JISC

D4.3 Report on the development of version 2 of the Personalisation Engine

Project Information			
Project Acronym	MyPlan		
Project Title	MyPlan - Personal Planning for Learning throughout Life		
Start Date	1st September 2006	**End Date**	30th November 2008
Lead Institution	Birkbeck College, University of London		
Project Director	Prof. Alex Poulovassilis & Dr George Magoulas School of Computer Science and Information Systems, Birkbeck College, University of London, Malet Street, London WC1E 7HX, UK {ap,gmagoulas}@dcs.bbk.ac.uk		
Project Manager & contact details	Dr Nicolas Van Labeke Birkbeck College, University of London London Knowledge Lab 23-29 Emerald St London WC1N 3QS, UK nicolas@dcs.bbk.ac.uk		
Partner Institutions	Institute of Education; Community College Hackney; UCAS; Linking London Lifelong Learning Network		
Project Web URL	www.lkl.ac.uk/research/myplan/		
Programme Name (and number)	e-Learning Capital Programme		
Programme Manager	David Kernohan		

Document Name			
Document Title	D4.3 - Report on the development of version 2 of the Personalisation Engine		
Reporting Period	April 2007 – December 2007		
Author(s) & project role	Nicolas Van Labeke (project manager – WP4 leader)		
Date	15/10/2008	**Filename**	D4.3.doc
URL	*N/A*		
Access	☒ Project and JISC internal	☐ General dissemination	

Document History		
Version	**Date**	**Comments**
0.1	14/07/2008	Document created
0.2	13/08/2008	First draft ready
1.0	15/10/2008	Final Version

Table of Contents

Figures and Tables

1 Introduction

As outlined in deliverable D3.1 (MyPlan Personalisation Specification), the personalisation functionalities that were prioritised for development during the MyPlan project are as follows:

1. Personalised search of timelines from "people like me"

2. Personalised recommendation of which course(s) to study next

3. Customisation of the L4All system

4. Automatic update of users' profiles

5. Ability to record and display ratings of search results

Version 1 of the L4All personalisation engine was described in detail in Deliverable D4.1 of the MyPlan project, and focussed on the first three issues above: search for people like me, customisation of the system and recommendation (only partially described in that deliverable).

This document describes Version 2 of the personalisation engine and also aims at giving a general description of the current state of the L4All system.

This document is organised as follows. First, the current architecture of the L4All system is described. Next, the overall GUI of the system is presented, highlighting the significant changes since Version 1. Third, the implementation of the personalisation engine, Version 2, is described, focusing on the new approach that is now being used for providing a personalised recommendation mechanism.

2 System Architecture

The overall architecture of the L4All system has remained basically the same over the various phases of the MyPlan project, with a few minor changes and additions to reflect the evolution of the project and the different shifts of focus (see for example L4All Deliverable D6.2 and MyPlan Deliverable D4.1). The current system architecture can be seen in Figure 1.

L4All is deployed over a Tomcat web server. It consists of two different components: the web portal (the front-end of the L4All system) and the web services and related core components (the back-end of the system).

The web portal consists of several JSP pages, providing users with an interface for the core functions of the system: user and timeline management (creation, modification, etc.), user and timeline search (search by keyword, search for "people like me", search for "what next", etc.), course search (within in the L4All database or through the LearnDirect service) and overall GUI management (registration forms, timeline visualisation, etc.).

The back-end of the L4All system consists of a set of dedicated web services, implemented as Java Servlets (see Figure 2). The user model has been designed as an RDF ontology and implemented using the JENA Semantic Web framework, ensuring an easy mapping between the JavaBeans representing the various objects of the user model (user profile, timeline, episode, etc.) and their storage in a MySQL database.
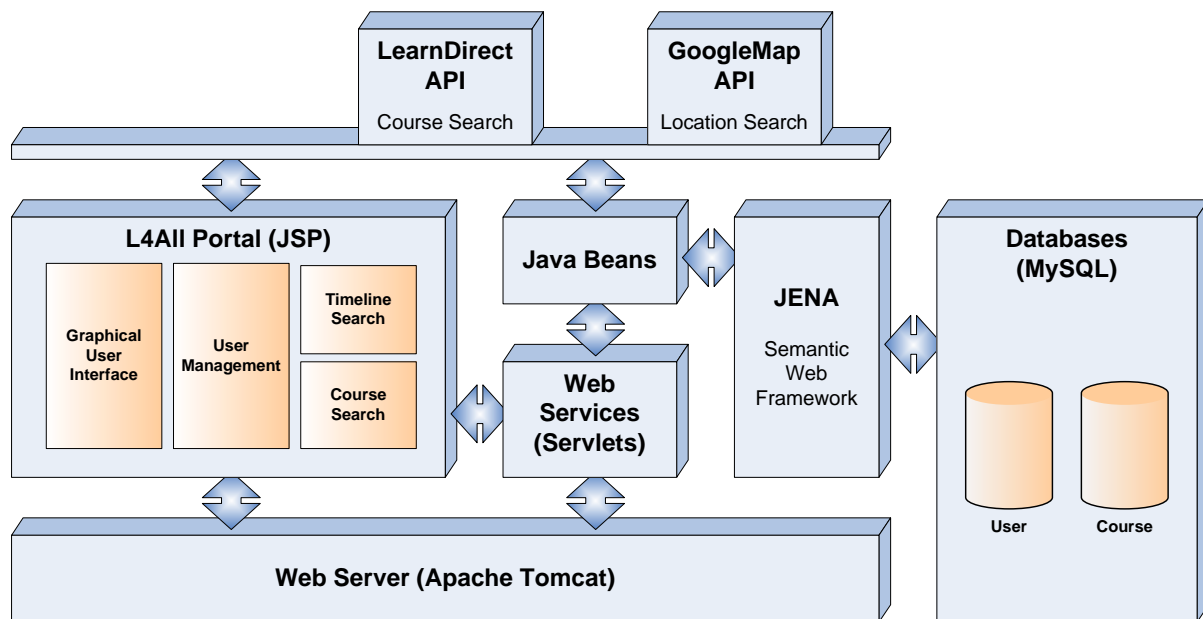
Figure 1. Overview of the L4All architecture

The L4All system is also integrated with two off-the-shelf external web services:

- The **LearnDirect API** allows parameterised search within their directory of courses. It was deployed during the earlier L4All project as a proof-of-concept for interoperability between the L4All system and other external services relevant for lifelong learners. The LearnDirect search API is used to give users access to course descriptions, enabling them to add relevant courses into their timeline (as course episodes).

- The **GoogleMap API** provides several services for manipulating location-based information, as well as managing their (now ubiquitous) interactive map. It is used in the L4All system to display locations associated with users (e.g. their current or past location), timelines (e.g. location of particular episodes) and courses (e.g. location of course providers), and for computing distances between locations.
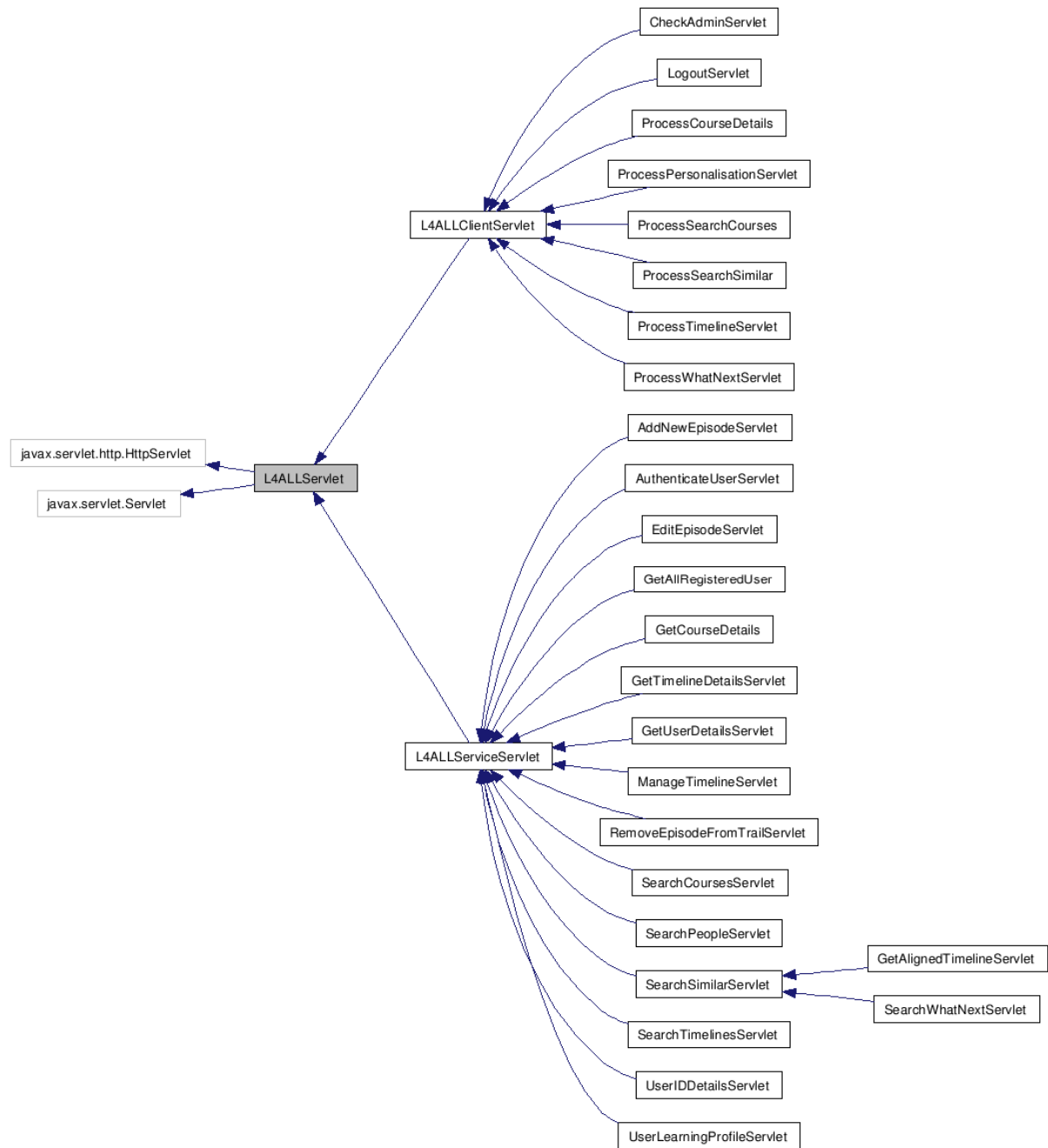
Figure 2. Hierarchical view of the Servlets supporting the L4All system.

