DELIVERABLE D2.6

FINAL SYSTEM RELEASE

Document Version:	v2.2 – 11 October 2016
Document Status:	Final
Document Type:	Report
Diss. Level:	Public
Lead Partner:	TUGraz
Delivery Date:	M31 (September 2016)
Authors:	Michael Kickmeier-Rust, Arslan Khurshid, Kolja Debus, Angela Fessl

TABLE OF CONTENTS

1. INTRODUCTION

In the very heart of WP2 is the design and development of a Learning Analytics web portal. Based on the requirements specification document, the focus groups and design studies, we designed, planned, and developed such platform and had a second and in principle complete release in M22. In the focus of Y3 was a polishing of the functionality in terms of stability and computational performance as well as a completion in terms of features.

The entry portal is available at http://css-kmi.tugraz.at/mkrwww/leas-portal2/.
Demo login credentials are User: Lea, Password Inthebox

Please note: The project's prototype release supports Google's Chrome browser only, it has, however, widely been tested with Firefox and Microsoft Edge browsers. There are several known issues with Microsoft's Internet Explorer.

2. DOCUMENTATION

The general look and feel and the basic design didn't change. There is a central landing page that provides access to all the functions of the portal. The design is a typical "metro style" design. This design is intended to experience the system a bit like Windows 10 and its tiles of functions and features. This design is easily customizable. A new feature developed in year 3 was the option to configure the tools in the My Tools section individually through the configuration tool (cf. to the manual for details). This allows the right configuration for each user type; for students, for example, certain tools can be hidden, and teacher can select only the needed tools. In a future step, in the context of exploitation, the configuration can easily be done with a wizard.

The main menu grants access to the following components:

2.1. MY TOOLS

The first, and perhaps most important, feature is the "My Tools" tile. This functionality allows users to define a set of internal and external tools to primarily record and gather data for the analytics functions and the open learner modelling. The final system release within the project includes the following tools:

- **Lea's API Interface:** The portal features a well-defined open API to allow external tools to communicate with the portal. This tool is a demonstration and test mechanisms for the use of the API. The API as well as the tool are described in detail below!

- **myClass light:** This tool is a version of myClass with reduced functionality but in turn fully integrated into the design of the portal. The purpose of this tool is to allow teachers to make manual records about learning progress and competence developments.

  When selecting the tool, the teacher can select one of the assigned subjects (meaning that a teacher cannot access the subjects of other teachers).

  A teacher first selects the group and the subject from the menus and finally a student. The tool lists all competencies linked to this particular subject. By clicking on the textbox, holding the value for each competency, the teacher can manually adjust the value with a slider control. Adjustments in this tool directly influence the analyses and the OLM.

- **myActivities light:** Similar to myClass light, this tool allows teacher to add certain activities manually. Activities are linked to competencies, which allows the CbKST-type analyses but activities can also be analyzed and displayed through the OLM.

  A teacher first selects the group, then the subject and finally the student. For this student the teacher can use the display to add or delete new activities. The activities listed here are those associated with this particular subject.

- **FCA Learning Analytics:** This tool allows teachers to define FCA structures and display the FCA analyses of the available data. The details are explained below.

- **Flower App Teacher View:** The flower app is a tool for providing a graphically neat and simple app for students' self-assessment (see description below). For the teachers, we provide the same functionalities and scales, only in a graphically neutral way.

  The teacher only selects the student group and an individual student. This enables the evaluation of the student along the same scales.

  In the following, the teacher can review the individual results of one students or the entire results set of all students. This module allows comparing the teacher' evaluation, the student's self-assessment, as well as the results in the official (external) test.


Assess: RUHAU TAUHUTAUL
Motivation
1 = strongly disagree
6 = strongly agree
1. In my opinion, English is the most important subject at school.
2. I enjoy learning English.
3. Out of all the subjects, I find it most important to improve my English.
Load
1 = strongly disagree
6 = strongly agree
1. I often feel I will never be able to learn English well.
2. It is difficult for me to learn new English vocabulary.
3. I believe it is more difficult for me to learn English than for most of my classmates.

2.2. LEA'S API

An open and accessibly system is a key success factor. Working with APIs inevitably means preparing an external tool for pushing data into the Lea's Box system. While **xAPI/Tincan** is a powerful standardized, however, rather complex approach, the simplest way of bringing data into the system is Lea's API, an internal API that can be configured easily.

API SPECIFICATIONS

Call URL	http://css-kmi.tugraz.at/mkrwww/leas-portal2/api/pushin.php
secrettoken	A unique character string; this character string is automatically generated when an external data source is registered via the configuration tool – module 'data source'. This value to avoid invalid entries by bots, etc. and to assure data are not compromised.
studentid	Identifier of a particular student in an external tool. This value is a unique character string. The string is specified when students are registered for external tools in the configuration tool – module 'external link'. This value is used in a scenario where platform features (e.g., OLM visualizations) are accessed within the portal and – in contrast to a looped service where anonymous data are retrieved, analyzed, and reported back to an external tool.
teacherid	Identifier of a particular teacher in an external tool. This is optional, if no specific teacher is specified, use the value '0'. This value is a unique character string. The string is specified when tool are registered in the configuration tool – module 'external link'. This value is used in a scenario where platform features (e.g., OLM visualizations) are accessed within the portal and – in contrast to a looped service where anonymous data are retrieved, analyzed, and reported back to an external tool.
datasourceid	Identifier (integer) of an external data source as specified in the configuration tool – module 'data source'.
activityid	Identifier (integer) of a specific activity in an external data source as specified in the configuration tool – module 'activities'.
value	The principle of Lea's API is that an external tool can pass any value (which may vary substantially; it can be a right/wrong
minvalue	

maxvalue	statements, the time spent on a task, school marks, the number of clicks in an app, etc.) into the system. This value is then 'normalized' by the related minimum and maximum values. This offers a percentage range which gives an indication about how 'well' an incoming information is and how much the *believe model* of the related competencies much be adjusted. The passed value can also be negative. With a single API call up to three such value/min/max combinations can be passed to the system.

The API call returns nothing (an empty string) in the case of success or alternatively an error message in case of failure.

