ADS API Specification

Version: 7.0.0

Revision History

Date	Version	Description	Author
October 2005	1.0	Initial Draft. Updated from Phase1b Documentation.	Mark Lassiter
January 31, 2006	1.1	Revised based on feedback from LCG testers. Added TOC, Revision History and Cover page. Renamed the document to ADS API Specification.	Mark Lassiter
February 13, 2006	1.2	Removed PenaltyBidSegment field. Added bidDelay field. Added Change Log	Mark Lassiter
May 31, 2006	1.3	Add MSS LF methods and payloads. Added AS Award Type field to instruction payload.	Mark Lassiter
May 31, 2006	1.4	Added MSS data dictionary items. Updated TOC.	Mark Lassiter
October 12, 2006	1.5	Added XSD to Section 4.0 as per PMO standards request.	Karen Holly-Asper
October 13, 2006	1.6	Added Sample XML Added Batch Type for OOS Instructions	Mark Lassiter
February 19, 2007	1.7	Modified spec for CR3B changes.	Mark Lassiter
February 21, 2007	1.8	Adjusted batch type definition and updated Market simulation reference. Added note regarding MSS LF.	Mark Lassiter
February 22, 2007	1.9	Removed Price field from Detail Segments	Mark Lassiter
April 4, 2007	2.0	Updated ads.caiso.com.xsd. Corrected data dictionary.	Jason Fleming
October 5, 2007	2.1	Removed PassIndicator from XSD.	Troy Siegel
November 5, 2007	2.2	Changed data types for UID fields to string.	Mark Lassiter
August 22, 2008	3.0	Removed DA AS Award from XSD and data definition. Added new ADS API Response Web	Mark Lassiter

		Service methods.	
September 17, 2008	3.1	Added DOP sequenceNumber and Trajectory Batch bindingFlag elements.	Mark Lassiter
December 7, 2009	4.0	Added MSG Configuration elements. Added ADS API Transition Support section. Updated UIDs to correctly reflect GUIDs instead of Integers.	Mark Lassiter
December 7, 2009	4.1	Added configuration id to Trajectory data.	Mark Lassiter
June 24, 2010	4.2	Add AGC and RMR flag	Ivan Loh
July 10, 2012	5.0	Add contingency dispatch elements.	Ivan Loh
August 28, 2012	5.1	Update CAISO header logo.	Ivan Loh
September 6, 2012	5.1.1	Update schema definitions	Ivan Loh
July 9, 2013	5.2.0	Add ADS Reason Codes	Ivan Loh
October 25, 2013	6.0.0	Add hourlyMwThreshold to instruction	Ivan Loh
March 17, 2014	7.0.0	EIM Release Add baseSchedule Add registeredIntertieFlag	Ivan Loh

Table of Contents

1. Overview

This document explains in detail the semantics of the function calls you can make using the ADS API web services. In this document the following are described:

- Syntax of requests.
- Format of the response data.

The following documents are referenced throughout this document:

Document	Description
ADSAPIWebService.wsdl	WSDL description of the ADS API Web Service.
ADSAPIResponseWebService.wsdl	WSDL description of the ADS API Response Web Service.
ads.caiso.com.xsd	XML Schema Definition describing the payload returned via the ADS API

1.1. Request Overview

The following requests are available through the ADS API Service:

Request	Description
getDispatchBatchesSinceUID	Retrieves a list of batch headers dispatched after a specified batch UID (unique identifier). Batches will not include instruction data.
getBatchStatus	Returns the batch status for a specified batch UID.
getBatchHeader	Returns the batch header for the specified batch UID.
getDispatchBatch	Retrieves the dispatch batch including the requestor's set of instructions for the specified batch UID.
isNewTrajData	Returns true if new trajectory data has become available since a specified batch UID.
getTrajectoryData	Returns all new trajectory data received since the specified batch UID.
submitMSSLFRequest	Receives an MSS Load Following Request into the ADS system and returns validation results.
validateDispatchBatch	Informs the ADS system that the specified batch UID has been successfully received by the requestor.

The following requests are available through the ADS API Response Web Service:

Request	Description
acceptInstruction	Fully accepts the specified Instruction.
declineInstruction	Declines the specified Instruction.
partialAcceptInstruction	Partially accepts the specified Instruction.

1.2. ADS API Response Overview

Each ADS API request will return a single response that will contain the requested data or a fault condition. The following custom response types are described below:

Response	Description
APIDispatchResponse	XML document containing a list of batches and associated information about each batch (e.g. batch UID, batch status, etc.).
DispatchBatch	XML Document containing a single ADS Dispatch Batch. May or may not contain instruction data depending on the call.
APITrajectoryResponse	XML document containing all trajectory data relevant to a particular client.
MSSLFResponse	XML document containing the validation results for each MSS LF Request submitted via **submitMSSLFRequest.**

1.3. ADS API Response Service Result Codes

The ADS API Response Web Service methods all return a single xsd:int value. The result codes are defined below:

Response	Code	Description
RC_SUCCESS	0	The Instruction Response was received and accepted by the ADS system.
RC_INVALID_RESPONSE	1	The Instruction Response submitted is invalid. For example, the MW value may be outside the acceptable range for the instruction.
RC_INVALID_RESPONSE_PERIOD	2	The Instruction Response was received by ADS outside of the Market Participant response period.
RC_BATCH_NOT_FOUND	3	The Batch UID specified does not exist.
RC_INSTRUCTION_NOT_FOUND	4	The Instruction UID specified does not exist in the Batch specified.
RC_UNAUTHORIZED	5	You do not have sufficient privileges to respond to the resource associated with the submitted Instruction UID.

1.4. Errors and Data Types

All error conditions will be reported as SOAP <fault> elements.
All other response types are standard xsd type definitions defined in the **ads.caiso.com.xsd** document.

