EUROPEAN COMMISSION

DIGIT
Connecting Europe Facility

Guidelines to Access Point PMode Configuration for PNRGOV

Version [0.02]

Date: 22-08-2019

Document Approver(s):

Approver Name	Role
Adrien FERIAL	Product Owner

Document Reviewers:

Reviewer Name	Role
Maarten DANIELS	Test Manager
Cosmin BACIU	Lead Developer

Summary of Changes:

Version	Date	Created by	Short Description of Changes
0.01	14/05/2019	Vlad VEDUTA	Initial version
0.02	27/05/2017	Vlad VEDUTA	Updated the service , action and party naming sections to reflect the changes suggested by PIM

Table of Contents

1. INTRODUCTION

1.1. Purpose

This document describes the eDelivery PMODE configuration guidelines addressed to the airlines companies and government authorities that exchange PNRGOV messages over eDelivery AS4.

1.2. References

Ref.	Document	Content outline
[REF1]	PNRGOV-EDIFACT-Implementation-Guide-13_1.pdf	The purpose of this document is to describe the recommended usage of the Passenger and Airport Data Interchange Standards PNRGOV EDIFACT Message Standards. These messages are intended to facilitate the exchange of data relevant to government requirements on PNR data and Airlines reservation systems. This document was developed, and will be maintained, by the IATA/ATA PNRGOV Sub-Group in coordination with the Passenger and Airport Data Interchange Standards Reservations Sub-Group.
[REF2]	AS4 Profile of ebMS 3.0 Version 1.0	While ebMS 3.0 represents a leap forward in reducing the complexity of Web Services B2B messaging, the specification still contains numerous options and comprehensive alternatives for addressing a variety of scenarios for exchanging data over a Web Services platform. The AS4 profile of the ebMS 3.0 specification has been developed in order to bring continuity to the principles and simplicity that made AS2 successful, while adding better compliance to Web Services standards, and features such as message pulling capability and a built-in Receipt mechanism. Using ebMS 3.0 as a base, a subset of functionality is defined along with implementation guidelines adopted based on the “just-enough” design principles and AS2 functional requirements to trim down ebMS 3.0 into a more simplified and AS2-like specification for Web Services B2B messaging. In addition to addressing EDIINT requirements, a Minimal Client conformance profile is provided that addresses lower-end exchange requirements. This document defines the AS4 profile as a combination of conformance profiles that concern an implementation capability, and of a usage profile that concerns how to use this implementation. A couple of variants are defined for the AS4 conformance profile - the AS4 ebHandler profile, the AS4 Light Client profile and the AS4 Minimal Client profile - which reflect different endpoint capabilities.
[REF3]	eDelivery AS4 -	The eDelivery AS4 Profile is a profile of the ebMS3 and AS4 OASIS Standards. It has provisions for use in four-corner topologies, but

Ref.	Document	Content outline
	1.14	can also be used in point-to-point exchanges. This specifications profile can be implemented using open source or closed source commercial software products compliant with these standards. It is designed to support both One Way and Two Ways (Request-Response) exchanges. The profiling is heavily based on the ENTSOG (the European Network of Transmission System Operators (TSO) for Gas) AS4 profile for TSOs and on e-CODEX specifications.
[REF4]	IATA Airline Code	IATA Airline Code is a 2-letter code that identifies airlines.

1.3. Acronyms

Acronym	Description
Access Point	An AS4 AP is referred to as an Access Point. An Access Point can send messages to other Access Points, and receive messages from other Access Points. In a PMode file an Access Point is identified as a party.
AS4	Applicability Statement 4. AS4 is a Conformance Profile of the OASIS ebMS 3.0 specification, and represents an open standard for the secure and payload-agnostic exchange of Business-to-business documents using web services.
DES	Data Encryption Standard
ebXML	Electronic Business XML. Project to use XML to standardise the secure exchange of business data.
Endpoint	A party which receives messages is an endpoint.
MEP	Message Exchange Pattern A Message Exchange Pattern describes the pattern of messages required by a communications protocol to establish or use a communication channel.
MPC	Message Partition Channel. Different MPCs can exist, each defined as a Container. An MPC allows the flow of messages from a Sending MSH to a Receiving MSH to be partitioned into several flows, each of which is controlled separately. An MPC also allows flows from several Sending MSHs to be merged into a single unique flow that will be treated as such by a Receiving MSH.
MSH	Message Service Handler The MSH is an entity that is able to generate or process messages that conform to the ebMS specification, and which act in at least one of the two ebMS roles: Sender and Receiver. In terms of SOAP processing, an MSH is either a SOAP processor or a chain of SOAP processors. In either case, an MSH has to be able to understand the eb:Messaging header (qualified with the ebMS namespace).
OASIS	Advancing open standards for the information society Non-profit, international consortium creating interoperable industry specifications based on public standards.
PMode	Processing Mode

