, a simplified graphical notation in the shape of a block diagram is used in this document. Even though the notation is graphical, it still uses some printable characters, e.g. to denote the number of occurrences of a node. These are mainly borrowed from the syntax of DTDs for XML. The characters and their meaning are defined in the following table.

Character	Meaning
+	one or many occurrences
*	zero or more occurrences
?	zero or one occurrences

If none of these characters is used the default occurrence is exactly once.

There is one more feature of the DDF that needs to have a corresponding graphical notation, the un-named block. These are blocks that act as placeholders in the description and are instantiated with information when the nodes are used at run-time. Un-named blocks in the description are represented by a lower case character in italics, e.g. x.

Each block in the graphical notation corresponds to a described node, and the text is the name of the node. If a block contains an x, it means that the name is not known in the description and that it will be assigned at run-time. The names of all ancestral nodes are used to construct the URI for each node in the Management Object. It is not possible to see the actual parameters, or data, stored in the nodes by looking at the graphical notation of a Management Object.

For a further introduction to this graphical notation, please refer to [DMStdObj].

6.3 DDF Compliance

The Management Object descriptions in this document are normative. However, the descriptions also contain a number of informative aspects that could be included to enhance readability or serve as examples. Other informative aspects are, for instance, the ZeroOrMore and OneOrMore elements, where implementations may introduce restrictions. All these exceptions are listed here:

- All XML comments, e.g. "<!–some text □", are informative.
- The descriptions do not contain an RTProperties element, or any of its child elements, but a description of an actual implementation of this object MAY include these.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

- If a default value for a leaf node is specified in a description, by the DefaultValue element, an implementation MUST supply its own appropriate value for this element. If the DefaultValue element is present in the description of a node, it MUST be present in the implementation, but MAY have a different value.
- The value of all Man, Mod, Description and DFTitle elements are informative and included only as examples.
- Below the interior nodes Ext and BearerParams, an implementation may add further nodes at will.
- The contents of the AccessType element MAY be extended by an implementation.
- If any of the following AccessType values are specified, they MUST NOT be removed in an implementation: Copy, Delete, Exec, Get, and Replace.
- If the AccessType value Add is specified it MAY be removed in an implementation if the implementation only supports a fixed number of child nodes.
- An implementation MAY replace the ZeroOrMore or OneOrMore elements with ZeroOrN or OneOrN respectively. An appropriate value for N must also be given with the …OrN elements.

6.4 Conformance Definitions

The status definition in the node definitions indicates if the client supports that node or not. If the status is “Required” then the client MUST support that node in the case the client supports the parent node.

