



Project Document Cover Sheet

Project Information			
Project Acronym	PIOP		
Project Title	Portfolio interoperability projects		
Start Date	November 2008	End Date	31 st March 2009
Lead Institution	VESA 13-19 Education Support Agency [part of Leicester LA]		
Project Director	Tim Farthing		
Project Manager & contact details	Tim Farthing VESA 13-19 Education Support Agency Knighton Fields Centre Herrick Rd Leicester LE2 6DH tim.farthing@vesa.org.uk 0116 2707942		
Partner Institutions	Loughborough College Connexions Leicester Shire		
Project Web URL	http://wiki.cetis.ac.uk/PIOP_LeicesterShire		
Programme Name (and number)			
Programme Manager			
Document Name			
Document Title	Final Report Leicestershire JISC PIOP project – March 2009		
Reporting Period	<i>Up to March 2009</i>		
Author(s) & project role	Tim Farthing – Project Manager Megan Jones – VESA IT Administrator Sam Persse – Web Master Loughborough College		
Date	17 th March 2009	Filename	JISC PIOP final project report 17 th March 2009
URL	<i>if document is posted on project web site</i>		
Access	<input type="checkbox"/> Project and JISC internal		<input type="checkbox"/> General dissemination

Document History		
Version	Date	Comments

Final Report Leicestershire JISC PIOP project – March 2009

Introduction to our e-PF/ILP – eprogressfile systems

The Leicestershire PIOP has 3 versions of their e-PF/ILP installed in schools across Leicester City and Leicestershire.

- KS3 version for schools with pupils aged 11-13
- KS4 version for schools with pupils aged 14-16
- Post 16 version for schools, FE Colleges and 6th Form Colleges with pupils aged 16-19

The version we used for the purposes of this JISC project was the KS4 version. This was because schools using the application would use both the exporter for learners leaving and going on to Post 16 Education and an importer for those schools taking in new cohorts at the start of KS4 (age 14). The personal statement is also used heavily in the KS4 version when Learners apply through the Common application form to Post 16 institutions.

A brief introduction to how the system works

Our e-PF/ILP product is primarily led and driven by school and college staff as part of a planned curriculum process. It is therefore fairly structured and learner engagement relies significantly on encouragement and support of staff, often in a small group and one to one context. It is very much about “the process” where learners undertake a number of exercises and activities which feed through the system and populate text areas in the final ILP section. The personal statement is a good example of this, learners can click on an import button and a series of prompts appear in the personal statement text area acting as a kick start for the final more formal edited down version. These prompts are especially important for younger learners who may find it difficult to fill in a blank text area.

Although much of the interoperability between our own 3 versions would be in transition from one key stage to another, we also had to take into account sideways movements for learners leaving schools in our area and moving to other areas and wishing to take their e-PF/ILP with them.

This led to the question, what exactly should be exported?

- In transition moving between the Key stages the learner would only want to take with them the final edited down completed version
- In a sideways transition situation where, for example, a learner moves school mid way through a Key Stage the final editing process might not be complete. Therefore the learner would want to take all parts of the ILP from the final versions to the prompts.

For the purposes of this project we decided to export everything in the ILP, as this would more fully test out the LEAP2a specification. This is, in essence, the web pages making up the ILP.

- All About Me
- My Learning
- Career Plan
- Personal Statement
- Progress Reviews
- Achievement Log

- Draft CV

Please see the Leicestershire pages on the wiki to get the login details to access a demonstration of our system

http://wiki.cetis.ac.uk/PIOP_LeicesterShire

The Export Process

The exporter was created as a web service developed in C# Microsoft Visual Studio.NET 2008. After initially trying to develop the service using basic XML classes within .NET framework we discovered a new class as part of the .NET Framework 3.5. The [AtomFeedFormatter](#) class allows the creation of atom feed elements more easily with a set of in built functions.

The main issue with development was to create some of the more complex elements (leap2a specific ones). These couldn't be created with the standard atom functions, so had to be created as "Element Extensions" using the .NET [XMLElement](#) class.

