

Transparency Production Unit (“TransProd”)

Data exchange scenarios

Version 3.1 – 12/03/2020

Versioning

Version 1.0	Cédric Charlier	23/08/2009
Version 2.0	Cédric Charlier	20/09/2009
Version 2.1	Cédric Charlier	26/01/2010
Version 2.2	Cédric Charlier	17/02/2010
Version 2.3	Cédric Charlier	24/04/2011
Version 3.0	Jean-Michel Reghem	8/05/2014
Version 3.1	Sylvain Baurès	12/03/2020

Editorial board

IT Application Leader for Reporting Applications (ETAP)	Jean-Michel Reghem
--	--------------------

Approbation list

Business Project Leader	Thomas Oldenhove	<input checked="" type="checkbox"/>
Business Analyst	José Gonzalez Pastor	<input checked="" type="checkbox"/>
IT Project Leader	Jean-Michel Reghem	<input checked="" type="checkbox"/>
Test Manager	Frédéric Williaume	<input checked="" type="checkbox"/>

Distribution list

All Parties implementing the transparency production process

1 Table of Contents

1	TABLE OF CONTENTS	2
2	PURPOSE OF THE DOCUMENT AND OVERVIEW	3
2.1	Purpose	3
2.2	Overview	3
3	SUBMIT TRANSPARENCY PRODUCTION DATA	5
3.1.1	Actors	5
3.1.2	When	5
3.1.3	What	5
3.1.4	How	5
3.1.5	Response	5
3.1.6	Diagrams	6
4	RECEIVE BUSINESS ACKNOWLEDGE	7
4.1	Push strategy	7
4.1.1	Actors	7
4.1.2	When	7
4.1.3	What	7
4.1.4	How	7
4.1.5	Response	7
4.1.6	Diagram	8
4.2	Pull strategy	8
4.2.1	Actors	8
4.2.2	When	8
4.2.3	What	8
4.2.4	How and Response	8
4.2.5	Diagram	10
5	GET LIST OF POWER UNITS	11
5.1.1	Actors	11
5.1.2	When	11
5.1.3	What	11
5.1.4	How	11
6	GET LIST OF HYDROSTORAGE PLANTS	12
6.1.1	Actors	12
6.1.2	When	12
6.1.3	What	12
6.1.4	How	12

2 Purpose of the document and overview

2.1 Purpose

The goal of this document is to explain the different ways data for transparency production can be supplied to Elia. This document explains also the different possibilities provided to data supplier to retrieve the status of their submissions. The last chapter explains how to retrieve the list of Power Units for which Elia is expecting data.

2.2 Overview

The main idea of this project is to provide Web Services exposed and hosted by Elia. Data for transparency production must be supplied by each ARP to Elia by the means of these Web Services. Elia will perform different quality checks on data submitted and then return an acknowledgement to confirm the treatment of the previously sent data. In case of mistake in the submitted data, the acknowledgement will contain information about treated and not treated values.

To avoid unnecessary tuning in the respective firewalls and too much security concerns it has been decided that the tunnel of communication will be the same than the one used for Probid.

Some variants have also been implemented by Elia to let the ARP choose the technical implementation they wanted and suiting their internal strategy. For this, the usage of Xml files (streamed) or Domain objects are allowed to submit data through the WebService. The possibility for the ARP to retrieve the acknowledgement directly through the Web Service hosted by Elia (Pull) or to receive it on its own Web Service (Push) have also been developed.

When possible, and with a minimum set of change, Enso-E (formally ETSO) document have been used to formalize the data contracts between ARP and Elia.

2.3 URL of external WebSite and Webservice

- **Prod:**

- Web site

<https://transprod.elia.be>

- External Service

<https://transprod.elia.be/services/ExternalService.svc>

- External Service WSDL <https://transprod.elia.be/services/ExternalService.svc?wsdl>