2. ADS API Service Operations

2.1. getDispatchBatchesSinceUID

Returns all batches processed by the ADS system since the batch UID specified. Batches are returned in a list without instructions. The list will be empty if no batches are found. Batches should be processed in the order received. No assumptions should be made regarding the magnitude or ordering of the Batch UID. Always use the last batch UID processed for your next call or use -1 to retrieve the entire ADS API cache. At present, this would be all batches received by ADS in the last 24 hours.

The batch list will be returned as an XML document conforming to the **ads.caiso.com.xsd *APIDispatchResponse*** schema definition.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Last batch UID processed by your system.
Output	*response*	xsd:APIDispatchResponse	XML document containing a list of dispatches received since the batch UID specified.

2.2. getBatchStatus

This request returns the batch status associated with the provided batch UID.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Valid ADS batch UID.
Output	*response*	xsd:integer	See the section 4 for a list of valid Batch Status values.

2.3. getDispatchBatch

Returns the Dispatch Batch and associated instructions in Base64 encoded compressed XML for the batch UID specified. Only those instructions viewable by the requestor will be returned. This batch will be returned as an XML document conforming to the **ads.caiso.com.xsd *DispatchBatch*** schema definition. The data is returned as Base64 Encoded Compressed (GZIP) XML.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Valid ADS batch UID.
Output	*response*	xsd:DispatchBatch	XML document containing the requested batch and associated instructions viewable by the requestor.

2.4. getBatchHeader

Returns the batch header for the batch UID specified. This method can be used to check the status of batch as well as other batch attributes without the overhead of requesting the entire instruction set. The batch will be returned as an XML document conforming to the **ads.caiso.com.xsd *DispatchBatch*** schema definition. This data **will not** be compressed or encoded.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Valid ADS batch UID.
Output	*response*	xsd:DispatchBatch	XML document containing the requested batch. Instructions will not be returned.

2.5. isNewTrajData

Returns **true** if more recent trajectory data exists since the batch UID specified, otherwise returns **false**. Periodically execute to check for new trajectory data. When True is returned, execute the *getTrajectoryData* method to retrieve all new trajectory data in a single XML document.

Note that the Trajectory Batch UID is tracked separately form the Dispatch Batch UID. You should also track these values separately in your code.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Valid ADS trajectory batch UID.
Output	*response*	xsd:boolean	**True** if new trajectory exists, otherwise **false.**

2.6. getTrajectoryData

Retrieves trajectory data received by the system since the trajectory batch UID specified. This batch list will be returned as an XML document conforming to the **ads.caiso.com.xsd *APITrajectoryResponse*** schema definition. The data is returned as Base64 Encoded Compressed (GZIP) XML.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Valid ADS trajectory batch UID.
Output	*response*	xsd:APITrajectoryResponse	XML document containing a list of trajectory batches received since the batch UID specified. Each batch will contain the DOP and Compliance data associated with the batch.

2.7. validateDispatchBatch

Informs the ADS system that the specified batch UID has been successfully received by the requestor. Only those instructions for which the requestor has primary access to the resource will be validated. If the instruction has already been validated, the instruction is skipped. This method should ONLY be passed ADS Dispatch Batch UIDs. Do not pass trajectory UIDs.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Valid ADS batch UID.

2.8. submitMSSLFRequest

Use to submit an MSS Load Following Request into the ADS system. This method performs a simple validation on each submitted instruction and forwards valid instructions to IFM. This method will return an MSS LF Response with the newly assigned CAISO batch identifier and a list of the MSS LF Instruction Response items. Each item also includes the newly assigned CAISO Instruction identifier and a flag indicating whether or not the instruction was valid.

The submitted request is an XML document conforming to the **ads.caiso.com.xsd *MSSLFRequest*** schema definition. The response is an XML document conforming to the **ads.caiso.com.xsd *MSSLFResponse*** schema definition.

NOTE: This functionality is **ONLY available** to **MSS Load Following Market Participants**.

I/O	Name	Type	Description
Input	request	xsd:MSSLFRequest	XML document containing a list of MSS LF Instruction requests.
Output	*response*	xsd:MSSLFResponse	XML document containing a list of MSS LF Instructions submitted along with an assigned CAISO identifier and a flag indicating whether or not the instruction passed validation.

3. ADS API Response Service Operations

3.1. acceptInstruction

Accepts the specified ADS Instruction for the instructed DOT value. You must have a Primary or Secondary role for the resource associated with the instruction. The batch must not be expired and must be accepting responses. For hourly pre-dispatch instructions, this means the batch status must be 1.

A result code as defined in section 1.3 will be returned.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Batch UID of DispatchBatch associated with Instruction.
Input	instructionUID	xsd:string	Instruction UID of instructions to be accepted.
Output	*response*	xsd:int	Result code as defined in section 1.3.

3.2. declineInstruction

Declines the specified ADS Instruction for the entire instructed DOT value, returning the instruction to its Hour Ahead schedule or 0 if not scheduled. You must have a Primary or Secondary role for the resource associated with the instruction. The batch must not be expired and must be accepting responses. For hourly pre-dispatch instructions, this means the batch status must be 1.

A result code as defined in section 1.3 will be returned.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Batch UID of DispatchBatch associated with Instruction.
Input	instructionUID	xsd:string	Instruction UID of instructions to be accepted.
Input	reasonCode	xsd:int	A valid ADS reason code for declining the instruction. Valid values are: 10 - Line down 11 - Economic Considerations 12 - Bad Bid Submitted

			13 - Unit Derate 14 - No Available Transmission 15 - Timed Out: Minimum Accepted 16 - Timed Out: Forced Decline
Output	*response*	xsd:int	Result code as defined in section 1.3.

3.3. partialAcceptInstruction

Partially accepts the specified ADS Instruction for the specified MW amount. The specified MW amount must be between the Instruction DOT value and the resources Hour Ahead schedule.

You must have a Primary or Secondary role for the resource associated with the instruction. The batch must not be expired and must be accepting responses. For hourly pre-dispatch instructions, this means the batch status must be 1.

A result code as defined in section 1.3 will be returned.