We decided to use ASP.NET rather than the classic ASP which the progress file site had been developed in because we found the built in functions and development environment allowed us to build a web service and get it running a lot quicker, which was important due to the project's tight deadlines.

Mapping

To see a spread sheet of our mapping to the LEAP2a specification, please go to the wiki:

http://wiki.cetis.ac.uk/PIOP_LeicesterShire

Here you will see the final version plus some formative ones where you can track how this developed as the project progressed.

The process followed was:

- 1) Determined all the data to export (please see above), documenting from where in the database this came from and its datatype.
- 2) Drew a pictorial representation of the data and its relationships to each other as an initial starting point (please see on wiki the original mapping exercise- please note that this is purely a kick off point and you will see things changed quite considerably from this point to the final mapping document, due to both our further understanding of the specification and also due to the specification changing and adapting)
- 3) Carefully looked at the content of data being entered (not always as what would be expected with a field's title.)
- 4) Started the mapping process by first trying to determine the LEAP2a type
- 5) Worked through and documented the relationships between entries
- 6) Added literals where appropriate
- 7) Categorised according to the LEAP2a specification where relevant.
- 8) Finally completed a diagrammatic version of our final export.

2009-03/LEAP2A types

- **Ability**

The final mapping did not have any object types classified as *abilities*, however the area of Key Skills at certain points were classified this way but our final decision was to classify as an *entry*. The rationale behind this is as follows:

- The learner's competency with the six Key Skill areas in our application (*working with others, problem solving, improving learning outcomes, communication, number application, information technology*) is a self rating process. Therefore, this falls into the specification under the *ability* type '*the portfolio contains claims (which are entries) to abilities*'.
- Following on from this rationale, for a period of time we mapped the six key skill areas as entries with a claim to impersonal definitions of *ability* which would be defined outside the portfolio, typically in a skills or competency framework.
- On reflection and discussion we felt that overall, with the ever changing '*skills framework*' [for example], the emergence of Functional Skills and Personal Learning and Thinking Skills we were unsure as to whether other portfolios would categorize Key Skills as ours, or would use a different framework for skill definitions such as Functional Skills.
- Therefore, we finally it was decided that it would be better for our application to keep this item as one collective item of "Key Skills" and present it for export as it is used in the application – all Key Skills listed as one text string that is used as a prompt in the unedited *personal statement*.

I would like to stress that this was purely our thought processes for this and by no means definitive or correct, but I wanted to record our thinking as hopefully this could be useful to help others when trying to define their data accord to the specification.

- **Achievement**

The *achievement log* within our application fell into place readily with the LEAP2a specification with literal fields for the date and a field for the learner to record the evidence.

A foot note to this is that our application does have a field entitled "*my achievements so far*" in the '*My learning web page*'. This in fact is a text area where the learner can input anything they like it does not have to be qualified by *evidence* or *dates* therefore according to the specification would be an *entry* rather than an *achievement*. Again this highlights the need to look further than a field's name. Developers when importing from another system must look at how the field has been defined by the specification rather than its content or what it is called.

It is exactly in cases as this that I feel the specification works well in that only fully qualified achievements will be exported/imported as thus.

- **Activity**

The final mapping had no items classified as *activities*. The only field we looked at and considered as to whether it could be an activity was our '*work experience*' however our system had no qualifiers for *activity* to further describe our work experience – there are no *date* fields or any references to *organizations* it is merely a text area where the learner could input something as "*I did work experience in ASDA for a week in June*". Therefore was left as an entry.

- **Meeting**

Progress Reviews constituted *meeting* in our application. These would be a series of pre planned meetings between the author (learner) and their progress tutor. Initially it was difficult to find the qualifying literals and relationships to further describe this apart from the *date* of the *Progress Review*. Under closer examination the review does have an agenda – this is the '*learner summary statement*' that the learner would prepare beforehand. Any outcomes from the meeting would be in the form of targets.

- **Organization**

There are no references to organization in our application

- **Person**

Address and *personal detail* fields easily fell in line with this Leap type, there were no problems here merely followed the specification using the literals to further describe the data - spatial, person type etc.