3 Graphical User Interface

As described in Deliverable D4.1, the interface of the L4All system has been thoroughly redesigned during the MyPlan project, in order to better support the deployment of the personalisation functionalities that are the focus of the MyPlan project.

Version 2 of the system does not radically change any aspect of the new GUI and the reader is referred to D4.1 for a description of the various components of the interface.

Overall, the interface maintains the visualisation of the user's timeline as the central part of the display (see Figure 3). The various functionalities (e.g. profile edit, course or timeline search, personalisation of webpage, etc.) are made accessible through a pop-up window containing the relevant interface (see Figure 4). These popup HTML pages are deliberately kept simple and small, so that their purpose is limited to data input, while the exploitation of

their results – and potential feedback – takes place within the main page.
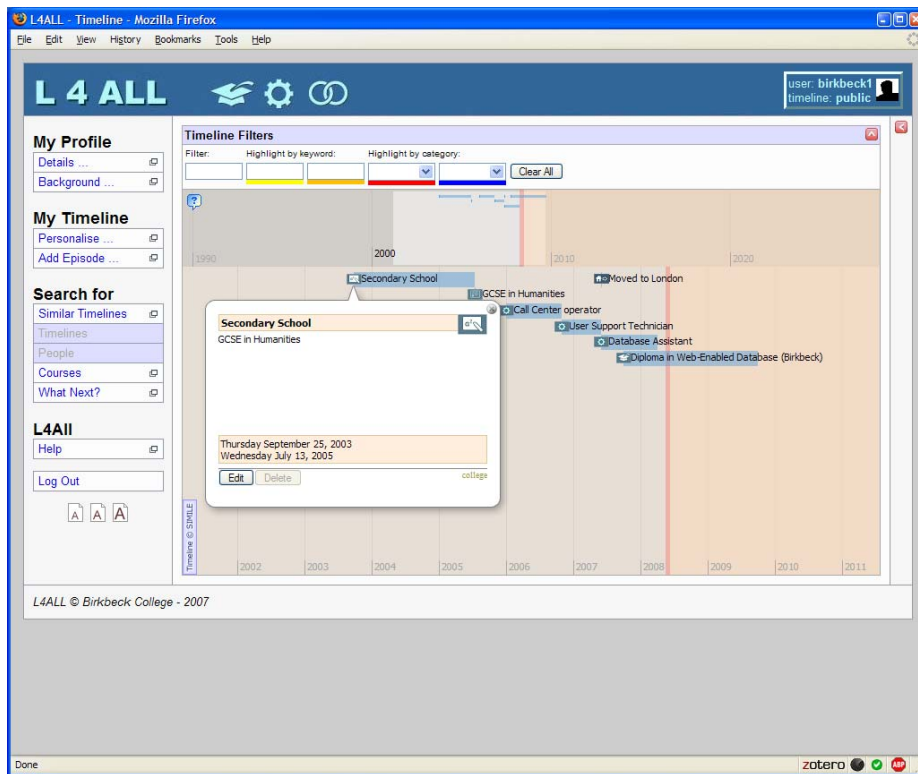
Figure 3. The main interface of the L4All system.

For example, searching for "what to do next" initially provides the user with a simple form, which then returns a list of all the matching timelines, in a tabular and condensed way (see Figure 11). The full exploration of these timelines takes place within the main page, with the selected timeline displayed below the user's timeline (see Figure 12). Upon selection of a timeline to be visualised on the main page, the popup window moves to the background so that the user's attention can be fully focussed on the two timelines being displayed.
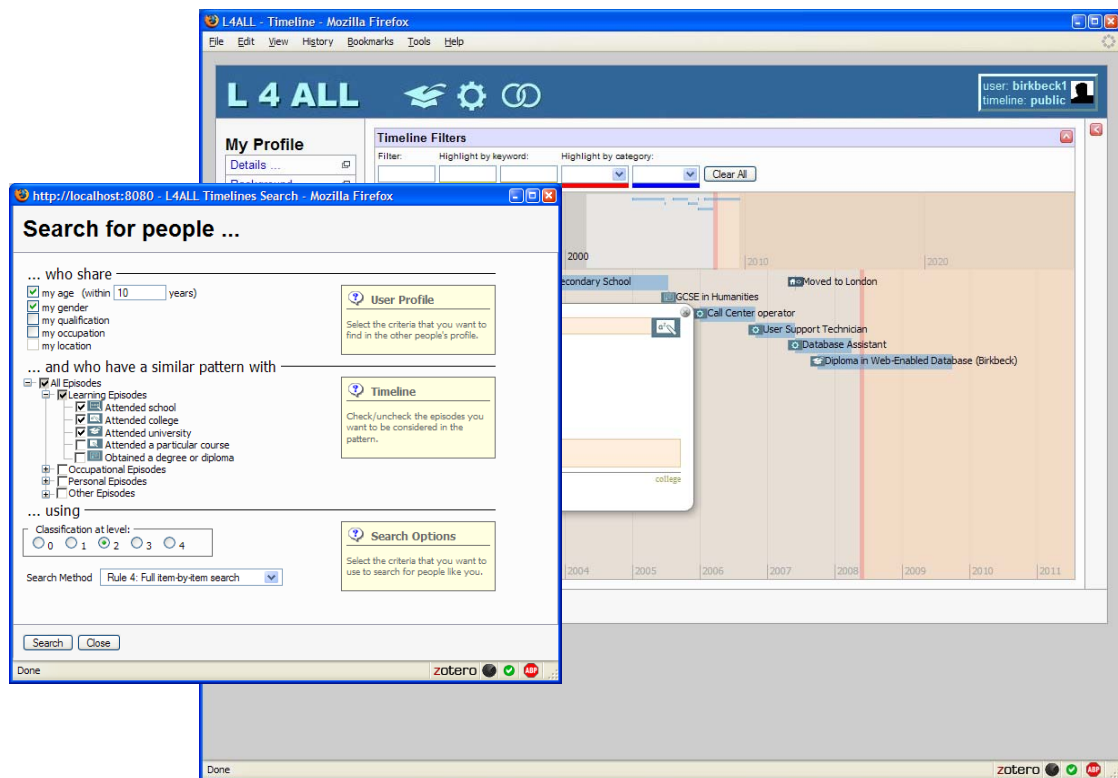
Figure 4. Interface for "search for people like me" available in pop-up window.

The Phase I evaluation, as reported in Deliverable D5.1, highlighted a number of issues related to the interface (e.g. unfamiliar terminology, confusing layout of the web portal, etc.). Most of these issues have been addressed in Version 2 and an up-to-date list of these issues can be found in Appendix A.

4   Personalisation Engine V2

The development of Version 1 of the L4*All* personalisation engine (described in Del 4.1) included the following aspects:

- Redesign of the GUI of the L4*All* system, using DHTML/javascript for the front-end and JSP/servlets for the back-end.

- Redesign of several aspects of the ontology underlying the L4*All* system in order to accommodate the new personalisation functionalities: different categories of user (learner, expert, institution), a two-axis taxonomy of episodes, etc.

- Design and implementation of a similarity measure engine for the comparison of learners' timelines. The mechanism is based on converting timelines into strings of comparable tokens and using string metrics for ranking them.

- Design and implementation of a "search for people like me" functionality, using the similarity engine for proposing matches between the user's timeline and others.

