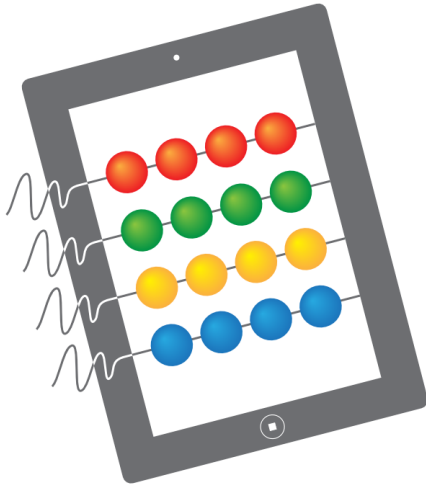




FP7 ICT STREP Project



LEARN PAd

Deliverable D9.5

Technology-oriented Learn PAd whitepaper

First version

<http://www.learnpad.eu>



LINAGORA



No Magic Europe

n|w

Fachhochschule
Nordwestschweiz



*WIKI



Project Number	: FP7-619583
Project Title	: Learn PAd Model-Based Social Learning for Public Administrations

Deliverable Number	: D9.5
Title of Deliverable	: Technology-oriented Learn PAd whitepaper
Nature of Deliverable	: Report
Dissemination level	: Public
Licence	: Creative Commons Attribution 3.0 License
Version	: 2.5
Contractual Delivery Date	: 24 January 2016
Actual Delivery Date	: 24 January 2016
Contributing WP	: WP9
Editor(s)	: Jean Simard
Author(s)	: Jean Simard (XWIKI)
Reviewer(s)	: Andrea Polini (UNICAM)

Abstract

In this white paper, you'll be shown a technical overview of the Learn PAd platform. Learn PAd platform is targeting learning of processes in Public Administration with innovative approach combining model-driven learning and collaborative learning. This white paper will first expose the different functionalities of the platform through a scenario that covers a variety of use cases. Then it will present the components and communications involved in order to orchestrate these use cases.

Keyword List

platform, white paper

Document History

Version	Changes	Author(s)
0.1	ToC	Jean Simard
0.2	ToC	Benjamin Lanciaux, Jean Simard
1.0	Write the Introduction	Jean Simard
1.1	Add Flow view and Component View	Jean Simard
1.2	Remove sections	Jean Simard
2.0	Improve after a review from XWiki	Benjamin Lanciaux, Jean Simard
2.1	Improve after a review from NME	Joaldas Januškevičius, Jean Simard
2.2	Add graphics	Jean Simard
2.3	Improve with contributions	Francesco Basciani, Antonello Calabrò, Alessio Ferrari, Francesca Lonetti, Alfonso Pierantonio, Andrea Polini, Gianni Rosa, Darius Šilingas, Jean Simard
2.4	Improve with contributions from LINAGORA	Tom Jorquera, Jean Simard, Sarah Zribi
2.5	Improve with contributions from NME, FHNW	Joaldas Januškevičius, Andrea Polini, Andrea Sergiacomi, Barbara Thönssen

Document Reviews

Release	Date	Ver.	Reviewers	Comments
ToC	2 Dec. 2015	0.2		
Draft	27 Dec. 2015	1.2		
Internal	10 Jan. 2016	2.5	Andrea Polini, Antonia Bertolino	
Candidate Final	24 Jan. 2016	3.0	Antonia Bertolino	

Table Of Contents

1	E-Learning and Knowledge Management of processes in Public Administration .	1
1.1	Why Learn PAd?	1
1.2	What is Learn PAd?	1
1.3	Who will benefit Learn PAd?	1
1.4	When will Learn PAd exist?	2
1.5	What is in this white paper?.....	2
2	Learn PAd platform: how to use it?	3
2.1	Overview of Learn PAd Platform	3
2.2	Kind of users in Learn PAd Platform.....	3
2.3	Modeler use cases	3
2.3.1	Modeling.....	3
2.3.2	Publishing the models	5
2.3.3	Verification of the models	5
2.3.4	Resolve feedbacks	5
2.4	Learner use cases	5
2.4.1	Discover a new process.....	5
2.4.2	Getting recommendations.....	6
2.4.3	Collaborate to the documentation	6
2.4.4	Reviewing collaborative content	6
2.4.5	Formulate a feedback to Modelers.....	6
2.4.6	Simulate a process	7
3	Learn PAd platform: solutions and technologies	9
3.1	Learn PAd Core.....	10
3.2	Modeling Environment.....	10
3.3	Collaborative Workspace	11
3.4	Transformations.....	11
3.5	Ontology Recommender.....	12
3.6	Simulation Environment	13
3.7	Model Verification	13
3.8	Content Analysis.....	14
4	About partners	15
4.1	Consiglio Nazionale delle Ricerche.....	15
4.2	BOC Asset Management GmbH	15
4.3	Linagora Grand Sud Ouest SA.....	16
4.4	No Magic Europe UAB	16
4.5	Regione Marche	17