-

- **Demo :**

- Web site <https://transproddemo.elia.be>

-

- External Service <https://transproddemo.elia.be/services/ExternalService.svc>

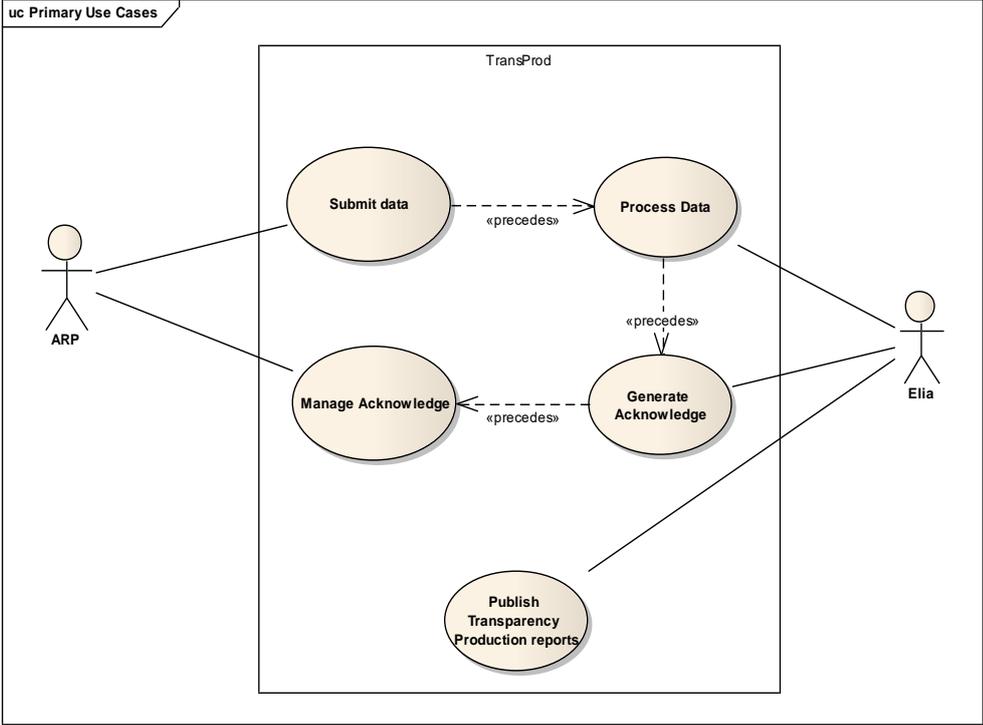
-

- External Service WSDL

<https://transproddemo.elia.be/services/ExternalService.svc?wsdl>

-> Please use your ISOEXT account for connection:

Reset of your password is possible on this page: <https://pwd.elia.be/>



3 Submit transparency production data

3.1.1 Actors

- Sender: ARP
- Receiver: Elia

3.1.2 When

Once by day ... The ARP must send a schedule document each day before 9 o'clock. Moreover several schedule documents can be sent per day. For example, the ARP can send a new version of the schedule document if the previous version needed corrective actions.

Once a week ... ARP's owning a water reservoir or an hydro storage plant must send a weekly stored energy value of the previous week (average) in MWh. It should be send before W+1 Wednesday 9 o'clock, with week W beginning the Monday and ending the Sunday.

On event ... An ARP should also send outage documents when a planned or unplanned outage happen. The goal is to let Elia publish this information on the Transparency Production reports.

3.1.3 What

See Xml specifications document(Version 3.0)

3.1.4 How

Each ARP is allowed to choose between the two strategies described here under, to submit its data. It's not necessary (and useless) to develop both of them.

3.1.4.1 Using Xml files

For the needs of some Arp, Elia has created a set of methods on its Web Service to deal with Xml files.

The methods named *ProcessScheduleMessageStream*, *ProcessOutageDocumentStream* and *ProcessHydroStorageDocumentStream* provides three entry points to submit bytes array to Elia. The first one should be used for schedule data, the second one for outages information and the last one for HydroStorage Weekly value information. The bytes array submitted must be Xml files. These Xml files must validate against XSD provided by Elia. These XSD are derivation from ENSO-E standards.

3.1.4.2 Using Domain Objects

Elia has also created a set of methods on its Web Service to deal with a Domain Model. In this case, the contract between ARP and Elia is a contract using strongly-typed parameters.

The methods named *ProcessScheduleMessage*, *ProcessOutageDocument* and *ProcessHydroStorageDocument* provides two entry points to Domain Objects to Elia. The first one should be used for a schedule message, the second one for outage document and the last one for HydroStorage document. The classes exposed by the Web Services are based on the XSD provided by Elia and are defined in the WSDL.