I/O	Name	Type	Description
Input	batchUID	xsd:string	Batch UID of DispatchBatch associated with Instruction.
Input	instructionUID	xsd:string	Instruction UID of instructions to be accepted.
Input	acceptDOT	xsd:double	The accepted DOT amount which must be between the instructed DOT MW and the Hour Ahead schedule.
Input	reasonCode	xsd:int	A valid ADS reason code for partially accepting the instruction. Valid values are: 10 - Line down 11 - Economic Considerations 12 - Bad Bid Submitted 13 - Unit Derate 14 - No Available Transmission 15 - Timed Out: Minimum Accepted 16 - Timed Out: Forced Decline
Output	*response*	xsd:int	Result code as defined in section 1.3.

4. Recommended Usage

The ADS API was designed for real time usage and not as a historical query tool. For the latter, please use the ADS Query Tool.

The following is pseudo code demonstrating the expected usage of the ADS API in a real time client application. This logic should keep your client application current with all ADS Dispatch and Trajectory Batch data. ADS API methods are highlighted in red.

Pseudo Code

```
Load LastDispatchUID    // or default to -1 the first time
Load LastTrajectoryUID  // or default to -1 the first time

Start Loop
      // Check for new batches
      Batches = getDispatchBatchesSinceUID( LastDispatchUID )
      // Iterate batches returned (may be zero if no new)
      For Each Batch in Batches
            // Retrieve Instructions
            BatchData = getDispatchBatch( Batch.BatchUID )
            // Decode and decompress
            DecodeAndDecompress( BatchData )
            // Optional Step: Validate receipt
            validateDispatchBatch( Batch.BatchUID )
            // Process Batch Data (your logic)
            Process( BatchData )
            // Update the last batch uid processed
            Set LastDispatchUID = Batch.BatchUID
      End for each

      // Check for new trajectory batches
      If isNewTrajData( LastTrajectoryUID ) then
            // Get New Trajectory data
            TrajBatches = getTrajectoryData( LastTrajectoryUID )
            // Decode and decompress
            DecodeAndDecompress( TrajBatches )
            // Iterate Trajectory Batches
            For Each TrajBatch in TrajBatches
                  // Process Batch Data (your logic)
                  ProcessTrajectory( TrajBatch )
                  // Update the last batch uid processed
                  Set LastTrajectoryUID = TrajBatch.BatchUID
            End for each
      End if

      // Sleep
      Sleep 10 Seconds
End Loop
```

5. XSD

Below is the ADS XML schema definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<schema
    targetNamespace="http://ads.caiso.com/api/schema/v6"
    xmlns:tns="http://ads.caiso.com/api/schema/v6"
    xmlns="http://www.w3.org/2001/XMLSchema"
    xmlns:tns="http://ads.caiso.com"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified">

 <element name="DispatchBatch">
  <complexType>
   <sequence>
    <element name="marketID" type="xsd:string"/>
    <element name="batchStatus" type="xsd:int"/>
    <element name="batchReceived" type="xsd:dateTime"/>
    <element minOccurs="0" name="batchSent" type="xsd:dateTime"/>
    <element minOccurs="0" name="batchExpires" type="xsd:dateTime"/>
    <element name="batchType" type="xsd:int"/>
    <element name="startTime" type="xsd:dateTime"/>
    <element name="dispatchMode" type="xsd:int"/>
    <element name="bindingFlag" type="xsd:string"/>
    <element name="revisionNo" type="xsd:int"/>
    <element minOccurs="0" name="contingencyType" type="xsd:string"/>
    <element minOccurs="0" name="pathExclusion" type="xsd:string"/>
    <element minOccurs="0" name="instructions">
     <complexType>
      <sequence>
       <element maxOccurs="unbounded" minOccurs="0" ref="tns:instruction"/>
      </sequence>
     </complexType>
    </element>
   </sequence>
   <attribute name="batchUID" type="xsd:string" use="required"/>
  </complexType>
 </element>

 <element name="instruction">
  <complexType>
   <sequence>
    <element name="batchUID" type="xsd:string"/>
    <element name="resourceId" type="xsd:string"/>
    <element minOccurs="0" name="startTime" type="xsd:dateTime"/>
    <element minOccurs="0" name="endTime" type="xsd:dateTime"/>
    <element minOccurs="0" name="dot" type="xsd:double"/>
    <element minOccurs="0" name="oosEnergyCode" type="xsd:string"/>
    <element minOccurs="0" name="asType" type="xsd:string"/>
    <element name="instructionType" type="xsd:int"/>
```

```
      <element minOccurs="0" name="preGoto" type="xsd:double"/>
      <element minOccurs="0" name="bidDelay" type="xsd:int"/>
      <element minOccurs="0" name="minAccept" type="xsd:double"/>
      <element minOccurs="0" name="acceptDot" type="xsd:double"/>
      <element minOccurs="0" name="acceptStatus" type="xsd:string"/>
      <element minOccurs="0" name="responder" type="xsd:string"/>
      <element minOccurs="0" name="reasonCode" type="xsd:int"/>
      <element minOccurs="0" name="oprAcceptDot" type="xsd:double"/>
      <element minOccurs="0" name="oprAcceptStatus" type="xsd:string"/>
      <element minOccurs="0" name="oprResponder" type="xsd:string"/>
      <element minOccurs="0" name="oprReasonCode" type="xsd:int"/>
      <element minOccurs="0" name="validated" type="xsd:dateTime"/>
      <element minOccurs="0" name="validatedBy" type="xsd:string"/>
      <element minOccurs="0" name="apiValidated" type="xsd:dateTime"/>
      <element minOccurs="0" name="apiValidatedBy" type="xsd:string"/>
      <element name="revisionNumber" type="xsd:int"/>
      <element name="statusCode" type="xsd:int"/>
      <element minOccurs="0" name="clearedMW" type="xsd:double"/>
      <element minOccurs="0" name="awardMW" type="xsd:double"/>
      <element minOccurs="0" name="selfSchedMW" type="xsd:double"/>
      <element minOccurs="0" name="hourlyMw" type="xsd:double"/>
      <element minOccurs="0" name="rmrTestRequestor" type="xsd:string"/>
      <element minOccurs="0" name="configurationId" type="xsd:string"/>
      <element minOccurs="0" name="transitionFromConfigId" type="xsd:string"/>
      <element minOccurs="0" name="transitionToConfigId" type="xsd:string"/>
      <element minOccurs="0" name="agcFlag" type="xsd:string"/>
      <element minOccurs="0" name="rmrFlag" type="xsd:string"/>
      <element minOccurs="0" name="hourlyMwThreshold" type="xsd:double"/>
      <element minOccurs="0" name="baseSchedule" type="xsd:double"/>
      <element minOccurs="0" name="registeredIntertieFlag" type="xsd:string"/>
      <element minOccurs="0" name="detail">
        <complexType>
          <sequence>
            <element maxOccurs="unbounded" minOccurs="0" ref="tns:instructionDetail"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
    <attribute name="instructionUID" type="xsd:string" use="required"/>
  </complexType>
</element>