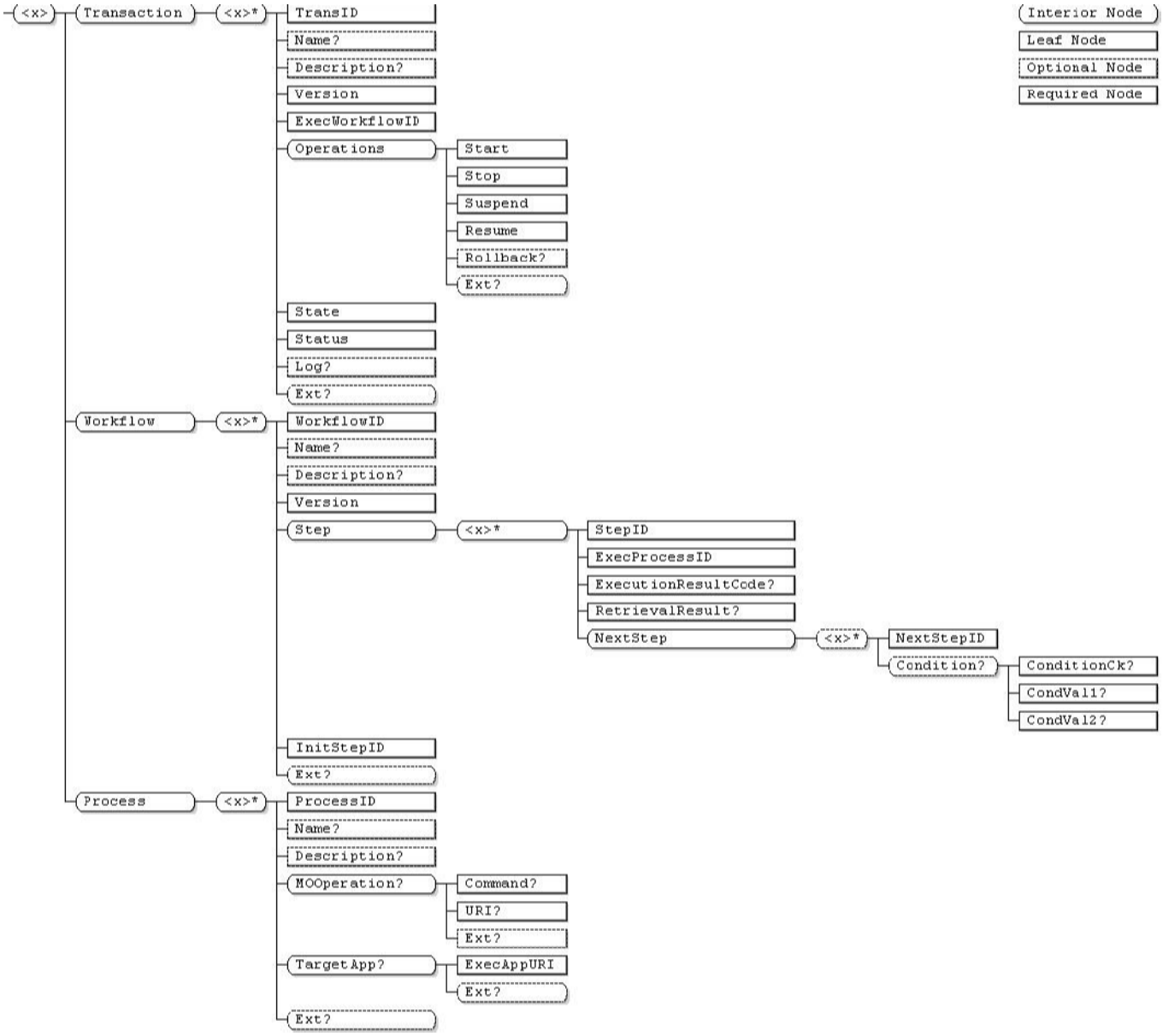
© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

7. Software and Application Control Management Object

The Management Objects associated with Software and Application Control management are assembled under an unnamed interior node x, dynamically or statically created.

Protocol Compatibility: This object is compatible with OMA Device Management protocol specifications, version 1.2 [DMPRO].

7.1 Figure of the Management Object (Informative)



© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

7.2 SACMO Parameters

Software and Application Control Management Object consists of following parameters:

<x>

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node groups together the parameters of a SACMO. The ancestor elements of this node define the position in the Management Tree of this Management Object. However, the structure of the DM tree and hence positions in the tree of Management Objects are out of scope of this specification.

The type of this node MUST be the SACMO Management Object ID "urn:oma:mo:oma-sacmo:1.0".

Transaction

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node groups together the transaction nodes.

Transaction/<X>

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrMore	node	Get

This interior node groups the information of one transaction.

Transaction/<X>/TransID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

The TransID uniquely identifies the Transaction within the SACMO tree. It is provided by the SACMO Server.

Transaction/<X>/Name

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the Name of a Transaction. It is provided by the SACMO Server.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Transaction/<X>/Description

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the Description of a Transaction. It is provided by the SACMO Server.

Transaction/<X>/Version

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies the Version of a Transaction.

The format of the version string is implementation specific and MAY depend on the characteristics of the Transaction, however this version string SHOULD encapsulate information that can differentiate between different versions (e.g. different editions) of the same Transaction.

A common and widely accepted version string contains a sequence of two or more positive integers, separated by '.' character, for example: "2.1.1".

A Transaction that does not have any version information at the time of download SHOULD use an empty string, to denote that it is version-less.

Transaction/<x>/ ExecWorkflowID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a workflow to be executed. This value refers to one of <WorkflowID>s listed under Workflow/<x>.

Transaction/<X>/Operations

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node is a parent node for operation that can be executed on a Transaction.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Transaction/<X>/Operations/Start

Status	Occurrence	Format	Min. Access Types
Required	One	null	Exec

This node is used with Exec command to start the Transaction operation; Once started, the Transaction transits to the Active state.

Transaction/<X>/Operations/Stop

Status	Occurrence	Format	Min. Access Types
Required	One	null	Exec

This node is used with Exec command to stop the Transaction operation; Once terminated, the Transaction transits to the Inactive state.

Transaction/<X>/Operations/Suspend

Status	Occurrence	Format	Min. Access Types
Required	One	null	Exec

This node is used with Exec command to suspend the Transaction operation; Once suspended, the Transaction transits to the Suspend state.

