



Whitepaper

B2B Implementation Over AS2 in Boomi



www.jadeglobal.com



info@jadeglobal.com



+1-408-899-7200

Contents

Introduction	02
Challenges	02
AS2 Message Format	03
Requirements for AS2 implementation	05
Implementation of AS2 in Boomi	06
Monitoring Transactions:	10
AS2 Benefits	11



Introduction



This document describes AS2 message format, configuration and steps to implement B2B integration solution using AS2 protocol in Boomi.

AS2: AS2 (Applicability Statement 2) is a popular protocol to transmit data securely and reliably over the internet. AS2 supports the encryption of messages that are then exchanged with trading partners and vendors via HTTPS. These messages are built using the S/MIME format. Mainly, As2 is used to exchange different data such as EDI, XML, or Custom documents between partners.

B2B: Business-to-business (B2B) integration is the automation of business processes and communication between two or more organizations. It allows them to work and trade more effectively with their customers, suppliers and business partners by automating key business processes.

Challenges

We face below common challenges while implementing AS2.

- Identifying required info between partners to setup AS2 connection
- Processing of multiple data formats with same partner using single AS2 ID in Boomi

AS2 Message Format

The AS2 layer is wrapped in the HTTP/S protocol as shown below. AS2 protocol will be having Identification info, AS2 message packing details, payload and MDN details which are wrapped inside the HTTP. We can configure separate Http server certificates which will be used by the server when sending MDN back to the AS2 Client application. The server certificate is not the same thing as the AS2 certificate. Someone could use the same certificate for both purposes but that is very uncommon.

Following are the AS2 protocol layer details:

Identification data:

- As2 ID: AS2 system Identifier (ID) is a unique value which agreed between partners to identify who is sending the data.
- AS2 Sender cert: which is a server public key shared with a trading partner to encrypt the data before sending to Partner
- AS2 Receiver cert: is a client public key shared with Partner and used to encrypt the data before sending to the client application

AS2 Message Packaging:

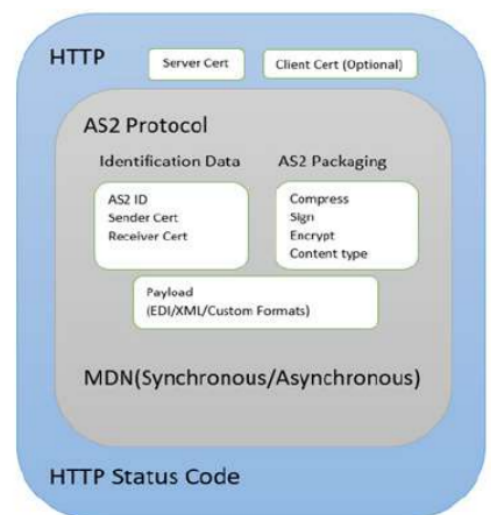
- Sign Message: Message will be signed using the Signature Secure Hash Algorithm and then send to partner
- Encrypt Message: Message will be encrypted with Encryption Algorithm and then send to partner
- Compress Message: Message will be compressed and then send to partner

Payload:

The payload is the actual business data which will be exchanged with partner in any format like EDI/XML/custom Formats

MDN:

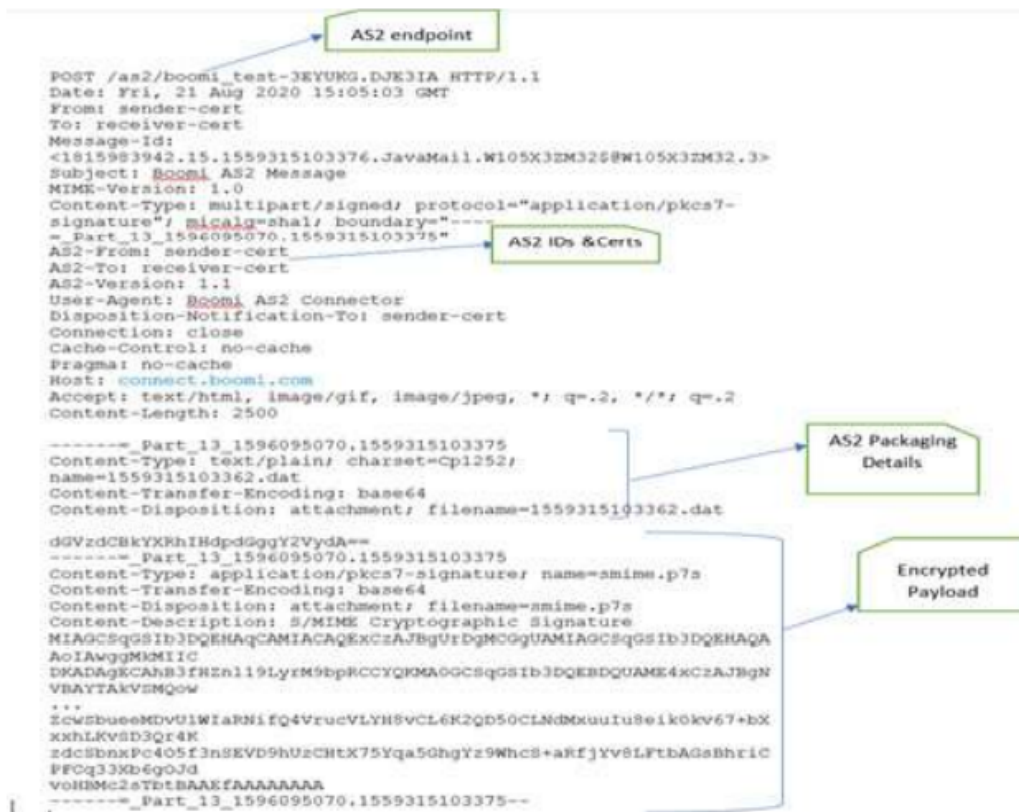
MDN (Message Disposition Notification) is an electronic acknowledgment of receipt that is sent to the sender via AS2 after an electronic message has been sent. This acknowledgment of reception confirms that the message has been transmitted completely and received by the partner.



Contact us to know more about
Our Industry Solution



The AS2 layer will have AS2 specific headers and content. Here's an example of an HTTP request using AS2 with certificates.