4.6	<i>Fachhochschule Nordwestschweiz</i>	18
4.7	<i>Università degli Studi di Camerino</i>	18
4.8	<i>Università degli Studi dell'Aquila</i>	19
4.9	<i>XWiki SAS</i>	19

1 E-Learning and Knowledge Management of processes in Public Administration

1.1. Why Learn PAd?

In modern society public administrations (PAs) are undergoing a transformation of their perceived role from controllers to proactive service providers, and are under pressure to constantly improve their service quality while coping with quickly changing context (changes in law and regulations, societal globalization, fast technology evolution) and decreasing budgets. Civil servants are challenged to understand and put in action latest procedures and rules within tight time constraints.

1.2. What is Learn PAd?

Learn PAd is an innovative holistic e-learning platform for PAs that enables process-driven learning and fosters cooperation and knowledge-sharing. Learn PAd technical innovation is based on four pillars:

- 1) a new concept of model-based e-learning (both process and knowledge)

Learning of processes by Civil Servants is driven by a very structured information: models

- 2) open and collaborative e-learning content management

Learning should be based on open data and everybody can participate to improve these data

- 3) automatic, learner-specific and collaborative content quality assessment

Every new content is verified by automatic systems and also by other Civil Servants, collaboratively

- 4) automatic model-driven simulation-based learning and testing

Civil Servants can run a process in a risk-free simulation in order to learn and be evaluated on their new knowledge

1.3. Who will benefit Learn PAd?

Learn PAd is designed with and for the Public Administration. Civil Servants from these Public Administrations implements processes everyday: they must be aware of any evolution or any new process.

1.4. When will Learn PAd exist?

Learn PAd is a European Union funded project and started on February 2014. It's a two and half years long project so it will end on August 2016. In the meantime, a first prototype has been evaluated by Regione Marche Administration (Italy) on November 2015. A second release of the platform will be release on April 2016. A final release is expected for April 2016.

1.5. What is in this white paper?

This white paper will present the technical solutions that have been merged together to meet the requirements of this Learn PAd platform. In the next section, you'll meet with the Learn PAd platform. In the last section, you'll meet the partners that participate in the elaboration of this Learn PAd platform.

2 Learn PAd platform: how to use it?

In this section, we'll provide different use cases which will give an overview of Learn PAd Platform. These use cases are user-driven: you'll see Modeler use cases and Learner use cases. But first, let's explain what is the Learn PAd Platform and what are the kind of users that will use this Learn PAd Platform.

2.1. Overview of Learn PAd Platform

The Figure 2.1 shows the global workflow happening inside the Learn PAd platform. The Modeling Environment component ❶ is the place where the models are designed. Then they are verified by the Model Verification component ❷ before being transformed by the Transformations component ❸ in different kinds of representations:

- Wiki pages for the Collaborative Workspace component ❹
- Ontologies for the Ontology Recommender component ❺
- Business Process files for the Simulation Environment component ❻

Eventually, Collaborative Workspace is checking the content co-created by Civil Servants by sending them to the Content Analysis component ❼.

2.2. Kind of users in Learn PAd Platform

Learn PAd platform has the goal to document processes in the Public Administration for Civil Servants to learn them. It means that the first users of the platform will be Civil Servants: they will learn or improve their existing knowledge about processes.

In the meantime, someone has to document these processes: the Modeler. Based on information of an experienced Civil Servant, the Modeler will translate the process into models that will be injected in the Learn PAd platform.

The two following sections will be about the typical use cases of a Modeler and of a Civil Servant, and more specifically, a Learner.

2.3. Modeler use cases

2.3.1. Modeling

The first task of the Modeler is to create and document a existing process from the Public Administration. Working on a Modeling tool, he will model Business Processes. But in Learn PAd, the Modeler brings a lot more than Business Process modeling. The Modeler will also complete Business Process model with: Organization Structure, Documents and Knowledge, Competencies, Case Management, KPI, Motivations.