API CALL CODE

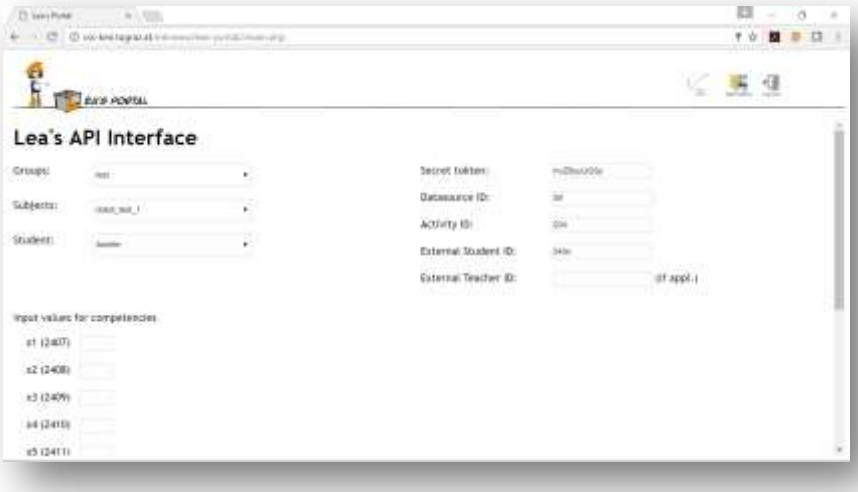
```
// Saving data through API
if($saveit!=0) {

                $sql = "SELECT * FROM subject_has_competencies WHERE subjectid = $subjectid";
                $result =  mysql_query($sql);
                if (!$result) {echo 'Error 1: ' . mysql_error(); exit; }
                $sum = 0;
                $cnt = 0;
                while($row = mysql_fetch_assoc($result)) {
                        $num_rows = mysql_num_rows($result);
                        if ($num_rows != 0) {
                                $cid = $row['compid'];
                                $cid1 = "min$cid";
                                $cid2 = "max$cid";

                                // Check if a value is oming for a given competency and sum the values up
                                if (isset($_REQUEST[$cid])) { $x= $_REQUEST[$cid]; } else { $x = ""; }
                                if($x!="") {
                                        $sum += $x;
                                        $cnt++;
                                }
                        }
                }

                // Make the average
                $x = $sum / $cnt;
                $x = round($x, 3);

                // Generate the API call
                $url = "http://css-kmi.tugraz.at/mkrwww/leas-portal2/api/pushin.php?";
                $url .= "secrettoken=$st&studentid=$stud&datasourceid=$ds&activityid=$aid&teacherid=$tid&value
    =$x&minvalue=$minx&maxvalue=$maxx";
                $ch = curl_init();
                curl_setopt($ch, CURLOPT_URL, $url);
                curl_setopt($ch, CURLOPT_POST, 1);
                curl_setopt($ch, CURLOPT_POSTFIELDS, $postvars);
                curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
                curl_setopt($ch, CURLOPT_CONNECTTIMEOUT, 3);
                curl_setopt($ch, CURLOPT_TIMEOUT, 20);
                $response = curl_exec($ch);
                curl_close($ch);
}
```

API TESTER

With this tool, Lea's Box provides the users with a needful feature to test the connection of external tools with the portal and allows an easy debugging in case of problems.

The API tester allows manually filling in the requested value. To facilitate the generation of API statements, the user can select groups, subjects, and students from menus. This automatically retrieves the correct ids. Also, the related competencies are listed (including displaying the correct internal id). The user can now manually add all relevant information in the given text boxes. The competency values can be filled in the text boxes manually or with a slider control (that pops out when clicking on a text box). AT the button of the form, the respective minimum and maximum values can be filled in. Clicking the push button at the bottom of the form triggers the API call and returns either a success message or the full error message.

Please note that the API tester affects the 'real' student data as stored in the system database! Do not test API calls with real students, use test students only! There is not direct undo function for API calls available!



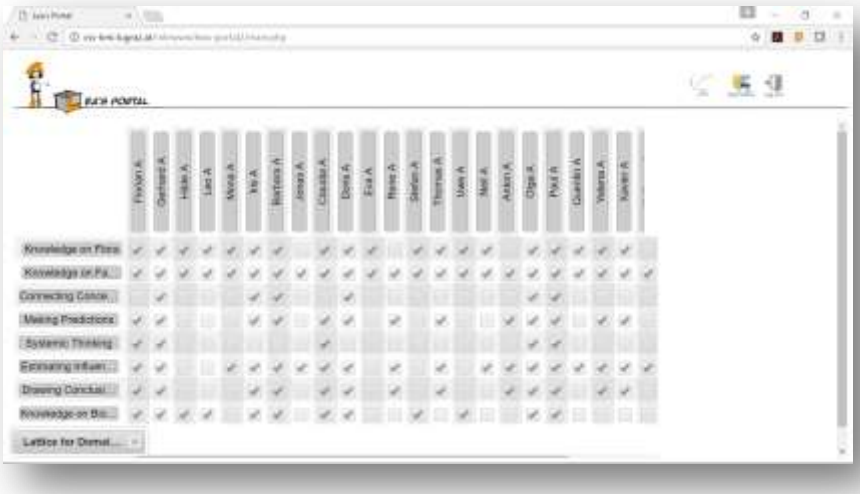
2.3. FCA LEARNING ANALYTICS

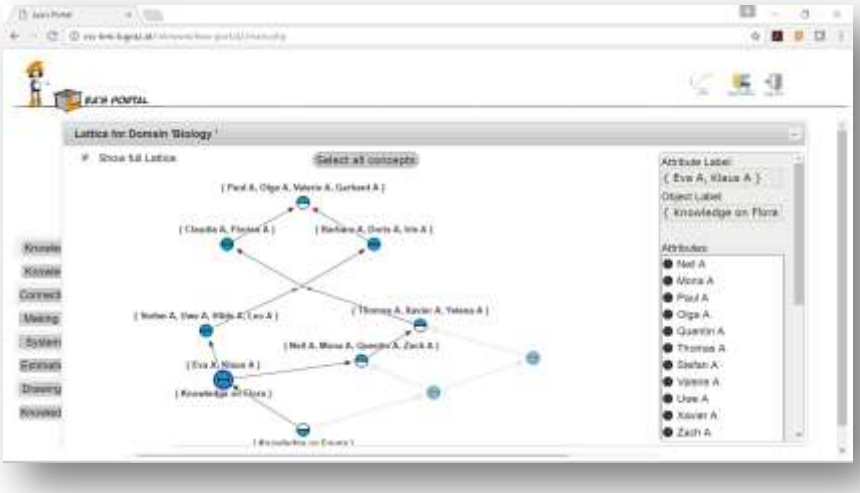
The FCA related tool conducts data analyses on the basis of the system database (and the available data) in real time. First, a teacher selects the group and the subject. Besides the field competencies appears the link ‘Show’. This link displays all competencies related to a given subject. A teacher can now define the competencies that shall be part of the FCA structure by checking the boxes. In many cases not all of a subject’s (e.g., 4th grade Maths in the summer term) competencies can be represented meaningfully using FCA. Often it is more reasonable to make the analyses only over smaller but related topics and sub-topics.