<element name="instructionDetail">
  <complexType>
    <sequence>
      <element minOccurs="0" name="serviceType" type="xsd:string"/>
      <element minOccurs="0" name="mw" type="xsd:double"/>
    </sequence>
    <attribute name="segNo" type="xsd:int" use="required"/>
  </complexType>
</element>
```

```
<element name="trajectoryBatch">
 <complexType>
  <sequence>
   <element name="batchReceived" type="xsd:dateTime"/>
   <element name="bindingFlag" type="xsd:string"/>
   <element minOccurs="0" name="batchSent" type="xsd:dateTime"/>
   <element minOccurs="0" name="dopList">
    <complexType>
     <sequence>
      <element maxOccurs="unbounded" minOccurs="0" ref="tns:trajectoryDop"/>
     </sequence>
    </complexType>
   </element>
   <element minOccurs="0" name="complianceList">
    <complexType>
     <sequence>
      <element maxOccurs="unbounded" minOccurs="0" ref="tns:trajectoryCompliance"/>
     </sequence>
    </complexType>
   </element>
  </sequence>
  <attribute name="batchUID" type="xsd:string" use="required"/>
 </complexType>
</element>

<element name="trajectoryDop">
 <complexType>
  <sequence>
   <element name="resourceId" type="xsd:string"/>
   <element name="dop" type="xsd:double"/>
   <element name="targetTime" type="xsd:dateTime"/>
   <element name="sequenceNumber" type="xsd:int" />
   <element minOccurs="0" name="configurationId" type="xsd:string"/>
  </sequence>
  <attribute name="dopUID" type="xsd:string" use="required"/>
 </complexType>
</element>

<element name="trajectoryCompliance">
 <complexType>
  <sequence>
   <element name="resourceId" type="xsd:string"/>
   <element name="startTime" type="xsd:dateTime"/>
   <element name="mwh" type="xsd:double"/>
   <element name="complFlag" type="xsd:string"/>
   <element minOccurs="0" name="configurationId" type="xsd:string"/>
  </sequence>
  <attribute name="complianceUID" type="xsd:string" use="required"/>
 </complexType>
</element>

<element name="APIDispatchResponse">
```

```
  <complexType>
   <sequence>
    <element name="dispatchBatchList">
     <complexType>
      <sequence>
       <element maxOccurs="unbounded" minOccurs="0" ref="tns:DispatchBatch"/>
      </sequence>
     </complexType>
    </element>
   </sequence>
  </complexType>
</element>

<element name="APITrajectoryResponse">
  <complexType>
   <sequence>
    <element name="trajectoryBatchList">
     <complexType>
      <sequence>
       <element maxOccurs="unbounded" minOccurs="0" ref="tns:trajectoryBatch"/>
      </sequence>
     </complexType>
    </element>
   </sequence>
  </complexType>
</element>

<element name="MSSLFRequest">
  <complexType>
   <sequence>
    <element name="scMSSBatchId" type="xsd:string"/>
    <element minOccurs="0" name="mssLFInstructionRequests">
     <complexType>
      <sequence>
       <element maxOccurs="unbounded" minOccurs="0" ref="tns:MSSLFInstructionRequest"/>
      </sequence>
     </complexType>
    </element>
   </sequence>
  </complexType>
</element>

<element name="MSSLFInstructionRequest">
  <complexType>
   <sequence>
    <element name="scMSSBatchId" type="xsd:string"/>
    <element name="scMSSLFInstructionId" type="xsd:string"/>
    <element name="resourceId" type="xsd:string"/>
    <element name="startTime" type="xsd:dateTime"/>
    <element name="endTime" type="xsd:dateTime"/>
    <element name="loadFollowingMW" type="xsd:double"/>
   </sequence>
```

```
    </complexType>
  </element>

  <element name="MSSLFResponse">
    <complexType>
      <sequence>
        <element name="caisoMSSBatchId" type="xsd:string"/>
        <element name="scMSSBatchId" type="xsd:string"/>
        <element minOccurs="0" name="mssLFInstructionResponses">
          <complexType>
            <sequence>
              <element maxOccurs="unbounded" minOccurs="0" ref="tns:MSSLFInstructionResponse"/>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </complexType>
  </element>

  <element name="MSSLFInstructionResponse">
    <complexType>
      <sequence>
        <element name="caisoMSSBatchId" type="xsd:string"/>
        <element name="caisoMSSLFInstructionId" type="xsd:string"/>
        <element name="scMSSBatchId" type="xsd:string"/>
        <element name="scMSSLFInstructionId" type="xsd:string"/>
        <element name="validated" type="xsd:boolean"/>
      </sequence>
    </complexType>
  </element>
</schema>
```

5.1. Sample XML

APIDispatchResponse

