Erasmus Without Paper Network — from development to production

Janina Mincer-Daszkiewicz (jmd@mimuw.edu.pl), Faculty of Mathematics, Informatics, and Mechanics, University of Warsaw, Banacha 2, 02-097 Warszawa, Poland

Keywords

Erasmus+, EWP Network, Registry, Erasmus+ Dashboard, EWP Competence Center, digital mobility, data portability, API, Student Information System

1. ABSTRACT

EWP (*Erasmus Without Paper*) is the European project co-financed in years 2015-2017 by the *Erasmus+ Programme, Key Action 3 (Prospective Initiatives — Forward Looking Cooperation Projects*). EACEA has prolonged funding for years 2018-2019.

In December 2018, three years after the start of the first EWP project, in front of 350 participants from across Europe gathered in the stunning aula of the University of Ghent, and many more following the event via live stream, the Erasmus Without Paper Network was officially launched (see https://www.erasmuswithoutpaper.eu/launch-conference). Teams representing SOP from Austria, QS Unisolution from Germany, SIGMA from Spain, MUCI from Poland, University of Ghent from Belgium, and University of Porto from Portugal conducted live demo showing exchange of mobility data between their Student Information Systems.

The event does not mark the end of the EWP project, but is the milestone turning the development stage into production. The most challenging activities are still ahead — industrializing the Network and its elements, integrating data transfer with business processes of student mobility carried at higher education institutions, increasing network coverage by accepting new institutions, supporting new partners in joining the network, on a political, organizational and technical level. This support will be offered by the Competence Center to be established in 2019.

While the network grows, new mobility scenarios are recognized and new APIs to support them are designed and developed. New use cases show up and get support, like reporting from HEIs to the Mobility Tool+, or integrating services offered by other networks (like ESMO or EMREX) into one digital platform for education sector.

The aim of this paper is to show the Erasmus Without Paper Network running in production and supporting student mobility in partner institutions. The components of the Network will be presented and their role explained. Planned tasks and activities of the Competence Center will be discussed. Directions of extension of the Network functionality beyond basic scenarios of the Erasmus mobility will be shown. Plans for the European-wide roll out of the EWP Network will be shared.

The ultimate goal is to show the benefits of the EWP solution and encourage higher education institutions to become part of it.

2. INTRODUCTION

EWP (*Erasmus Without Paper*) [5] is the European project funded in years 2015-2017 by the *Erasmus+ Programme, Key Action 3 (Prospective Initiatives — Forward Looking Cooperation Projects)*. The application for a follow-up has been approved and EWP continues as the European project in years 2018-2019.

In December 2018 at the EWP conference in Ghent, the EWP Network had been officially launched. Teams representing software providers from Austria, Germany, Spain, Belgium, Portugal and Poland conducted live demo showing exchange of mobility data between their Student Information Systems ([12]). Live demo was carried in the development environment, but some of the partners have already joined the production EWP Network opening up to the exchange of the production data stored in their production SISs. Possible user scenarios showing EWP support for Erasmus mobility were demonstrated not only during the conference but also in numerous presentations, videos, webinars, and workshops (see e.g. [11]).

The event has been an important milestone but also a beginning of the even more important follow-up. It seems that stakeholders are convinced of the importance and — in some way — inevitability of the paperless mobility. What may not be so widely shared is the belief that the network is ready for production use. Who is in charge of the network components? Will partners respect *Service Level Agreement*? Is the functionality of the Network limited to already recognized scenarios of Erasmus mobility or will it grow when new requirements are recognized? Is privacy of data respected? Should my institution wait for her partners in mobility to join before making the first step? That may look like a typical chicken-egg dilemma. The new partners would prefer to join the larger network so might be willing to wait until it reaches some critical number of participants. Will the development and deployment effort pay off and when (if at all) can we expect return on investment (meaning not only funds but also organizational effort incurred to change processes)?

The goal of this paper is to dispel doubts and help to make a decision. We want to show that the EWP infrastructure had been developed according to highly professional IT standards, the software design specification located in GitHub is open for everybody, there are numerous tools for developers, the Competence Center will offer help to all groups of stakeholders. The network functionality increases gradually. There are plans to integrate services offered by various digital networks into one set of services for academic community. IT companies delivering mobility software to higher education institutions already plan development. Finally yet importantly, the risk of failed investment is limited since software can be developed in stages, along with the growing awareness of the end-users needs.

In chapter 3, we explain what the EWP Network is and why — in our opinion — it is fully ready for the production use. In chapter 5, we describe tools and resources offered to support developers. The role and tasks of the Competence Center are explained in chapter 6. Network functionality is described in chapter 4. Plans for rollout, dissemination and sustainability of the project results are outlined in chapter 7, which concludes the paper.

3. WHAT IS THE EWP NETWORK?

For many the term “*EWP Network*” may sound like a buzzword used in many contexts to describe any digital services for the education market. Let us explain what this term means, especially after the official launch of the EWP Network. In the following sections, we describe the components of the network.

3.1. Design

Software development starts with description of a software system to be developed in a document called **software requirements specification**. In case of EWP, gathering requirements had a form of a wide-scale survey with Higher Education Institutions from 31 different countries. The final report (that plays a role of a software requirements specification) can be found at [9].

This report had been the basis for the technical team to work out a software design, consisting of data design, architecture design, interface design and procedural design. In case of EWP, a **software design specification** is stored in GitHub [7], publicly available for all, to read, follow, comment and contribute. It comprises 62 repositories (as in February 2019), with data models, description of the EWP Network architecture, design of authentication and security protocols (in particular client and server authentication methods), design of *Application Programming Interfaces* (APIs) for network services, rules of error handling, many examples of requests and responses, and even source code for parts of the network. The design is of the highest professional standard, it is carried according to widespread common practices in use in both closed and open-source software design, and follows rules of semantic versioning, which allows keeping track of backward compatibility.