Based on the students and competencies, the tool computes the formal concepts for three distinct views:

- **Competencies x Students:** the formal context highlighting distinct clusters of which students and groups of students hold which competencies and competency cluster.
- **Competencies x Activities:** the formal context of clusters that indicate the relationship between activities and related competencies. This can be useful, for example, when planning of activities (e.g., exercise or homework); e.g., to assure that the entire competency structure is covered.
- **Students x Activities:** this module displays the formal context of activities and students indicating the distinct clusters of which students have already completed which activities. This can be useful for an adaptive group formation or the planning of activities (e.g., exercises or homework).

The formal context, that is computed in real-time, is represented as a matrix of concepts and attributes. This matrix includes already al information, however, sorting and ordering the clusters is a non-trivial problem. This is the particular strength of FCA lattices. This means, the matrix can also be displayed as the typical FCA lattice diagrams. Such visualization reads from bottom to top and indicate existing “concept x attribute” clusters and adds also such, that can be logically derived from the lattice paths but that do not actually existing in the data. The details about using the FCA tool are specified in the FCA Tool Manual that is attached as an annex.





2.4. LEARNING SPACES

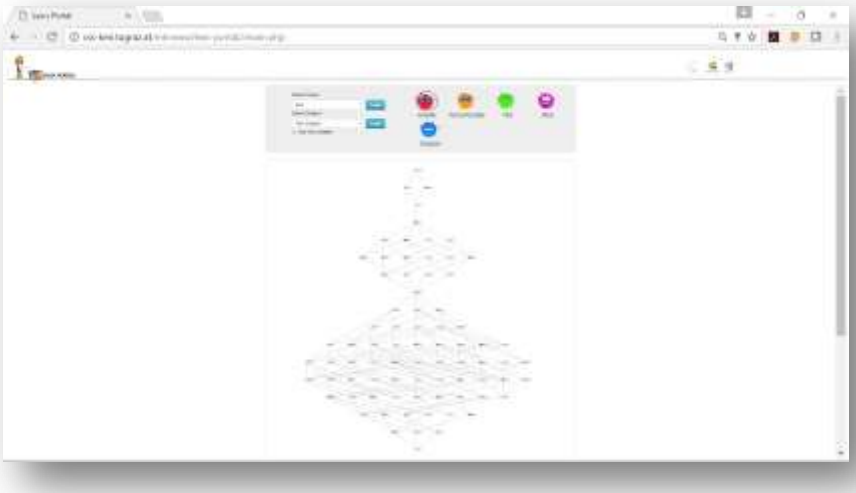
The next tile refers to the “Learning Spaces”, this is the feature that allows teachers to access the CbKST analyses visualization module as the front end for CbKST related analyses and visualizations. The diagrams represent the competence spaces of learners and their learning paths. Learning Spaces can be defined for groups of students and subjects, on the basis of the competencies defined for the subjects. CbKST-based analyses occur on the basis of the activity and achievement data held by the system.

LEARNING SPACES

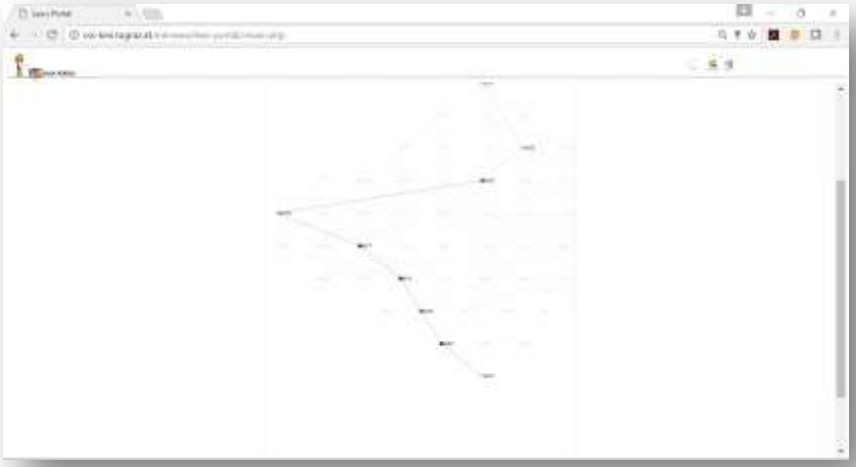
This module offers the classical Hasse diagrams of CbKST competence models. The teacher selects a group of students and a subject. The related competence structure is automatically generated on the fly and displayed as an interactive Hasse diagram. The diagram reads from bottom to top and it is composed of nodes and lines. The nodes indicate competency states and the lines the learning paths. The bottom most node is the start state of a competency model where, in most cases, none of the competencies of a subject are learned. The lines, subsequently, indicate the possible learning paths, whereas, not all transitions are possible but only such that are in line with the defined competency model. The node colors code the probability of a competence state (in other words, the likelihood that specific competencies are held by a learner, based on all the evidences in the system). The darker the color of a node, the higher is the likelihood.

> *[Color schemes can be adjusted in the code, by default the color coding ranges from hex colors #AAC4D1 (0%) to #07238C (100%).]*

At the beginning, as long as no student is selected, the tool automatically displays the average competency likelihoods for all students. If a student is selected, the values of this particular student are displayed immediately. The diagram tool is interactive, so scrolling, panning, and zooming is possible. The displayed competency names are cut off (so that too long names cannot spoil the visualization). The full names can be displayed when clicking on the text in the diagram.

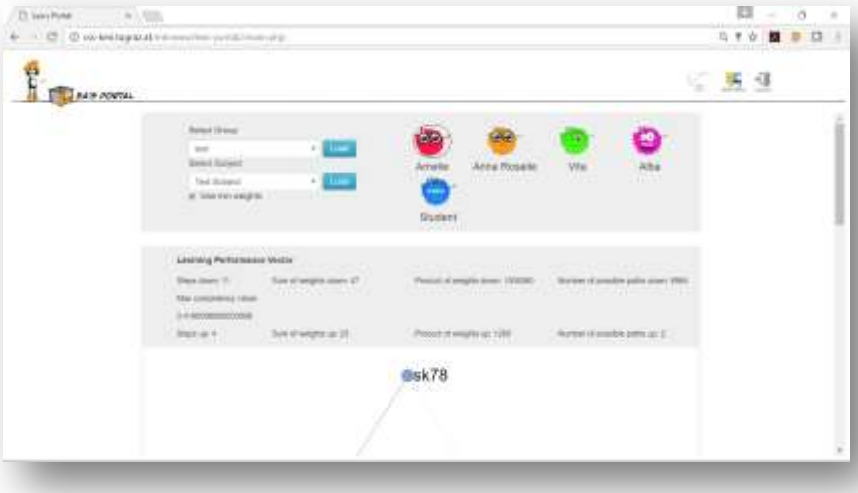


An important feature is displaying a student's learning paths. When clicking directly on a node, the likelihoods of learning paths are computed and the most likely paths is displayed. All other paths are greyed out. This path denotes the learning steps of a student from the start state to the current competency state. The learning path is indicated by the lines towards the bottom of the clicked node.