3.1.5 Response

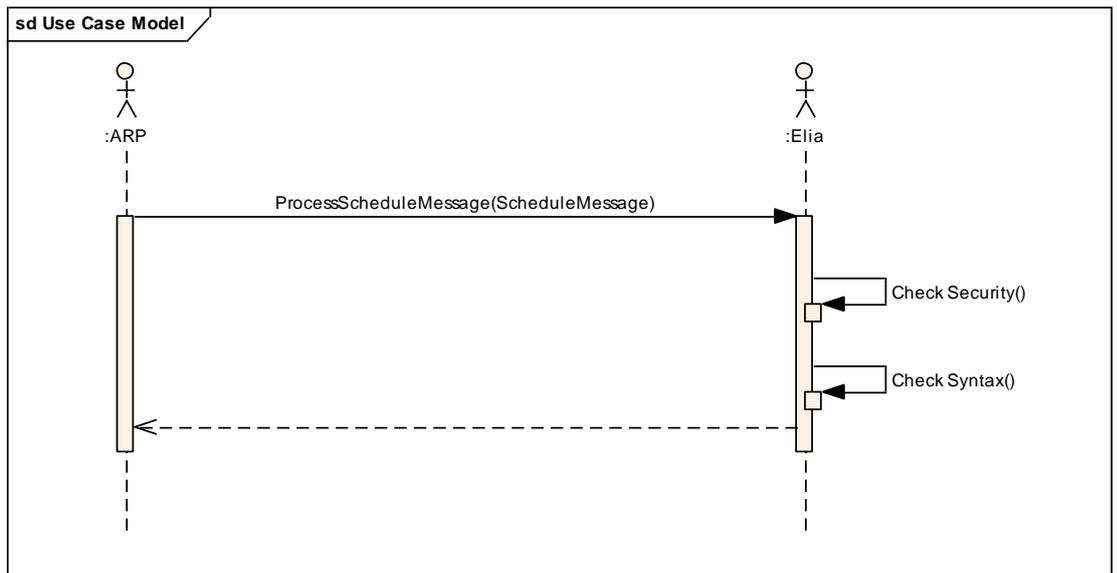
Elia is treating immediately the submitted data from two perspectives: security checks and syntax validation (useful in case of stream strategy). This response is strictly technical. If the WebService exposed by Elia generate no exception, it means

that submitted data have validated technical constraints (In the case of Xml files, it means that the validation with the XSD has been successful) and user is identified and authorized. A Fault means that your submission is incorrect from a syntax point of view and will not be treated further by Elia (You should receive technical details about the error in this acknowledge) or your credentials have not been validated.

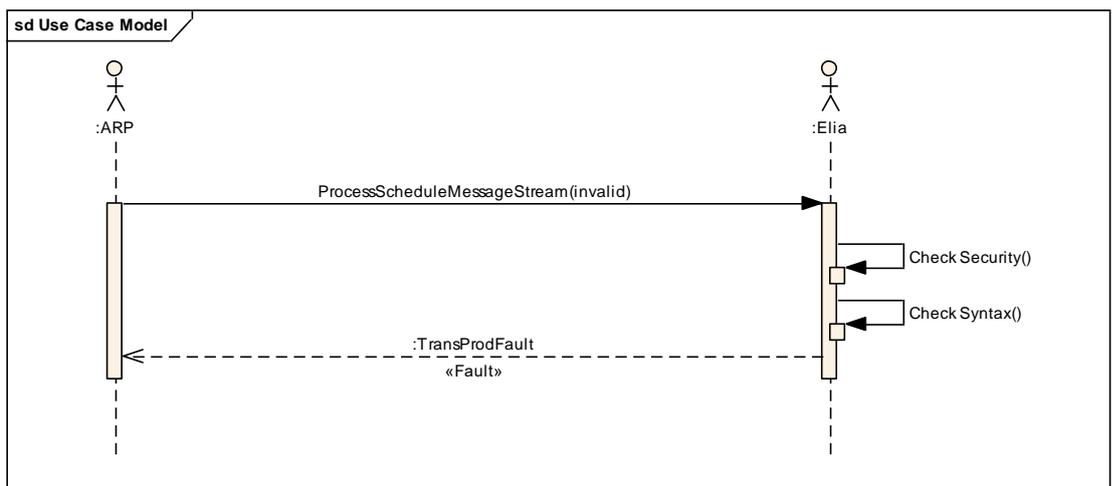
This response is not the end of the process. Elia still needs to validate the business validity of your data. So you should always use the process described in next chapter to know if submitted data have been validated from a business point of view.

3.1.6 Diagrams

This operation is performed in a synchronic scheme until this step.



In case of exception during security checks or validation syntax



4 Receive business acknowledge

Elia has also defined two strategies to retrieve the business acknowledge. Implementing one of them is enough.

