IVY – INTERPRETING IN VIRTUAL REALITY

Report to accompany Deliverable 4.4

IVY Virtual Environment (Prototype MK.II)

Panagiotis D. Ritsos
[p.ritsos@bangor.ac.uk]

Robert Gittins
[rgittins@bangor.ac.uk]

Jonathan C. Roberts
[j.c.roberts@bangor.ac.uk]

December 2012

Project coordinator: Sabine Braun

The IVY project has been funded with support from the European Commission. This report reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Table of Contents

1. Introduction

1.1 Purpose

This document describes the final design and implementation of the IVY Virtual Environment (denoted Prototype MK.II). The description covers:

a) the Head-Up Display used as the main graphical user interface for controlling the IVY Virtual Environment
b) the implementation of the virtual world locations and their underlying modular architecture
c) the interpreting scenarios employed and their structure, encompassing audio and textual information
d) the scenario management system used to create, edit and update users and content for the IVY Virtual Environment.

This document should be read in connection with the IVY 3D Virtual Environment, implemented in Second Life.

1.2 Intended Audience and Reading Suggestions

This document is addressed to the persons and organisations involved in the IVY project, in particular researchers, developers, project managers, users, testers and documentation writers. It includes a detailed technical description of the IVY Virtual Environment (Prototype MK.II). Section 2 provides an overview of the requirements and important features of IVY-VE. Section 3 presents the prototype's architecture. Section 4 presents the implemented system in detail.

1.3 Project Scope

The purpose of the IVY project is to explore the usage of Virtual Reality in the training of interpreters and their potential clients. The IVY 3D Virtual Environment encompasses a number of virtual interpreting scenarios, where the participants/users can perform various interpreting tasks and other learning activities, as well as challenging simulations. The IVY 3D Virtual Environment also offers a powerful tool for interpreter trainers by allowing scenario and dialogue customisation. One of the major pedagogical aspects driving the development and evaluation of the environment was the enhancement of the users' sense of presence through the immersion in a virtual world. Ultimately, the IVY environment was developed to provide a platform for examining (in future research) the merits of using Virtual Environments in educational contexts compared to traditional face-to-face teaching or using other online learning environments.

511862-LLP-1-2010-1-UK-KA3-KA3MP

1.4 References

[1] IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications

[2] IEEE Std 1233, 1998 IEEE Guide of Developing System Requirements Specifications

[3] IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology

[4] L. Gonze, M. Friedrich, and R. Kaye, "*XML shareable playlist format version 1*," 2008. [Online]. Available: http://www.xspf.org/xspf-v1.html

[5] Ritsos, P.D.; Gittins, R.; Roberts, J.C.; Braun, S.; Slater, C., "Using Virtual Reality for Interpreter-mediated Communication and Training," *Cyberworlds (CW), 2012 International Conference on*, pp.191-198, 25-27 Sept. 2012, doi: 10.1109/CW.2012.34

[6] D.4.3 IVY-VE Prototype MK.I

2. Project Description

2.1 IVY 3D Virtual Environment Outline

Addressing the needs of trainee interpreters and users of interpreting services (in different educational contexts), project IVY has used the exciting features of 3D virtual environment technology to create an innovative virtual educational space that supports the acquisition and application of skills required in interpreter-mediated communication. This educational space is referred to as the IVY Virtual Environment (IVY-VE) and comprises:

- A dedicated 3D virtual environment for:

    a) interpreting students
    b) future/potential users of interpreters ('clients')

- A range of virtual interpreting scenarios (e.g. business meetings, presentations, sales pitches, interviews, guided tours, training seminars) that share a range of virtual locations (e.g. meeting room, presentation area, outdoor area, seminar room) and that can be run in different working modes:

    a) Interpreting and Learning Activity mode
    b) Exploration mode, where users of interpreting services and novice interpreters can learn about interpreting and how to work with an interpreter
    c) Live Interaction mode, where participants can engage in interpreting role-plays

- Multilingual audiovisual content for the interpreting scenarios, created by:

    a) adapting and supplementing the corpora (in EN, DE, FR, PL) from the LLP project BACKBONE[1]
    b) adding new language corpora (GR, EL and RU).