```
<?xml version="1.0" encoding="utf-8"?>
<APIDispatchResponse
      xmlns:xsd=”http://www.w3.org/2001/XMLSchema”
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns="http://ads.caiso.com/api/schema/v6">
  <dispatchBatchList>
    <DispatchBatch batchUID="DISP-59F16A91-46CA-402B-FFDC-0A090014DA8C">
      <marketID>5MinDOT</marketID>
      <batchStatus>3</batchStatus>
      <batchReceived>2006-10-13T10:55:36.0000000-04:00</batchReceived>
      <batchSent>2006-10-13T10:55:37.0000000-04:00</batchSent>
      <batchExpires>2006-10-13T10:09:45.0000000-04:00</batchExpires>
      <batchType>0</batchType>
      <startTime>2006-10-13T10:10:00.0000000-04:00</startTime>
      <dispatchMode>0</dispatchMode>
```

```
    <bindingFlag>Y</bindingFlag>
    <revisionNo>4</revisionNo>
   </DispatchBatch>
   <DispatchBatch batchUID=" DISP-59F16A91-46CA-402B-FFDC-0A090015FF5DE">
    <marketID>5MinDOT</marketID>
    <batchStatus>3</batchStatus>
    <batchReceived>2006-10-13T10:55:36.0000000-04:00</batchReceived>
    <batchSent>2006-10-13T10:55:37.0000000-04:00</batchSent>
    <batchExpires>2006-10-13T09:59:45.0000000-04:00</batchExpires>
    <batchType>0</batchType>
    <startTime>2006-10-13T10:00:00.0000000-04:00</startTime>
    <dispatchMode>0</dispatchMode>
    <bindingFlag>Y</bindingFlag>
    <revisionNo>3</revisionNo>
   </DispatchBatch>
  </dispatchBatchList>
</APIDispatchResponse>
```

DispatchBatch

```
<?xml version="1.0" encoding="UTF-8"?>
<DispatchBatch xmlns="http://ads.caiso.com/api/schema/v6"
        batchUID="DISP-59F16A91-46CA-402B-FFDC-0A090014DA8C ">
   <marketID>5MinDOT</marketID>
   <batchStatus>3</batchStatus>
   <batchReceived>2006-10-13T14:55:36Z</batchReceived>
   <batchSent>2006-10-13T14:55:37Z</batchSent>
   <batchExpires>2006-10-13T14:09:45Z</batchExpires>
   <batchType>0</batchType>
   <startTime>2006-10-13T14:10:00Z</startTime>
   <dispatchMode>0</dispatchMode>
   <bindingFlag>Y</bindingFlag>
   <revisionNo>4</revisionNo>
   <instructions>
      <instruction instructionUID="DISP-59F16A91-46CA-402B-FFDC-0A090014DA8C-I0">
         <batchUID> DISP-59F16A91-46CA-402B-FFDC-0A090014DA8C </batchUID>
         <resourceId>TEST_RESOURCE_1</resourceId>
         <startTime>2006-10-13T14:10:00Z</startTime>
         <dot>12.0</dot>
         <instructionType>0</instructionType>
         <preGoto>0.0</preGoto>
         <bidDelay>0</bidDelay>
         <acceptDot>12.0</acceptDot>
         <apiValidated>2006-10-13T14:55:50Z</apiValidated>
         <apiValidatedBy>TEST_John  Smith</apiValidatedBy>
         <revisionNumber>2</revisionNumber>
         <statusCode>3</statusCode>
         <hourlyMw>3.0</hourlyMw>
          <configurationId>TEST_RESOURCE_1_1x1</configurationId>
          <agcFlag>Y<agcFlag>
         <rmrFlag>N<rmrFlag>
         <detail>
            <instructionDetail segNo="1">
```

```
            <serviceType>SCHD</serviceType>
            <mw>3.0</mw>
         </instructionDetail>
         <instructionDetail segNo="2">
            <serviceType>SUPP</serviceType>
            <mw>0.0</mw>
         </instructionDetail>
         <instructionDetail segNo="3">
            <serviceType>RMPS</serviceType>
            <mw>3.0</mw>
         </instructionDetail>
         <instructionDetail segNo="4">
            <serviceType>SPIN</serviceType>
            <mw>0.0</mw>
         </instructionDetail>
         <instructionDetail segNo="5">
            <serviceType>NSPN</serviceType>
            <mw>3.0</mw>
         </instructionDetail>
         <instructionDetail segNo="6">
            <serviceType>HASE</serviceType>
            <mw>0.0</mw>
         </instructionDetail>
         <instructionDetail segNo="7">
            <serviceType>MSS</serviceType>
            <mw>3.0</mw>
         </instructionDetail>
         <instructionDetail segNo="8">
            <serviceType>TBD</serviceType>
            <mw>0.0</mw>
         </instructionDetail>
      </detail>
   </instruction>
   <instruction instructionUID="DISP-59F16A91-46CA-402B-FFDC-0A090014DA8C-I1">
      <batchUID>DISP-59F16A91-46CA-402B-FFDC-0A090014DA8C</batchUID>
      <resourceId> TEST_RESOURCE_2</resourceId>
      <startTime>2006-10-13T14:10:00Z</startTime>
      <dot>11.0</dot>
      <instructionType>0</instructionType>
      <preGoto>0.0</preGoto>
      <bidDelay>0</bidDelay>
      <acceptDot>11.0</acceptDot>
      <apiValidated>2006-10-13T14:55:50Z</apiValidated>
      <apiValidatedBy>TEST_John Smith</apiValidatedBy>
      <revisionNumber>2</revisionNumber>
      <statusCode>3</statusCode>
      <hourlyMw>0.0</hourlyMw>
      <detail>
         <instructionDetail segNo="1">
            <serviceType>SCHD</serviceType>
            <mw>0.0</mw>
         </instructionDetail>
```