Here's an example of an MDN saying the AS2 message has been received and processed.

```
Date: Fri, 21 Aug 2020 15:20:40 GMT
From: receiver-cert
To: sender-cert
Message-Id:
<822688681.70057.1559316040813.JavaMail.clouduser@966643-
ordatom09.ord.booml.com>
Subject: Booml AS2 Message
MIME-Version: 1.0
Content-Type: multipart/report; report-type=disposition-
notification;
boundary="-----_Part_70056_119245328.1559316040813"
AS2-From: receiver-cert
AS2-To: sender-cert
AS2-Version: 1.1
User-Agent: Booml AS2 Connector

-----_Part_70056_119245328.1559316040813
Content-Type: text/plain
Content-Transfer-Encoding: binary

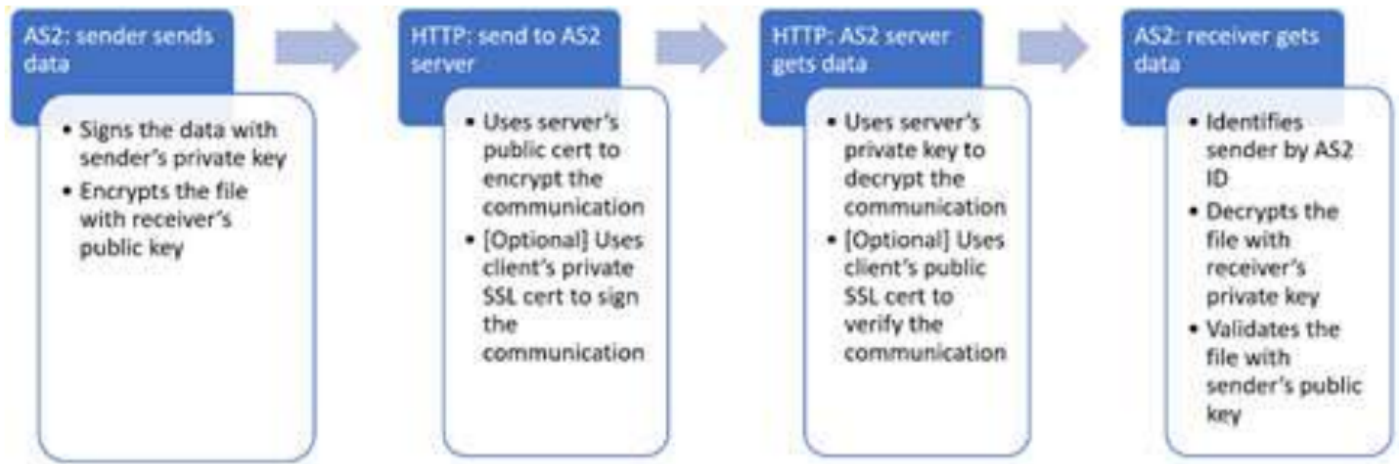
The message sent on Fri, 31 May 2019 15:20:39 GMT
From [sender-cert] to [receiver-cert]
with subject [Booml AS2 Message] and
message ID of
[<1374172721.5.1559316039880.JavaMail.clouduser@localhost.1>]
has been [processed]

-----_Part_70056_119245328.1559316040813
Content-Type: message/disposition-notification
Content-Transfer-Encoding: binary

Reporting-UA: receiver-cert (booml.server)
Original-Recipient: rfc822;receiver-cert
Final-Recipient: rfc822;receiver-cert
Original-Message-ID:
<1374172721.5.1559316039880.JavaMail.clouduser@localhost.1>
Disposition: automatic-action/MDN-sent-automatically; processed
Received-Content-MIC: gahLlFgmHDigwJQgD3jzTrFpaRA=, sha1

-----_Part_70056_119245328.1559316040813--
```

Below is the diagram shows how the components are generally used when sending an AS2 message:v



The data is signed using a private key to ensure the sender's identity as the creator of the document. The public key of the sender is used by the receiver to verify the sender. The data is encrypted using a public key so only the owner of the private key (which should be the receiver) will be able to decrypt the document.

Requirements for AS2 implementation

Sender Need below info from the receiver:

- URL of the receiver's AS2 server
- Public SSL server certificate used by receiver's AS2 server, if applicable (if the URL is HTTP, then no need for a certificate)
- Receiver's AS2 ID
- Receiver's public AS2 certificate
- Any specific algorithms for signing and encrypting and any specific MDN options

Receiver Need below info from Sender:

- URL of your AS2 server to send back the MDN
- Public SSL server certificate of your AS2 server
- Your AS2 ID
- Your public AS2 certificate

Implementation of AS2 in Boomi

Boomi AtomSphere is an on-demand multi-tenant cloud integration platform for connecting cloud and on-premises applications and data. Below explained the implementation steps which used to processing of Inbound and outbound B2B transactions over AS2 in Boomi.

Inbound Transactions:

We have below two methods to process Inbound documents. In both the methods, if the partner uses the single AS2 ID to send multiple documents types, then we need to create a gateway service which will be the first process to accept all the documents and route to the internal process based on the content-type and transaction type since Boomi will not allow having multiple listeners with same AS2 ID.

- AS2 shared server Listener
- Trading Partner AS2 Start Shapes

When a process containing these connectors is deployed to an Atom which are listen-only connectors that accept AS2 requests in real-time and initiate Boomi AtomSphere processes based on the AS2 ids and configuration made in the AS2 Shared Server operation/Trading Partner AS2 shape.