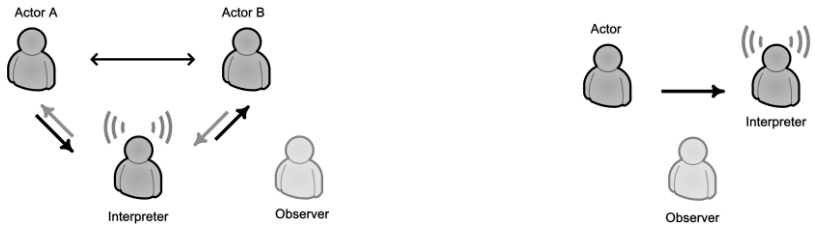
- Two sets of pedagogical material for

    a) the interpreting students
    b) the 'clients'.

IVY-VE supports a range of scenarios based on dialogues and monologues. In the Interpreting mode, the actors are avatars controlled by the environment, i.e. robots or non-player characters (NPCs), representing the interlocutors in each scenario. The user/interpreting student is present in the scenario with his/her own avatar, either to

[1] BACKBONE Corpora for Content and Language Integrated Learning, EU LLP 2009-10 project number 143502-2008-LLP-DE-KA2-KA2MP
[http://u-002-segsv001.uni-tuebingen.de/backbone/moodle/mod/resource/view.php?id=258]

interpret or observe. In the Live mode, the avatars of interpreting students, tutors and observers/visitors replace the NPC avatars. The Learning Activity and Exploration modes are complementary in that they provide a range of exercises and tasks and an induction to working with an interpreter respectively.



(a) Dialogue (b) Monologue

Fig.1 Communicative situations supported in IVY-VE

2.2 Virtual World Platform

The choice of virtual world for the first series of prototypes was Second Life. Second Life (2003) is a freely accessible 3D multi-user, collaborative, virtual world developed by Linden Labs, that has millions of subscribers around the world. Institutions, colleges and universities often use it for training and educational purposes. It falls into the category of a MultiUser Virtual Environment (MUVE).

The main advantages that led to its usage for the IVY project prototypes, when compared to other virtual worlds (such as OpenSim, ActiveWorlds, etc.) are:

- Second Life (SL) provides a large community of users, developers and enthusiasts. These users have created many add-ons and plugins, and therefore there are many examples of customisations.

- SL is platform for social interaction and education used by numerous institutions, colleges, universities – thus it was anticipated that the use of SL would increase the chances of exposure and information dissemination of the IVY project.

- The use of SL also enabled the consortium to build on prior work conducted by some of the IVY consortium partners and others. In particular, the Bangor IVY team has an established SL Island; this experience and prior experience allowed the virtual interpreting scenarios to be developed more quickly than in a purpose-built environment.

- SL does not require a user to run the VW him/herself, but can be accessed through public servers; consequently, prototypes were quickly built and shared between the developers and partners in IVY.

2.3 Scalability Limitations

The use of Second Life introduces a series of restrictions, due to the limitation in the number of objects (primitives) an island can hold and the lack of features such as instancing, found in modern massive multiplayer online games (MMOGs).

To compensate for these limitations and to maintain consistency in the virtual world, only one scenario can be executed per virtual location at a given time. IVE-VE has a unique setting for each type of scenario (e.g. classroom, meeting room). A particular scenario may share its location with another. For example, the meeting room location is used for business meetings, interviews and other similar dialogues; the seminar room is used for talks, lectures and other educational situations.

Upon a scenario selection and launch by a user, all scenarios sharing the same location become unavailable for other users, by means of a database lock. Once the user exits the selected scenario, all scenarios sharing the same location become available again. We preferred to maintain consistency in the virtual environment and explore scalability in future implementations. Figure 2 (below) shows the relationships and sequence of actions in IVY-VE. However, the locations have been duplicated to increase the opportunities for running scenarios in parallel. For example, the IVY environment uses three identical meetings rooms.