Transaction/<X>/Operations/Resume

Status	Occurrence	Format	Min. Access Types
Required	One	null	Exec

This node is used with Exec command to resume the Transaction operation; Once resumed, the Transaction transits to the Active state.

Transaction/<X>/Operations/Rollback

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	null	Exec, Get

This node is used with Exec command to Rollback the Transaction operation; Transaction is rolled back from previous Active state.

Transaction/<X>/Operations/Ext

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

This Optional interior node designates a branch of the Transaction operations sub-tree into which platform or vendor extensions MAY be added, permanently or dynamically. However, vendor extensions MUST NOT be defined outside of one of these Ext sub-trees.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.
[OMA-Template-Spec-20110101-I]

Transaction/<X>/State

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

This leaf node specifies the state of a Transaction. The value of this node MUST be one of the following:

Integer Value	**State**	**Description**
10	Inactive	The Transaction is in the Inactive state
20	Active	The Transaction is in the Active state
30	Suspend	The Transaction is in the Suspend state
40	Rollback	The Transaction is in the Rollback state

Transaction/<X>/Status

Status	Occurrence	Format	Min. Access Types
Required	One	int	Get

This leaf node specifies the Status of a Transaction. The value of this node MUST be one of the following:

Integer Value	**Status**	**Description**
10	Idle	The Transaction has not been started
20	Start Failed	The Transaction failed to activate
30	Transaction Progressing	The Transaction is running
40	Rollback	The Transaction is in the process of rollback
50	Transaction failed	The Transaction failed
60	Rollback failed	The Transaction failed to rollback
70	Transaction stopped	The Transaction has been stopped
80	Transaction suspended	The Transaction has been suspended

Transaction/<X>/Log

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	xml	Get

This leaf node stores the log of Transaction Operation encapsulated in XML format. The XML schema of the data is left to implementation.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Transaction/<X>/Ext

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

This Optional interior node designates a branch of the Transaction sub-tree into which platform or vendor extensions MAY be added, permanently or dynamically. However, vendor extensions MUST NOT be defined outside of one of these Ext sub-trees.

Workflow

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node groups together all the workflow nodes.

Workflow/<X>

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrMore	node	Get

This interior node is the multi-workflow node.

Workflow/<X>/WorkflowID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

The WorkflowID uniquely identifies the Workflow within the SACMO tree. It is provided by the SACMO Server.

Workflow/<X>/Name

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the name of a Workflow. It is provided by the SACMO Server.

Workflow/<X>/Description

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the description of a Workflow. It is provided by the SACMO Server.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Workflow/<X>/Version

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies the Version of a Workflow. It is provided by the SACMO Server.

The format of the version string is implementation specific and MAY depend on the characteristics of the Workflow, however this string SHOULD encapsulate information that can differentiate between different versions (e.g. different editions) of the same Transaction.

A common and widely accepted version string contains a sequence of two or more positive integers, separated by '.' character, for example: "2.1.1".

A Workflow that does not have any version information at the time of download SHOULD use an empty string, to denote that it is version-less.

Workflow/<X>/Step

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node groups together all the Step nodes

Workflow/<X>/Step/<X>

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrMore	node	Get

This interior node specifies the multi Step of Workflow.

Workflow/<X>/Step/<X>/StepID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

The StepID uniquely identifies the Step within the SACMO tree. It is provided by the SACMO Server.

Workflow/<X>/Step/<X>/ExecProcessID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies the ProcessID of the Process to be executed.

Workflow/<X>/Step/<X>/ExecutionResultCode

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

This leaf node specifies the result code of the Process executed in this Step. This node is created after a Process is executed. The result code SHOULD refer to other MO Enablers which is executed in this Step such as SCOMO, DiagMon. If the Process targets a non-DM client on the device, the value of this node MUST be one of the following:

Integer Value	**Status**	**Description**
1200	Successful	Successful – The Request has Succeeded
1250-1299	Successful – Vendor Specified	Successful Operation with vendor specified Result
1400	Failed	Failed – the Request has Failed
1450-1499	Failed – Vendor Specified	Failed Operation with vendor specified Result

Workflow/<X>/Step/<X>/RetrievalResult

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	Get

This leaf node specifies the retrieval result. The result is stored after the *Get* command of *Process/<X>/MOOperation/Command* is executed. The content of this node MUST be a text string encoded such as integer, boolean, float, character, date and time, and the transfer-encoding method described in section 5.2.4 of SyncML Common Meta-Information [META] MUST be used. Other formats are not supported.