Another innovative output of Lea's Box is the so-called Learning Performance Vector (LPV) and the Learning Horizon. The LPV is a prediction of a particular student's chances to reach a final learning goal of a course. In contrast to many existing prediction algorithms, the LPV approach is based on the CbKST-approach paired with the OLM-type weighting of activities and competencies. A detailed description of the approach is reported in deliverable D3.5. In simple words, the algorithm compares the learning steps of a student taken in a course so far with the remaining necessary steps to reach the final learning goal, depending on the remaining course time. The Learning Spaces tool provides two versions: (i) by default the most expensive paths is computed, based on the number of steps to the goal and on the basis of the weights of the involved competencies (in other terms, their difficulty). In other terms, this is the path that requires the most learning efforts. Alternatively, (ii) the least costly path can be calculated. For this, the user can tick the box 'use min weights' in the head section of the tool.

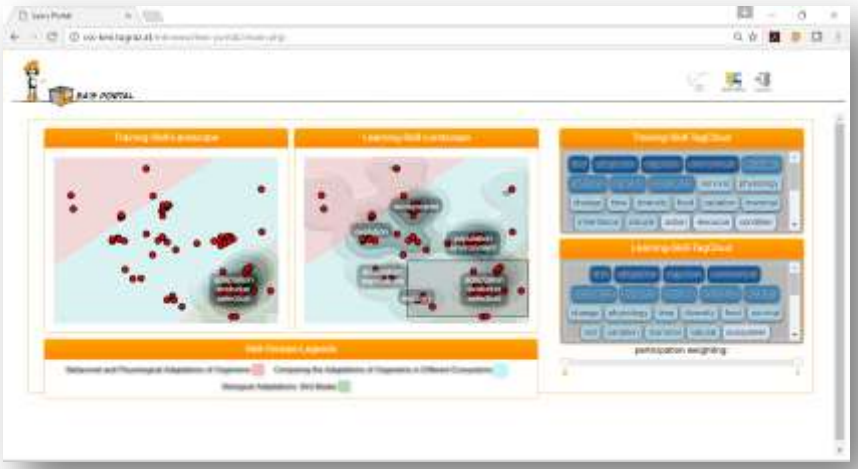
The respective path is immediately displayed as dotted line towards the top of a clicked node. Along with this visualization, a box appears reporting the key variables of for the algorithm (number of steps, weights, products of weights, competency values). This box also displays the Learning Horizon, the aggregated value indicating as student's chance to reach the final goal.



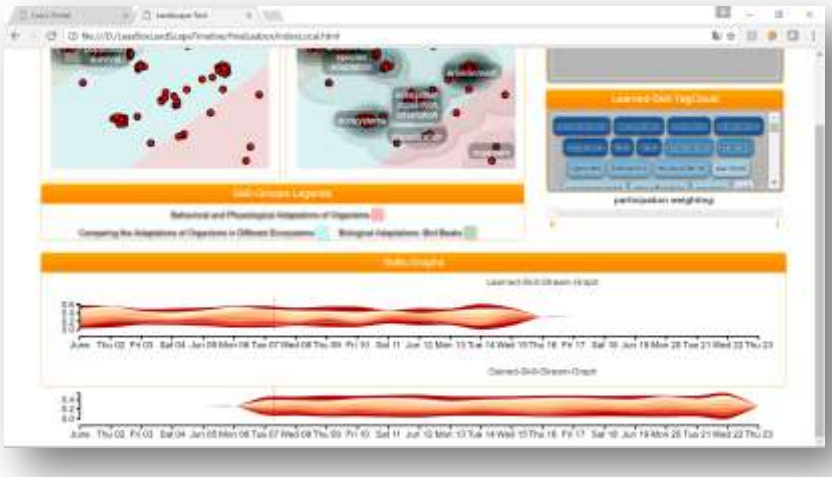
LEARNING LANDSCAPE

The Learning Landscape tool is a method to display CbKST-type structures in form of 3D-ish landscape visualizations. The reason for providing such alternative form of visualization is that the typical Hasse diagram and FCA lattice visualizations are very difficult to read and understand by novices. The Learning Landscape tool separates learning from assessment

activities and displays the attendance to learning objects and activities and elements providing evidence for competencies (i.e., internal and external activities). The individual elements are displayed as red dots which are order and placed on the plane based on the value (e.g., number of attendances) and conceptually (in an FCA sense). The coloring of the landscape background indicates the values; high values are displayed as peaks and low values as valleys.



A new feature added in Y3 is a time-based analysis. Along with the focus on temporal analyses of the FALSH data analysis warehouse, the Learning Landscape now displays the temporal sequence of learning and assessment activities. By this means the user can identify the intensity and progress of learning activities. A detailed description of this tool is attached as external annex.



2.5. MIND MAPPING TOOL

The mind mapping tool can be used for making open projects and structuring competence domains. This can be useful for teachers and students. The final release allows using the tool to create hierarchical competency models. A detailed manual is attached as an external annex.

2.6. OPEN LEARNER MODELLING

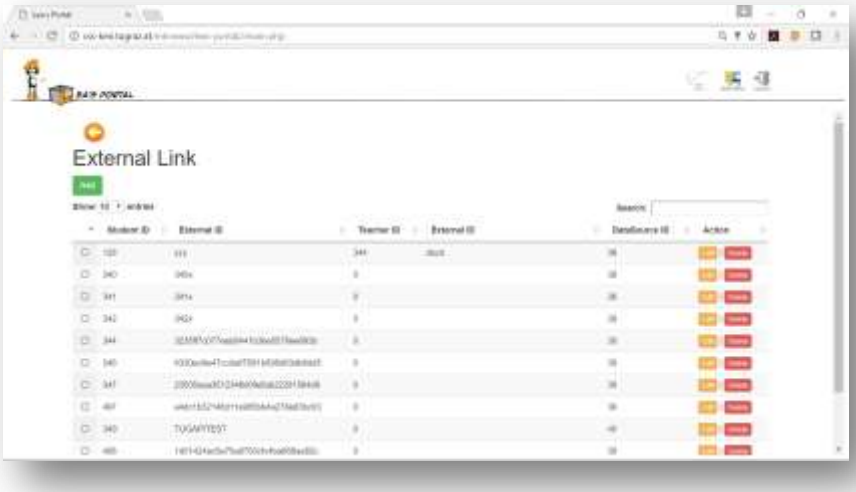
A very prominent feature of Lea's Box is the OLM, the Open Learner Modelling tool. It is the key instrument to display the analyses in a suitable way for users such as teachers, learners, or parents. The tool itself has new visualization services and the new release includes negotiation features that allow students to negotiate the results of analytics. The details are described in deliverable D4.5.