- Design and deployment of several customisation features (colour/shapes used in the timeline visualisation, bookmarks for interesting timelines, etc.)

The development of this Version 1 was described in detail in Del 4.1, and a report on its evaluation was given in Del D5.1.

The subsequent development of Version 2 has had two objectives:

- Addressing, as much as practically possible within the scope of this project, the issues identified during the Phase I evaluation, and improving the existing functionalities.

- Designing and implementing the "recommendation" mechanism

Most of the bug and issue fixing has not changed the overall structure and operation of the L4All system, and Del D4.1 remains accurate for a description of the design and implementation of the Version 1 personalisation functionalities. The following sections will therefore focus on describing the new recommendation functionality that has been developed within Version 2.

4.1 Recommendations vs. "What Next"

The Advisory Group (AG) meeting held at the London Knowledge Lab on 30/04/2008 was instrumental in refocusing our ongoing work on the "recommendation" mechanism. In particular, two considerations arose from that meeting.

Firstly, the terminology used for this facility needed to be carefully reconsidered: "recommendation" is a strong term related with career advice, which is not what the L4*All* system aims to provide. A more neutral term – such as "what next" – is more appropriate, especially since the mechanism we are providing will support the user in exploring others' timelines in a particular way i.e. *"this is what people have gone on to do after following a pathway similar to mine; why not consider similar future choices to them?"*).

Secondly, the source of information to use for this "what next" functionality needed to be feasible from a pragmatic viewpoint. As mentioned in Deliverable D4.1 (Section 5.2), our initial aim was to provide a template-based version of a timeline, not representing full timeline information, but consisting of a chain of episodes that could be viewed as a sequence of prerequisites and a final goal. Such an approach may be appropriate for overcoming the lack of formal connections between episodes within timelines, but it does however have practical limitations in the sense that it needs significant expert effort in building up a repository of such templates for the full learner community. The recommendation from the AG meeting on 30/04/2008 was to leverage instead the repository of users' own timelines in order to present possibilities of "what next" to individual users, building on the timeline alignment mechanism described in Del D4.1.

The deployment of the "what next" functionality provided in Version 2 comprises three steps:

- Finding a possible alignment between the user's timeline and a target timeline extracted from the database; such alignment is made possible by the Needleman-Wunsch similarity metrics discussed in Del 4.1 and below;

- Identifying, within that alignment, the episodes from the target timeline that could be of interest to this user;

- Presenting such information (i.e. both the alignment and the potentially interesting episodes) to the user within the interface.

4.2 The Needleman-Wunsch Similarity Metric

The two timelines to be compared are encoded following the methodology described in Del D4.1 (section 4.1). Figure 5 shows two examples of timelines to be compared: S1 represent the user's timeline and S2 a potential timeline, extracted from the L4All database, to be matched with the user.

S1	Cl-6.4-4.1	Wk-J.0-2.1	Un-9.1-6.3	Wk-R.0-2.4	Wk-C.0-3.5	Wk-P.0-2.3	Un-6.4-6.1	Wk-G.0-4.2	Wk-K.0-3.1

S2	Cl-0.0-4.1	Wk-R.0-2.4	Un-6.4-6.3	Wk-N.0-9.2	Un-6.4-6.1	Wk-S.0-3.1	Wk-J.0-3.1	Wk-J.0-2.1

Figure 5. Two encoded timelines – see Deliverable 4.1 for the encoding details.

These two examples are real timelines, built from the CVs of current students and alumni from Birkbeck, all of whom have Birkbeck's Foundation Degree in IT as a common episode (labelled as **Un-6.4-6.1** in timelines S1 and S2). Using real-life data proved to be useful for identifying the potential and problems of our approach (discussed below).

		Target's Timeline (S2)							
		Cl-0.0-4.1	Wk-R.0-2.4	Un-6.4-6.3	Wk-N.0-9.2	Un-6.4-6.1	Wk-S.0-3.1	Wk-J.0-3.1	Wk-J.0-2.1
User's Timeline (S1)		**0**	1	2	3	4	5	6	7	8
	Cl-6.4-4.1	**1**	2	3	4	5	6	7	8	9
	Wk-J.0-2.1	**2**	3	4	5	6	7	8	9	8
	Un-9.1-6.3	**3**	**4**	5	6	7	8	9	10	9
	Wk-R.0-2.4	4	5	**4**	5	6	7	8	9	10
	Wk-C.0-3.5	5	6	**5**	6	7	8	9	10	11
	Wk-P.0-2.3	6	7	**6**	**7**	**8**	9	10	11	12
	Un-6.4-6.1	7	8	7	8	9	**8**	9	10	11
	Wk-G.0-4.2	8	9	8	9	10	**9**	10	11	12
	Wk-K.0-3.1	9	10	9	10	11	**10**	**11**	**12**	**13**

Figure 6. A similarity matrix computed by the Needleman-Wunsch algorithm.

The system generates a similarity matrix between the two timelines (see Figure 6). Once the matrix has been completed, the potential alignments between episodes can be computed by back-tracking the Needleman-Wunsch computation and determining the "path" that led to the final (unnormalised) distance value between the two timelines. Moving along the back-tracking path indicates how the episode tokens from both strings are matched:

- A "left" move indicates no matching with the token in S2 and therefore corresponds to its alignment with a gap in S1;

- An "up" move indicates no matching between the token in S1 and therefore corresponds to its alignment with a gap in S2;

- A "diagonal" move indicates a match between the two tokens and therefore their alignment.

There may be several possible alignments resulting from the back-tracking of the similarity matrix. Choosing which one is an open issue that will be discussed below but for the time being, let us consider one path, as indicated in Figure 6. The corresponding alignment of tokens from both timelines can be seen in Figure 7.

S1	CI-6.4-4.1	Wk-J.0-2.1	Un-9.1-6.3		Wk-R.0-2.4	Wk-C.0-3.5	Wk-P.0-2.3		Un-6.4-6.1	Wk-G.0-4.2	Wk-K.0-3.1			
S2				CI-0.0-4.1	Wk-R.0-2.4			Un-6.4-6.3	Wk-N.0-9.2	Un-6.4-6.1		Wk-S.0-3.1	Wk-J.0-3.1	Wk-J.0-2.1
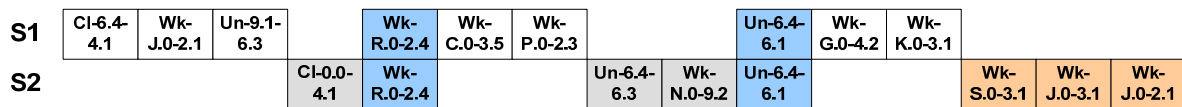
Figure 7. Alignment of two timelines.

The purpose of the alignment mechanism is to provide the user with information about possible future choices, as identified from others' timelines. As such, the focus should be on the target timeline (S2), identifying several important aspects:

- The similarity mechanism identifies two matches between the two timelines S1 and S2: a work episode (Wk-R.0-2.4) and a university episode (Un-6.4-6.1), both displayed in blue. It also indicates their relative position within both timelines.

- The mechanism indicates the episodes of the user's timeline S1 that have no match in the target timeline S2, displayed in grey. This information has no particular relevance at this stage.

- More important, the mechanism finally identifies the episodes of the target timeline that have no match with the user's timeline, displayed in orange. These are the focus of the procedure, as they potentially represent episodes that the user may be inspired to explore or may even consider for their own future personal development.

An alignment mechanism is more effective than other comparison approaches (such as set overlap, for example) because it not only identifies common episodes between the two timelines but also their relative positions and how the two timelines could be aligned. Since the encoding of the timeline into token strings preserves the chronological order of the episodes, this alignment also maintains some connection with the "real-life" experience of the users. This property allows us to apply some "rule-of-thumb" heuristics for the detection of relevant episodes.

First, the matching episodes (identified in blue in Figure 7) define a "common ground" between the two timelines, i.e. a commonality between the life experiences of the two users. This common ground delimits the alignment of the two timelines into three zones that can be used to identify potential sources of inspiration from the target timeline:

1. All episodes before the common ground can be deemed irrelevant (identified in grey in Figure 7), since they will mostly include earlier life experiences that will be "obsolete" for the user.

2. All episodes after the common ground are deemed relevant (identified in orange in Figure 7) as a source of inspiration for future personal development and are therefore the main outcomes of the "what next" functionality.

3. All episodes within the common ground (but not mapping with one of the user's episodes) have an ambiguous status, being potentially either relevant or irrelevant. In such cases, looking at their position in the alignment is not enough to make a judgement; and a more in-depth analysis of these episodes will be clearly needed to remove the ambiguity about their relevance. This introduces some difficulties that will be discussed in Section 4.5 below. In the current version of the system, and in the absence of a deeper mechanism for judging episodes' relevance, they are deemed irrelevant (and identified in grey in Figure 7).