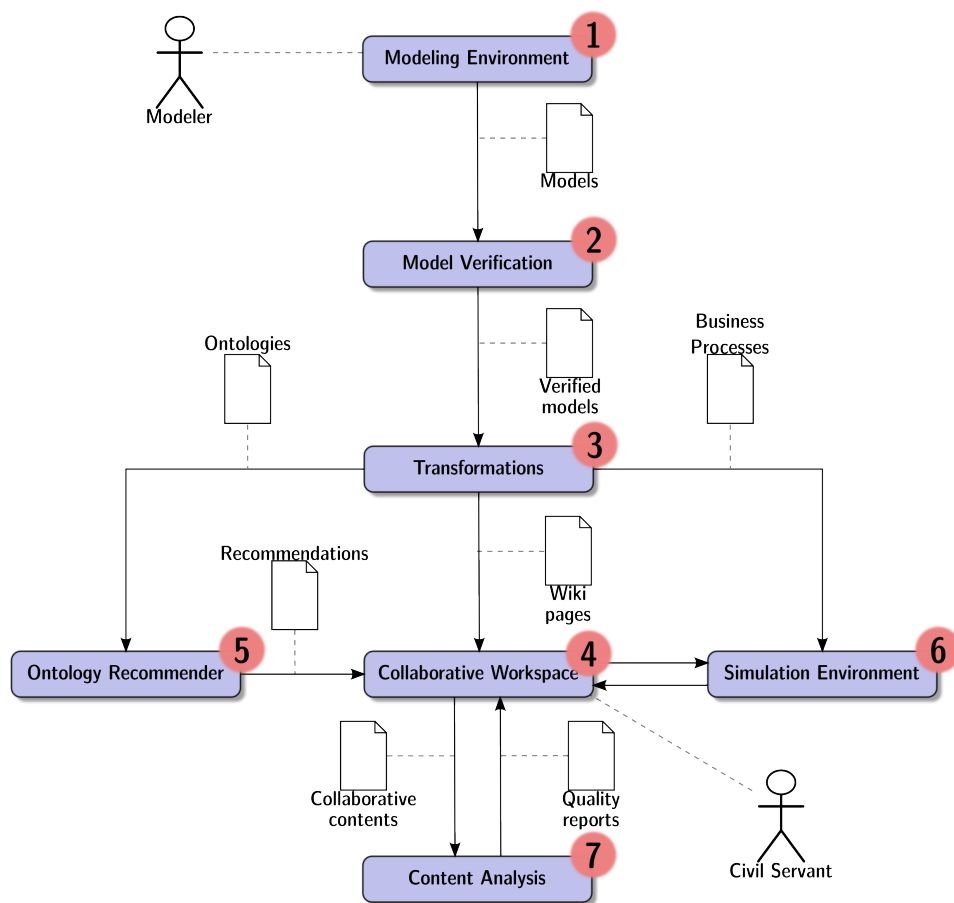


Figure 2.1: Flow view of the Learn PAd platform

Behind the scene The Modeler will work on the Modeling Environment component [3.2]. In Learn PAd, two of these components have been tested: Adoxx from BOC, and MagicDraw from No Magic. Both these softwares have implemented specific meta-models for Business Process, Organization Structure, Documents and Knowledge, Competencies, Case Management, KPI, Motivations. On top of that, they also include a weaving meta-model that links all these meta-models together.

2.3.2. Publishing the models

Once the models has been finalized, the Modeler will publish them in order to make them accessible to the Learners.

Behind the scene Once the Modeler decide to publish the models, the Modeling Environment component [3.2] will push the models toward the Learn PAd Core Platform component [3.1]. The Learn PAd Core Platform component [3.1] will have to check the models then import them into the other components (more details about these processes in the next sections).

2.3.3. Verification of the models

When the Modeler is publishing the models, the Learn PAd Platform may reject them because of errors. In fact, the Learn PAd Platform will run a list of automatic verifications to check the consistency of models.

Behind the scene In the Learn PAd Platform, a Model Verification component [3.7] is in charge of formal verifications of the models. It will provide help to the Modeler by spotting errors and providing warnings. Once Learn PAd Core Platform component [3.1] has check that models don't contain errors, it will pushed them towards other component of the Learn PAd Platform, mainly Collaborative Workspace component [3.3], Ontology Recommender component [3.5] and Simulation Environment component [3.6] (more details on these component in the following sections).

2.3.4. Resolve feedbacks

Civil Servants may spot inconsistencies in the models. In this case, they'll fill a feedback intended for the Modeler. When starting his modeling tool, the Modeler will see these feedbacks, he will either fix them or reject them, then publish the new model toward the Learn PAd Platform.