2.7. SETTINGS MODULE

The settings module is now a complete and fully integrated tool to make all relevant configurations and setup work. For tasks that usually are demanding and are based on many data sets, such as students or competencies, batch import features of external csv files are available. Although the tool is complete, it widely cannot be considered particularly user-friendly. This however, is out of the scope of the project and a clear task for exploitation

activities beyond the project.

The detailed manual for the configuration tool is available as an external annex.

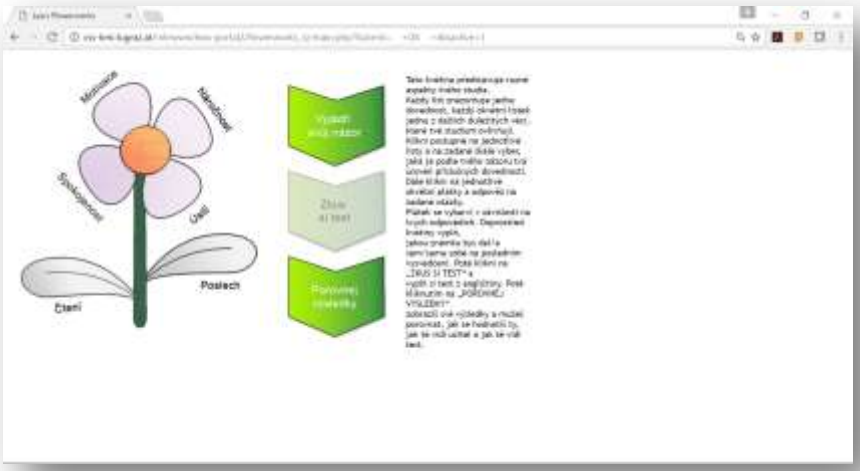


2.8. INFORMATION TILES

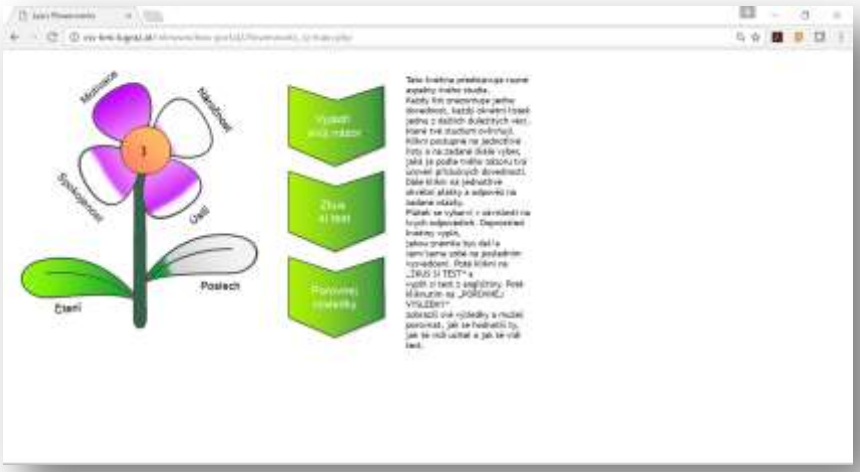
In addition to the core functionalities, the new system provides tiles for access further information and help. The Info tile leads to background information of the project, the help tile grants access to all manuals of the (internal) tools. In addition there is a contact tile and a link to the projects Facebook site. This site (/LeasLearning) is a very active hub of recent information about Learning Analytics and an easy way to get in touch with the project.

3. THE FLOWER APP FOR STUDENTS

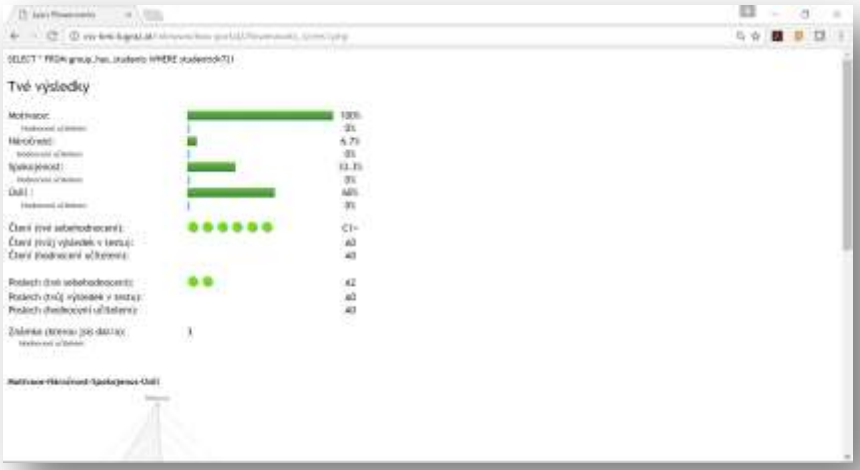
The flower app is a student self-assessment tool that has been developed in 2 versions. In year 2 we used an initial version of the flower app for pilot studies in the Czech Republic and in Austria. This app comprised *English as second language* and, in Austria, *Biology*.



The fundamental idea of the app is to provide students with a nice graphical interface for conducting self-assessments in a specific topic. The leafs of the flower indicate sub-topics; when clicking on a leaf, the student can access a questionnaire for this specific topic and can assess herself. Depending on the value of self-assessment, the leafs fill with color (a very positive assessment leads to a fully filled and colored leaf).



The students can then take an official external test. This is realized by linked the SCIO *Testovani* product, a commercial test app, widely used in the Czech Republic. Using the Lea's API, the results are fed back to the Lea's Box system. Students can then display and compare their self-assessment with the test score and also with the teacher's evaluation.

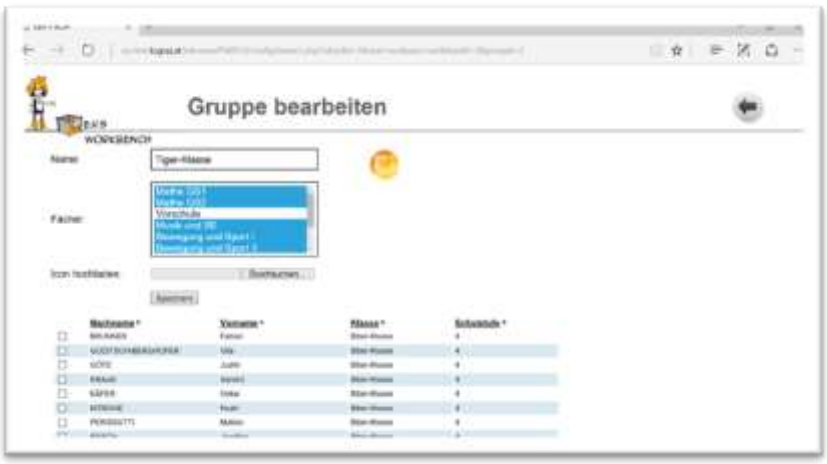


In year 3, we revised and extended the and, on the basis of the year 2 feedback, we slightly re-designed the app. The new flower app come with a greenish design and it is tailored to be used with more than one topic. For the year 3 pilots in the Czech Republic we will use Maths, English, Czech, and Meta-Competencies. The principles and functionalities are widely the same.