	A collection of parameters that determine how User Messages are exchanged between a pair of APs with respect to Quality of Service, Transmission Mode and Error Handling.
X.509	Cryptography standard for PKI.
W3C	World Wide Web Consortium The main international standards organisation for the World Wide Web.
IATA	The International Air Transport Association (IATA) supports aviation with global standards for airline safety, security, efficiency and sustainability.
PNR	Passenger name record
EDIFACT	Electronic Data Interchange For Administration, Commerce and Transport. EDIFACT is accepted as the international EDI standard that has been adopted by organisations wishing to trade in a global context. A standard set of syntax rules have been ratified by the United Nations.
PNRGOV	PNR for GOVERNMENT
PADIS	Passenger and Airport Data Interchange Standards

#	Parameter	Type[1]	Description	Value in Profile
1	PMode.ID : optional	n/a	Identifier for the PMode. e.g. the name of the business transaction.	No guidelines
2	PMode.Agreement : optional	n/a	The MEPs which the message exchanges will conform to.	No guidelines
3	PMode.MEP: required	S	Message Exchange Pattern: • **Two-Way MEP** governs the exchange of two User Message Units in opposite directions. • **One-Way MEP** governs the exchange of a single User Message Unit unrelated to other User Messages.	Chosen value: • **Two-Way MEP** Used with *RefToMessageId*.
4	PMode.MEPBinding: required	S	A Binding is the transport channel binding URL assigned to the MEP: • Push • Push-and-Push	No guidelines
4	PMode.Initiator.Party (optional if PMode.Responder is present)	S	Qualifies the party initiating the MEP. Specifies an identifier (and possibly a *type* attribute value)	Chosen type/values: for **airline parties** (valid also connecting via a AS4 service provider) • **type:** urn:oasis:names:tc:ebcore:partyid-type:unregistered:iata:airline **value: <2 letters IATA code>**

[1] Static (S), (E) Enforced, (n/a) Not profiled or unused

				for government authorities (valid also when connecting via a AS4 service provider) • **type:** urn:oasis:names:tc:ebcore:partyid-type:unregistered:national_authority **value: <authority name>** This type/value is the name for the production
5	PMode.Initiator.Role	E	The Message Producer carries out the initiator role, which is the role of the party sending the first message of the MEP.	"Carrier" for message initiated by a carrier and "State" for messages initiated by a state.
6	PMode.Responder.Party : optional if PMode.Initiator is present.	E	Qualifies the party responding to the initiator party in the MEP. Specifies an identifier (and possibly a *type* attribute value) compliant with e-SENS Addressing ABB.**Error! Bookmark not defined.**	Chosen type/values: for **airline parties** (valid also connecting via a AS4 service provider) • **type:** urn:oasis:names:tc:ebcore:partyid-type:unregistered:iata:airline **value: <2 letters IATA code>** for government authorities (valid also when connecting via a AS4 service provider) • **type:** urn:oasis:names:tc:ebcore:partyidtype:unregistered:national_authority **value: <authority name>** This type/value is the name for the production
7	PMode.Responder.Role	E	The Message Consumer carries out the responder role.	"Carrier" for response message from a carrier and "State" for response messages from a state.
8	PMode[1].Protocol.	S	Endpoint URL of the Receiver MSH (or Receiver	No Guidelines