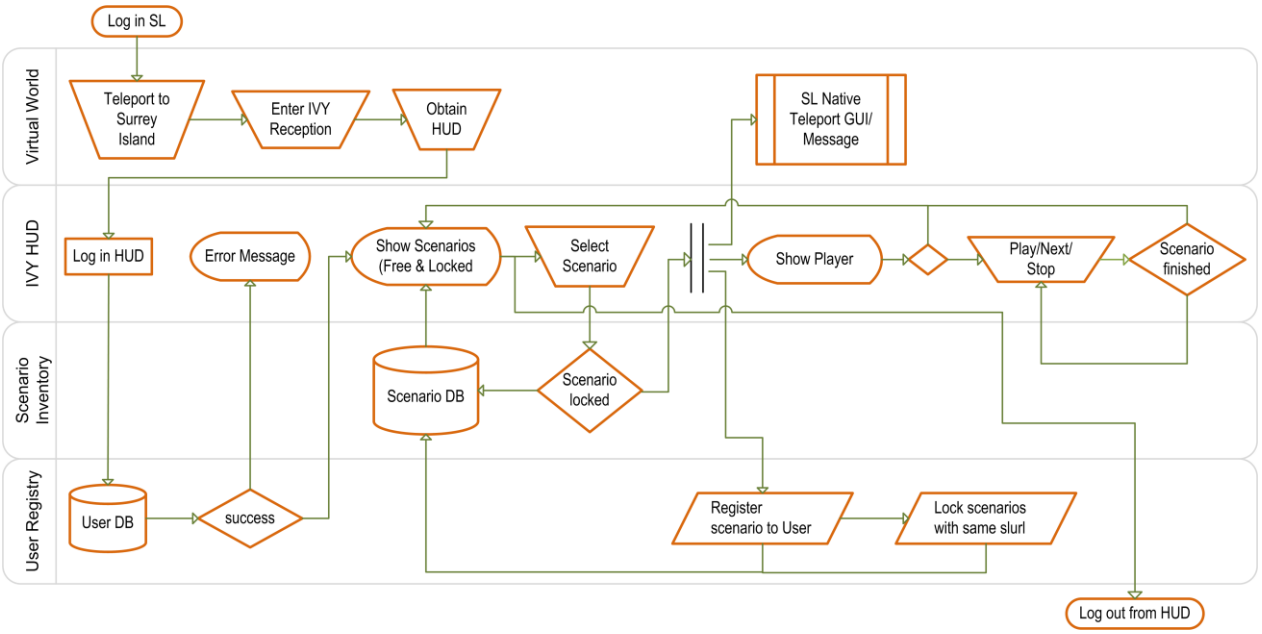


Fig 2 IVY-VE Cross-functional flowchart

2.4 User Classes and Characteristics

IVY-VE has the following user classes, mapping to individuals or groups from its intended audience:

- **Users**: the participants in the IVY-VE whose purpose is to explore, participate and practise with the scenarios in it. These are typically interpreting students.
- **Administrators/Content Managers**: the person(s) responsible for the management of the scenario and audio content. These are typically interpreting tutors and researchers.
- **Developers**: the person(s) responsible for the IVY-VE maintenance and support.
- **Observers**: the participants in the IVY-VE whose purpose is to observe other users. These include clients of interpreters.

3. IVY VE MK.II Design and Architecture

The IVY-VE MK.II merges useful features from web technologies and native coding in Second Life, using Linden Scripting Language (LSL), resulting in a hybrid solution. It is built around a web application with two entry points.

The main entry point into the IVY-VE is accessible from within SL in the form of a Heads-up Display (HUD). The HUD features a menu-driven system and is populated from a database. It shows the available working modes, language combinations and scenarios (including monologue/dialogue options) to the users. Upon scenario selection, the HUD gives access to an audio player and initiates in-world teleport events. The other entry point is an administration panel which remains independent of SL and is used by content managers to manage scenarios and user information. The following diagram (Fig.3) presents an architectural overview of the system's sub-components, separated into three functionality layers, corresponding to implementation modules:

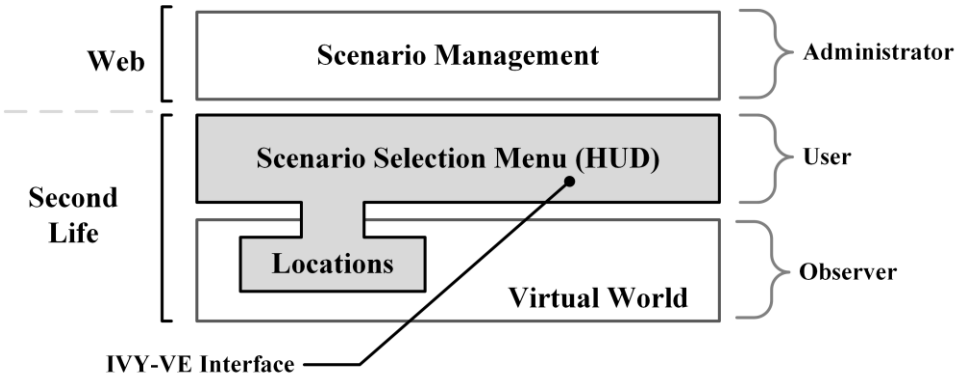


Fig.3 IVY - VE Architecture

4. IVY-VE MK.II Implementation

4.1 Overview

Bearing in mind the aforementioned architecture, IVY-VE is implemented with elements in two different domains:

- the web domain, where a web application, encompassing the HUD and administration panel, interfaces with the database;
- the virtual world of Second Life, where scenario locations were created.

The HUD can be obtained in the virtual world as an object, providing an extension to the native SL interface and an in-world view of the contents of the database, where scenario-specific audio and text reside. Each scenario consists of a sequence of audio files and a playlist with additional textual information – title, keywords (domains), location where the monologue/dialogue takes place and a description of the communicative situation ("brief"). Providing these resources, i.e. the audio files sequence and the textual information, is performed through the aforementioned administration panel, through a simple, intuitive interface, outside of the VE. The audio files themselves are uploaded separately, using FTP in a repository, which is accessible from the web application.

The hierarchically arranged menu in the HUD allows the user to make a series of choices until the user reaches a list of audio materials that match his/her selections. Choices, in order of importance, are Form (dialogue/monologue), Language (EN, DE, FR, PL, GR, EL and RU) and Dialogue/Monologue Title. An information panel presents the aforementioned textual information relating to the monologue/dialogue at hand. A launch button initiates a teleport function in the virtual world to the scenario's location, allowing the user to move instantly in situ, and launches the player controls. Controlling the scenario is done much as it is in any player, with the usual functions of play, stop, fast-forward and rewind. The fast-forward and rewind buttons are used to navigate between dialogue turns and monologue sections. The functions of the player are described in the Handbook (Deliverable 4.4). The pedagogical rationale for the functions of the player is outlined in Deliverable 3.3.

4.2 IVY-VE Web Application

The IVY web application, which effectively includes both the HUD and Administration interfaces, is implemented using the Appfuse 2 open source Java EE framework[2]. Appfuse uses industry-standard Java technologies such as Spring Framework[3] and Apache Struts[4], both of which are employed in this implementation. It (Appfuse) also has JPA support for database operations and uses Apache Maven[5] integration for project. Our prototype is

[2] http://www.appfuse.org
[3] http://www.springsource.org/
[4] http://www.springsource.org/
[5] http://maven.apache.org/

deployed using Apache Tomcat 6.x open source web server and employs the MySql 5.x database.

Appfuse includes the following features, which are used in IVY-VE:

- **Generic CRUD backend,** allowing the design and implementation of a scenario management system with mechanisms for creating, uploading, editing and deleting scenarios.
- **Authentication and authorisation**, allowing easy implementation of access control based on Spring Security.
- **User management**, enabling system and content administrators to easily manage the pool of users (students) and their access level.
- **Strong internationalisation support**, providing a way of translating the web application in all the languages supported in our corpus.

4.3 Audio File Format, Segmentation and Management

As a basis for the prepared monologues and dialogues that are used in the Interpreting mode, the IVY-VE uses audio extracts (segments), in MPEG-2 Audio Layer III format, adapted and/or created from the LLP project BACKBONE and other audio/video corpora, wrapped in XSPF[6], (*XML Shareable Playlist Format*) [4] play-lists (scripts). An example of the XML playlist can be seen in Appendix C.

The configuration of the system for handling dialogues and monologues is based on the pedagogical design outlined in Deliverable 3.3. Thus, dialogues are made up of 'A' turns and 'B' turns, whereas monologues have only 'A' turns. In the dialogues, the turns do not necessarily need to follow an ABAB pattern (i.e. questions and answers in alternating order), but can have an arbitrary order to allow for greater periods, where a speaker talks, to be broken in segments, thus allowing an interpreter to focus on shorter segments. Similarly, there are no technical requirements relating to which speaker starts or ends the dialogue. However, it is assumed that only one actor talks per audio segment and there is no overlap between actors' speech.