Two further points should be made about the mechanism

- The "common ground" is a fluid notion that is only defined by the existence of matching episodes between the two timelines. Depending on the user's timeline and on the encoding of the timelines (see discussion in Section 4.5), it may be reduced to one single episode or even none, and the three zones (before, within and after) may not exist at all. Our algorithm tries to deal with such situations. For example, a target timeline that offers

10

no matching episodes with the user's timeline is not presented as a potential source of information (its similarity value is set to zero).

- The mechanism is not intended to be authoritative or prescriptive in its identification of potential sources of inspiration. This is why we dropped the term "recommendation" in favour of a more neutral "what next". In this view, the discrimination of episodes into relevant/irrelevant groups is not definitive: they will still all be presented to the user, as the user is the only one in a position to make a judgement about their relevance. The "what next" mechanism aims to provide users with computational and visual support to facilitate that judgement.
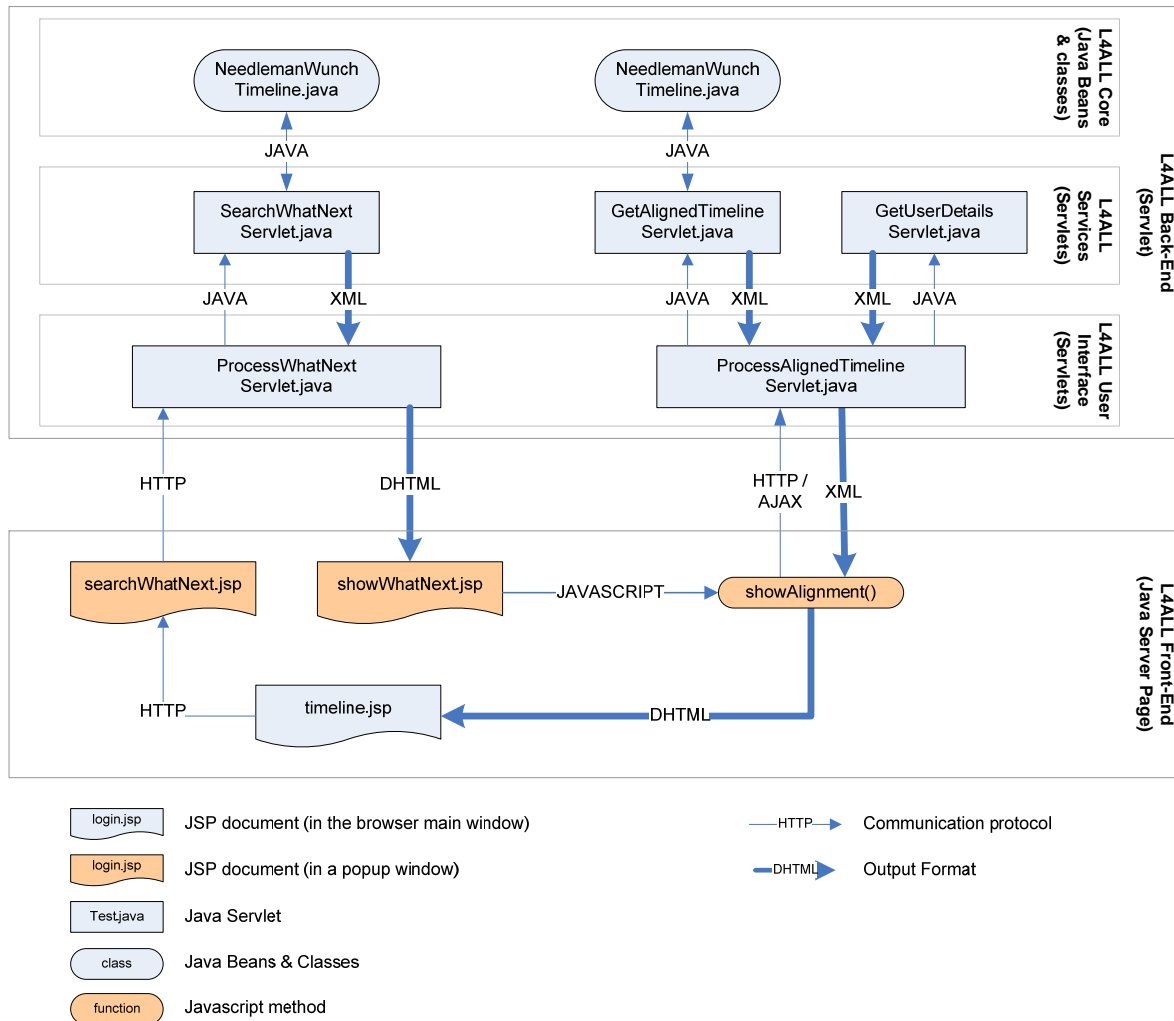
Figure 8. Flowchart of the "What Next" functionality

4.3 Implementation

The "what next" functionality has been implemented as a specialisation of the "search for people like me", described in Deliverable D4.1. Both functionalities several characteristics, notability their use of similarity metrics. The "what next" functionality as a whole is distributed within two distinct threads (see Figure 8):

11

- One thread searches for relevant timelines in the database and presents the resulting list to the user (implemented by the two servlets[1] `SearchWhatNextServlet.java` and `ProcessWhatNextServlet.java`)

- Upon selection of one of the resulting timelines by the user, the alignment of the user's timeline and the selected timeline, and the presentation of the processed information to the user (implemented by the two servlets `GetAlignedTimelineServlet.java` and `ProcessAlignedTimeline.java`)

The service for requesting a list of "what next" timelines is invoked as follows:

/searchWhatNext?username=user1**&epd_level=**2

---

where:

- **username** is the (unique) identifier of the user initiating the request (*mandatory*)
- **epd_level** is the depth of encoding used for the episodes, between 0 for a wide comparison to 4 for a narrow comparison[2] (*optional, default is 2*).

The response to the request is an XML document containing a summarised list of all the timelines containing relevant information for the user (see Figure 9). For each relevant target timeline, the output contains the identifiers of the timeline and its owner (**timeline_id** and **timeline_owner**), the description associated with the timeline (**timeline_desc**), age and full name of the timeline's owner (**user_age** and **user_fullname**) and the encoding of the timeline (**timeline_encoding**). The similarity measure between the user's timeline and each target is also given (**score**).

```
<advices cardinality='5' for='user1'>
    <advice score='0.17'>
        <timeline_id>alumni2</timeline_id>
        <timeline_owner>alumni2</timeline_owner>
        <timeline_desc><![CDATA[Currently working as a senior Project Manager in London.
                Former student at Bikbeck (FD-IT in 2006).]]>
        </timeline_desc>
        <user_age>1975</user_age>
        <user_fullname>Alumni 2</user_fullname>
        <timeline_encoding><![CDATA[Cl-0.0-4.1 Un-3.7-6.3 Wk-R.0-3.1 Un-6.4-6.1 Wk-P.0-2.1
                Wk-J.0-2.1 ]]>
        </timeline_encoding>
    </advice>

    …
    <advice score='…'>
        …
    </advice>
    …

</advices>
```

Figure 9. Output of the searchWhatNext service.

---

[1] Following the architecture principles established in Deliverable D4.1, a clear distinction between the service-based back-end architecture and the web-based front-end has been put in place, implemented by a dual servlet approach. For any given functionality, one servlet is implementing the service-based back-end (typically generating XML), while the second servlet – invoking the back-end one – is maintaining the interface (typically generating DHTML).

[2] Following the encoding mechanism described in Deliverable D4.1, the depth of encoding relates to the precision given to the classifications used for each episode, e.g. **Wk-1.2.3.4-1.2.3.4** for a work episode at depth 4, **Wk-1.2-1.2** for the same episode at depth 2, **Wk--** at depth 0.

The service for requesting the alignment between two timelines is invoked as follows:

> **/getAlignedTimeline?username=**user1**&matchname=**alumni1**&epd_level=**2**&mode=**ALIGN

where:

- **username** is the (unique) identifier of the user initiating the request (*mandatory*)
- **matchname** is the (unique) identifier of the target timeline to compare with (*mandatory*)
- **epd_level** is the depth of encoding used for the episodes (*optional, default is 2*).
- **mode** describes the format of the request's output, among the following options:
  - **ALIGN** returns a list of the alignments of the episodes of both timelines, as an ordered set of pairs of episodes' positions in their own timeline (with the value -1 used to indicate a gap in the timeline); see Figure 10.
  - **TIMELINE_HIGHLIGHT** returns the target timeline as usual, but with an additional element for each episode indicating their relevance for the "what next" functionality (IGNORED, MATCHED or SUGGESTED).
  - **TIMELINE_MATCH** also returns the target timeline but with a "lens" applied to it, shifting and rescheduling episodes in the target timeline so that they are synchronised with the user's timeline, in an attempt to represent the alignment.
  (*optional, default is ALIGN*).

```
<alignments cardinality='12' user='user1 target='alumni1'>
    <alignment order='1'>
        <user_episode position='1'/>
        <target_episode position ='-1'/>
        <description><![CDATA[The first episode in the user's timeline is not aligned with any
                    episode in the target timeline]]>
        </description>
    </alignment>
    <alignment order='2'>
        <user_episode index='-1'/>
        <target_episode index='1'/>
        <description><![CDATA[The first episode in the target timeline is not aligned with any
                    episode in the user's timeline]]>
        </description>
    </alignment>
    <alignment order='3'>
        <user_episode index='2'/>
        <target_episode index='2'/>
        <description><![CDATA[The second episode in the user's timeline is aligned with the
                    second episode in the target timeline]]>
        </description>
    </alignment>
    …
    < alignment order='…'>
        …
    </advice>
    …

</advices>
```

Figure 10. Output of the getAlignedTimeline service (ALIGN mode).