	Address: required		Party) to which Messages under the PMode leg are to be sent.	
9	PMode[1].Protocol.SOAPVersion	E	The SOAP version to be used.	Chosen value: **1.2**
10	PMode[1].BusinessInfo.Service	S	**Name of the service** to which the User message is intended to be **delivered**. It identifies a set of related business transactions or other message exchanges in the context of a business process or use case. .	Services on both initiating and responding parties: Chosen type/value: • **type: empty** • **values:** urn:iata::service:Supply urn:iata: service:Request Ping service on both initiating and responding APs: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/service
11	PMode[1].BusinessInfo.Action	S	Name of the action(s) the User message is intended to invoke. Identifies the different types of business transactions or other message exchanges in the context of an *Identified Service*. This may be an identifier of a document type, if the exchange of a document of that type unambiguously identifies the purpose and requested action in the context of the Service.	Actions on both initiating and responding: **values:** urn:iata:pnrgov:action:SupplyTravelData urn:iata:pnrgov:action:RequestTravelData urn:iata:pnrgov:action:AcknowledgeTravelData urn:iata:pnrgov:action:RejectTravelData Ping action on both initiating and responding APs: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/test

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Valid “Service – Action” pairs for the PMode LEGs:
<table>
<tr><th>Service</th><th>Action</th></tr>
<tr><td>urn:iata:service:Supply</td><td>urn:iata:pnrgov:action:SupplyTravelData
urn:iata:pnrgov:action:AcknowledgeTravelData
urn:iata:pnrgov:action:RejectTravelData</td></tr>
<tr><td>urn:iata:service:Request</td><td>urn:iata:pnrgov:action:SupplyTravelData
urn:iata:pnrgov:action:RequestTravelData
urn:iata:pnrgov:action:AcknowledgeTravelData
urn:iata:pnrgov:action:RejectTravelData</td></tr>
<tr><td>http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/service</td><td>http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/test</td></tr>
</table>
</td>
</tr>
<tr>
<td>12</td>
<td>PMode[1].BusinessInfo.Properties</td>
<td>S</td>
<td>A list of properties.
A property is a data structure that consists of four values within the User message:
1. The property name, which can be used as an identifier of the property.
e.g. A required property named messagetype can be noted as:
Properties[messagetype].required="true"
2. The property description.
3. The property data type.
4. Boolean parameter indicating (if true) that the property is mandatory. Otherwise (if false) the property is optional.

In four corner exchanges, there is a mandatory inclusion of originalSender and finalRecipient and</td>
<td>1. originalSender:
• This property addresses the original (end entity) sender party.
• String
• Mandatory

2. finalRecipient
• This property addresses the final (end entity) recipient party.
• String
• Mandatory</td>
</tr>
</table>

			optional inclusion of *trackingIdentifier.*	
13	PMode[1]. BusinessInfo.MPC	E	Identifier of the MPC (Message Partition Channel) to which the message is assigned.	Default value: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/defaultMPC
14	PMode[1].BusinessInfo.PayloadProfile	S	A list of payload parts. A payload part is a data structure that consists of five properties: 1. **name** (or Content-ID) that is the part identifier, and can be used as an index in the notation PayloadProfile[]; 2. **MIME data type** (text/XML, application/PDF, etc.); 3. name of the applicable **XML Schema** file if the MIME data type is text/XML; 4. **maximum size** in kilobytes; 5. Boolean parameter indicating (if *true*) that – within the User message – the **part is mandatory**. Otherwise (if *false*) the part is optional. The message payload(s) must match this profile. Optionally a maximum size for the total AS4 message, including all payloads, can be chosen.	The message body will be empty. There will be exactly one attachment to contain the PNRGOV-EDIFACT message 1. **cid:edifactMessage** 2. MIME data type **application/EDIFACT** n/a 3. n/a 4. n/a 5. **TRUE**
15	PMode[1].ErrorHandling.Report.SenderErrorsTo	n/a	The address – or comma-separated list of addresses – to which to send errors generated by the MSH that was trying to send the message in error.	No guidelines
16	PMode[1].ErrorHandling.Report.ReceiverErrorsTo	n/a	The address– or comma-separated list of addresses – to which to send ebMS errors generated by the MSH that receives the message in error.	No guidelines