- **Plan**

The *Target Diary* worked well with the *plan* leap type in that each target has a title and the field “*How I intend to do it*” becomes the content of this target, the target *start* and *end dates* further qualify the *plan*. The field “*what I Intend to do*” is the goal of the target. This becomes an *entry* with a “*supported by*” relationship to the target.

- **Resource**

There are no references to resources in our application

- **Selection**

To structure our export we made each web page of the ILP a selection so that they can be used individually for example as with the CV or personal statement or as a complete whole making a full ILP.

- **Entry**

Where an item can not be defined as above it was classed an entry, in some cases this was further qualified by its category.

Categories

The project decided to use only those categories as defined in the LEAP2a specification. We did not seek to make any local category schemes as we sought to implement the specification purely in the terms in which it is defined.

The categories we used were as follows

- common item
- life area
- person type
- selection type
- SWOT

The category terms further described entries, hopefully resulting in importers that will be able to qualify each entry and hence be able to place more easily.

For example the central part of our E-ILP is the learner’s personal statement. At KS4 this is used for Post 16 applications, in the Post 16 version it is used for UCAS applications. Many parts of the process act as prompts for this, being able to define it as a *common item* category and give a further term of *personal statement* which will make sure that this then imports into the personal statement of another eportfolio. It is imperative that such an important item in the eportfolio of those 14-19 is properly placed.

Issues and learning points from the export

- 1) When determining the type of item, use the specification’s predicates to fully define the LEAP2a type, for example field headings can be misleading, for example our database has a field entitled ‘*Education and Qualifications*’ which is a simple text area *entry* made by the learner. On analysis this was classified as ‘*entry*’ as it lacked the predicates that could define it as an *achievement*, – ‘*dates*’, ‘*targets*’ or ‘*evidence*’. Further into the ILP there is

an '*Achievement Log*' where learners can fully log their achievements with dates and evidence this therefore can truly be determined as an '*achievement*' type.

- 2) Some parts of our export are not entries made by the learners themselves but are feedback from a series of activities undertaken. For example, with Key Skills the learner clicks a series of check boxes, the application then feeds back to the learner a rating of how competent they are in each skill area. The application totals up their own self assessment, the higher the score the greater the competency. The result is fed back to the learner as a prompt in the personal statement. The raw data for the self assessment of these Key Skills is held in the database by way of a numeric value which on its own is meaningless, only having context when fed back as a text string prompt. There are similar issues with the *learning style* section. The learner is asked to check a series of boxes if they agree with each statement and this is held in the database as a series of true and false inputs which, on their own have no meaning, their meaning coming from a text string interpreted by the application.

The two examples cited above could have been categorised within the specification as '*check*' and/or '*priorities*'. However, on reflection it was felt this would not be useful as the resulting text string would be more viable to other eportfolios than as numeric or true/false inputs.

A direct outcome of this is that we were unable to import exactly as we export.

- 3) Interoperability within our own three versions will always lend itself to being able to export more than we can import. However, for our systems we do not envisage this as being problematic because of their inherent nature, in that they are process led. Therefore as each learner moves up through each key stage of their education they need to undergo the process embodied in each version, the resulting import from the previous version will act as an initial kick start.

The Import process

The importer has been created in classic ASP. This was due to the nature of the import process we decided upon. We decided that a user would login to our system (for the first time) and they would be presented with two options, continue with an empty ILP or import from previous system. Future developments would include updating existing data, allowing the user to choose, but for the purposes of this pilot we decided to concentrate on the import into an empty account.

The importer upload the user's profile using an online form, then queries the XML file directly. The elements are scanned in the following hierarchy to determine which data should go where...

- **leap:type** – this is the first port of call to map elements of specific leap types to the areas that we've mapped (see mapping section below for details)
- **categories** – the leap categories are then used to determine basic entries into the appropriate areas of the ILP (individual learning plan).
- **relationships** – relationships are then used to link entries together in terms of progress reviews, targets etc.
- **everything else** – everything else that isn't picked out will be imported into a general "important things about me" area, that students can look at and manipulate later.