4.1 Push strategy

4.1.1 Actors

- Client: Elia
- server: Arp

4.1.2 When

On event ... after complete processing by Elia. The process (business validation and storage) of a set of data is depending on the size of this set and is influenced by other sets available at the same moment. So Elia can't guarantee a maximum response time. In most case the response from Elia should be available in less than 10 minutes.

4.1.3 What

See Xml specifications document about acknowledge document

4.1.4 How

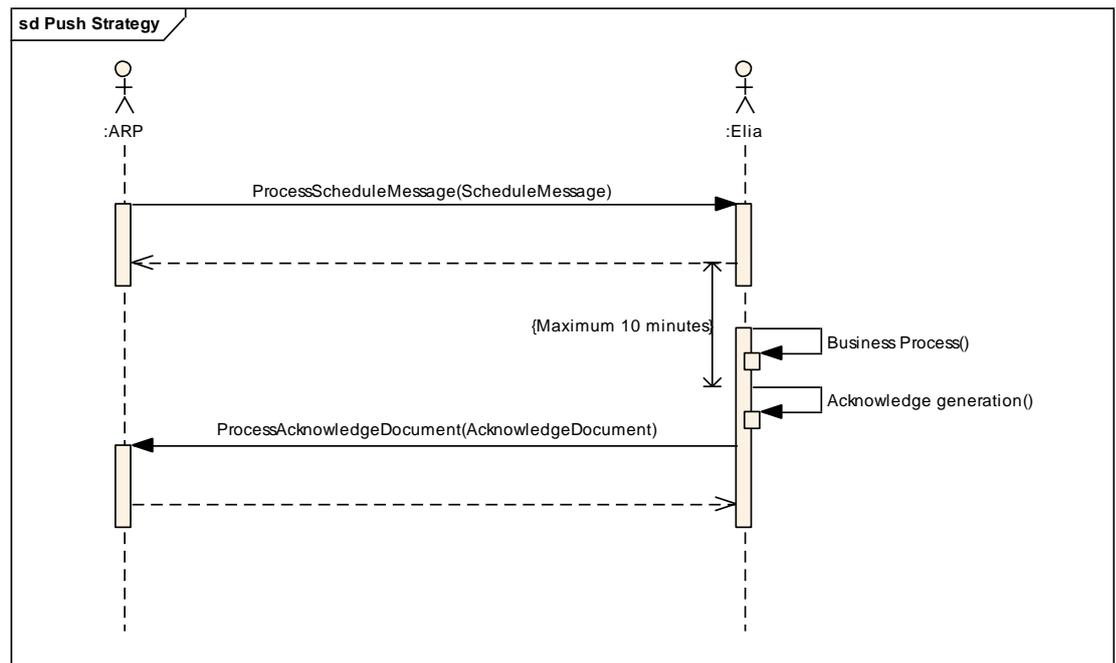
The ARP deciding to implement this strategy must provide a Web Service implementing the contract defined in *TransProd.wsdl*. Also it must provide connection information (url, security settings, ...) to Elia and a support contact during connection issues.

4.1.5 Response

Elia will connect to the Web Service operation and execute it. If the operation fail or succeed is not influencing further the process. Elia will not retry in case of failure.

In case of failure it's still possible for the Arp to retrieve the acknowledge document, check [4.2 Pull strategy](#) for more information

4.1.6 Diagram



4.2 Pull strategy

4.2.1 Actors

- Client: Arp
- Server: Elia

4.2.2 When

On Request (by Arp) after complete processing by Elia. The process (validation and storage) of a set of data is depending on the size of this set and is influenced by other sets available at the same moment. So Elia can't guarantee a maximum response time. In most case the response from Elia should be available in less than 10 minutes.

4.2.3 What

Two information are available to check the treatment of a set of data submitted by an Arp.

First information is the status of the treatment. This status is available right after the submission of a set of data.

When the treatment is finished, another information is available. This information is the acknowledge document. *See [Xml specifications document about acknowledge document](#)*

4.2.4 How and Response

The ARP deciding to implement this strategy must connect to Elia Web Service and query the different methods.