			e.g. This may be the address of the MSH sending the message in error.	
17	PMode[1].ErrorHandling.Report.AsResponse	S	Boolean parameter indicating (if *true*) that errors generated from receiving a message in error are sent over the back-channel of the underlying protocol associated with the message in error.	Chosen value: **TRUE**
18	PMode[1].ErrorHandling.Report.ProcessErrorNotifyConsumer	S	Boolean parameter indicating (if *true*) that the Consumer (application/party) of a User Message matching the PMode should be notified when an error occurs in the Receiving MSH.	Chosen value: **TRUE**
19	PMode[1].ErrorHandling.Report.DeliveryFailuresNotifyProducer	S	Boolean parameter indicating (if *true*) that – during processing of the User Message to be sent – the Producer (application/party) of a User Message matching the PMode should be notified when an error occurs in the Sending MSH.	Chosen value: **TRUE**
20	PMode[1].ErrorHandling.Report.BusinesssErrorNotifyProducer	S	Boolean parameter indicating (if *true*) that – the backend sender is notified if the delivery of a message to fails, either at the C3 side or while sending the message. In the latter case, the message fails to be sent at C2 side, the backend is notified	Chosen value: **TRUE**
21	PMode[1].Reliability	n/a	Reliability contracts – or references to them – that will govern the reliability of messages exchanged.	No guidelines
22	PMode[1].Security.WSSversion	E	The value represents the version of WS-Security to be used, and it has one possible value: 1.1 Chosen value: **1.1.1**	Enforced: All security values are enforced by eDelivery AS4 profile
23	PMode[1].Security.X509.Sign	E	Boolean parameter indicating (if *true*) that the message is to be signed. Chosen value: **TRUE**	Enforced: All security values are enforced by eDelivery AS4 profile
24	PMode[1].Security.X509.Signature.	E	Public certificate to use when verifying signed data:	Enforced: All security values are enforced by eDelivery AS4

	Certificate		Signing Certificate of the Sender	profile
25	PMode[1].Security.X509.Signature.HashFunction	E	Algorithm used to compute the digest of the message being signed: http://www.w3.org/2001/04/xmlenc#sha256 The definitions for these values are in the [XMLDSIG] specification.	Enforced: All security values are enforced by eDelivery AS4 profile
26	PMode[1].Security.X509.Signature.Algorithm	E	Identifies the algorithm to compute the value of the digital signature: http://www.w3.org/2001/04/xmldsig-more#rsa-sha256 The definitions for these values are found in the [XMLDSIG] or [XMLENC] specifications.	Enforced: All security values are enforced by eDelivery AS4 profile
27	PMode[1].Security.X509.Encryption.Encrypt	E	Boolean parameter indicating (if *true*) that the MSH will encrypt: • All payload parts: ○ Any SOAP Body will also be encrypted. • Attachments. Chosen value: **TRUE** **N.B.** The MSH will not encrypt the Messaging header. If confidentiality of data in the Messaging header is required, this can be achieved through **transport level security**.	Enforced: All security values are enforced by eDelivery AS4 profile
28	PMode[1].Security.X509.Encryption.Certificate	E	Public certificate to use when encrypting data: Encryption Certificate of the Receiver	Enforced: All security values are enforced by eDelivery AS4 profile.

29	PMode[1].Security.X509.Encryption.Algorithm	E	Encryption algorithm to be used. The definitions for these values are found in the [XMLENC] specification. Chosen value: http://www.w3.org/2009/xmlenc11#aes128-gcm => **AES-GCM**	Enforced: All security values are enforced by eDelivery AS4 profile
30	PMode[1].Security.X509.Encryption.MinimumStrength	E	Integer value for the effective strength which the encryption algorithm must provide in terms of *effective* or random bits. The value is less than the key length in bits when check bits are used in the key. Chosen value: **128** e.g. The 8 check bits of a 64-bit DES key would not be included in the count. Setting MinimumStrength to 56 is required in order to have a minimum strength equal to that supplied by DES.	Enforced: All security values are enforced by eDelivery AS4 profile
31	PMode[1].Security.UsernameToken.username	E	Username to include in a WSS *Username* Token. Not used	No Guidelines
32	PMode[1].Security.UsernameToken.password	E	Password to use inside a WSS Username Token. Not used	No Guidelines
33	PMode[1].Security.UsernameToken.Digest	E	Indicates whether a password digest will be included in the WSS UsernameToken element. Not used	No Guidelines
34	PMode[1].Security.	E	Indicates whether the WSS UsernameToken element	No Guidelines