Behind the scene Civil Servants can fill a feedback from the Collaborative Workspace component [3.3]. The Modeling Environment component [3.2] will ask feedback to the Learn PAd Core Platform component [3.1] which will gather them from the Collaborative Workspace component [3.3]. Once the Modeler has modified and fixed the models, then the publish procedure will be the same as already described in Section 2.3.2. See also Section 2.4.5 for how Civil Servants are creating feedbacks in the Collaborative Workspace component [3.3].

2.4. Learner use cases

2.4.1. Discover a new process

At some point, a Learner will want to learn about a new process from the Public Administration. Once the Learner is logged onto the Learn PAd Platform, he'll see a list of the available processes on his dashboard. He'll be able to navigate through the process, going forward and backward in the series of activities of the process and all of the contextual information linked to this process.

Behind the scene The models are imported into the Collaborative Workspace component [3.3] which is wiki-based. When the Learn PAd Core Platform is pushing models towards the Collaborative Workspace component [3.3], the models are transformed into wiki pages by a Transformations component [3.4]: each element of the model is becoming a structured wiki page. Once the models have been imported into the Collaborative Workspace component [3.3], the Learner can navigate through the models directly in the wiki.

2.4.2. Getting recommendations

In order to improve the learning experience of the Learner, the wiki will also display recommendations to the Learner as he browses through the models in the wiki. Recommendations could be about people to get help from, existing example that could help to understand the process, multimedia materials, etc.

Behind the scene In the Learn PAd Platform, the Ontology Recommender component [3.5] is able to give recommendations based on contextual information from the browsing of the Learner. For example, information about who he is and what he's looking at could be used to suggest colleagues that may help him on this particular activity of the learned process. In order to infer recommendations from the contextual informations, Ontology Recommender component [3.5] is transforming models into an ontology through a Transformations component [3.4].

2.4.3. Collaborate to the documentation

At some point, the Learner will get experience on the process and may point out the weaknesses of the documentation. Looking at a particular activity in the process, he'd like to complete with some pertinent information. He can create a new document which will be linked to this specific activity of the process, and write, based on his experience, new content that could help other Learners. Note that everybody is able to collaborate so the documentation of the Learn PAd Platform is not only tight to the Modeler but also to the collaboration of every Civil Servant in the Public Administration.

Behind the scene In the Collaborative Workspace component [3.3], each wiki page will display a possibility to create a collaborative content attached to this page. The collaborative content will create a new wiki page that will be editable by any Civil Servant willing to improve the documentation.

2.4.4. Reviewing collaborative content

Collaborative contents give the ability to Civil Servants to cooperate in the improvement of the documentation of the processes. Learn PAd Platform will execute automatic text verifications in order to improve the quality of the produces content. The results of these executions will be shown to an experienced Civil Servant who may improve the collaborative content accordingly or reject the propositions.

Behind the scene Learn PAd Platform provides a Content Analysis component [3.8] which is able to run semantic analysis on the textual content and detects text vagueness. A specific Graphical User Interface provide a review workflow inside the wiki, only available to Civil Servant experienced on the given process.

2.4.5. Formulate a feedback to Modelers

Even if Learners can improve documentation with these collaborative content (see previous section), it's also possible they think there is a mistake in the model and would like to contact the Modeler in order to fix it. Like collaborative content, each wiki page is providing the functionality to write a feedback

directly to the Modeler. Usually, these functionality will be restricted to any experienced Civil Servant on the concerned process.

Behind the scene Based on the role of the user, his experience of the process, he will or not be able to propose a feedback to Modelers. When the Modeler will start his modeling tool, the Modeling Environment component [3.2] will ask Learn PAd Core Platform about any new feedback to display to the Modeler.

2.4.6. Simulate a process

Learn PAd Platform provides a way to run a process which is very efficient way to learn. The Learner will be taught by simulating a process with simulated data; he'll be able to execute the activities of the process, do his own mistakes.

Behind the scene From the Collaborative Workspace component [3.3], the Learner will be able to access a simulation of the process. The process is simulated by the Simulation Environment component [3.6], which provides its own Graphical User Interface embedded into the wiki. The Simulation Environment component [3.6] uses documents and data in order to execute a full process. During the simulation, two things are happening:

- the context of the simulation (who, what is currently simulated, what are the data, etc.) is sent to the Ontology Recommender component [3.5] in order to provide real-time recommendations to the Learner during the simulation
- the Learner will be monitored during the simulation (time of achievement, number of successes and errors, etc.) by the Monitoring Component which is part of the Simulation Environment component [3.6]

3 Learn PAd platform: solutions and technologies

In the previous section, we explained what were the functionalities of the Learn PAd Platform and how the users were interacting with it. A lot of different components are implied in these processes. This section will present the different components of the Learn PAd platform. This component view will show which are the implemented components and how they are connected to each other into the Learn PAd platform (see Figure 3.1).