To query the status of a set of data use the method named *GetRequestStatus*. The response will answer by one of the following status:

Exception with submitted document:

- Refused
- Incorrect syntax

Treatment still running:

- Received
- Syntax validated
- In business validation

Treatment done, acknowledge document available:

- Fully accepted
- Fully rejected
- Partially rejected

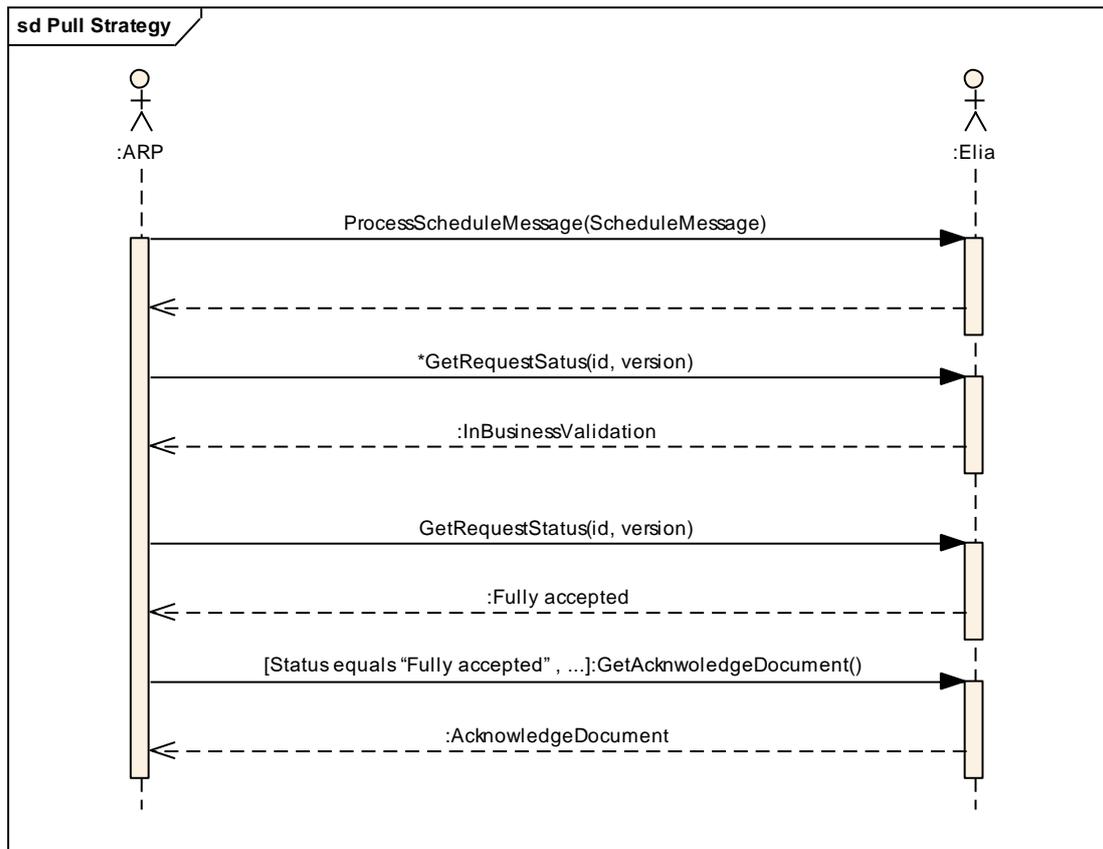
If you reach the status Refused or Incorrect Syntax, you should have received an exception following your request (for instance, a badly formatted xml which didn't respect the xsd, or a problem with the login ...) . The status will not change anymore.

If the status is "Received" or "Syntax validated" or "In business validation", the treatment is still running, retry **30 seconds** later.

If the status is "Fully accepted", "Fully rejected" or "Partially rejected" then the acknowledge document is available.

When the status is final, you can use the method named *GetAcknowledgeDocument* or *GetAcknowledgeDocumentStream* to retrieve the acknowledge document. If you try to retrieve the acknowledge before the final is "Fully accepted", "Fully rejected" or "Partially rejected" then an exception will be thrown. So we definitively ask you to check the status before trying to retrieve the acknowledge document, in a pulling strategy.

4.2.5 Diagram



5 Get list of Power Units

5.1.1 Actors

- Requester: ARP

5.1.2 When

On event ... An Arp can request this method (*GetPowerUnits*) when needed

5.1.3 What

Get the list of Power Units defined in Elia's system and associated to the ARP connected to the system.

5.1.4 How

Request the method (no parameter expected) and get the response.

6 Get list of HydroStorage Plants

6.1.1 Actors

- Requester: ARP

6.1.2 When

On event ... An Arp can request this method (*GetHydroStoragePlants*) when needed

6.1.3 What

Get the list of HydroStorage plants defined in Elia's system and associated to the ARP connected to the system.

6.1.4 How

Request the method (no parameter expected) and get the response.