In the project, XML (*eXtensible Markup Language*) is used for description of data format, and XSD (*XML Schema Definition*) for formal specification of XML documents.

The most business-oriented part of the design stored in GitHub is a document, which briefly describes – with help of some flowcharts – how the Student Mobility Business Process is modeled within the EWP Network. It helps to get a quick grasp on which APIs are used, by whom, and when.

Each repository with the specification of a particular API contains:

a) ReadMe part – description of the request parameters and error handling.
b) Manifest entry in XSD format (see section 3.2).
c) Response specification in XSD format.
d) Response examples in XML format.

Every development team can read the specification and develop its own implementation. Transparency of the project design is crucial for building trust and getting the best results.

Documents posted on GitHub are available for review, change notifications help to keep developers informed, built-in issue tracker can be used for asking questions, reporting errors or posting suggestions. GitHub supports formal versioning of documents and source code. The project partners approve official versions of data format and APIs, and – once the APIs are released – backward compatibility is guaranteed. The data formats are compliant with commonly accepted standards, e.g. for transcripts of records we use ELMO ([3]) worked out by the EMREX group ([4]).

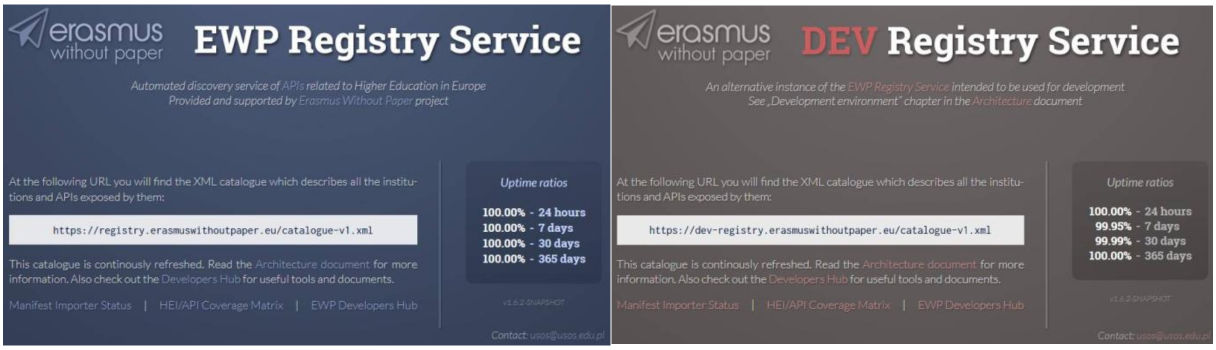
3.2. Registry

The Registry is the only central part of the EWP Network. It is kind of the address book, which gathers in one place information about:

a) **who** – is connected,
b) **where** – how to find him,
c) **what** – services are offered.

In fact, there are two registries available under fixed addresses (Figure 1):

a) PROD: https://registry.erasmuswithoutpaper.eu,
b) DEV: https://dev-registry.erasmuswithoutpaper.eu.



EWP Registry Service

Automated discovery service of APIs related to Higher Education in Europe
Provided and supported by Erasmus Without Paper project

At the following URL you will find the XML catalogue which describes all the institutions and APIs exposed by them:

https://registry.erasmuswithoutpaper.eu/catalogue-v1.xml

This catalogue is continously refreshed. Read the Architecture document for more information. Also check out the Developers Hub for useful tools and documents.

Manifest Importer Status | HEI/API Coverage Matrix | EWP Developers Hub

Uptime ratios

100.00% - 24 hours
100.00% - 7 days
100.00% - 30 days
100.00% - 365 days

v1.6.2-SNAPSHOT

Contact: usos@usos.edu.pl

DEV Registry Service

An alternative instance of the EWP Registry Service intended to be used for development
See „Development environment" chapter in the Architecture document

At the following URL you will find the XML catalogue which describes all the institutions and APIs exposed by them:

https://dev-registry.erasmuswithoutpaper.eu/catalogue-v1.xml

This catalogue is continously refreshed. Read the Architecture document for more information. Also check out the Developers Hub for useful tools and documents.