4. MYCLASS

So far, an integral part of LEA'S web platform was the myClass3 tool. It's tailored to the concrete demands arising from the focus group work with our partner school in Graz, Austria and also (even with a slightly different look 'n feel) our partner school in Stuttgart, Germany. The original tool stems from the earlier Next-Tell project and has been significantly advanced and improved during Lea's Box. We developed 2 versions, intended to facilitate an easy tracking of learning performance, the related analyses, and an individualized reporting of results. The tool has been described in the previous deliverables.

A significant extension was to provide the teachers with a tailored configuration tool (based on the modules of the system), that allows the school to use the system in an autarkic manner, independent from the project. Also, the schools host their system on an external, independent server. The features include the configuration of students, classes, subjects, competence models, teachers, indicial groups for certain projects. In addition, there is also a database backup function (teachers / editors were instructed to make backups of the database whenever they perform major changes with the tool to avoid any loss of data).



While myClass was only one part of the entire system, for the final release we took myClass out of the system (except for the light version, as described above). Instead we provide myClass in two preconfigured variants as an open source stand-alone tool. Due to the work in Next-Tell and the pilot studies conducted in Lea's Box myClass can be considered being on TRL 7 to 8. The tools can be downloaded from the Lea's Box platform, including example data and manuals. Of course, the tools provide the built in lnk to the Lea's Box system via

the Lea's API. In summary, myClass is one of the key elements of the exploitation strategy of the project.

5. DATA STRUCTURE AND FLASH DATA ANALYSIS WAREHOUSE

A, on the first sight simple, task is to define a data architecture that is capable of handling all the demands of various internal and external tools. A specific challenge is the handle the partly large data sets in a scalable and performant way. The project's answer to this problem is twofold. On the one hand we have a thoroughly and iteratively developed data model. On the other hand, for task and analyses that are computationally highly demanding, we put a data analyses warehouse solution in place.

A particular difficulty for the main data model lies in the heritage of OLM and myClass. Both are/were integral parts of the system, however, conceptual different. In addition we had to accommodate the demands from external tools and administrative needs of users. An additional challenge is the need for meeting all relevant data protection and data safety principles and regulations. The developed data architecture perhaps is one of the most precious outcomes of the project – providing a maximum of flexibility and performance at the same time. The following image illustrates the recent version:

The data architecture is based on the following logic:

- Learners are contained in groups
- Learners are considered internal and external entities
- Competencies are grouped into subjects
- Competencies can have conventional tree hierarchies as well as CbKST-type prerequisite structures

- Teachers are linked to subjects and groups
- Activities are linked to competencies
- Data sources (internal / external) are linked to activities
- Access rights are bound to schools – administrators – teachers (in this order)
- Performance data are stored as log data in encrypted form (in complog and activitylog tables)
- Data ownership is school entity or learner entity; the vulnerability to data drop out is minimized (e.g., when students withdraw their data).
- Peripheral apps and tools (e.g., the mind mapping tool or the flower app) use separated databases
- Performance relevant data are mirrored in the data warehouse

It shall be highlighted that in addition to the complexity of data that must be stored by the Lea's Box system, to meet all the requirements, we also have to think in terms of analytics. Many analytics and many data sets are computationally extremely demanding. This means, that a data warehouse approach must be included to account for the analytics that often must be done in real time. Consequently, the data architecture for performance data and evidences links to a data analysis warehouse. The principle idea is that performance data are stored redundantly in forms that match the analytics that are called by the users. For example, the data are stored aggregated and summarized over certain time intervals so that analyses and visualization can be realized in a highly performant way.

5.1. FAST LEARNING ANALYTICS FOR SUPERIOR HARNESSING (FLASH)

During the final year, a relational database for a joint analysis of all data was constructed. Main goal is to provide a flexible and lasting infrastructure for gaining insight into data from various and multiple sources.

Example analyses have been performed for data

- from the platform adaptvive curriculum (SEBIT)
- from the platform RAUNT (SEBIT)
- that had been pushed into the mylea_beta database by external tools and internally generated data from the portal

To achieve this goal, the database was planned as both relatively simple in structure and open-ended. Data to be pushed into the database should be easily identified regarding its origin, while remaining cleansed from all attributes that would prevent an analysis with data from other sources.

A full-fledged data warehouse, the best solution for achieving this, was deemed to costly in time, working hours and hardware to being finished during the lifetime of the project. A likewise approach was pursued in order to gain the maximum outcome for the resources at hand.

Regarding the original plan, the database was adapted during the final year as follows:

- additional information about students was made possible to be used as filter criteria or independent variables for analyses;
- groups were added as a dimension, as they showed to be important for users;
- schools or, more generally speaking, institutions/organizations were included, since they constituted a central orientation point for end-users, while being useful for research questions that are interesting for future use.

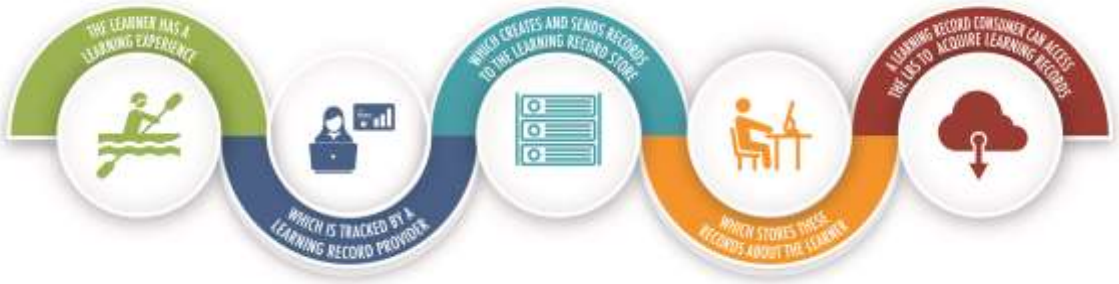
Summarized, the database services have been honed to firstly fit more snugly the structures that single end-users are familiar with from their daily practice. Secondly, additional layers of organizationally interesting dimensions have been added to expand the field of usage to administration.

6. CANVAS IMPORT TOOL

Canvas is a popular commercial learning management system (www.canvaslms.com). It is used by many schools and universities. In the context of the Lea's Box studies (the Italian language learning studies of Y2) we used data from the Canvas LMS. To have a quick and practical solution to get the log file data from Canvas into the Lea's Box system (without the need of implementing APIs at the Canvas side), we used a csv import tool. This tool is not an official part of the Lea's Box portal and there is no further support of the tool. On the other hand, however, the functionalities might be useful for other users. Thus, we provide the source code openly but without any warranty or liability. The package is available through the Lea's Box website. It includes sample csv files for the import.