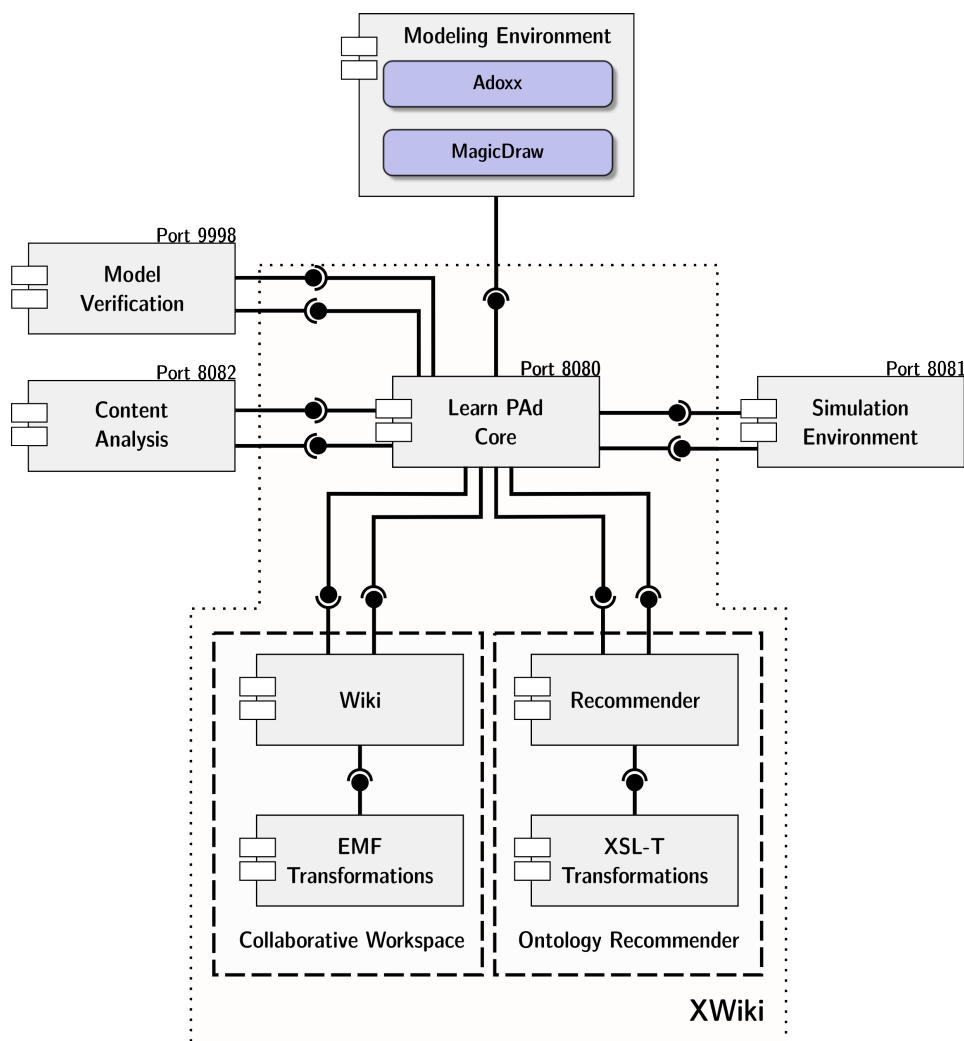


Figure 3.1: Component view of the Learn PAd platform

3.1. Learn PAd Core

Description Learn PAd platform has been designed as modular as possible. Everything is plugged onto this Learn PAd Core component. In order to be independent as much as possible from the technologies used in each component, each component is communicating with the Learn PAd Core component with REST APIs.

This component is developed inside the XWiki software. This means that storage is shared with all components that are part of XWiki. However, it communicates with all the components with REST which make it a replaceable component.