Manifest Importer Status | HEI/API Coverage Matrix | EWP Developers Hub

Uptime ratios

100.00% - 24 hours
99.95% - 7 days
99.99% - 30 days
100.00% - 365 days

v1.6.2-SNAPSHOT

Contact: usos@usos.edu.pl

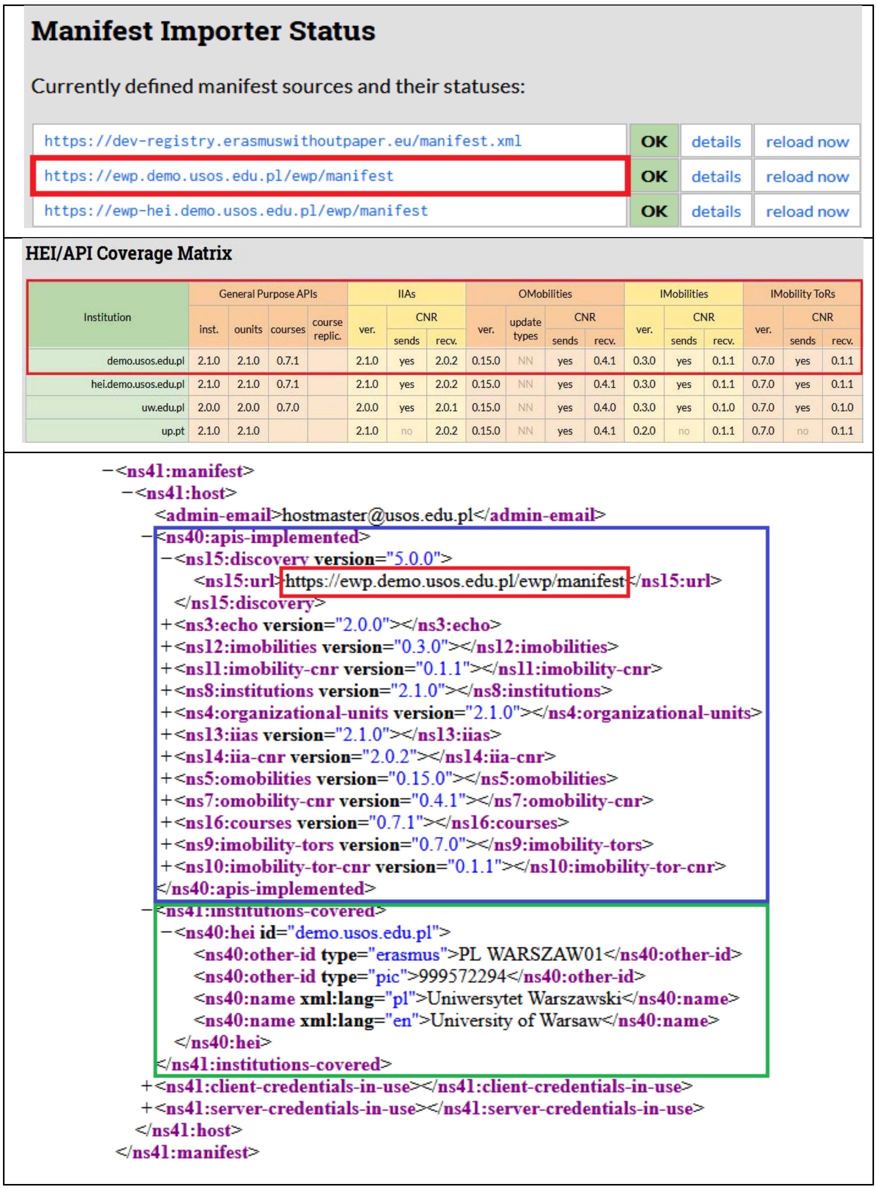
Figure 1 The PROD and DEV Registries of the EWP Network

The Registry is filled manually with the URLs of the new nodes. The Registry updates its content automatically by periodically reading **Manifest** files (services) located under these URLs and getting information about institutions covered and APIs implemented (**Discovery API** is used for that). The Registry keeps track of all the implemented APIs in order to avoid unnecessary requests. Most changes in the Registry can be performed simply by updating the manifest on the partner's server (and the Registry will fetch these changes automatically). This approach supports the scalability of the solution.

Registry is the only node in the network, which implement the **Registry API**. The other nodes call this API to get information about institutions and APIs.

EWP Hosts are not required to implement *all* features of the EWP Network.

The Registry has its web page (Figure **1**). Under the link '*Manifest Importer Status*' one can find the list of manifest files with their current status, and under the link '*HEI/API Coverage Matrix*' – the list of institutions and supported APIs, with the version numbers (Figure **2**).



Manifest Importer Status

Currently defined manifest sources and their statuses:

https://dev-registry.erasmuswithoutpaper.eu/manifest.xml	OK	details	reload now
https://ewp.demo.usos.edu.pl/ewp/manifest	OK	details	reload now
https://ewp-hei.demo.usos.edu.pl/ewp/manifest	OK	details	reload now

HEI/API Coverage Matrix

Institution	General Purpose APIs				IIAs			OMobilities				IMobilities			IMobility ToRs		
	inst.	ounits	courses	course replic.	ver.	CNR		ver.	update types	CNR		ver.	CNR		ver.	CNR	
						sends	recv.			sends	recv.		sends	recv.		sends	recv.
demo.usos.edu.pl	2.1.0	2.1.0	0.7.1		2.1.0	yes	2.0.2	0.15.0	NN	yes	0.4.1	0.3.0	yes	0.1.1	0.7.0	yes	0.1.1
hei.demo.usos.edu.pl	2.1.0	2.1.0	0.7.1		2.1.0	yes	2.0.2	0.15.0	NN	yes	0.4.1	0.3.0	yes	0.1.1	0.7.0	yes	0.1.1
uw.edu.pl	2.0.0	2.0.0	0.7.0		2.0.0	yes	2.0.1	0.15.0	NN	yes	0.4.0	0.3.0	yes	0.1.0	0.7.0	yes	0.1.0
up.pt	2.1.0	2.1.0			2.1.0	no	2.0.2	0.15.0	NN	yes	0.4.1	0.2.0	no	0.1.1	0.7.0	no	0.1.1

```
−<ns41:manifest>
  −<ns41:host>
     <admin-email>hostmaster@usos.edu.pl</admin-email>
    −<ns40:apis-implemented>
      −<ns15:discovery version="5.0.0">
          <ns15:url>https://ewp.demo.usos.edu.pl/ewp/manifest</ns15:url>
       </ns15:discovery>
      +<ns3:echo version="2.0.0"></ns3:echo>
      +<ns12:imobilities version="0.3.0"></ns12:imobilities>
      +<ns11:imobility-cnr version="0.1.1"></ns11:imobility-cnr>
      +<ns8:institutions version="2.1.0"></ns8:institutions>
      +<ns4:organizational-units version="2.1.0"></ns4:organizational-units>
      +<ns13:iias version="2.1.0"></ns13:iias>
      +<ns14:iia-cnr version="2.0.2"></ns14:iia-cnr>
      +<ns5:omobilities version="0.15.0"></ns5:omobilities>
      +<ns7:omobility-cnr version="0.4.1"></ns7:omobility-cnr>
      +<ns16:courses version="0.7.1"></ns16:courses>
      +<ns9:imobility-tors version="0.7.0"></ns9:imobility-tors>
      +<ns10:imobility-tor-cnr version="0.1.1"></ns10:imobility-tor-cnr>
     </ns40:apis-implemented>
    −<ns41:institutions-covered>
      −<ns40:hei id="demo.usos.edu.pl">
          <ns40:other-id type="erasmus">PL WARSZAW01</ns40:other-id>
          <ns40:other-id type="pic">999572294</ns40:other-id>
          <ns40:name xml:lang="pl">Uniwersytet Warszawski</ns40:name>
          <ns40:name xml:lang="en">University of Warsaw</ns40:name>
       </ns40:hei>
     </ns41:institutions-covered>
    +<ns41:client-credentials-in-use></ns41:client-credentials-in-use>
    +<ns41:server-credentials-in-use></ns41:server-credentials-in-use>
   </ns41:host>
</ns41:manifest>
```