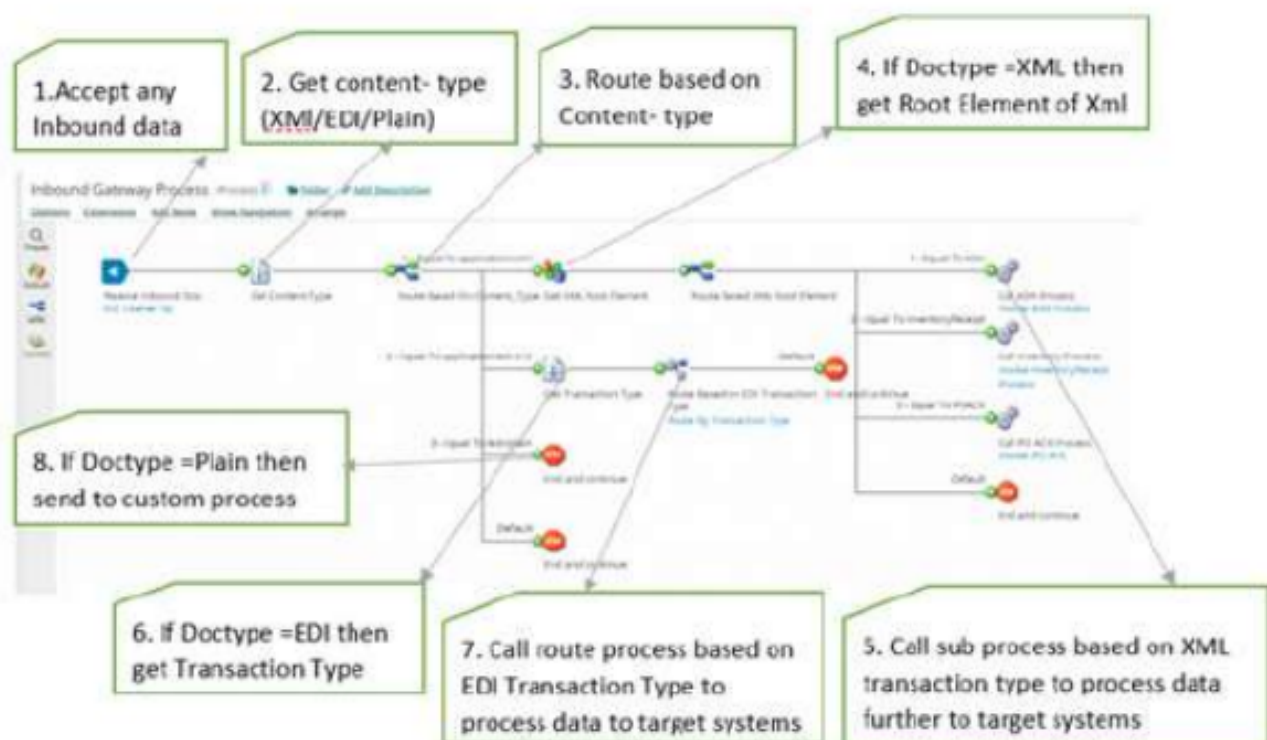
Once the Inbound process is deployed to an on-premise server, the process is ready to accept inbound transactions via AS2 endpoint. The endpoint can be found in the shared webserver of atom management and this endpoint can be shared with the partner to post the data.

Following are the examples of AS2 endpoint for on-premise servers and cloud atom.

On-Premise AS2 End point - http://<shared_web_server_URL>:<port>/as2.

Cloud atom AS2 End Point - https://c01-usa-east-integrate.boomi.com/as2/<atom_instance_ID>;.

Implementation approach with AS2 Listener process



Code snippet to get Root element from XML

```
import java.util.Properties;
import java.io.InputStream;
import org.jdom.Document;
import org.jdom.Element;
import org.jdom.input.SAXBuilder;
import org.xml.sax.SAXException;
import com.boomi.execution.ExecutionUtil;

for( int i = 0; i < dataContext.getDataCount(); i++ ) {
    InputStream is = dataContext.getStream(i);
    Properties props = dataContext.getProperties(i);
    SAXBuilder builder = new SAXBuilder();
    Document doc = builder.build(is);
    Element rootNode = doc.getRootElement();
    String rootElementName=rootNode.getName()
    if(rootElementName=="InventoryReceiptNote"){
        ExecutionUtil.setDynamicProcessProperty("Inbound_MessageTypeIDP","InventoryReceiptNote", false);
    }
    else if(rootElementName=="ASN"){
        ExecutionUtil.setDynamicProcessProperty("Inbound_MessageTypeIDP","ASN", false);
    }
    else if(rootElementName=="POACK"){
        ExecutionUtil.setDynamicProcessProperty("Inbound_MessageTypeIDP","POACK", false);
    }
    else { ExecutionUtil.setDynamicProcessProperty("Inbound_MessageTypeIDP","Unsupported XML message received
from partner, MessageType is - "+rootElementName, false); }
    dataContext.storeStream(is, props);
}
```