Provided services

- Orchestrator for inter-component communications
- Storage system for the Learn PAd Platform¹

Technologies

Java for platform server development

Jax-RS/Restlet Framework to implement REST services

Jax-B Framework to serialize Java class to XML

Component Manager XWiki library for a component-oriented architecture

Jetty Platform webserver

3.2. Modeling Environment

Description Modeling Environment component is a specific component. It's used for modeling but once used, it can be shutdown. However, the Learn PAd Core component is assumed to be always on-line. Therefore, communication between these two components are always a one way communication: the Modeling Environment component send or ask information to the Learn PAd Core component.

The format Modeling Environment component is pushing toward the Learn PAd platform is a ZIP archive (files *.lpzip) which contains:

- Set of models [format:XML/Adoxx or XML/MagicDraw]
- Images [format: BMP, JPG, PNG, SVG]
- Images map [format:HTML-Map]
- Business Process models [format:BPMN 2.0]

Provided services

- TO BE DONE

Technologies

NoTech TO BE DONE

¹Note that each component may also have its own storage system.

3.3. Collaborative Workspace

Description The Collaborative Workspace is divided into two main components. Since the Learn PAd Core component is not doing any transformation on the files (Learn PAd Core component is more like a router), the Collaborative Workspace component is getting raw files. Therefore, the Wiki component is downloading the raw files from the Learn PAd Core component, then use an EMF Transformation component (see Section 3.4) to transform models into wiki pages.

The Wiki component is the most important part of the Collaborative Workspace component. The Graphical User Interface will display models. Wiki component will download models, ask for transformation, and import the resulting wiki pages from the transformation.

Provided services

- Graphical User Interface for Civil Servants
- Embedding simulation in the Graphical User Interface
- Provide widget for recommendations to the Civil Servants

Technologies

Velocity/Groovy Scripting languages for server-side rendering in the wiki

HTML5 Web technologies for interactive applications in the wiki

Bootstrap for client-side web interface design

Java for platform server development

3.4. Transformations

Description The models respectively, produced by the ADOxx, and Magic Draw Modelling Environment are transformed to the corresponding Wiki pages by means of techniques and tools provided by the Eclipse Modeling Framework (EMF). Such models, describe the same business process, and are not directly comparable due to the different standard representation. In order to normalize the artifacts, UDA developed a transformational architecture which provide an Ecore representation for them and bridge the abstraction distance between the Learn Pad models and the wiki structures. The architecture performs several operations, among them:

- a model-to-model transformation
- a model-to-code transformation in order to produce the XWiki structure which are sent to the Wiki component.

Provided services

- Pre-processing of models for preparation to EMF parsing
- Transformation of models from one meta-model to another one
- Serialization of model

Technologies

EMF (Eclipse Modeling Framework) is a modeling framework and code generation facility for building tools and other applications based on a structured data model

ATL (Atlas Transformation Language) is a model-to-model transformation engine where a source model is translated into a target model Mb according to a transformation definition

Acceleo is a model-to-code transformation engine that produce the code according to a template-based rules

3.5. Ontology Recommender

Description The ontology recommender component provides a combined ontology representation of the Learn PAd meta models, models and their relationship. All models from the modelling environments can be transformed into instances of the platform independent meta models (LPIMM) represented by specific ontologies for BPMN, BMM, KPI's etc. With rules the associations to the conceptual models (LCIMM) are inferred and recommendations of experts, learning objects and learning material are provided in a context-sensitive and personalized manner. The integrated case based reasoning (CBR) component suggests similar cases to learn from past experiences.

Provided services

- Modelset transformations: A new set of models from the modelling environment is transformed and stored in the ontology as instances
- Context based expert recommendations (ex. performers most often executed a task, officers with same role in another organisation unit, etc.)
- Recommendation of learning material supporting learner to reach next level of competency (derived from learners acquired competencies and the required competencies given by the role or organisational targets)
- Retrieval of past cases and knowledge (experiences) to learn how to treat a similar new case

Technologies

RDF/RDFS/OWL Ontology languages used to model and represent knowledge of Learn PAd²

SPARQL W3C recommendation and query language for RDF applied to query the ontology for recommendations³

SPIN Inferencing rules language based on SPARQL and applied to inference associations from the imported models and the conceptual enterprise model⁴

ArchiMEO Extended enterprises ontology representing mainly the OMG Archimate enterprise architecture plus additional enterprise concepts

XSLT Used to transform model set exports from XML into instances of the platform independent meta model⁵

ICEBERG An ontology based CBR component with an annotation based configuration developed by the FHNW

²<https://www.w3.org/TR/2004/REC-owl-ref-20040210/>

³<http://www.w3.org/TR/rdf-sparql-query/>

⁴<http://spinrdf.org/>

⁵<https://www.w3.org/TR/xslt20/>

3.6. Simulation Environment

Description The simulation framework provides the subsystem where learners can simulate Business Processes interactively and is used by one or multiple civil servant(s) in order to learn processes. It proposes a simulation platform providing a flexible and interoperable simulation environment with support for collaboration. The simulation engine supports both the software emulation of the involved parties, and the provisioning of dedicated means for gathering learners willing to train on the business process by interacting with other learners. The infrastructure also includes monitoring feature providing feedbacks for the evaluation of learners, business processes, and associated learning contents.