Figure 2 Fragments of the manifest file showing institutions covered and APIs implemented

The Registry may also be used for projects unrelated to EWP, as long as these projects have similar architecture.

Both Registries are monitored on a regular basis. Eventually PROD registry will have a couple of instances, located on different servers, synchronized in real time, to offer high availability under the agreed SLA (see chapter 6).

3.3. Connector (EWP server)

It is a piece of software, which is a SIS's agent in the EWP Network. It *has to* implement:

a) Security transport protocols (**ECHO**) — to transfer data in the network.
b) **Discovery** API (**Manifest** file) — to be discovered by the Registry (see section 3.2).

The EWP server *may* implement some **business oriented** APIs — to offer services to other nodes.

One **EWP node** can host more than one HEI (recognized by domain names = SCHAC codes). In order to join the network, the host should publish a valid **Discovery Manifest file** somewhere on its servers and send the URL of this file to the EWP Registry administrator. It will be stored in the Registry allowing partners to identify and verify future requests.

Some of the APIs are **event listener APIs**. If Host 2 wants to receive change notifications from other hosts, it indicates in its Manifest file that it has implemented a specific event listener API. Now, if Host 1 is able to broadcast such notifications, then it asks the Registry which hosts have implemented the proper listeners, and posts the proper notifications at the listener URLs.

The Figure **3** shows part of the network with the Registry and some hosts covering various numbers of institutions, which implement various sets of APIs.

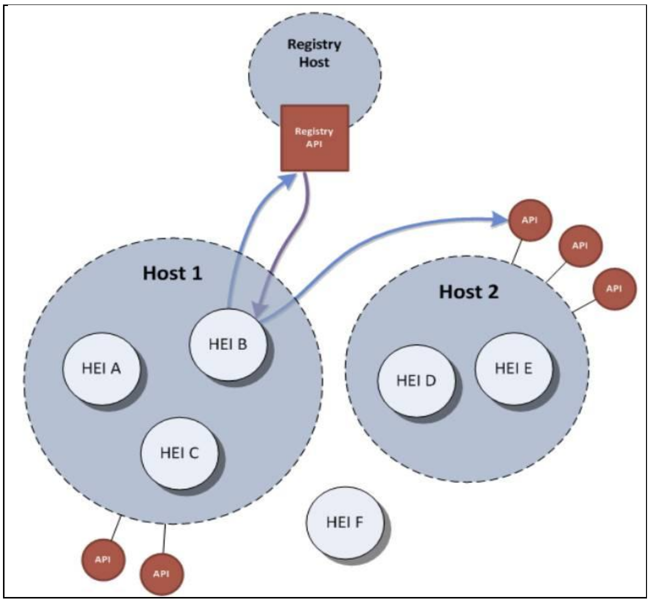


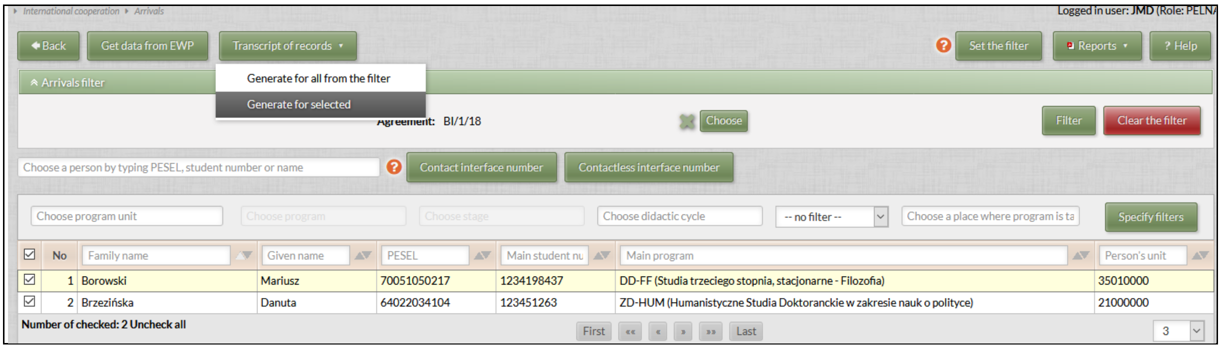
Figure 3 Part of the EWP network with the registry and nodes

3.4. EWP client

The EWP client is composed of the EWP functionalities spread over the **User Interface** of the Mobility module used by IRO, being it part of the local SIS or stand-alone software. It is the most **business oriented** part of the project and should be designed in close contact with the **business owner** (mostly IRO staff).

The specification covers APIs for the most important procedures of the mobility process, like signing bilateral agreement, sending list of the nominated students or exchange of transcript of records when the mobility ends (see chapter 4). It is up the developers to decide which APIs to integrate, in what way etc.