```
            <instructionDetail segNo="2">
               <serviceType>SUPP</serviceType>
               <mw>2.0</mw>
            </instructionDetail>
            <instructionDetail segNo="3">
               <serviceType>RMPS</serviceType>
               <mw>4.0</mw>
            </instructionDetail>
            <instructionDetail segNo="4">
               <serviceType>SPIN</serviceType>
               <mw>1.0</mw>
            </instructionDetail>
            <instructionDetail segNo="5">
               <serviceType>NSPN</serviceType>
               <mw>3.0</mw>
            </instructionDetail>
            <instructionDetail segNo="6">
               <serviceType>HASE</serviceType>
               <mw>1.0</mw>
            </instructionDetail>
            <instructionDetail segNo="7">
               <serviceType>MSS</serviceType>
               <mw>0.0</mw>
            </instructionDetail>
            <instructionDetail segNo="8">
               <serviceType>TBD</serviceType>
               <mw>0.0</mw>
            </instructionDetail>
         </detail>
      </instruction>
   </instructions>
</DispatchBatch>
```

APITrajectoryResponse

```
<?xml version="1.0" encoding="utf-8"?>
<APITrajectoryResponse xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns=" http://ads.caiso.com/api/schema/v6">
 <trajectoryBatchList>
   <trajectoryBatch batchUID="TRAJ-59F16A91-46CA-402B-FFDC-0A090014DA8C">
     <batchReceived>2006-10-13T10:55:37.0000000-04:00</batchReceived>
     <bindingFlag>Y</bindingFlag>
     <batchSent>2006-10-13T10:55:37.0000000-04:00</batchSent>
     <dopList />
     <complianceList>
       <trajectoryCompliance complianceUID="TRAJ-59F16A91-46CA-402B-FFDC-0A090014DA8C-C0">
         <resourceId>TEST_RESOURCE_1</resourceId>
         <startTime>2006-10-13T09:55:00.0000000-04:00</startTime>
         <mwh>0</mwh>
         <complFlag>Y</complFlag>
         <configurationId>TEST_RESOURCE_1_1x1</configurationId>
       </trajectoryCompliance>
```

```
      <trajectoryCompliance complianceUID="TRAJ-59F16A91-46CA-402B-FFDC-0A090014DA8C-C1">
        <resourceId> TEST_RESOURCE_2</resourceId>
        <startTime>2006-10-13T09:55:00.0000000-04:00</startTime>
        <mwh>0</mwh>
        <complFlag>N</complFlag>
      </trajectoryCompliance>
    </complianceList>
  </trajectoryBatch>
  <trajectoryBatch batchUID="TRAJ-59F16A91-46CA-402B-FFDC-0A090014BB4BE">
    <batchReceived>2006-10-13T11:08:36.0000000-04:00</batchReceived>
    <batchSent>2006-10-13T11:08:36.0000000-04:00</batchSent>
    <dopList>
      <trajectoryDop dopUID="TRAJ-59F16A91-46CA-402B-FFDC-0A090014BB4BE-D0">
        <resourceId> TEST_RESOURCE_1</resourceId>
        <dop>16</dop>
        <targetTime>2006-10-13T11:08:00.0000000-04:00</targetTime>
         <sequenceNumber>1</sequenceNumber>
         <configurationId>TEST_RESOURCE_1_1x1</configurationId>
      </trajectoryDop>
      <trajectoryDop dopUID="TRAJ-59F16A91-46CA-402B-FFDC-0A090014BB4BE-D1">
        <resourceId> TEST_RESOURCE_2</resourceId>
        <dop>14</dop>
        <targetTime>2006-10-13T11:07:00.0000000-04:00</targetTime>
        <sequenceNumber>1</sequenceNumber>
      </trajectoryDop>
    </dopList>
    <complianceList />
  </trajectoryBatch>
 </trajectoryBatchList>
</APITrajectoryResponse>
```

MSSLF Request

```
<?xml version="1.0" encoding="utf-16"?>
<MSSLFRequest xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://ads.caiso.com">
 <scMSSBatchId>TestBatch1</scMSSBatchId>
 <mssLFInstructionRequests>
  <MSSLFInstructionRequest>
   <scMSSBatchId>TestBatch1</scMSSBatchId>
   <scMSSLFInstructionId>TestInstruction1</scMSSLFInstructionId>
   <resourceId>test</resourceId>
   <startTime>2006-09-25T01:00:00.0000000-04:00</startTime>
   <endTime>2006-09-25T02:00:00.0000000-04:00</endTime>
   <loadFollowingMW>100</loadFollowingMW>
  </MSSLFInstructionRequest>
 </mssLFInstructionRequests>
</MSSLFRequest>
```

MSSLF Response

```
<?xml version="1.0" encoding="UTF-8"?>
<MSSLFResponse xmlns="http://ads.caiso.com">
  <caisoMSSBatchId>8</caisoMSSBatchId>
```