	UsernameToken.Nonce		will contain a Nonce element. Nonce => A number or bit string used only once in security engineering. Not used	
35	PMode[1].Security.UsernameToken.Created	E	Indicates whether the WSS UsernameToken element will have a *Created* timestamp element. Not used.	No Guidelines
36	PMode[1].Security.PModeAuthorize	E	Boolean parameter indicating (if *true*) that a message on the MEP leg has to be authorised for processing under the PMode. Chosen value: **FALSE** If the parameter is *true* this implies that the following has to be used for this purpose: • PMode.Responder.Authorization.{username/password}, **if the message is sent by Responder**. • PMode.Initiator.Authorization, **if the message is sent by Initiator**. e.g. When set to *true* for a *PushRequest* message sent by the Initiator, the pushing will only be authorised over the MPC indicated by this *Push* signal if: • the MPC is the same as specified in the PMode leg for the pushed message; **and** • the signal contains valid credentials (i.e. username/password).	No Guidelines
37	PMode[1].Security.S	E	Boolean parameter indicating (if *true*) that a signed	No Guidelines

	endReceipt		receipt (Receipt ebMS signal) containing a digest of the message has to be sent back.	
38	PMode[1].Security.SendReceipt.NonRepudiation	E	Boolean parameter indicating (if *true*) that **non-repudiation of receipt** is required. Otherwise (if *false*) only reception awareness is required. Non-repudiation prevents a recipient from denying receipt of a message. Non-repudiation receipts have to be sent synchronously for each message type. **N.B.** Non-repudiation is only *per hop* in the case of the four-corner-model, in particular the hop from corner two to corner three.	No Guidelines
39	PMode[1].Security.SendReceipt.ReplyPattern	E	Indicates whether the Receipt signal is to be sent: • as a callback on a separate connection. (value "**callback**"); **or** • synchronously on the HTTP response or back-channel (value "**response**"). If absent from the PMode, any pattern may be used.	No Guidelines
40	PMode[1].PayloadService.CompressionType	E	Type of compression used in the payload.	No Guidelines
41	PMode[1].ReceptionAwareness	S	If present, indicates that steps must be taken to seek to ensure reception of a message.	No Guidelines
42	PMode[1].ReceptionAwareness.Retry	S	If present, indicates that the steps taken to seek to ensure reception of a message will be repeated if necessary.	No Guidelines
43	PMode[1].ReceptionAwareness.Retry.P	S	Contains information about the maximum number of retries, the retry timeout or a retry rule.	No Guidelines

	arameters		These parameters vary with the AP implementation.	
44	PMode[1].ReceptionAwareness.DuplicateDetection	E	Boolean parameter indicating (if *true*) that duplicate messages are to be detected. Profile value: **TRUE**	Chosen value: **TRUE**
45	PMode[1].ReceptionAwareness.DetectDuplicates.Parameters	S	A implementation-dependent composite string showing (for example): • Maximum size of message log over which duplicate detection is supported. • Maximum time window over which duplicate detection is supported. The string contains a sequence of parameters of the form: name=value, separated by either comas or ';'. e.g. "maxsize=10Mb,checkwindow=7D", if the duplicate check window is guaranteed of 7 days minimum. To determine if a message is a **duplicate**, only the **message identifier is checked**.	No Guidelines
46	PMode[1].BusinessInfo.subMPCext		Sub-channel extension to be used. e.g. If PMode[1].BusinessInfo.MPC is "http://sender.example.com/mpc123" **and** subMPCext is "subc42" **then** the sub-channel is: http://sender.example.com/mpc123/subc42	No Guidelines

			and the **sub-channel extension** is: subc42	

3. CONTACT INFORMATION

CEF Support Team

By email: CEF-EDELIVERY-SUPPORT@ec.europa.eu

Support Service: 8am to 6pm (Normal EC working Days)