Let us follow an example. There are some students from **UW** (sending institution) on mobility at **HEI** (receiving institution). When the mobility ends, **HEI** generates **Transcript of Records** for incoming students. This can be done per student or for a group of students (see checkboxes in the column on the left on Figure **4**).



Logged in user: JMD (Role: PELN

Agreement: BI/1/18

☑	No	Family name	Given name	PESEL	Main student nu	Main program	Person's unit
☑	1	Borowski	Mariusz	70051050217	1234198437	DD-FF (Studia trzeciego stopnia, stacjonarne - Filozofia)	35010000
☑	2	Brzezińska	Danuta	64022034104	123451263	ZD-HUM (Humanistyczne Studia Doktoranckie w zakresie nauk o polityce)	21000000

Number of checked: 2 Uncheck all

Figure 4 SIS in **HEI** — generating Transcript of Records for the incoming students

HEI notifies the partner about the available transcripts. **UW** obtains the notification (Figure **5**).

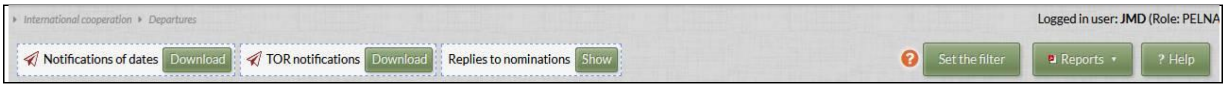
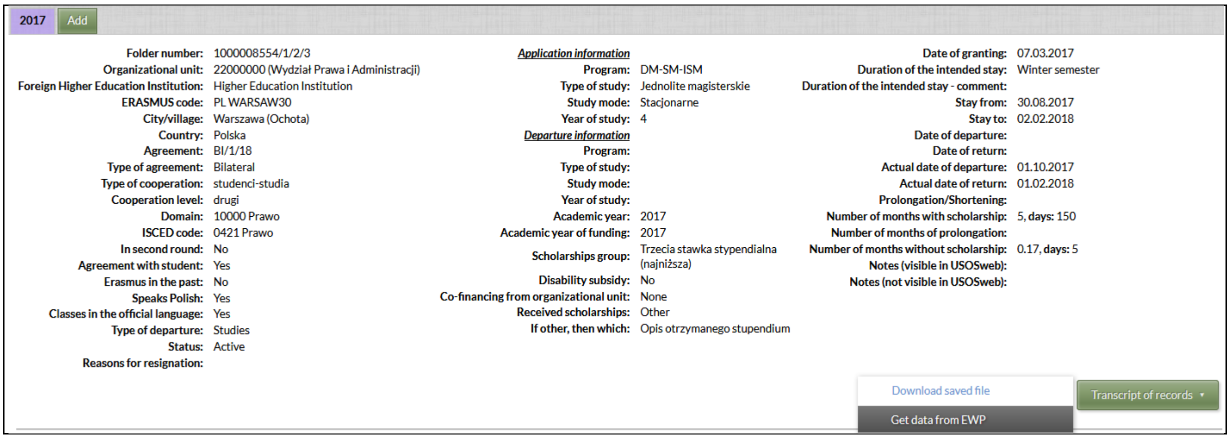


Figure 5 SIS in **UW** — being notified about the available transcripts

UW uses **Get data from EWP** to get Transcript of Record for the outgoing student straight to the local system in the context of the outgoing mobility (Figure **6**). The transcript is transferred in the ELMO format (designed for EMREX, see [3], [10]) which means that it contains courses, grades, ECTS points in a structed XML from which data can be copied to the local tables and used for further processing (e.g. automatic recognition or composing diploma supplement). PDF version for pretty-printing in embedded in XML. Both XML and PDF can be digitally signed.



Folder number: 1000008554/1/2/3
Organizational unit: 22000000 (Wydział Prawa i Administracji)
Foreign Higher Education Institution: Higher Education Institution
ERASMUS code: PL WARSAW30
City/village: Warszawa (Ochota)
Country: Polska
Agreement: BI/1/18
Type of agreement: Bilateral
Type of cooperation: studenci-studia
Cooperation level: drugi
Domain: 10000 Prawo
ISCED code: 0421 Prawo
In second round: No
Agreement with student: Yes
Erasmus in the past: No
Speaks Polish: Yes
Classes in the official language: Yes
Type of departure: Studies
Status: Active
Reasons for resignation:

Application information
Program: DM-SM-ISM
Type of study: Jednolite magisterskie
Study mode: Stacjonarne
Year of study: 4
Departure information
Program:
Type of study:
Study mode:
Year of study:
Academic year: 2017
Academic year of funding: 2017
Scholarships group: Trzecia stawka stypendialna (najniższa)
Disability subsidy: No
Co-financing from organizational unit: None
Received scholarships: Other
If other, then which: Opis otrzymanego stupendium

Date of granting: 07.03.2017
Duration of the intended stay: Winter semester
Duration of the intended stay - comment:
Stay from: 30.08.2017
Stay to: 02.02.2018
Date of departure:
Date of return:
Actual date of departure: 01.10.2017
Actual date of return: 01.02.2018
Prolongation/Shortening:
Number of months with scholarship: 5, days: 150
Number of months of prolongation:
Number of months without scholarship: 0.17, days: 5
Notes (visible in USOSweb):
Notes (not visible in USOSweb):

Figure 6 SIS in **UW** — getting Transcript of Record for the outgoing student

More scenarios are available in [11]. Of course, various SISs may have different user interface and may use EWP APIs in a different way (see [12], [13]).

Notifications play an important role in data synchronization between partner institutions. Data objects are identified by **GUIDs** (*global unique identifiers*) which can be stored in a local system to identify objects coming from the network and recognize them in the local system.