Acceptable speaker turn sequences for dialogues are, for example:

ABABAB	(strict alternating turn sequence)
ABBAABAABA	(irregular turn sequence)

[6] http://www.xspf.org/

Audio segments are uniquely named and can be interchanged — within each script — to derive further language combinations of that scenario. Dialogue and Monologue files follow the naming pattern shown below:

B_DEN_000010_0000

```
 |  |  |          |        |
 |  |  |          |        |------   Audio File and Text File Sequence Number (0-N) [5]
 |  |  |          |---------------   Dialogue number (From Look-up table) [4]
 |  |  |----------------------------   Language [3]
 |  |-------------------------------   Form (Monologue/Dialogue) [2]
 |----------------------------------   Actor A or B [1]
```

[1] Actor: Notation for the VE participant. “A” is used for the person asking the questions; “B” is used for the person responding to the questions.

[2] Form: Dialogue or Monologue (D for Dialogue, M for Monologue)

[3] Language: (maintain 2-letter designation).

[4] Dialogues and monologues are uniquely numbered per form AND per language. The last digit of the sequence number is always 0 (i.e. the above example designates dialogue 1 not 10). This allows a 'filler' digit in case other dialogues are added in the future within some additional classification or category.

[5] The files corresponding to the turns of each actor within each dialogue are numbered consecutively.

Separate number sequences are therefore created for the A and B turns. There is also a separate numbering sequence for each language. These are start as follows:

English:	000010_0000
French:	010010_0000
German:	020010_0000
Polish:	030010_0000
Greek:	040010_0000
Hebrew:	050010_0000
Russian:	060010_0000

The file naming conventions for monologues are the same as those for the dialogues, except that they always begin with ‘A_M’, as there are only A turns in monologues. For more examples see Appendix A.

Each script has textual information associated with it, namely title, keywords (domains), location where the monologue/dialogue takes place and a description of the communicative situation. Last but not least, the order of audio files is also required and is provided through the Administration Panel (described in Section 4.6, below).

4.4 Scenario Environment and Actors

A series of virtual locations were developed on the IVY Island, using several geometric models and textures. Photorealistic landscaping of both immediate scenario locations and distant views provided a generic setting for the project. Created locations include a reception area, meeting rooms, classrooms, office settings, an exhibition community centre, shops and conference rooms (Fig. 4, below). The goal throughout was to use the smallest number of primitives to allow scalability and replication in the future. The reception area serves as the central hub for visitors where they can obtain the HUD, view introductory videos and read notice boards with information.

The IVY virtual locations are separated into two main groups: those used for the Live mode and those used for the Interpreting Practice mode. Live mode locations, along with an exhibition hall that was deliberately designed to accommodate the Exploration mode, are located on the IVY Island floor. Users can access them easily, allowing increased opportunities for social interaction. Interpreting mode locations are placed on sky platforms.

Locations that are used in multiple scenarios, such as classrooms and meeting rooms, have been duplicated to match the 'demand' created by the available content. For example, the IVY monologue/dialogue corpus contains a relatively large number of communicative situations that make use of the meeting room location. In accordance with this, the meeting room has been duplicated so that a number of students can use monologues/dialogues that take place in the meeting room at the same time, thereby increasing practice opportunities. Each location is a modular component that can be replicated, depending on the frequency of usage. Also, by placing these in the sky, other (non-participating) avatars are less likely to walk into them, introducing an element of exclusivity for each location and requiring users to have to use the HUD to teleport to them.

Each scenario is manually populated with Second Life 'robots' (bots) as the actors, according to the corpus' requirements. These 'bots' are from Pikkubot[7] and Thoys[8]. Each bot group is controlled by a dedicated server, either using in-world chat or, in the case of Pikkubot, command lines in the control server's telnet prompt. The bots also use recursive animation overrides (AO) that make them appear life-like.

A series of information and presentation boards have been placed in the aforementioned exhibition area, for use in the Exploration mode. Two types of presentations, one using Microsoft PowerPoint presentations and one using HTML-on-a-prim boards are available. The latter use jQuery to display textual information grouped into sections and presented as a collapsible accordion menu, minimising the need for vertical scrolling. Informative videos are also embedded on virtual boards, using the same strategy.