```
   <scMSSBatchId>TestBatch1</scMSSBatchId>
   <mssLFInstructionResponses>
      <MSSLFInstructionResponse>
         <caisoMSSBatchId>8</caisoMSSBatchId>
         <caisoMSSLFInstructionId>7</caisoMSSLFInstructionId>
         <scMSSBatchId>TestBatch1</scMSSBatchId>
         <scMSSLFInstructionId>TestInstruction1</scMSSLFInstructionId>
         <validated>true</validated>
      </MSSLFInstructionResponse>
   </mssLFInstructionResponses>
</MSSLFResponse>
```

6. Data Dictionary

This section contains a data dictionary of those attributes and elements found in the ADS XSD document.

Element / *Attribute*	XML Schema Type	Description
DispatchBatch		
batchUID	xsd:string	Unique identifier for ADS Dispatch Batch.
marketID	xsd:string	Market Run Identifier
batchStatus	xsd:int	The batch status. One of the following integer values: 0 = New 1 = Active 2 = Time Out (hourly only). 3 = Closed 4 = Emergency Cancelled 5 = Operator Response Period (hourly only) 6 = Deferred.
batchReceived	xsd:dateTime	Time batch was received into the ADS system.
batchSent	xsd:dateTime	Time batch was set to Active (batchStatus = 1).
batchExpires	xsd:dateTime	Time the batch expires/times out or otherwise transitions the batch status from its current status to the next. This differs depending on the batch type.
batchType	xsd:int	The batch type. 0 = 5 minute dispatchable 1 = Hourly Pre-Dispatch

		2 = Commitment (Startup/Shutdown) 3 = AS Awards 4 = OOS Instructions 5 = Pre-Dispatch Hourly AS Awards
startTime	xsd:dateTime	The start time associated with the batch. This time should correspond to the start time of the interval for which the batch is targeted.
dispatchMode	xsd:int	The source system's dispatch mode: 0 = Interval 1 = Manual 2 = Contingency
bindingFlag	xsd:string	Whether or not the batch is a binding batch. For the first release of ADS MRTU, this will always be Y (binding).
revisionNo	xsd:int	A revision number used to track incremental changes to the batch. Each time an instruction within the batch changes or the batch status changes, this value is incremented.
contingencyType	xsd:string	Valid contingency type values are: RTCD = Real-Time Contingency Dispatch RTDD = Real-Time Disturbance Dispatch
pathExclusion	xsd:string	Valid path exclusion values are: NTE = Northern Ties Excluded STE = Southern Ties Excluded NSTE = Both Northern & Southern Ties Excluded
Instruction		
instructionUID	xsd:string	ADS Unique identifier for each instruction. Unique across all batches.
resourceId	xsd:string	Either the MF registerd resource ID or the dynamic Intertie

		TransactionID
configurationId	xsd:string	MSG Configuration ID if applicable.
startTime	xsd:dateTime	Start time for instruction. Aka target time for DOT instructions.
endTime	xsd:dateTime	End time for instruction. Does not apply to all instruction types.
Dot	xsd:double	DOT for 5 minute, hourly and OOS instructions.
awardMW	xsd:double	AS MW Award for Ancilary Service Award instructions.
clearedMW	xsd:double	AS Cleared MW for AS Award instructions. Cleared MW is the total MW amount; equal to awardMW + daMW + selfSchedMW.
selfSchedMW	xsd:double	AS Self Scheduled MW for AS Award instructions.
oosEnergyCode	xsd:string	OOS Energy Code for Out-Of-Stack Instructions.
asType	xsd:string	AS Type for Ancillary Service Award instructions: EN = Energy RU = Regulation Up RD = Regulation Down SR = Spinning Reserve NR = Non-Spinning Reserve RC = Residual Unit Commitment SR = Spinning Reserve LFU = Load Following Up LFD = Load Following Down
instructionType	xsd:int	Instruction Type. One of the following values: 0 = DOT (5 minute/Hourly/OOS) 1 = Min Constraint (OOS) 2 = Max Constraint (OOS) 3 = Fixed Constraint (OOS) 4 = Start up (Commitment) 5 = Shut down (Commitment) 6= Capacity Award (AS Award) 7 = MSG Transition Instruction
transitionFromConfigId	xsd:string	Defines the “from” configuration id for an MSG transition

		instruction (instructionType=7). Note that the startTime and endTime elements will define the transition period.
transitionToConfigId	xsd:string	Defines the “to” configuration id for an MSG transition instruction (instructionType=7). Note that the startTime and endTime elements will define the transition period.
agcFlag	xsd:string	The AGC flag will communicate when a resource was on AGC (Automatic Generation Control) at the start of the real-time market run. Y/N.
rmrFlag	xsd:string	The RMR flag will communicate when a resource is being incremented above the day- ahead schedule in real time, due to a RMR (Reliability Must Run) contract. Y/N.
hourlyMwThreshold	xsd:double	Accept/decline MW threshold
baseSchedule	xsd:double	EIM resource hourly base schedule
registeredIntertieFlag	xsd:string	Y: Physical resource has a MF registered resource Id. N: Physical resource does not have a MF registered resource Id.
preGoto	xsd:double	Previous DOT if available.
minAccept	xsd:double	Minimum Accept DOT for Intertie instructions. May be null if not applicable.
acceptDot	xsd:double	Accepted DOT for intertie instructions.
acceptStatus	xsd:string	Accept Status for intertie instructions.
responder	xsd:string	Responder for intertie instructions.
reasonCode	xsd:int	Reason Code for declined or partially accepted intertie instructions. Valid values are: 10 - Line down

		11 - Economic Considerations 12 - Bad Bid Submitted 13 - Unit Derate 14 - No Available Transmission 15 - Timed Out: Minimum Accepted 16 - Timed Out: Forced Decline
oprAcceptDot	xsd:double	Operator Accept DOT for intertie instructions.
oprAcceptStatus	xsd:string	Operator Accept Status for intertie instructions.
oprResponder	xsd:string	Operator responder for intertie instructions.
oprReasonCode	xsd:int	Operator Reason Code for declined or partially accepted intertie instructions.
validated	xsd:dateTime	Time instruction was validated by ADS Client. If null, the instruction was not received by an ADS Client with Primary access to the resource.
validatedBy	xsd:string	The user common name responsible for the instruction validation.
apiValidated	xsd:dateTime	Time instruction was validated by ADS API Client. If null, the instruction was not received by an ADS API Client with Primary access to the resource OR the ADS API client has not been coded to call *validateDispatchBatch*. See the pseudo code for an example.
apiValidatedBy	xsd:string	The client certificate common name responsible for the instruction validation.