It is up to the local stakeholders how to automate data exchange and in particular, whether the operator or a system daemon should trigger it. Eventually, when the local users will start trusting the network, more and more data may be synchronized between the institutions fully automatically.

4. FUNCTIONALITY OF THE NETWORK

There are many APIs supporting the **EWP business model**:

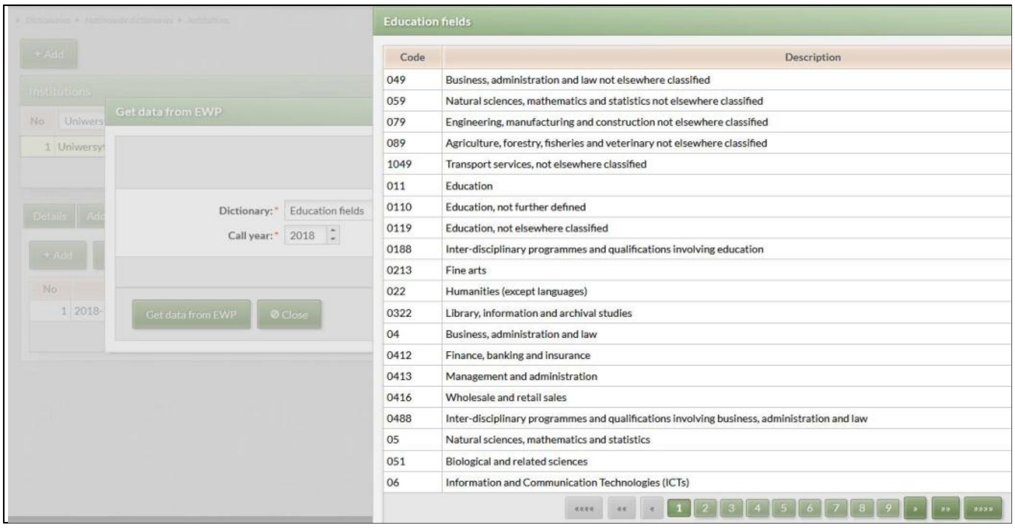
a) **Primary Network** APIs (Discovery Manifest, Echo, Registry — see sections 3.2 and 3.3).
b) **General Purpose** APIs (Institutions, Organizational Units, Courses).
c) **Erasmus Mobility** APIs (Interinstitutional Agreements, Outgoing Mobilities, Incoming Mobilities, Nominations Approval, Mobility Arrival and Departure, Transcripts of Records, CNRs — see the example in section 3.4).
d) **MT+** APIs (Institutions, Projects, Dictionaries).

Each API is relatively simple and can be developed separately from the others. APIs may be implemented one by one, gradually building support for the whole mobility process.

There are APIs for **sending** (pushing) data and for **requesting** (pulling) data in response to **change notifications** (CNR). It is possible to handle EWP data exchange **automatically** by subscribing to notifications and firing triggers when they come.

The special functionality in the network is offered by the **Mobility Tool+** (MT+). MT+ is a platform tool for collaboration, management and reporting for mobility projects under the Lifelong Learning Programme (LLP) and under the Erasmus+ Programme. It is developed by the European Commission to be used by the beneficiaries of Erasmus+ Projects managed by Erasmus+ National Agencies. MT+ is connected to the EWP Network and offers APIs for retrieving:

a) general information on institutions,
b) list of projects for a particular institution and call year,
c) terms from a particular MT+ dictionary and call year (see Figure 7).



Dictionary:* Education fields

Call year:* 2018

Code	Description
049	Business, administration and law not elsewhere classified
059	Natural sciences, mathematics and statistics not elsewhere classified
079	Engineering, manufacturing and construction not elsewhere classified
089	Agriculture, forestry, fisheries and veterinary not elsewhere classified
1049	Transport services, not elsewhere classified
011	Education
0110	Education, not further defined
0119	Education, not elsewhere classified
0188	Inter-disciplinary programmes and qualifications involving education
0213	Fine arts
022	Humanities (except languages)
0322	Library, information and archival studies
04	Business, administration and law
0412	Finance, banking and insurance
0413	Management and administration
0416	Wholesale and retail sales
0488	Inter-disciplinary programmes and qualifications involving business, administration and law
05	Natural sciences, mathematics and statistics
051	Biological and related sciences
06	Information and Communication Technologies (ICTs)

Figure 7 Getting the dictionary of educational fields from the MT+

The most highly expected functionality of the MT+ node, which is under development, is the API for reporting mobilities. Mobility data, which is now transferred by export/import of static files via web interface, will soon (planned for 2019) be sent digitally from SIS of the institution to MT+.

More functionality will be added to the EWP Network in 2019, like grade conversion (using results of the Egracons project, http://egracons.eu/). There are also plans to combine services posted by EWP with services provided by other digital networks, like ESMO [8] and EMREX [4], and offer them all in one unified digital Network.