Workflow/<X>/Step/<X>/NextStep

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This interior node is a parent node for next step and the condition information.

Workflow/<X>/Step/<X>/NextStep/<X>

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrMore	node	Get

This interior node is a placeholder for next step and the condition information.

Workflow/<X>/Step/<X>/NextStep/<X>/NextStepID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies the StepID of the next Step to be executed.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Workflow/<X>/Step/<X>/NextStep/<X>/Condition

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

This inferior node specifies a condition to the next Step to be applied. The Condition is provided by the SACMO Server.

Workflow/<X>/Step/<X>/NextStep/<X>/Condition/ConditionCk

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

This leaf node specifies the Condition Check to be performed on the value contained in the ExecutionResultCode node or RetrievalResult node.

Before performing the Condition Check with RetrievalResult node, the value of RetrievalResult node and the value of CondVal1 node (and CondVal2 node in case value is 70 or 75) SHALL be transformed into the original format.

Before performing the Condition Check with ExecutionResultCode node, the value of CondVal1 node (and CondVal2 node in case value is 70 or 75) SHALL be transformed into the int format.

The value of this node MUST be one of the following:

Integer Value	**Status**	**Description**
10	>	The value of ExecutionResultCode is greater than CondVal1
15	>	The value of RetrievalResult is greater than CondVal1
20	<	The value of ExecutionResultCode is less than CondVal1
25	<	The value of RetrievalResult is less than CondVal1
30	> or =	The value of ExecutionResultCode is greater than or equal to CondVal1
35	> or =	The value of RetrievalResult is greater than or equal to CondVal1
40	< or =	The value of ExecutionResultCode is less than or equal to CondVal1
45	< or =	The value of RetrievalResult is less than or equal to CondVal1
50	=	The value of ExecutionResultCode is equal to CondVal1

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Integer Value	Status	Description
55	=	The value of RetrievalResult is equal to CondVal1
60	Not =	The value of ExecutionResultCode is not equal to CondVal1
65	Not =	The value of RetrievalResult is not equal to CondVal1
70	Between	The value of ExecutionResultCode is between CondVal1 and CondVal2
75	Between	The value of RetrievalResult is between CondVal1 and CondVal2

Workflow/<X>/Step/<X>/NextStep/<X>/Condition/CondVal1

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	Get

This leaf node specifies the first value used by Condition Check to compare with the value in the ExecutionResultCode node or in the RetrievalResult node. The content of this node MUST be a text string encoded for the comparing value such as integer, Boolean, float, character, date and time, and the transfer-encoding method described in section 5.2.4 of SyncML Common Meta-Information [META] MUST be used. Other formats are not supported.

Workflow/<X>/Step/<X>/NextStep/<X>/Condition/CondVal2

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	Get

This leaf node specifies the second value used by Condition Check to compare with the value in the ExecutionResultCode node or in the RetrievalResult node. The content of this node MUST be a text string encoded for the comparing value such as integer, Boolean, float, character, date and time, and the transfer-encoding method described in section 5.2.4 of SyncML Common Meta-Information [META] MUST be used. Other formats are not supported.

Workflow/<X>InitStepID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies the StepID of the initial Step to be executed in the workflow.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Workflow/<X>/Ext

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

This Optional interior node designates a branch of the Workflow sub-tree into which platform or vendor extensions MAY be added, permanently or dynamically. However, vendor extensions MUST NOT be defined outside of one of these Ext sub-trees.

Process

Status	Occurrence	Format	Min. Access Types
Required	One	node	Get

This node is a parent node for all the Processes associated with Workflow.

Process/<X>

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrMore	node	Get

This node is an internal node for all the Processes associated with Workflow.

Process/<X>/ProcessID

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

The ProcessID uniquely identifies the Process within the SACMO tree. It is provided by the SACMO Server.

Process/<X>/Name

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the Name of a Process. It is provided by the SACMO Server.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Process/<X>/Description

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	chr	Get

This leaf node specifies the Description of a Process. It is provided by the SACMO Server.

Process/<X>/MOOperation

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	node	Get

This interior node groups parameter of operation for a MO node. When *Process/<x>/TargetApp* is not present, this node MUST be present. In case both *Process/<x>/MOOperation* as well as *Process/<x>/TargetApp* exist for a given *ProcessID*, then SACMO Client MUST use *Process/<x>/MOOperation* and ignore *Process/<x>/TargetApp node*.