Provided services

- BPMN process simulation session management, in single-user, collaborative, or mixed modes
- graphical interface for interacting with a simulation session
- Automated tasks verification and validation
- Simulation sessions dynamic monitoring via events notifications

Technologies

Java for simulator server development

Junit for automated testing

Activiti engine for BPMN execution

Jetty for simulator webserver

Resteasy for REST API creation and consumption

Jackson for object/json conversion between client and server

HTML5 for client-side web interface

Bootstrap for client-side web interface design

JSON form for client-side form generation

Glimpse for monitoring messages processing

ActiveMQ for monitoring messages dispatch

3.7. Model Verification

Description The Model Verification component is in charge of verifying correctness of models with respect to relevant structural properties such as deadlock and livelock. The components applying well studied reachability analysis algorithms permits to assess the models also with respect to model related properties. In particular the component includes a plug-in based architecture that permit to add new properties according to different model domains. The component in case a issue on the model is identified returns back a clear description of the problem to be presented to the modeler.

Provided services

- Model verification service
- Verified properties extension

Technologies

JavaEE platform as base programming framework.

JAX-RS technology used to define the REST service interfaces

JAX-B Framework in order to manage XML input/output of the services

Grizzly server used to deploy the services

Apache common exec framework in order to manage the execution of the external model checker
LOLA

LOLA (Low Level PetriNet Analyser) model checker in order to perform formal verification of LTL/CTL properties over a petri net

3.8. Content Analysis

Description The Content Analysis component is a Web application with a REst interface, to analyse natural language content according to six quality criteria (Correctness, Presentation Clarity, Completeness, Simplicity, Non Ambiguity and Content Clarity). Given a text, produced by civil servants and content managers, the component checks for language defects and, for each defect, returns the defective part of the text, together with a suggestion for improvement.

Provided services

- Detection of textual defects concerning Correctness, Presentation Clarity, Completeness, Simplicity, Non Ambiguity and Content Clarity
- Suggestion for defects removal

Technologies

LanguageTool a class library used to check the Correctness of a sentence, in terms of grammar.

GATE a class library implemented in Java for processing natural language. It is used to check Simplicity, Non Ambiguity and Content Clarity.

JSOUP a Java library for working with HTML. It is used to parse HTML Content for the Presentation Clarity and Completeness.

4 About partners

4.1. Consiglio Nazionale delle Ricerche

Leading partner of the Learn PAd project, CNR is an Italian Public Research Institute. Learn PAd is carried out by Software Engineering and Dependable Computing (SEDC) and Formal Methods and Tools (FMT) laboratories, that provide the Content Analysis and Model Verification components, as well as monitoring for simulation.

Competencies

- Identification of defects in natural language text
- Formal verification of business processes
- Monitoring of business processes and learning assessment

Contact

Stefania Gnesi stefania.gnesi@isti.cnr.it or +39 050 621 2918



Consiglio Nazionale
delle Ricerche

<http://www.cnr.it>



<http://fmt.isti.cnr.it/doku.php>



ISTITUTO DI SCIENZA E TECNOLOGIE
DELL'INFORMAZIONE "A. FAEDO"

<http://www.isti.cnr.it/>

4.2. BOC Asset Management GmbH

Based in Austria, it's a leading company in Modeling. It's providing the software Adoxx for the Modeling Environment component.

Competencies

- TO BE DONE

Contact

TO BE DONE TO@BE.DONE or PHONE



<http://www.boc-group.com>

4.3. Linagora Grand Sud Ouest SA

LINAGORA is a French leading company in Open Source Software and a Professional Services Provider. Its novice support solution (OSSA, Open Source Software Assurance) is a unique solution on a French market that provides a full line of services available through our one-stop shop in order to secure the FLOSS oriented information systems.