4.4 Deployment of the "What Next" functionality

The front-end of the L4All system has been upgraded in Version 2 to incorporate this "what next" functionality, providing the user with an enhanced graphical interface to access its various components. The interface for searching for relevant timelines is very similar to the "search for people like me" interface. Deployed as a popup window (see Figure 11), it offers the user the possibility to specify some parameters for the search. In the current version, only the depth of the episode encoding (wide to narrow) is taken into account but provision

has been made for defining a specific episode to look for, and searching over timelines of other users, experts or recommendation "templates" (see Del D4.1 for discussion of these different types of users and timelines).

Upon parameter completion by the user, a summary of all relevant timelines is presented as a table, with the potential timelines ordered by their similarity (see Section 4.3 above) and summarised by the name and age of their users and by a short description. A "show" button allows the learner to display the selected timeline in the main window, below the user's own timeline. Two modes of visualisation of the target timeline are supported.
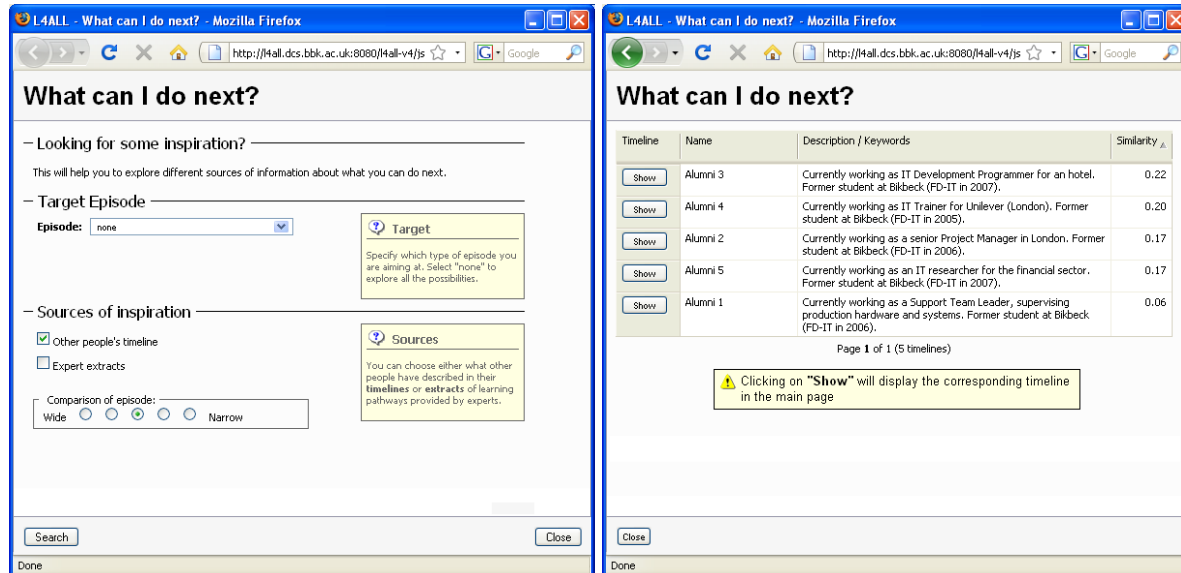

Figure 11. Search interface of the "what next" functionality.

The first one (called "highlighting", see Figure 12) consists of the traditional linear representation of a timeline, episode by episode, synchronised with their original timestamps. The information coming from the alignment between the two timelines is used to indicate, using different colours, the status of each individual episode in the target timeline: blue for matching episodes, grey for episodes judged to be irrelevant, and orange for episodes judged to be a potential source of inspiration. As with all timelines, the user has the possibility to further explore individual episodes by accessing their description (presented in a popup balloon, activated by clicking on the episode in the timeline widget).

The second mode of visualisation (called "recommendation", see Figure 13) allows the user to explore the same timeline but with the system explicitly representing the potential sources of inspiration as "future" episodes. To build this representation, the same colour scheme is used for individual episodes but the target timeline is this time synchronised with the user's timeline. Matching episodes in the target timeline, as detected during the alignment procedure, are synchronised with their corresponding episode in the user's timeline (start and end dates are modified accordingly). Irrelevant episodes are reduced to instant events and approximately located in the appropriate portion of the timeline[3]. Episodes judged to be a relevant source of inspiration for the user are shown as potential future goals by shifting them into the "future" part of the timeline (i.e. beyond the current date). Since the duration of these episodes has no particular meaning in this context, they are arbitrarily redefined as year-long episodes (i.e. long enough to be visible); their relative position to each other is maintained.

---

[3] Since these "irrelevant" episodes have no correspondence in the user timeline, their location in the "aligned" timeline can only be defined relatively to the next or previous alignments of episodes.
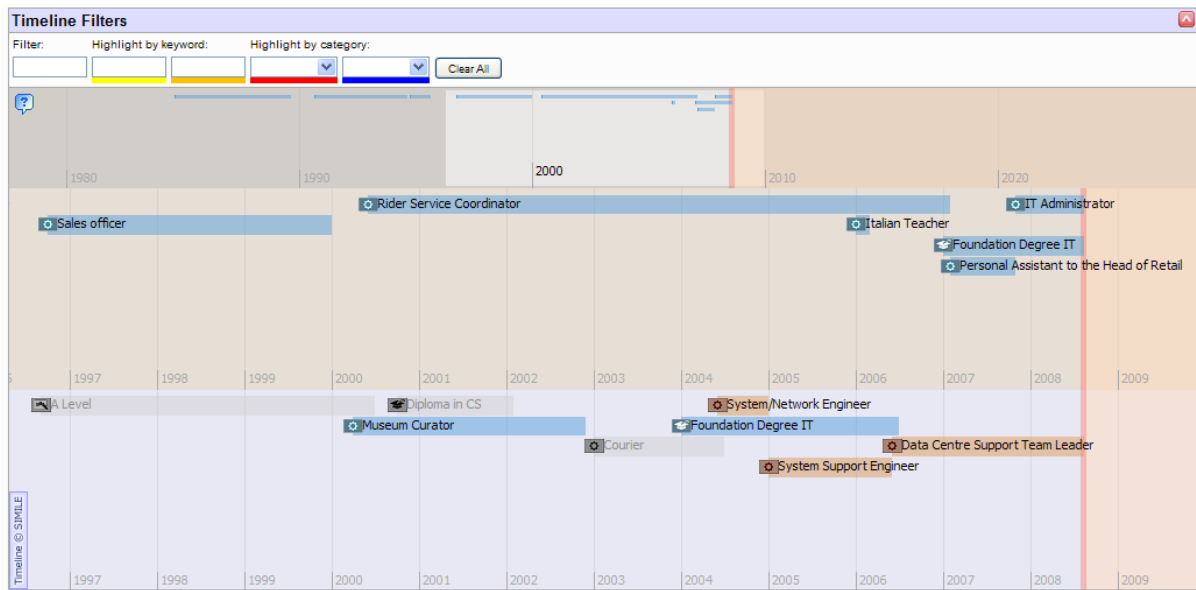
Figure 12. "What Next" timeline presented in highlight mode.
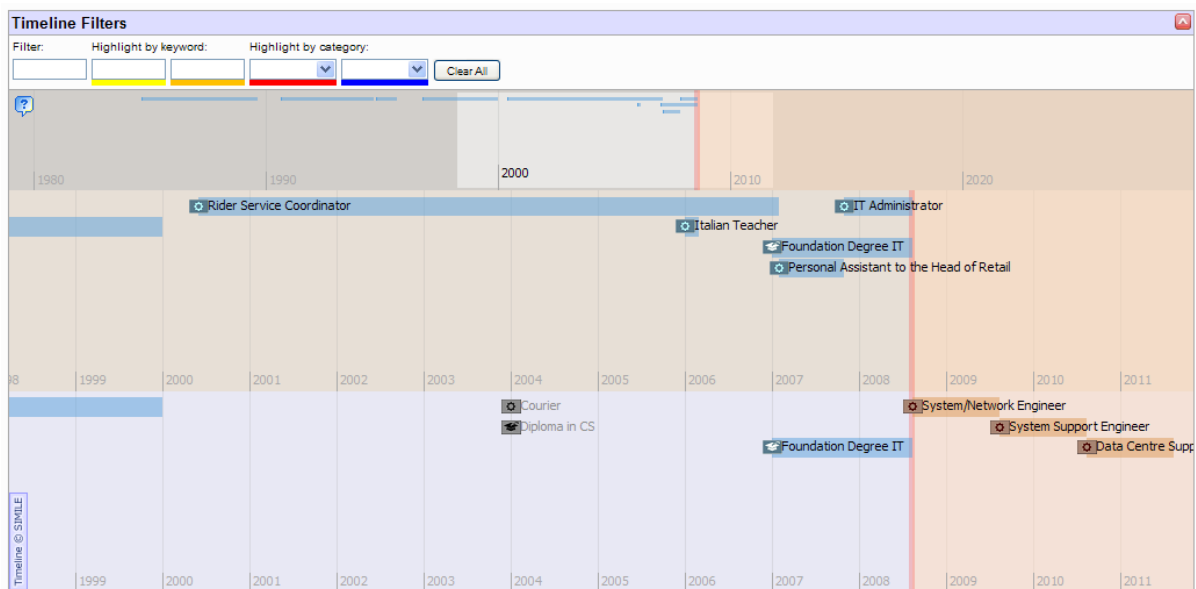
Figure 13. "What Next" timeline presented in recommendation mode.