Process/<X>/MOOperation/Command

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	int	Get

This leaf node specifies a command for a MO node. The command of this node MUST be one of the following:

Integer Value	**Description**
10	Exec Command. Operate an DM Exec command on a node of a MO
20	Get Command. Operate a DM Get command on a node of a MO and store data in the Workflow/<X>/Step/<X>/RetrievalResult node.
30 – 100	Reserved

Process/<X>/MOOperation/URI

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	chr	Get

This leaf node specifies an address to a node which resides in the Management Tree of the DM Client, and the node addressed by MOOperation/*URI* should be able to be used with MOOperation/Command. For example, MOOperation/*URI* can point to *Download/<x>/Operations/Download* under the Management Tree of SCOMO and is accompanied by *Exec* value in the MOOperation/Command node.

Process/<X>/MOOperation/Ext

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

This optional interior node is for vendor-specific extensions. This node can contain additional parameters which will be used in the MO operation. The content and format of this node are out of scope of SACMO.

Process/<X>/TargetApp

Status	Occurrence	Format	Min. Access Types
Required	ZeroOrOne	node	Get

This interior node groups information that is used for executing an application which can be accessed by SACMO Client. When *Process/<x>/ExecMOURI* is not present, this leaf node MUST be present.

Process/<X>/TargetApp/ExecAppURI

Status	Occurrence	Format	Min. Access Types
Required	One	chr	Get

This leaf node specifies a URI for an application to be executed. The application could be a native executable in the local storage of the End Device or a script stored in the remote file server. The content and format of this node are out of scope of SACMO, and they are implementation-specific.

Process/<X>/TargetApp/Ext

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

This optional interior node is for vendor-specific extensions associated with the application identified by *ExecAppURI*. This node can contain additional parameters which will be delivered to the application for processing, or any metadata needed during the execution of the application. The content and format of this node are out of scope of SACMO.

Process/<X>/Ext

Status	Occurrence	Format	Min. Access Types
Optional	ZeroOrOne	node	Get

This Optional interior node designates a branch of the Process sub-tree into which platform or vendor extensions MAY be added, permanently or dynamically. However, vendor extensions MUST NOT be defined outside of one of these Ext sub-trees.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

8. Behavior associated with the Management Object

In the SACMO tree, the Exec command is only allowed to target Primitives under Operations nodes. After an Exec command to one of the Primitives under the Operations node, the SACMO Client MUST send a response to the SACMO Server in either of the following ways:

1. Asynchronously if the Exec command is acceptable and will be executed asynchronously.
    - The SACMO Client MUST return status code 202 ("Accepted for processing") for the Exec command as defined in [DMREPRO].
    - Upon completion of the asynchronous operation, the SACMO Client MUST send an alert back to the SACMO Server by using the Generic Alert defined in [DMPRO].
2. Synchronously: If the Exec command is acceptable and executed synchronously.
    - If the Exec command is performed successfully, the SACMO Client MUST return SACMO Result Code 1200 ("Successful – The Request has Succeeded") or Result Code in the range 1250-1299 ("Successful Operation with vendor specified Result") by the Status command.

Before running a Transaction, a SACMO Client MUST check existence of all components related to the Transaction. Then once the Transaction has started, the SACMO Client MUST also check for existence of a given component before it is to be used in the runtime. If a component does not exist, the SACMO Client MUST stop the Transaction and act as follow:

- If the Exec command is executed asynchronously, the SACMO Client MUST send a Generic Alert to the SACMO Server with a corresponding Result Code in the range 1415-1417.
- If the Exec command is executed synchronously, the SACMO Client MUST return SACMO Result Code in the range 1415-1417 for the Exec command.

In SACMO Client's response message for reporting purpose, the SACMO Client MUST include the Transaction, Workflow, Step or Process information in addition to the status code described above:

- If a Transaction is completed, the /Transaction/<x*>/TransID URI was invoked in <Item>/<Target>/<LocURI>.
- If a Transaction is failed due to component existence check, an URI of the node that refers to an inexistence Workflow/Step/Process SHALL be invoked in <Item>/<Target>/<LocURI>. For example: If a Workflow is missing in the Workflow check of a Transaction, the /Transaction/<x*>/ExecWorkflowID URI is given. If a Process is missing in the Process check of a Step, the /Workflow/<x*>/Step/<x*>/ExecProcessID URI is given. If a Step is missing in the Step check of a NextStep, the /Workflow/<x*>/Step/<x*>/NextStep/<x*>/NextStepID URI is given.
- If a Transaction is failed due to Step execution failure, the /Workflow/<x*>/Step/<x*>/StepID URI was invoked in <Item>/<Target>/<LocURI>
- URI of the Primitive node on which the Exec command was invoked in <Item>/<Source>/<LocURI> element
- For asynchronous reporting, SACMO Result Code of the Primitive executed in <Item>/<Data> element which format is 'xml'

The detailed Exec command semantics and SACMO Client responsibility are described below.