7. xAPI / TINCAN

xAPI (and Tincan was the project within which xAPI has been developed and often is used synonymously) is a successor of SCORM and are quasi standards in the exchange of human performance data. xAPI is sometimes called the first step to the “next-gen SCORM” because it’s designed to achieve the amount of interoperability the was promised by SCORM. But where SCORM works primarily with learning management systems, xAPI supports more diverse platforms and less clean and straight-forward settings. The principle data flow is illustrated in the following image (courtesy of ADLnet).



In the context of Lea’s Box, this means that various data sources produce micro information (in principle log-type data, e.g., every click in an online learning system) which are passed to the Lea’s Box xAPI end point. An example statement is illustrated below. Technically speaking, the API is based on a JSON formatted statement following the xAPI standards and using the RESTful technology (http://restfulapi.net/).

To setup xAPI connectivity, a user must configure an activity provider which defines the call URL, a key and a secret token.

```
{
    "id": "deb23024-5a46-419a-aec3-8114c4667252",
    "actor": {
        "mbox": "mailto:m.arslan.khurshid@gmail.com",
        "objectType": "Agent"
    },
    "verb": {
        "id": "http://adlnet.gov/expapi/verbs/completed",
        "display": {
            "und": "completed"
        }
    },
    "timestamp": "2016-10-11T13:18:33.957Z",
    "stored": "2016-10-11T13:18:35.140Z",
    "authority": {
        "name": "Unnamed Account",
        "account": {
            "homePage": "http://cloud.scorm.com/",
            "name": "ECEwtxWLcSwnSfRCQvY"
        },
        "objectType": "Agent"
    },
    "version": "1.0.0",
    "object": {
        "id": "http://tincanapi.com/activities/sending-my-second-statement",
        "objectType": "Activity"
    }
}
```

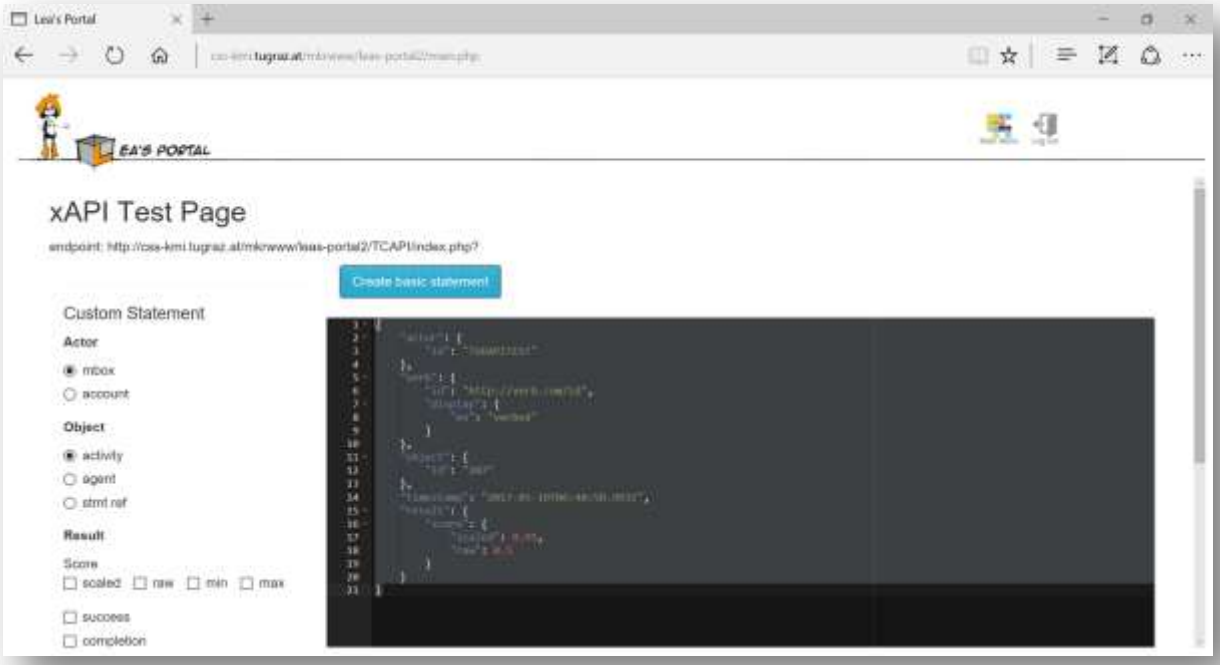

xAPI Statement

For setup and testing purposes we recommend the “SCORM Cloud platform” (which is a free tool to work with xAPI calls) with is an open tool for setting up, and retrieving xAPI calls (https://cloud.scorm.com/sc/user/LRSView).

xAPI TEST PAGE

With the final release of the system we provide an xAPI test page (see image below). With this tool a user can generate a basic xAPI statement I a form that is supported by the Lea’s Box xAPI endpoint. In this release, the Lea’s Box system has become a full xAPI data store (making aforementioned approach obsolete).

The test page allows experimenting and trying out different “recipes” to send data from different external tools. As shown in the screenshot, the tool enables to define actors, objects, results and contexts. Every recipe (i.e., form of xAPI statement) can be tested directly on with this tool. The result (success, failure, including description) is shown immediately. When a statement works, the user can copy the code and use it in their own external tools. The Lea’s Box endpoint supports the full standard, detailed information is given here: https://github.com/adlnet/xAPI-Spec/blob/master/xAPI-Communication.md



In principle we have 2 situations, a teacher has configured everything and sends results or a teacher hasn't configured all activities and external data sources.

CASE 1:

```
{
  "actor": {
    "name": "Sally Glider",     // This is our userid, isn't it? What would be best?
  },
  "verb": {
    "id": "http://adlnet.gov/expapi/verbs/completed",
    "display": { "en-US": "completed" } // We could support
  },
  "object": {
    "id": "http://example.com/activities/hang-gliding-test",
```

```
    "definition": {
      "type": "http://adlnet.gov/expapi/activities/assessment",
      "id": { "123" }, // The external activity id
      "name": { "en-US": "Hang Gliding Test" },
      "description": {
        "en-US": "The Solo Hang Gliding test, consisting of a timed flight from the peak of
Mount Magazine"
      },
      "extensions": {
        "http://example.com/gliderClubId": "test-435"
      }
    }
  },
  "result": {  // This is what we usually take to adjust the complog value:
    "completion": true,
    "success": true,    // if success is true we update the competencies in complog according
to the score below (assuking score is from 0 to 1).
    "score": {
      "scaled": 0.95
    },
    "extensions": {
      "http://example.com/flight/averagePitch": 0.05
    }
  },
  "context": {
```

```
    "instructor": {
      "name": "Irene Instructor", // the teacher id so that we have the teacher also
      "mbox": "mailto:irene@example.com"
    },
    "contextActivities":{
      "parent": { "id": "http://example.com/activities/hang-gliding-class-a" }
      "grouping": { "id": "http://example.com/activities/hang-gliding-school" }
    },
    "extensions": {
      "http://example.com/weatherConditions": "rainy"
    }
  },
  "timestamp": "2012-07-05T18:30:32.360Z",
  "stored": "2012-07-05T18:30:33.540Z",
  "authority": {
    "name": "Irene Instructor",
    "mbox": "mailto:irene@example.com"
  }
}
```


In case there is no score between 0 and 1 we can also take:


```
"result": {
    "completion": true,
    "success": true,
```

```
    "score": {
      "minimum": 1,
         "maximum": 10,
         "raw value": 6
    },
```

This is described in https://tincanapi.com/statements-101/ at section “results”.