Both these visualisation modes allow the user to explore the same target timeline, but with different emphasis: the highlight mode for locating the different episodes in the target timeline, preserving their initial timescale, and the recommendation mode for locating potential future goals. However, both modes only represent the *outcome* of the "what next" functionality, not the *process* itself.

A first attempt to present the process has been implemented, aiming at providing the user with explanation about how the target timeline episodes have been deemed to be relevant or not. In the current version of the system, this explanation is given as a simple representation of the alignment itself (see Figure 14), where episodes from both timelines are displayed in a list and their mapping indicated by different arrows:

- Left arrow (dim blue) for episodes in the user's timeline without a matching in the target;

- Double arrow (dark blue) for matching episodes;

- Right arrow (grey) for episodes in the target timeline judged irrelevant;

- Right arrow (red) for episodes in the target timeline judged relevant.
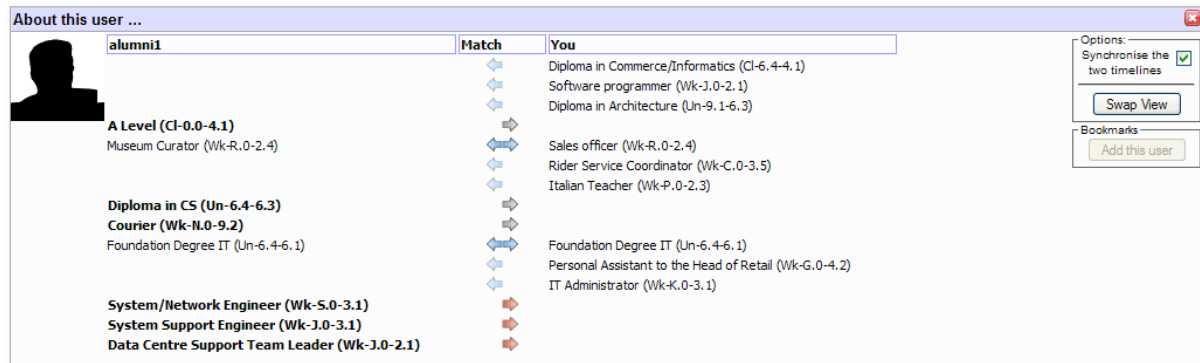
Figure 14. An alternative visualisation of the timelines' alignment.

4.5 Issues

The mechanism for exploring "what next" possibilities from others' timeline, as described in this document, is a first prototype, a proof-of-concept than a fully-fledged functionality. Our aim was to exploit the principles of similarity matching of timelines, as initially deployed in the "search for people like me" functionality, but in a slightly different context. Provision of this functionality also raises four issues that will need to be addressed. Issue (i) is inherent to similarity measures and has been mentioned when discussing the "search for people like me" in Del D4.1), while the others are specific to eliciting inspiration from timelines.

(i) Level of encoding

One of the recurrent issues, already raised for the "people like me" functionality, is the conflicting aspect of episode classifications and their encoding as a string of tokens. Providing a deeper, fine-grained classification of episodes is a necessary step for a better semantic description of a user's timeline. As such, the introduction of a dual classification (e.g. job and activity sector for work episodes, degree and domain for educational, etc, see Deliverable D5.1, appendix D) is a step in the right direction.

However, how detailed an episode needs to be for the purpose of applying similarity measures is an open question, whether for "people like me" or for "what next" functionalities. A wide comparison (with classifications ignored, e.g. **Un-0-0** for the Foundation Degree episode) or a narrow comparison (with classifications at maximum depth, e.g. **Un-6.4.0.0-6.1.0.0** for the same episode) will give a different similarity value, therefore different possible alignments and hence different potential sources of inspiration.

(ii) Multiple Alignments

As mentioned before, back-tracking the distance matrix computed by the Needleman-Wunsch algorithm may offer several possible alignments between the two timelines. Determining the "best" one is not an easy problem, as subjective factors related to the user's own definition of "relevance" and relations between episodes have to be taken into account. The current implementation of the mechanism is taking a straightforward approach by choosing the "left-first" alignment, hence privileging an alignment with the target's timeline. However, this might not always be a good choice, as exemplified in Figure 15.

User's Timeline (S1)		Target's Timeline (S2)				
		Wk-K.0-3.1	CI-0.0-4.1	Wk-G.0-3.1	Un-6.4-6.1	Wk-I.0-3.1
	0	1	2	3	4	5
CI-6.4-4.1	**1**	2	3	4	5	6
Wk-J.0-2.1	**2**	3	4	5	6	7
Un-9.1-6.3	**3**	4	5	6	7	8
Wk-R.0-2.4	**4**	5	6	7	8	9
Wk-C.0-3.5	**5**	6	7	8	9	10
Wk-P.0-2.3	**6**	**7**	**8**	**9**	10	11
Un-6.4-6.1	**7**	8	9	10	**9**	**10**
Wk-G.0-4.2	**8**	9	10	11	10	**11**
Wk-K.0-3.1	9	**8**	**9**	**10**	**11**	**12**

Alignment 1

	CI-6.4-4.1	Wk-J.0-2.1	Un-9.1-6.3	Wk-R.0-2.4	Wk-C.0-3.5	Wk-P.0-2.3	Un-6.4-6.1	Wk-G.0-4.2	Wk-K.0-3.1			
									Wk-K.0-3.1	CI-0.0-4.1	Wk-G.0-3.1	Un-6.4-6.1	Wk-I.0-3.1

Alignment 2

	CI-6.4-4.1	Wk-J.0-2.1	Un-9.1-6.3	Wk-R.0-2.4	Wk-C.0-3.5	Wk-P.0-2.3			Un-6.4-6.1		Wk-G.0-4.2	Wk-K.0-3.1
							Wk-K.0-3.1	CI-0.0-4.1	Wk-G.0-3.1	Un-6.4-6.1	Wk-I.0-3.1		

Figure 15. Impact of alternative alignments on the "what next" functionality.

The figure shows the distance matrix between the user's timeline previously used but a different target. As with the previous example, this target timeline was also built from an alumnus' real-life experiences and includes the same common episode (**Un-6.4-6.1**). But, by applying the "left-first" back-tracking approach (indicated as "alignment 1" and by the blue path), this episode is not detected as the "common ground", being preferred by another one (a work episode, **Wk-K.0-3.1**). As a result, the suggestion includes several potential episodes, some being clearly irrelevant (the **CI-0.0.-4.1** episode corresponds to a college episode, a degree much lower than the user's highest qualification). However, should an alternate back-tracking be taken (indicated as "alignment 2" and by the red path), a "better" matching (based on the **Un-6.4-6.1** episode) will be presented to the user, with a much better indication of relevant/irrelevant episodes.

The reason for this "mismatch" can be attributed to two factors. First, both timelines contain the same pair of episodes (**Un-6.4-6.1** and **Wk-K.0-3.1**) but in a reverse order. Aligning both episodes is therefore impossible, resulting in the two different paths indicated above. Second, and more critically, the only reason we were able to analyse the defect of the first alignment and prefer the second one was because of the subjective and expert knowledge we had about the real-life experience of the user, but not integrated into its timeline representation.

(iii) Identification of relevant episodes

The approach we have been taking for identifying episodes that are potentially a source of inspiration for users has been based exclusively on their relative position within the alignment between the user and the target timelines.

(iv) Similarity vs. dissimilarity

The core of the alignment mechanism described above is the Needleman-Wunsch similarity

metric, from which the back-tracking mechanism from the distance matrix is particularly important. However, the main purpose of these metrics, that is to compute the *similarity between two strings*, has proved to be not that useful in this context.

By definition, similarity metrics compute a (normalised) distance between the two timelines, a value ranging from 0 (totally different) and 1 (totally similar). We are using this value to rank the timelines returned by the search procedures (i.e. both the "search for people like me" procedure and the "what next" procedure). However, what seemed to be useful in the former case (albeit not really informative, see Deliverable D5.3) proved to be relatively useless in the latter case.