5. TOOLS FOR DEVELOPERS

The main reference document for the developers is the **Developers Guide** [6]. It gives the overview of resources and tools supporting development:

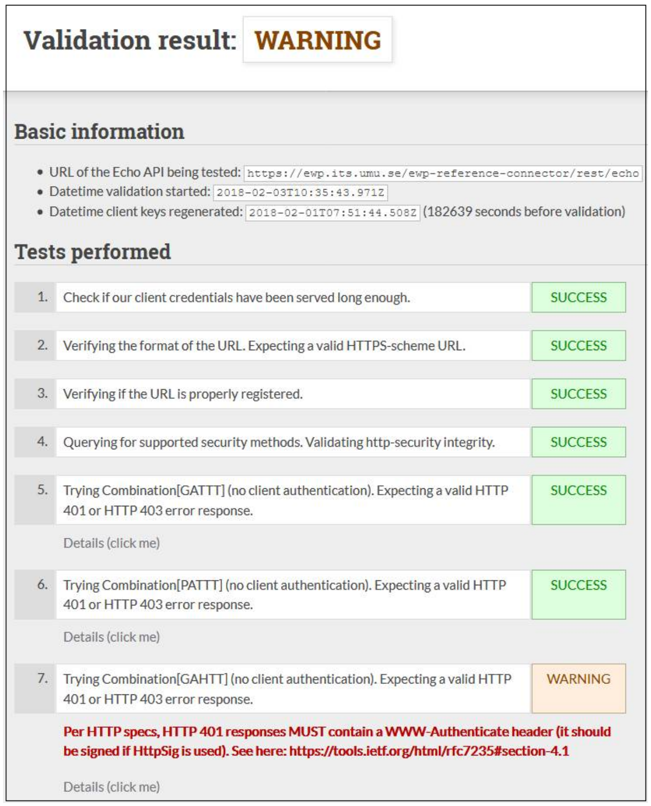
1. Documents and specifications.
2. Libraries and tools.
3. Echo API Validator.
4. Other APIs Validators.
5. XML Schema Validator.

The **Echo API Validator** helps to determine if the implementation meets the basic EWP standards (in particular, its security requirements). It requires the developer to implement the core security framework (which is needed by *all the other* APIs). **Echo API** has been designed to serve two purposes:

a. to make developers aware of the specific security features required by the EWP Network,
b. to allow running automated tests on all existing implementations (thus reducing the risk of security misconfiguration).

The example report produced by the Validator is shown in Figure **8**. The Validator performs thousands of tests, checking the implementation against the specification, only a small part of the report is presented.

Institutions API Validator is also available. More API Validators will be developed in 2019.



Validation result: WARNING

Basic information

- URL of the Echo API being tested: `https://ewp.its.umu.se/ewp-reference-connector/rest/echo`
- Datetime validation started: `2018-02-03T10:35:43.971Z`
- Datetime client keys regenerated: `2018-02-01T07:51:44.508Z` (182639 seconds before validation)

Tests performed

1. Check if our client credentials have been served long enough. SUCCESS
2. Verifying the format of the URL. Expecting a valid HTTPS-scheme URL. SUCCESS
3. Verifying if the URL is properly registered. SUCCESS
4. Querying for supported security methods. Validating http-security integrity. SUCCESS
5. Trying Combination[GATTT] (no client authentication). Expecting a valid HTTP 401 or HTTP 403 error response. SUCCESS
   Details (click me)
6. Trying Combination[PATTT] (no client authentication). Expecting a valid HTTP 401 or HTTP 403 error response. SUCCESS
   Details (click me)
7. Trying Combination[GAHTT] (no client authentication). Expecting a valid HTTP 401 or HTTP 403 error response. WARNING
   Per HTTP specs, HTTP 401 responses MUST contain a WWW-Authenticate header (it should be signed if HttpSig is used). See here: https://tools.ietf.org/html/rfc7235#section-4.1
   Details (click me)

Figure 8 Echo API Validator

Another available tool is the **XML Schema Validator**. This tool helps to write EWP XML documents. For example, a developer can check the contents of the local Manifest file before uploading it onto the production site. The Validator allows validation of any XML document described in all released specifications. This tool will validate against the schema only! Even if such validation succeeds, the file may still be invalid (if, for example, the developer did not adhere to the guidelines described in <*documentation*> elements included in the XSD files).

Developers should start with implementing **Discovery API** and **Echo API**. Having implemented these basic APIs, the institution may become part of the Network. Implementation of other APIs may follow according to the needs and priorities of the institution.

There is a reference connector developed by the Umea University, set up to help in testing the communication within the Network and to serve as the reference implementation. It is running at https://ewp.its.umu.se/ewp-reference-connector.

Open Source University Alliance (OSUA) helps the community by offering source code of some implementations, e.g. basic, ready to deploy EWP connector developed by the University of Porto and the complete reference connector. See https://open-source-alliance.erasmuswithoutpaper.eu.

Some of the partners offer stable demo versions of their installations to be used for testing, e.g. there are two demo installations of the University of Warsaw (demonstrated in section 3.4). They contain full set of data necessary for comprehensive testing. Data come from institutional databases but have been scrambled to protect privacy. Credentials needed to use this test bed are available upon request from the author of the paper.

Testing of the new implementation is crucial. Testing should start with automated tests, followed by the testing with the reference connector, demo installations of the partners, or your own. Testing is a necessary first step of technical requirements of the **entry procedure** (see chapter 6).

6. COMPETENCE CENTER

The **Competence Center** (CC) has been established to support all stakeholders: Erasmus mobility coordinators, Higher Education Institution leaders, IT teams and third party mobility software providers. In particular, its goal is to guide HEIs in choosing a most appropriate way to join the EWP community.

HEIs with no specific tool can use the **Erasmus+ Dashboard** to exchange data with their partners. The Erasmus+ Dashboard is a free to use management tool designed to support HEIs with the administration of mobility, which is used by more than 760 institutions (as in February 2019). It allows handling interinsitutional agreements, managing incoming and outgoing students, signing and reviewing their Online Learning Agreements as well as communicating with students and partner institutions. It also connects to the European Commission's Erasmus+ App. The tool has been developed with the support of the European Commission.

HEIs with their own in-house built mobility management software should develop their own connector to communicate with other systems. The local IT team should do the implementation and thus we encourage the decision makers to involve IT people in the process right from the beginning.

In case HEI is using third party mobility management software, chances are that the software provider is already connecting the software to the EWP Network. We encourage HEIs to get in touch with their respective software providers to learn more about their involvement in EWP.