8.1 Structure Elements of <Item>/<Data> for Reporting

As described above, the Result Code will be included in XML structure in <Item>/<Data> element. The XML schema is defined in the Appendix C:

- ResultCode: Specify the SACMO Result Code of the Primitive executed

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

8.2 Exec command semantics on Transaction

Before receiving the Exec command, the following pre-conditions need to be satisfied:

- Dynamic node creation
- At least the following nodes need to be set with an appropriate value:
  - /TransID
  - /Operations… (at least one Operation sub-node for the Primitive to be executed)

The Exec command targeting a child node of an Operations node starts the execution of a chosen Primitive. For example:

```
<Exec>
<CmdID>3</CmdID>
 <Item>
   <Target>
    <LocURI>./SAM/Transaction/Trans789/Operation/Start</LocURI>
   </Target>
 </Item>
</Exec>
```

If the Exec command is targeting Start Primitive of the Transaction, the SACMO Client MUST start the Transaction operation, the flow is described in the example Trans789.

8.3 Use of the Generic Alert

If the Exec command is executed asynchronously, the SACMO Client MUST send a notification to the SACMO Server about the outcome of the operation via a Generic Alert [DMPRO] message. The SACMO Client MAY send the alert message during the same or subsequent DM session. The alert message content is specified in the section 8 and MUST include an alert type specified in the section 8.5.

Alerts that are reporting an error or failure condition SHOULD report an importance level defined in [DMPRO] higher than Informational in the Mark field of the Meta information. If the SACMO Server needs to retrieve additional information, such as transaction status described in /Status node, then the SACMO Server MAY query the Device for those specific nodes.

8.4 SACMO Result Code

The Result Code of the operation MUST be sent as an integer value in Item/Data element of the Generic Alert [DMPRO] message or in response to an Exec command in case of synchronous execution. The Result Code MUST be one of the values defined below:

Result Code	Result Message	Informative Description of Status Code Usage
1200	Successful	Successful – The Request has Succeeded
1250-1299	Successful – Vendor Specified	Successful Operation with vendor specified Result
1400	Client Error	Client error – based on User or Device behaviour
1401	User cancelled	User chose not to accept the operation when prompted
1402	Transaction Failed	The Transaction failed

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

1403	Transaction failed due to Device is out of memory	The Transaction failed due to insufficient memory in the Device to run the Transaction.
1404	Start failed	The SACMO Start operation failed
1405	Stop failed	The SACMO Stop operation failed
1406	Suspend failed	The SACMO Suspend operation failed
1407	Resume failed	The SACMO Resume operation failed
1408	Rollback failed	The SACMO Rollback operation failed
1409	Undefined Error	Indicates failure not defined by any other error code
1410	Workflow does not exist	The Workflow to be executed does not exist.
1411	Step does not exist	The Step to be executed does not exist.
1412	Process does not exist	The Process to be executed does not exist.
1413	Workflow aborted – authorization failure	The Server requesting execution of a Workflow does not have access rights to an MO involved in a Process contained in the Workflow
1450-1499	Client Error – Vendor Specified	Client Error encountered for Operation with vendor specified Result Code
1500	Alternate Download Server Error	Alternate Download Server Error Encountered
1501	Alternate Download Server Unavailable	The Alternate Download Server is unavailable or does not respond
1550-1599	Alternate Download Server Error – Vendor Specified	Alternate Download Server Error encountered for Operation with vendor specified Result Code

8.5 Alert Types for SACMO

The following alert type MUST be used in a Generic Alert [DMPRO] message originating from a Software Application and Control Management Object. The alert type is used to identify the operation that was performed on the Device.

- urn:oma:at:sacmo:1.0:OperationReport

8.6 User Interaction Commands

The SACMO Server SHALL be capable of including User Interaction commands in the workflow which will be displayed to the User, prompting the user for a response Upon receiving the response the workflow SHALL resume.

The SACMO Server SHALL support DISPLAY Alert, CONFIRM OR REJECT Alert and USER CHOICE Alert as described in [DMPRO] in the workflow.

The SACMO Client, when installed on a device that supports user interaction, SHALL support DISPLAY Alert, CONFIRM OR REJECT Alert and USER CHOICE Alert as described in [DMPRO].