The top-ranked timelines in the returned list for the "what next" search will be the timelines that are very similar to the user's timeline, that is to say timelines that may offer very few suggestions of episodes for future development. On the other hand, bottom-ranked timelines, being increasingly dissimilar to the user's timeline, may not offer any "common ground", i.e. no common experiences with the user.

This is not to say that the matching mechanism in itself is not useful. As far as one-to-one comparison and alignment of timelines are concerned, it provides an interesting framework for finding common patterns of experience between users and suggesting future episodes as a source of inspiration. What is not really appropriate in this context is the ranking of timelines according to their *similarity.*

What could be more effective would be a measure that would take into account *both* the similarity between the common ground of the aligned timelines (combined with their prior experience) *and, at the same time,* the dissimilarity of the remaining episodes, proposed as suggestions for future development. The similarity component of the metrics will represent the similar development of both users, up to a certain point in time, whereas the dissimilarity will represent the variability of potential experiences that may arise from this point onward (see Figure 16).
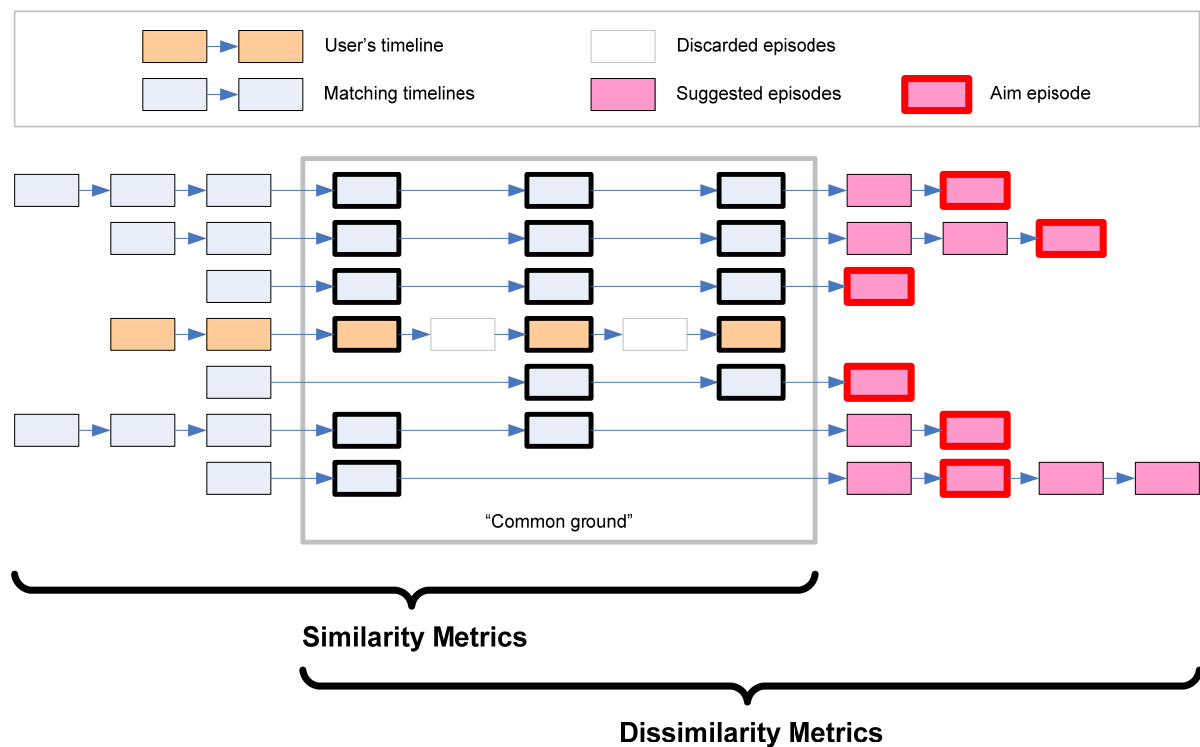

Figure 16. A holistic approach to timeline matching.

5 Outcomes and Conclusions

The initial personalisation requirements, as elaborated in Deliverable D3.1 (MyPlan Personalisation Specification), identified the five following aims for Workpackage 4 of the MyPlan project:

1. Personalised search of timelines from "people like me"

2. Personalised recommendation of which course(s) to study next

3. Customisation of the system

4. Automatic update of users' profiles

5. Ability to record and display ratings of search results

The first aim was the focus of the first version of the personalisation engine and has been described in Deliverable D4.1. The main outcome of this work has been the introduction of a method, based on string similarity measures, for identifying the degree of similarity between the timelines of two users. A second outcome has been a complete redesign of the system's interface, in particular the way timelines are displayed. Indeed, a major issue remains on how users can be supported in exploring graphically one timeline (their own) or several timelines simultaneously (e.g. "people like me").

The second aim of Workpackage 4 was reoriented as a result of the Phase I evaluation we undertook and following advice from the project advisory group (see Deliverable 5.1). The aim of a "recommendation" mechanism has been reoriented toward a less formal "what could I do next?" mechanism. The main outcome of this approach has been a proposal for reusing the same method as for finding "people like me", i.e. string similarity measures, but exploiting some of their capabilities for identifying token alignments, and therefore identifying timeline episodes that could represent a potential source of inspiration for the user.

The third aim has been integrated within the redesign of the systems' interface. By switching to a more homogeneous architecture for the front-end (DHTML rather than third-party technology such as Flash), more possibilities for offering users a personalised experience of the system have been put into place. In the current system, the focus has been more on customising the front-end (see Deliverable D4.1 for a description of some of these possibilities) rather than on personalising the back-end services, given their still exploratory status.

The last two aims above – automatic updates of users' profile and rating of search results – have not been addressed. Possibilities for automatically updating certain attributes of the user profile are evident in the system e.g. extracting the "highest qualification" attribute from the educational episodes in the timeline. However, the approach we took for searching for "similar" people is based on their timelines and raises questions about the need and indeed relevance of some of the information currently present in the user profile. Moreover, an automatic update such as for "highest qualification" relies on an ontology of educational qualifications; we have indeed introduced such an ontology in this project but this has not yet been carefully evaluated. Regarding the issue of rating of search results (aim 5 above), the envisioned approach of using expert-defined timelines and/or recommendations for courses has been replaced during the project by an exploration of peer-defined timelines. In such a context, rating becomes a very sensitive question that could not have been dealt with effectively within the timescale and resources of the current MyPlan project.

Despite the non-delivery of aims 4 and 5 above, we believe that the work undertaken within this Workpackage has made some important contributions to the overall aims of the MyPlan project. We also believe that the project will benefit in the long run by addressing two crucial points. The first point, as mentioned earlier, concerns the external representation of timelines. Evaluations of the L4All system have pointed out that the current dynamic widget,

although very attractive, is not necessarily the most intuitive way of visualising timelines – at least in the early stage of a user's timeline construction. Moreover, the diversity of tasks revolving round the timeline (e.g. managing your own timeline, searching for similar people, searching for inspirational timelines and episodes, reflecting on others' timelines, etc.), although based on the same information, strongly suggests that a single, well-designed visualisation is unachievable and that multiple, complementary or supplementary, dynamically connected representations, should be investigated.

The second point arising is the need to integrate the system's core functionalities within a social platform (or several), to supply the social features that are critically missing from L4All and, therefore, to allow users to share more effectively their timeline with peers and to comment on others' timelines. The purpose of this facility would be to encourage reflection, support professional and peer advice in the planning process, draw in more users by "viral distribution", and allow users to draw on each others' personal knowledge.

Appendix A – Implementation Technologies

For the development of the L4All system, the following technologies have been used:

- Java SE Development Kit version 6 (with compiler compliance level defined at v1.4 for backward compatibility). http://java.sun.com/

- Apache Tomcat version 5.5.23 for the server. http://tomcat.apache.org/index.html

- Java Servlet Framework version 2.3 and JavaServer Pages (JSP) version 2.0 (part of the Tomcat server).

- MySQL version 5.0 for storing the RDF metadata through the JENA framework. http://www.mysql.com/

- Apache Jakarta Commons DBCP (Database Connection Pools) version 1.3 to manage a shared pool of connections to relational databases. http://commons.apache.org/dbcp/

- JENA Framework version 2.1 for storing, querying and updating RDF metadata over relational databases. http://jena.sourceforge.net/

- The SimMetrics Java library for computing similarity measures between strings of token. http://www.dcs.shef.ac.uk/~sam/stringmetrics.html

- The SIMILE Timeline Javascript library version 1.2 for visualising the L4All timelines. http://simile.mit.edu/timeline/

- Several components of the Zapatec AJAX Framework for designing the user interface (e.g. calendar widget, tree widget, form validation). http://www.zapatec.com/

- Ruthsarian Layout as a source of inspiration for designing the CSS-based layout of the L4ALL system. http://webhost.bridgew.edu/etribou/layouts/index.html

A.1 Quality Assurance and Testing

The L4All pilot development has been carried out following the QA guidelines given in the UKOLN specifications.