The EWP **entry procedure** lists the steps to follow on a policy level and on a technical level. When the new EWP connector is ready or at least the basic network APIs are available, the URL of the local Manifest file should be entered to the DEV Registry. The EWP Technical Support Team should be contacted to make it happen. Then testing should start, in the development environment. Tools and resources described in chapter 5 may be used. The testing phase should be concluded with the acceptance testing of the new installation, carried out by the EWP Technical Support Team. After getting approval from the EWP Technical Support Team, the local installation is accepted in the production EWP Registry.

CC will work out all necessary documents and procedures to secure the smooth running of the EWP Network. We need a strict rule to verify and authorize the new participants (nobody wants to open the network to diploma mills). We also need a *Service Level Agreement* to be accepted and signed by all network participants, since our business process will depend on the accessibility of the nodes of our partners.

At the web page of CC, available at https://www.erasmuswithoutpaper.eu/cc, one can find useful resources, tutorial videos, presentations, tools and material offering.

7. SUMMARY

The EWP Network has evolved into a professional secure digital platform, connecting production SISs, with a high potential for Europe-wide roll out. It is open for all categories of users: HEIs with a homemade SIS (like UGent, UPorto), clients of the commercial mobility software providers (like SOP, QS Unisolution, Terra Dotta, Solenovo), members of a consortium of HEIs using the same SIS (like MUCI from Poland, SIGMA from Spain, CINECA from Italy, UNIT from Norway), or numerous HEIs handling their mobilities via the Erasmus+ Dashboard.

The Mobility Tool+, managed by the Directorate-General Education and Culture Unit (DG EAC) of the European Commission, offers services to the EWP hosts for getting the data needed for reporting results of the mobility projects, and — starting from 2019 — to deliver reports fully digitally.

Dissemination events going on all over Europe help in sharing experience and involving partners in mobility for joint actions.

The Competence Center will offer expertise and support, and will take care of the formal aspects of keeping the network in operation. In particular, it will state the requirements for new participants.

EWP constitutes a significant innovation in current practices for organizing student mobility and has a strong potential to be mainstreamed with a long-term impact. One aim pursued by the EWP project is the outreach to European and National policy makers to create a shift in administrative culture and the use of ICT tools, by not only proposing a publicly available network for the exchange of student data, but also engaging in policy dialogue in preparation of the follow-up programme of Erasmus+. This amounts to a significant contribution for the modernization of higher education, which is one of the tenets of the *Modernizing education in the EU* Communication put forward by the European Commission (see [1] and [2]). EWP is mentioned in both of the referenced documents.

8. ACKNOWLEDGEMENTS

EWP 2.0 project is co-funded by the Erasmus+ Programme of the European Union under the grant 590192-EPP-1-2017-1-LU-EPPKA3-PI-FORWARD. It is also co-financed by the Polish Ministry of Science and Higher Education from the funds allocated in the years 2018-2019 for science, granted to international co-financed projects.

Wojtek is the main architect of the EWP Network. Polish EWP node has been implemented by Marta, Michał and Kamil. All the EWP partners contributed to the success of the project.

9. REFERENCES

All links have been retrieved in February 2019.

[1] Communication from the commission to the European parliament, the council, the European economic and social committee and the committee of the regions on a renewed EU agenda for higher education (2017). https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52017DC0247&from=EN

[2] Communication from the commission to the European parliament, the council, the European economic and social committee and the committee of the regions on the Digital Education Action Plan (2018). https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52018DC0022&from=EN

[3] ELMO data standard, https://github.com/emrex-eu/elmo-schemas.

[4] EMREX project, http://www.emrex.eu.

[5] Erasmus Without Paper project, http://www.erasmuswithoutpaper.eu.

[6] Erasmus Without Paper Developers Guide, https://developers.erasmuswithoutpaper.eu.

[7] Erasmus Without Paper in GitHub, https://github.com/erasmus-without-paper.

[8] ESMO project, http://www.esmo-project.eu.

[9] Jahnke S. (2017). *Desk Research — Erasmus Without Paper*, Brussels, Belgium, https://www.erasmuswithoutpaper.eu/sites/default/files/pages/EWP%20desk%20research%20final%20version.pdf.

[10] Mincer-Daszkiewicz J. (2017). *EMREX and EWP offering complementary digital services in the higher education area*, EUNIS 2017, Münster, Germany. Published in EUNIS 2017: Shaping the Digital Future of Universities, Book of Proceedings, p. 354-357.

[11] Mincer-Daszkiewicz J. (2018). Mobility scenarios supported by the Erasmus Without Paper Network (full paper), EUNIS 2018, The 24th International Conference of European University Information Systems, 5-8 June 2018, Paris, France. Published in EUNIS 2018: Coming of age in a digital world, Book of Proceedings, p. 189-190.

[12] EWP development teams (2018). Presentation of the running EWP Network at the launch conference of the EWP 2.0 project, Live stream, 12.12.2018, Ghent, Belgium.

[13] Mincer-Daszkiewicz J. (2018). *EWP anatomy — an in-depth look at its architecture. EWP launch conference*, 13.12.2018, Ghent, Belgium.

10. AUTHOR BIOGRAPHY

Janina Mincer-Daszkiewicz graduated in computer science in the University of Warsaw, Poland, and obtained a Ph.D. degree in math from the same university. She is an associate professor in Computer Science at the Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw specializing in operating systems, distributed systems, performance evaluation and software engineering. Since 1999, she leads a project for the development of a student management information system USOS, which is used in more than 60 Polish Higher Education Institutions, gathered in the MUCI consortium. Janina takes an active part in many nation-wide projects in Poland. She has been involved in Egracons, EMREX and Erasmus Without Paper European projects.