8.7 Requesting User Confirmation

The SACMO Server MAY request user confirmation before performing SACMO operations within the Device. However it is important to ensure that the SACMO Client supports the ability to handle the user confirmation request.

In particular the SACMO Client:

- When installed on a device that supports user interaction, SHALL support DISPLAY Alert as described in [DMPRO].

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

- When installed on a device that supports user interaction, SHALL support CONFIRM OR REJECT Alert as described in [DMPRO].
- SHOULD support USER INPUT Alert as described in [DMPRO].
- When installed on a device that supports user interaction, SHALL support USER CHOICE Alert as described in [DMPRO].

In particular the SACMO Server:

- SHALL support triggering of DISPLAY Alert to DM Client
- SHALL support triggering of CONFIRM OR REJECT Alert to DM Client.
- SHOULD support triggering of USER INPUT Alert to DM Client.
- SHALL support triggering of USER CHOICE Alert to DM Client.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

9. Security Considerations (Informative)

Security for delivery and management of Software and Application Control Management Objects is of paramount importance. Authentication mechanisms supported by OMA DM protocol [DMPRO] SHALL be used to ensure that authenticated entities can deliver/perform management operations on Software and Application Management Objects on the Device. Similarly, authorization for management operations on Software and Application Control Management under the purview of this enabler SHALL be based on the ACL mechanisms defined by OMA DM TND [DMTND].

It is conceivable that a Workflow being executed under the authority of one Server may invoke Mos to which that Server does not have access rights. For each Process, the SACMO Client SHALL check that the Server requesting execution of the Workflow has access rights to all Mos involved in that Process. If the Server requesting execution of a Workflow does not have access rights to any MO involved in the Process being checked, the SACMO Client SHALL abort execution of the Workflow and SHALL return a Result Code of 1413 (Workflow aborted – authorization failure).

The SACMO enabler does not mandate nor restrict any mechanism to guarantee authenticity, confidentiality and integrity of Software and Application Control Management Objects delivered to the Device. It is envisioned that existing security mechanisms for this purpose such as Digital Signatures, SSL [SSL3.0], TLS [TLS1.0], etc. can easily work in conjunction with SACMO.

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version

A.2 Draft/Candidate Version 1.0 History

Document Identifier	Date	Sections	Description
Draft Versions: OMA-TS-SACMO-V1_0	26 Apr 2010	all	Initial Draft agreed in "OMA-DM-SACMO-2010-0003R02-INP_TS_baseline"
	27 Jul 2010	2.1, 6.2, 6.3, 7.8	Incorporated: OMA-DM-SACMO-2010-0010-CR_Reference_correction OMA-DM-SACMO-2010-0011R02-CR_SACMO_parameters_corrections OMA-DM-SACMO-2010-0012R03-CR_SACMO_User_Interaction_Commands Editorial Fix: Add a space in section 7.8
	14 Sep 2010	2.1, 2.2, 6.1, 6.3, 7, 7.1, 7.3, 7.4, 7.5, 7.6	Incorporated: OMA-DM-SACMO-2010-0015R02-CR_TS_Cleanup
	12 Oct 2010	3, 5, 6, 7, 8, 9, 10	Incorporated: OMA-DM-SACMO-2010-0018R01-CR_TS_Clarification OMA-DM-SACMO-2010-0019-CR_Remove_Tree_From_Table OMA-DM-SACMO-2010-0020R02-CR_Definition_and_New_Section OMA-DM-SACMO-2010-0021-CR_Bug_Fix
	17 Nov 2010	4, 5.3, 7.1, 7.2, 8.6	Incorporated: OMA-DM-SACMO-2010-0022-CR_SACMO_Result_Code OMA-DM-SACMO-2010-0023-CR_Add_missing_InitStepID_node OMA-DM-SACMO-2010-0024R01-CR_State_Transition OMA-DM-SACMO-2010-0025R01-CR_Add_TargetApp_Node OMA-DM-SACMO-2010-0026-CR_Intro_Cleanup OMA-DM-SACMO-2010-0027R02-CR_New_Node_for_Step_Result
	23 Nov 2010	4, 7.2	Editorial Fix & Add Figure Table
	28 Jan 2011	All	Editorial clean-up of hyerplinks and fonts by DSO, language set to English UK.
	15 Feb 2011	5.1.3, 5.2, , 7.1, 7.2 9 (deleted)	Incorporated: OMA-DM-SACMO-2010-0030-CR_Single_Workflow_per_a_Transaction OMA-DM-SACMO-2010-0031-CR_ExecURI_TargetAPP OMA-DM-SACMO-2010-0032-CR_Bug_Fix_for_Start_node OMA-DM-SACMO-2011-0004R02-CR_Transformation_of_Workflow
	21 Apr 2011	8.8, 8.9, Appendix B	Incorporated: OMA-DM-SACMO-2011-0003R03-CR_SCR OMA-DM-SACMO-2011-0005R02-CR_Inconsistency_of_User_Interaction_Commands OMA-DM-SACMO-2011-0006R03-CR_SACMO_Client_UI_Commands OMA-DM-SACMO-2011-0008R03-CR_SACMO_Srv_UI_Cmd_Support Editorial Clean-Up
	09 Jun 2011	4, 5.1.1, 8, 8.2, 8.4, 9	Incorporated: OMA-DM-SACMO-2011-0007R02-CR_Behaviour_of_checking_the_existence_of_SACMO_component OMA-DM-SACMO-2011-0009R01-CR_SACMO_Security_Fix OMA-DM-SACMO-2011-0010R01-CR_CONRR_C001_C002_fix