revisionNumber	xsd:int	Each time an instruction is updated, its revision number is incremented.
statusCode	xsd:int	The status of the instruction. Should always correspond to the instruction's batch status. However may differ during during Emergency cancel situations.

bidDelay	xsd:int	Minutes to sync
rmrTestRequestor	xsd:string	???
InstructionDetail		
segNo	xsd:int	The unique number used to order the instruction detail elements.
serviceType	xsd:string	The service type associated with the instruction detail element.
mw	xsd:double	The MW amount.
TrajectoryBatch		
batchUID	xsd:string	Unique identifier for ADS Dispatch Batch.
batchReceived	xsd:dateTime	Time batch was received into the ADS system.
bindingFlag	xsd:string	Y/N indicating whether or not the DOPs in this batch are binding.
batchSent	xsd:dateTime	Time batch was set to Active (batchStatus = 1).
TrajectoryDOP		
dopUID	xsd:string	Unique identifier for the ADS DOP record.
resourceId	xsd:string	Resource ID from the ISO Master File.
configurationId	xsd:string	MSG Configuration ID if applicable.
Dop	xsd:double	Dispatch Operating Point (DOP)
targetTime	xsd:dateTime	Target Time for DOP.
sequenceNumber	xsd:int	A sequence value to indicate correct ordering of DOP points when there are two points within the same batch with the same targetTime. To properly order points, use the following ordering: DOP Target Time, Trajectory Batch batchReceived time, DOP sequenceNumber.
TrajectoryCompliance		
complianceUID	xsd:string	Unique identifier for the ADS Compliance record.
resourceId	xsd:string	Resource ID from the ISO Master File.
configurationId	xsd:string	MSG Configuration ID if

		applicable.
startTime	xsd:dateTime	5 minute interval start time associated with this compliance record.
Mwh	xsd:double	Expected energy in MWh's for the 5 minute interval.
complFlag	xsd:string	Flag indicating compliance with the expected energy. Y/N.
MSSLFRequest		
scMSSBatchId	xsd:String	Unique identifier assigned by the SC for this batch of MSS Requests.
mssLFInstructionRequests	0..n xsd:MSSLFInstructionRequest	MSS Load Following Instruction Requests.
MSSLFInstructionRequest		
scMSSBatchId	xsd:String	Unique identifier assigned by the SC for this batch of MSS requests.
scMSSLFInstructionId	xsd:String	Unique identifier assigned by the SC for this Load Following Instruction request.
resourceId	xsd:String	Resource Id. ADS will verify that SC has sufficient privileges to submit on behalf of resource.
startTime	xsd:dateTime	Start time for LF MW.
endTime	xsd:dateTime	End time for LF MW.
loadFollowingMW	xsd:double	Load Following MW Positive for follow-up and negative for follow-down.
MSSLFResponse		
caisoMSSBatchId	xsd:String	Unique identifier assigned by the CAISO for this batch of MSS requests. Used internally by the ISO to track these requests.
scMSSBatchId	xsd:String	Unique identifier assigned by the SC for this batch of MSS requests passed back to SC so they can use either the newly assigned ISO id or their own.
MSSLFInstructionResponse		
caisoMSSBatchId	xsd:String	Unique identifier assigned by the CAISO for this batch of MSS requests. Used internally by the

		ISO to track these requests.
scMSSBatchId	xsd:String	Unique identifier assigned by the SC for this batch of MSS requests passed back to the SC for their internal coding.
caisoMSSLFInstructionId	xsd:String	Unique identifier assigned by CAISO for this Load Following Instruction request. Used internally by the ISO to track this request.
scMSSLFInstructionId	xsd:String	Unique identifier assigned by the SC for this Load Following Instruction request. Passed back to the SC for their internal coding.
validated	xsd: boolean	Flag specifying whether or not the submitted request was valid.

7. ADS API Transition Support

As of version 4.0 of the ADS API, we will be maintaining two versions of the ADS API to allow for an easy transition from the existing API to the latest release.

Version 5.0 has been deprecated and is no longer supported.

The supported ADS APIs are available at the following URIs:

https://adssta.caiso.com:447/ADS/APIWebService/v6
https://adssta.caiso.com:447/ADS/APIResponseWebService/v6

https://adssta.caiso.com:447/ADS/APIWebService/v7
https://adssta.caiso.com:447/ADS/APIResponseWebService/v7

8. Change Log

2009-12-7

- Added MSG Instruction data
- Updated example XML with GUIDs instead of integer UID values
- WSDL and XSD namespace changes

2008-09-17

- Added bindingFlag to TrajectoryBatch.
- Added sequenceNumber to TrajectoryDOP.

2008-08-22

- Removed DA MW.
- Added ADS API Response Service.

2006-02-13

- Removed **getBatchListByDate** method. Method is expensive. For historical queries, use the Query Tool. See the pseudo code for preferred real time API usage.
- Changed all references to **BatchSeq** to **BatchUID**. BatchSeq implies the numbers are always sequential. This may not be the case.
- **GetInstructionSet** was renamed to **GetDispatchBatch**.
- The return payloads have been changed for MRTU. New XSD attached.
- **ValidateDispatchBatch** has been added. This method may be used to programmatically confirm the receipt of a batch via the ADS API.

2006-05-31

- Added **submitMSSLFRequest**.
- Added AS Award Type. For now, the first two enumerations are reserved.

2006-10-12

- Added **XSD** to document in Section 4.
- Added Section 4.1 for Sample XML
- *New Section 4, which moved Data Dictionary and Change Log to Sections 5 and 6.*

2006-10-30

- Added Sample XML
- Added Batch Type for OOS Instructions

2007-02-19

- Removed AS Award Type
- Added new AS Breakdown fields:
    - awardMW (renamed asMW to awardMW)
    - clearedMW
    - selfProvidedMW

    - dayAheadMW
- Added QLFC to AS Type list.
- Updated XSD and Sample XML.

2007-02-22

- Removed Price from Instruction Detail.

2007-04-04

- Updated ads.caiso.com.xsd to the new version.
- Added new possible values to DispatchBatch.batchStatus.
- Changed DispatchBatch.bindingFlag from unsignedInt to string.
- Changed AS breakdown fields:
    - from Instruction.selfProvidedMW to Instruction.selfSchedMW (to reflect name in ResourceAwards.xsd v5.1)
    - from Instruction.dayAheadMW to instruction.daMW (to reflect name in ADS database)
- Changed Instruction.asType to string and filled in all possible values.

2007-10-05

- Removed passIndicator from ADS xsd to make consistent with published ads.caiso.com-release-4.1.7.xsd

2007-11-05

- Changed UID data types from xsd:int to xsd:string.