AS2 Listener Operation configuration:

server listen - AS2 Server Operation ? Folder A

AS2 Info Archiving Tracking

Connector Action Listen

AS2 From

AS2 From ID sender-ID

Signing Certificate Q sender-AS2-public

Client SSL Certificate i Q Choose...

Reject Duplicate Messages ☐ i

Use IP Filtering ☐ i

AS2 To

AS2 To ID receiver-ID

MDN Signing Certificate i Q receiver-AS2-private

Decryption Certificate i Q receiver-AS2-private

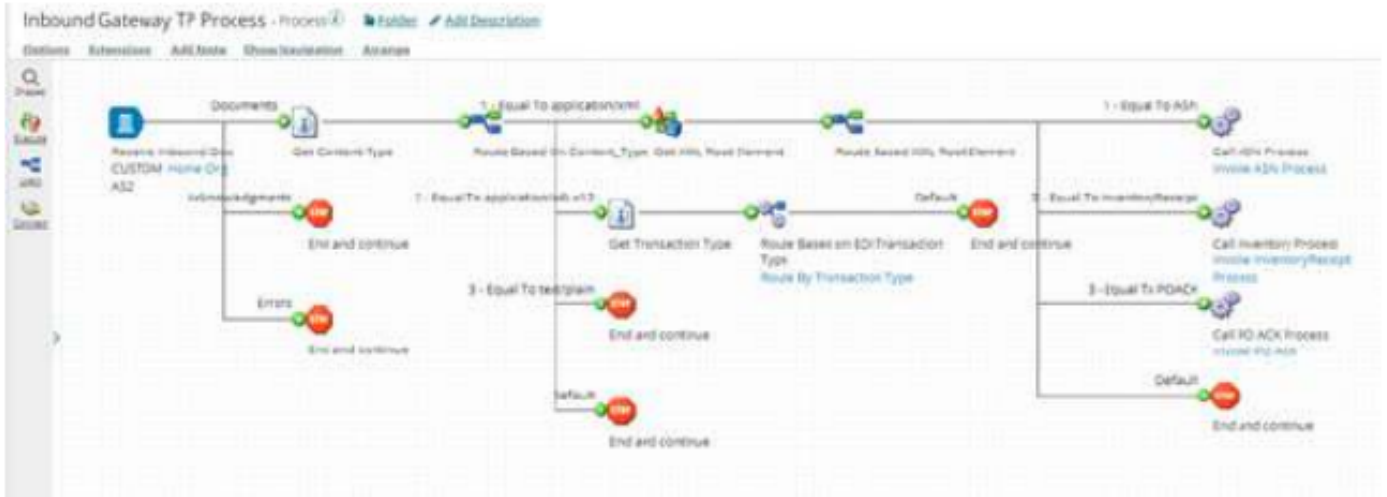
AS2 Shared Server operation configuration

The AS2 From ID is the trading partner's AS2 ID. The signing certificate is the sender's public AS2 certificate.

The AS2 To ID is the Home organization (the receiver's) AS2 ID. The corresponding certificates are Home organization private AS2 certificates.

Implementation approach with TP start shape

This process will start with the TP shape where we will configure My company profile and your trading parent profile. In the profile, we will configure the AS2 and certification details which are required for AS2 communication. The remaining Internal functionality will be the same AS2 listener process which identifies the documents and routing to the required subprocess based on the transaction type.