Competencies

- Research projects management
- Distributed and collaborative systems
- Business process related technologies
- Open Source projects management and dissemination

Contact

Jean-Pierre Lorré jplorre@linagora.com or + 33 688 346 385



<http://www.linagora.com>

4.4. No Magic Europe UAB

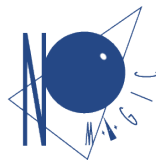
No Magic Europe is one of the leading software development companies in Lithuania. No Magic is best known as the vendor of the popular UML-based modeling platform MagicDraw TM. No Magic Europe provides custom software development services, professional training and consulting services focused on modeling languages and techniques. No Magic also plays a key role in Object Management Group consortium.

Competencies

- Business Architecture and Design
- Software Engineering
- Application and Process Integration
- Model-based Systems Engineering
- Experience in industrial training, including e-learning offerings
- Experts of OMG standards and modeling languages. Active roles in BPM practitioner communities and OMG standardization working groups

Contact

TO BE DONE TO@BE.DONE or PHONE



<http://www.nomagic.com>

4.5. Regione Marche

Marche Region is the institutional government of Marche Region, one of the 20 Regions of Italy. As a partner of the project which represents the public administration needs, we are experimenting the LearnPAD strategy and tools to enhance the organizational behaviours of the OSSs settled in our territory.

Competencies

- a specific knowledge on the business processes related to the OSSs (SUAP, in italian) and the experience of our local government bodies coordination board about:
 - standardization and simplification of forms and processes in favour of SMEs
 - implementation of the Marche digital agenda and of calls for funding for local authorities projects in order to increase digital growth
- a significant expertise of our PA School both in the management and assessment of training activities as well as in organizing specific courses (in elearning and in presence) for the OSSs
- technical skills of our ICT department (eg systems, organizations and workflows analysis; web and application development; accessibility and usability of graphical User Interfaces; help desk and PC maintenance; open source and fee software procurement management; installation and maintenance of information systems, social and collaborative platforms, web sites and content management systems; cloud data center, virtual and physical systems and networking management)



<http://www.regione.marche.it>

Contact

Andrea Sergiacomi andrea.sergiacomi@regione.marche.it or +39 071 806 3504

4.6. Fachhochschule Nordwestschweiz

Swiss University with expertise on ontologies. It's working on the recommendation engine.

Competencies

- TO BE DONE

Contact

Sandro Emmenegger sandro.emmenegger@fhnw.ch or +41 62 957 25 08

Barbara Thönssen barbara.thoenssen@fhnw.ch or +41 62 957 24 47



<http://www.fhnw.ch>

4.7. Università degli Studi di Camerino

University of Camerino is one of the oldest university in Italy, and is located in the Marche Region. The Computer Science division of University of Camerino has developed a considerable experience in exploring the role of formal analysis in particular when applied to Business Process Models.

Competencies

- Formal verification of BP models
- Modelling expertise for PA processes

Contact

Andrea Polini andrea.polini@unicam.it or +39 0737 402563

Barbara Re barbara.re@unicam.it or +39 0737 402524



<http://www.unicam.it>

4.8. Università degli Studi dell'Aquila

The University of L'Aquila (UDA) has developed a solid research and experience in the model-driven area (software engineering) on the metamodeling field, model manipulation and analysis. UDA concentrates his research in the specification of transformations and their application to several problem domains: web engineering, software architectures and middleware-based communication systems.

Competencies

- metamodeling and general model management
- the definition and instantiation of the Learn PAd metamodel that permits the specification of extended business processes in the domain of Public Administrations
- a transformational architecture where models produced in the different modeling environments are processed in order to generate the wiki structures associated to the business processes
- the adoption of state-of-the-art techniques and tools for designing and implementing model-to-model transformations
- the adoption of state-of-the-art techniques and tools for designing and implementing model-to-code transformations

Contact

Alfonso Pierantonio alfonso.pierantonio@univaq.it or +39 3283332668

Francesco Basciani francesco.basciani@graduate.univaq.it or +39 3336777151

Gianni Rosa gianni.rosa@univaq.it or +39 3803288267



<http://www.disim.univaq.it/main/index.php>

4.9. XWiki SAS

French open-source company, which is developing XWiki software. It's providing XWiki as the base of the Learn PAd platform and expertise about collaboration and knowledge management.

Competencies

- Expert on XWiki software used as basis for Learn PAd Platform
- Specialized in Collaborative Workspaces and Knowledge-Base Wikis
- Software development and integration
- Open-source software and community management

Contact

Jean Simard simard.jean@xwiki.com

Caleb Delisle calebjamesdelisle@xwiki.com



<http://www.xwiki.com>