CASE 2:

Nothing is configured internally (of course we have students and a subject with competencies):

We take the activity type objective instead (cf. http://xapi.vocab.pub/datasets/adl/). Objective, for us, defines the competencies that are addressed by an activity. This should look like this:

```
"activity": {
    "objective": {
         "id": "2344", // our id
          "display": { "en-US": "completed" } // name
    },
    "objective": {
         "id": "2344", // our id
          "display": { "en-US": "completed" } // name
    }
  }
```

It s is possible to get several ‘objectives’ at the same time. It is also possible to take an entire “course” which is a subject for us. So if the course is coming, we update all competencies according the value that is coming.

```
"activity": {
    “course”: {
         "id": "2344", // our id
          "display": { "en-US": "completed" } // name
    }
  }
```

It must be noted that Lea’s Box is a data store endpoint and receives data. Exporting data from the system is not implemented (yet).

EXTERNAL USE OF VISUALIZATION SERVICES

Lea’s Box supports the display of visualizations in external tools, for example, in iFrames. This support however, is limited.

LEARNING SPACES / OLM

Learning Spaces allow showing the results of analyses in form of Hasse diagrams (cf. p.11 of this document). The visualizations are available as a service. Competence Structures (i.e., the prerequisite relations between competences) can be accessed via this link: http://css-kmi.tugraz.at/mkrwww/leas-portal2/hassediagram/client/, competence spaces (the set of all competence states can be accessed via this link: http://css-kmi.tugraz.at/mkrwww/leas-portal2/hassediagram2/client/. He use, of course, requires login information. When no session variable is set, there is a logon screen. The login works with any user account of the main system. It is possible though, to directly pass the user credentials to the services bei using a session variable “un” with “un” as username and “pwd” as password. Again, this works with a Lea’s Box account.

The same works for the OLM system which is available as a stand-alone tool as well. This is accessible via http://52.33.42.241/leas-olm/gui/home.jsp. In the case of the OLM the session variables should be set manually.

8. DATA EXPORT

Data export features are provided through the configuration tool. The Lea's Box system provides several types for data backup and export function.

SQL FORMAT

The most important feature is a full-scale database backup function in sql format. This function dumps the entire database including all relevant tables and including all the stored data to a single file. This file can be used to restore the entire database in case of a loss of data. It allows moving databases or to further extract or re-format the entire data.

It is recommended to back up the database frequently in short periods in order to avoid a severe loss of data!

OTHER FORMATS

In addition to the full database export, data that is generally relevant can be exported in the formats *XML*, *JSON*, and *RDF*. These export functions cover the following data structures and datasets:

- Groups of a school
- Teachers of a school
- Students in groups
- Subjects of a school
- Competencies in subjects
- Teacher of subjects
- Students activities in relation to subjects
- Students competence values over time in relation to subjects

DELETING DATA

In addition to the general options provided by the configuration tool, administrators can delete the entire school from the system (e.g., in case an institution discontinues the use of Lea's Box). Alternatively, the administrator can delete all entries and data of a single student (e.g., when this student drops out of a school).

DATA OWNERSHIP

Data ownership is a critical aspect. The project offered a comprehensive guide for an appropriate treatment of data. From a technical perspective we reviewed data locker approaches, however, these ideas appear not mature and not suitable for the scope of Lea's Box. Further technical solutions are out of the scope of the project. Learners, however, have to possibility to export and delete their own data from the system, where this is appropriate, possible, or wanted. The export is made also in XML, JSON, or RDF format, however limited to a particular student's data (this includes subject, competencies, activities and achievements).

9. AVAILABILITY

In accordance with the Consortium Agreement, the Grant Agreement, and specifically the exploitation plan (deliverable D6.5), the software system developed in Lea's Box is in parts available as open source software to the public. In principle the entire system including the underlying central executive and the OLM package as well as the system database as an instance of the data model, are available as open source software. Considering the proposed system architecture, as of the DoW and including the extensions of years 1 and 2, all these components are provided openly (yellow and green coded components, as displayed in the following figure). Additionally developed components, specifically such that are based on extensive basic research activities, are subject to commercial and pre-commercial exploitation beyond the project, as described in the deliverable D6.5. Therefore, certain algorithms and solutions must be protected.

9.1. OPEN SOURCE PACKAGES

- The system portal including its main components
- The system database

- The OLM, as a complete package
- myClass (in two variants)
- The FCA analyses tool as a separate package
- The flower app
- The Canvas csv import tool

All packages are available for download at the Lea's Box Website as zip archives. All open source packages are distributed under the *Apache 2.0 Open Source License*. (https://opensource.org/licenses/Apache-2.0)

9.2. RESTRICTED PACKAGES

- **The Hasse diagram generation and visualization algorithm**
  Reason for protection: The algorithm derives competency structures on the fly in real-time and outputs an interactive diagram in an extremely preformat and quick way in web browsers. This is a valuable asset that is a clear advancement in comparison to all existing solutions in the CbKST / FCA communities.
- **The Learning Paths and Learning Performance Vector Algorithm**
  Reason: This algorithm performance the computation of the learning paths and the LPV in real-time and it's an extremely quick solution that works even in and for web browser. This is a valuable asset that is a clear advancement in comparison to all existing solutions in the CbKST / FCA communities.

- **The Learning Landscape Tool**
  Reason: This tool is a novel and innovative approach of displaying learning data. It is a close-to-market asset that is a part of the exploitation strategy. The related algorithms are novel and directly linked to the analyses components.
- **The FLASH analyses engine**
  Reason: Along with the data warehouse, we developed a query engine that is a highly performant solution to analyze large scale data and specifically large scale temporal data in real time. This is a valuable asset that is a clear advancement in comparison to all existing solutions in the CbKST / FCA communities.

10. ANNEXES

- FCA Tool Manual
- Learning Landscape Manual
- Mind Map Tool Manual
- Configuration Tool Manual