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Document Identifier	Date	Sections	Description
Draft Versions: OMA-TS-SACMO-V1_0	05 Jul 2011	7.2, 8, 8.1, 8.3, 8.4, 8.5, 9, App C (new)	Incorporated: OMA-DM-SACMO-2011-0011R01-CR_Process_Function_Extension OMA-DM-SACMO-2011-0012R01-CR_SACMO_Condition_Fix OMA-DM-SACMO-2011-0013-CR_SACMO_CONRR_C005 OMA-DM-SACMO-2011-0014R01-CR_SACMO_Result_Code_Correct
	06 Jul 2011	8	Re-application of CR OMA-DM-SACMO-2011-0013-CR_SACMO_CONRR_C005 (change 1)
Candidate Version: OMA-TS-SACMO-V1_0	19 Jul 2011	All	Status changed to Candidate by TP: OMA-TP-2011-0257-INP_SACMO_V1_0_ERP_for_Candidate_Approval

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Appendix B. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [SCRRULES].

B.1 SCR for SACMO Tree Structure

Item	Function	Reference	Requirement
SACMO-T-001-M	Use of appropriate Management Object identifier for the SACMO node	Section 7.2	
SACMO-T-002-M	Support for Required nodes under root node	Section 7.2	
SACMO-T-003-O	Support for Optional nodes	Section 7.2	
SACMO-T-004-M	Support for Required nodes under an Optional node if the Optional node is supported	Section 7.2	

B.2 SCR for SACMO Client

Item	Function	Reference	Requirement
SACMO-C-001-M	Support state transitions that appear to be atomic. If a state transition fails the Transaction remains in the previous state.	Section 5.3	
SACMO-C-002-M	Support Inactive State for Transaction	Section 5.3	
SACMO-C-003-M	Support Active State for Transaction	Section 5.3	
SACMO-C-004-M	Support Suspend State for Transaction	Section 5.3	
SACMO-C-005-O	Support Rollback State for Transaction	Section 5.3	
SACMO-C-006-M	Support result report	Section 8	
SACMO-C-007-O	Support asynchronous result reporting	Section 8	
SACMO-C-008-O	Support synchronous result reporting	Section 8	
SACMO-C-009-O	Support Generic Alert	Section 8.3	

B.3 SCR for SACMO Server

Item	Function	Reference	Requirement
SACMO-S-001-M	Support Software and Application Control Management Object	Section 7.2	
SACMO-S-002-M	Support triggering of	Section 8.7	

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Item	Function	Reference	Requirement
	DISPLAY Alert		
SACMO-S-003-M	Support triggering of COMFIRM OR REJECT Alert	Section 8.7	
SACMO-S-004-O	Support triggering of USER INPUT Alert	Section 8.7	
SACMO-S-005-M	Support triggering of USER CHOICE Alert	Section 8.7	

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.

Appendix C. XML Schema for Generic Alert Data (Normative)

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">

  <xs:element name="OperationReport">

    <xs:complexType>

      <xs:sequence>

        <xs:element ref="ResultCode" maxOccurs="1"/>

      </xs:sequence>

    </xs:complexType>

  </xs:element>

  <xs:element name="ResultCode" type="xs:unsignedInt" />

</xs:schema>
```

© 2011 Open Mobile Alliance Ltd. All Rights Reserved.
Used with the permission of the Open Mobile Alliance Ltd. under the terms as stated in this document.