The ASP then updates the necessary records in the database, so when the user opens the ILP forms they can see the data populated automatically.

We tested imports from Nottinghamshire's Passportfolio and Myprogressfile, as these were the other 14-19 partners.

Full details of how each export imported into our system can be found on the wiki http://wiki.cetis.ac.uk/PIOP_LeicesterShire#Import.

Our import followed the pattern, structure and style of our export in that it followed our interpretation of the specification and looked for the markers we had determined to classify our data namely following *leaptype*, *category*, *relationship* as indicated above.

Issue encountered with the import process.

- Differing interpretations of the specification.
For example we classified evidence of an '*achievement*' as an '*entry*' whereas other developers classified as an '*achievement*' marking it as the '*content*' to the achievement '*title*'. The result was a partial import resulting in reduced meaning.
- Differing system structures.
For example, our *targets/plans* do not have any hierarchical structure compared to a system such as Nottinghamshire's which has relationships between targets. Therefore the links between the targets are lost when importing into our system which may result in a loss of clarity and meaning of data.
- If data is not classified consistently between the two systems it will not import correctly. For example, our system classifies leap:type '*person*' further with a category term '*person type*' and a term '*self*' in order to distinguish between the portfolio holder and another '*person_type*' for example a referee. If there is only a classification of '*person*', both the details of the portfolio holder and any details of referees etc will not import correctly. Where we found we could relax our initial classification we did so, for example, we amended our importer so that with '*plans*' it did not look for both a start and end date but just a target date. However for '*leaptype: person*' our system must have further classification to determine what person they are.
- Need to include references in your importer for '*leap types*' you do not have in your own system so that they can be classified and imported into your default area. Our system has no references to abilities therefore in our initial imports these were completely ignored and did not even go through to the default field.
- The default field (in our system "*Important things about me*") although essential to receive all non classified data does in itself create issues
 - Potentially this could result in an extremely large text string. The end user would have to plough through this to edit down with a lot of unrelated points which would become increasingly tedious, especially for lower age groups.

Import/Export Issues

The very nature of our system meant it was definitely easier to export than to import. This, though, merely mirrors the system's structure and nature, in that it is process led. Essentially, it is not only an eportfolio but also a learning tool and an aid to the curriculum.

In practice the most common scenario of interoperability is that learners would use an import to kick start their e-ILP/eprogressfile as they start off at each key stage of their education, then through that key stage, undergo the system's process leading to a final export into either the next key stage or on to eportfolios held by HEIs, workplaces or personal ones.

Even with scenarios of learners transferring from one school to another mid Key Stage, learners still need to undergo the process to get the most out of this system.

Bearing this in mind, questions need to be raised as to how useful it is to the end user of our system to have a potentially large dump area or partially imported data that could have lost meaning and relevance.

- Should we therefore just import fully qualified data that ends up in exactly the right place fully in context? This then would mean a considerable data loss and from a developer's point of view something that initially many would baulk at.
- Should there be further development of the 14-19 eportfolios to further develop common terms so that interoperability between the 14-19 systems is more secure? However, if this were to be the case it would need to look further than the initial three 14-19 partners involved presently, and here there is always the proviso that the 14-19 curriculum developments are ever changing and terms developed today as common could in a short time be replaced by new terminology. For example, one can see this readily in the changing language of skills eg key skills, employability skills, PLTS etc etc. The challenge is which skill set to adopt.

We would recommend that in the future developers looking to use the specification need to think through these issues before importing, especially to systems used by younger learners.

The specification worked best where terms were classified fully through the leaptypes, category and further through the terms and relationships. Consequently, the data flowed seamlessly and ended in the right place. There is still a trade off in finding the right level of granularity as other developers might not have classified data similarly then it will not import correctly. The risk in loosening classifications is that data moving through the process might end up being less useful than hoped for.

Another strong recommendation for developers, especially those not familiar with the school and college educational landscape, is that they read through the specification fully. They should focus on looking at examples of how it is being used, examples of mappings etc, as this will be especially useful to get a feel for the terminology and the way the spec works.