The 'Edit AS2 Configuration' dialog box is shown. It contains several sections: 'General' with fields for 'Host' and 'Port'; 'AS2 Settings' with checkboxes for 'Enable AS2', 'AS2 Encryption', 'AS2 Signature', and 'AS2 Certificate'; 'Add New Settings' with a 'Name' field; 'Default Partner Settings' with fields for 'Host', 'Port', 'AS2 Encryption', 'AS2 Signature', 'AS2 Certificate', and 'AS2 Certificate Type'; 'Default Partner Information' with fields for 'Host', 'Port', 'AS2 Encryption', 'AS2 Signature', 'AS2 Certificate', 'AS2 Certificate Type', 'Report Duplicate Messages', and 'Use All Fields'; 'Default Partner Options' with a 'Transaction Authentication' dropdown; 'Message Options' with fields for 'Message ID', 'Content Type', 'In Message', 'Signature', 'Signature Algorithm', 'Encryption Algorithm', and 'Compression Algorithm'; 'Message Encryption/Authentication (PKI) Options' with fields for 'Recipient ID', 'Signature', 'Signature Algorithm', and 'Encryption Algorithm'; and 'Data Processing Options' with a 'Data' dropdown.

Sometimes the trading partner will request that an MDN be sent back either synchronously or asynchronously. Your trading partner will need to include this request for MDN in the original message when they send the data to your AS2 listener process or Trading Partner process. AtomSphere will interpret their request and if they have configured it properly, an MDN will get sent back.

If they desire to have the MDN sent back to a different URL or port, they will need to specify that in the Disposition-Notification-To header of the original message. Message Disposition Notifications (MDN) are tracked at the AS2-level under Process Reporting, for specific documents, to confirm that the communication exchange between AS2 partners was valid or invalid. However, the full message captured by the AS2 server is stored at the Atom-level. You should be able to view the MDN's in the work/AS2/inbound or outbound directories in your AtomSphere installation directory.

Outbound Transactions:

We can use the AS2 client connector or Trading partner to send the outbound documents to partners. Mostly, we use the trading partner to send EDI documents as TP will support to configure ISA and GS segments so EDI segment values will be configurable for each environment without changing in the code.

Here's a basic process to send any data using the AS2 client:



The AS2 client connection will define the AS2 server information. The AS2 server URL will go into the URL field. If the server has an SSL certificate, we'll use the public server certificate here

prod cloud AS2 URL - AS2 Connector ? Folder Add Description

Settings

URL	<input type="text" value="https://connect.boomi.com/as2/boomi_test-3GTWKG.DJESIA"/>
SSL Certificate	<input type="text" value="server.cert"/> 🔍 ✎ 🗑
Client SSL Certificate	<input type="text" value="Choose..."/> 🔍 ✎ 🗑
Authentication Type	<div>None ⓘ</div>

The AS2 client operation will define AS2 specific information. The AS2 From ID is Home organization(sender) AS2 ID. The signing certificate will be Home organization (sender) certificate. The AS2 To ID is the receiver's AS2 ID (trading partner). The encrypting and MDN signing certificate is the receiver's public AS2 certificates. They could be different and not the same public key.

send operation with cert - AS2 Operation ? Folder

AS2 Info Options Archiving Tracking Caching

AS2 From

AS2 From ID	<input type="text" value="sender-ID"/>
Signing Certificate	<input type="text" value="Sender-private"/> 🔍 ✎ 🗑

AS2 To

AS2 To ID	<input type="text" value="receiver-ID"/>
Encrypting Certificate	<input type="text" value="AS2-receiver-public"/> 🔍 ✎ 🗑
MDN Signing Certificate	<input type="text" value="AS2-receiver-public"/> 🔍 ✎ 🗑

send operation with cert - AS2 Operation ? Folder Add Description

AS2 Info Options Archiving Tracking Caching

Connector Action

Send ▼

Message Options

Subject	<input type="text"/>
Content Type	<div>txt/plain ▼</div>
Sign Message	<input checked="" type="checkbox"/>
Signature Secure Hash Algorithm	<div>SHA-1 ▼</div>
Encrypt Message	<input checked="" type="checkbox"/>
Encryption Algorithm	<div>Triple DES ▼</div>
Compress Message	<input type="checkbox"/>