Testing of the various components of the system has been carried out, using QA tools when appropriate: *JUnit* for repetitive Java testing (exception, expected results, etc.), *HttpUnit* for website testing (form submission, http authentication, cookies, automatic page redirection, etc.), *ServletUnit* for testing servlet containers.

The new Graphical User Interface and its integration with the back-end system have been designed using an incremental prototyping methodology, repetitively checked by on-site usability tests (using volunteers in the London Knowledge Lab) and complemented by the Phase I evaluation session, as described in Del D5.1. A list of bugs and issues, identified during these sessions, is summarised again in Appendix B.

All the project components, including the back-end Java code and front-end Javascript and DHTML code, have been thoroughly documented. The project code (and in particular the dependencies to third-party libraries) will be checked for compliance to open source distribution, under the terms of the GNU General Public Licence. Both the code and its documentation (Javadoc and JSDoc) will be made available at the end of the MyPlan project.

Appendix B – List of Faults and Improvements

Below is a list of all identified issues arising from the first evaluation phase with Birkbeck and CCH participants, as described in Del D5.1. Most of the bugs, erratic behaviour or interface issues have been fixed for the release of Version 2 of the system.

However, a number of long-term issues remain open. They include developing an online tutorial (or at list a comprehensive help facility); and supporting features that where not the focus of the MyPlan project (integration with further services, expanding the scope of the user's profile, etc.).

Overall System

1.	There are problems with the language used in the interface: too much "techies" jargon, too ambiguous in places. For example:	**FIXED**
2.	It may be a good idea to have a tutorial for new users, with a dummy "model" timeline, to briefly present the concepts and interface, before having them creating their own timeline (bbk3). An introductory text for the site/page may be enough for first-time users (bbk1,bbk4)	**TO DO**
3.	Some participants experienced problems with their timeline being data-corrupted and will not display at all.	**FIXED**
4.	The intended sequence of actions in L4*All* (i.e. first create/modify your personal details, then edit your learning/work experience, then edit your timeline) should be made more clear at the interface.	**FIXED**
5.	The organisation of the menus should be made clearer, with a clear distinction between functionalities (modify details, search, etc.) and visualisation (scale and colour of timeline, etc.).	**FIXED**
6.	It is not clear to users what advantages/benefits come from the need to spend time in filling their timeline. Also there is no hint of how detailed the timeline should be filled in (bbk3)	**TO DO**
7.	The "help" option to be completed (bbk3). Could also benefit from some sort of "tooltip" for the commands and widgets (bbk9).	**TO DO**

Layout and GUI

8.	The layout of the L4*All* system is problematic on a small screen such as in Birkbeck's B12 lab used for the evaluation session. The main page just fits in the screen but the left (menu) and right (bookmarks) columns seriously limit the width of the timeline widget. Moreover, when displaying two timelines simultaneously, the need to scroll down to see both timelines and the user's profile is a bit difficult.	**TO DO**
9.	In the edit popup windows (details, background, new episode, etc.), the "close" button (i.e. closing the popup WITHOUT saving) is to close to the "save" button, resulting in many participants loosing data unexpectedly.	**FIXED**
10.	The popup windows are satisfactory in terms of size and usage but there is a problem with the similarity search: when a returned timeline is selected, the search popup is sent to the background (to show the timeline now displayed in the main widget) but a few students believed the popup to be closed and had to start the search again.	**TO DO**
11.	The "bookmarks" section can be closed but cannot be opened again (bbk8).	**FIXED**

External Representation of the Timeline

12. The initial display parameters of the timeline (month scale, 200pt wide, centred on today's date) leave newly added episodes outside of the visible part (some participants started with episode 30 years ago!).	**TO DO**
13. When first logging into L4*All*, the timeline widget is empty, as no episode has been added yet. The widget has therefore no obvious meaning to the new user.	**FIXED**
14. The timeline filters did not make sense at first and seemed to be more highlighting than filtering (bbk1)	**TO DO**
15. There is a lack of indication that the timeline widget can be dragged back and forth, scrolling in time. (bbk5)	**FIXED**
16. Did not initially understand the purpose of the two bands in the timeline widget (bbk1)	**TO DO**
17. Manipulating the 'detailed' timeline by double-clicking on the top 'summary' timeline is quite a good feature (bbk1)	
18. It may be interesting to see more than two timelines at the same time (bbk1)	**TO DO**
19. Once a second timeline is displayed, the purpose of the "synchronise" option is unclear (bbk3).	**TO DO**

Registration Procedure

20. Tabbing between fields is not working (bbk1,bbk4,bbk6)	**FIXED**
21. All registration fields are mandatory; the note about "bold" fields is superfluous (bbk3).	**FIXED**
22. The icons used to indicate mandatory fields, errors and correct should be explained (bbk8).	**FIXED**
23. There is no guide about how to choose your user identifier and password (e.g. "there should be at least X characters") (bbk8).	**FIXED**
24. After registration, there are no hints about what to do next (bbk4).	**TO DO**

Profile Modification

25. Parts of the information in the "Personalise / My Timeline" are not properly saved or returned in the form (bbk3, bbk4, bbk5).	**FIXED**
26. "My Background – Needs": can only select one learning method	**TO DO**
27. "My Background – Needs": data validation on budget not working, crashing the system.	**FIXED**
28. There should be more drop-down options for other qualifications (bbk1)	**TO DO**
29. The purpose of "Future Learning Needs" part of the profile is unclear (bbk3).``	**TO DO**
30. The "about me" part seems to be unfinished (bbk3), does not contain enough options (bbk4).	**TO DO**
31. In the "Personalise / My Timeline", it is problematic to suggest user will die in 2030! (bbk3)	**FIXED**

32. In the classification popup (e.g. for a present occupation),

    a. When the search item fails to find an appropriate element, the "no node found" is not very user-friendly (bbk4).

    b. There is no "select/OK" button, need to double-click is not intuitive (bbk4)

FIXED

Adding Episode to the Timeline

33. What is the purpose of the URL field? (bbk3).   **TO DO**

34. When the "create episode" fails because of missing information, the error message is not consistent with the rest of the application.   **FIXED**

    a. Returning to the initial form loses all entered information (bbk3).

35. The distinction between "I have done that episode" and "I wish to do that episode" is not clear.   **FIXED**

    a. The "nature" option does not seem to be context sensitive (bbk1)

36. The calendar, despite being acknowledged as a "standard" widget is difficult to use, especially when you have to set a date a long time in the future or in the past.   **TO DO**

    a. May be easier just to be able to enter the date (bbk2,bbk4, bbk5)

    b. When adding a "start" date, the "end" date should be automatically set to this one, not today's (bbk1, bbk5)

37. For courses or degrees, there is no option for indicating whether the course is under way or was not finished (bbk1)   **TO DO**

    a. Option to select "up to current date", in particular for current job (bbk3)

38. When selecting an episode category from the list, it is easy NOT to see the scrollable list and miss the personal and other categories (bbk5).   **TO DO**

39. When editing a new episode, the system crashes if all mandatory fields are not properly entered.   **FIXED**

Classification of Episode

40. Several participants pointed out that the classifications, while presenting the main categories, do not contain examples of what the category relates to. The search functionality in the classification browser is fine but will be better if searching was also based on these examples.   **TO DO**

41. Episode classifications are, in general, difficult to understand. It is not necessarily clear which element corresponds to the user's episode. Ambiguities between items are hard to figure out.   **TO DO**

42. School episodes, as all educational ones, have both a qualification and subject classification; the latter is definitely not applicable (bbk4)   **TO DO**

43. Classifications not easy for foreign students (bbk4)   **TO DO**

44. "Unknown" element is not enough; there is a need for "none" (bbk4).   **TO DO**

45. For educational episode other than "obtained a degree/diploma", the qualification classification is irrelevant or ambiguous: does it relate to the objective of the episode or to its requirement?   **TO DO**

Similarity search

46. You need to convey the benefit of finding people with similar timelines; this is CRUCIAL (bbk3) — **TO DO**

   a. What does it tell me if I find someone who is like me, based on criteria provided?

   b. Can I conclude anything from this?

   c. Need to create a set of examples to demonstrate how this timeline comparison is useful

47. The search form is not clear (bbk1, bbk3). — **FIXED**

48. The initial information for returned timelines is not very specific; only opening the timelines in the widget will be useful (bbk4). — **TO DO**

49. Could not see the ranking of timelines (bbk3). — **TO DO**

50. Search needs to be based on aspiration/wish (bbk4). — **TO DO**

51. Search should take learning interests and hobbies into account (bbk2) — **TO DO**

52. Search methods (rules 1-4) are not clear (bbk1,bbk3,bbk4) — **TO DO**

53. Level of classification not clear at all (bbk1,bbk3,bbk4) — **TO DO**

54. See the point of searching for people but what about linking it to the search of appropriate courses? (bbk1) — **TO DO**