Message Disposition Notification (MDN) Options

Request MDN	<input checked="" type="checkbox"/>
Signed	<input type="checkbox"/>
Signature Secure Hash Algorithm	<div>SHA-1 ▼</div>
Sync/Async	<div><input checked="" type="radio"/> Synchronous <input type="radio"/> Asynchronous</div>
External URL	<input type="text"/>

There is an options tab in the operation to control AS2 message and MDN options. we'll see the options for signing, encrypting, and compressing the AS2 message here. The receiver might request a certain algorithm for signing and encrypting so we can adjust to the dropdown. The options for the MDN options allow us to request the MDN synchronously and asynchronously and request the receiver to sign it or not with an algorithm.

Contact us to know more about
Our Industry Solution



Monitoring Transactions



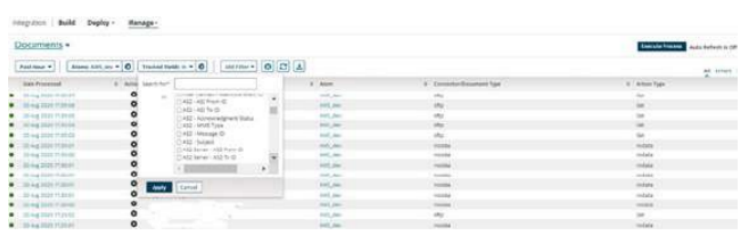
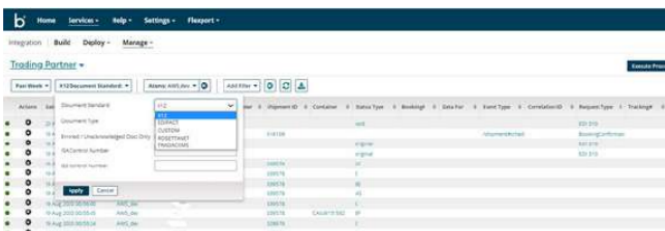
We can track all the transactions which are sent to partner and received from partner will be monitored through Process reporting. Documents will be tracked using standard tracked fields or customized fields which can be defined using the unique fields of the interface like, PO number Or shipment ID.

The following three options are available in process reporting to monitor the transactions which are processed in Boomi:

Executions: In the execution, it will list all the transactions processed in Boomi and available options to filter the transactions based on filter conditions such as process name, execution ID & date range

Documents: can use tracked filed to find the particular transaction is received or not with following standard tracked fields AS2 from ID, AS2 to ID, Subject, Message-ID, File Name, MIME Type

Trading partners: Process which use the trading partner shape can monitor the transactions with a filter condition



Message Disposition Notifications (MDN) are tracked at the AS2-level under Process Reporting, for specific documents, to confirm that the communication exchange between AS2 partners was valid or invalid. However, the full message captured by the AS2 server is stored at the Atom-level. we should be able to view the MDN's in the work/AS2/inbound or outbound directories in your Atomsphere installation directory.



AS2 Benefits



AS 2 will provide following benefits

Security: Provides End-to-end data encryption and SSL encryption and hashing

Non-repudiation: Confirmations and MDN

Flexibility: Large file transfer & Any data type including non-EDI payloads

**Contact us to know more about
Our Industry Solution**



Join in

Partner with Jade

Let us guide your path amidst the clouds and help you build the future enterprise.

Be an expert

Become a master in the interplay of domains and industries. Drive higher benchmarks of success.

A leader

Knowledge powerhouse we all hold together-clients, partners and Jade experts.

Contact us to know more about
Our Industry Solution



For more information

USA | CANADA | UK | AUSTRIA | INDIA



www.jadeglobal.com



info@jadeglobal.com



+1-408-899-7200

1731 Technology Drive, Suite 350 San Jose, CA 95110

Copyright 2022 Jade Global. All Rights Reserved. Jade and its logo are trademarks of Jade Global.