[7] http://www.pikkubot.de
[8] http://www.slbot.thoys.nl

Fig.4 Screenshots of some IVY-VE locations

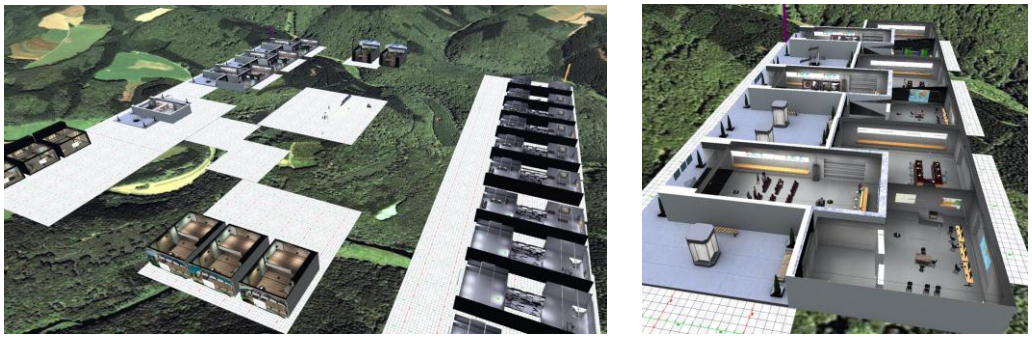


Fig.5 Aerial view of the sky platforms, where rooms for the Interpreting mode are placed

4.5 Heads-Up Display Menu Design

The HUD makes use of the 'HTML-on-a-prim' feature of Second Life, where HTML can be mapped on virtual objects (boards, panels, etc). Each view of the HUD is essentially a webpage, cropped to a small footprint and designed to match aesthetically the SL viewer interface.

The HUD (Fig. 6) is built using the jQuery JavaScript library. It displays a list of scenarios from the web-application's database as a drill-down menu and is, normally, placed at the lower left of the user's viewport. It is noteworthy that by using web technologies the text is rendered in-world in the best available quality, particularly when compared to text rendered on 'prims' using native LSL. In addition, the internationalisation features of Appfuse along with the character sets supported in the web enables our system to display textual information properly in all languages in our corpus.

The functionality of the HUD includes:

- **Navigation:** Navigating on the island and teleporting to each scenario is achieved using `slurls` (url-like links) of teleport destinations. Teleport events are initiated by passing the `slurl` to the SL client, upon scenario (or location for Live and Exploration modes) selection, by means of a JavaScript trigger in the HUD.

- **Mode Selection:** Working modes can be selected in the Mode Selection view. Each mode is denoted by an icon.

- **Utility Functions/Buttons:** Log-off and mode selection buttons are placed in all views, allowing for a quicker and more intuitive interface. A teleportation button to the IVY reception is available in the mode selection view.

- **Audio File Playback:** Audio is reproduced by means of a Flash player, embedded in the HUD's view, parsing the XSPF Playlists. The player interface allows users to start, pause, repeat and advance tracks, corresponding to each turn of the monologue/dialogue.

- **Textual Information:** Each mode view has textual information and icons to assist users. In the case of the Interpreting mode, upon scenario launch, users can see the scenario information, i.e., the title, keywords (domains), location where the monologue/dialogue takes place and a description of the communicative situation..

- **Access to SL GUI Functions** (limited): A limited number of `slurls` from the SL Viewer URI Name Space, having the prefix `secondliffe:///app`, allows access to elements of the native SL client's user interface. Such a `slurl`, launching the IVY group inspection pane is available in Live mode, allowing users to see the members of the group that are online and offer them teleports to where other members may be.

Fig.6 Screen captures of the IVY-VE HUD

The HUD can be expanded to a browser in SL, providing access to a series of supportive material for each scenario, especially the various types of learning activity in the Learning

Activity mode. Clicking on the right-most button on the HUD address bar duplicates the HUD in a browser. In this expanded form and underneath the player interface, the user can find different learning activities relevant to the monologue/dialogue that the user has selected. Generic preparatory, skills-based and reflective activities can be used with any monologue/dialogue that the user is working with. In addition, a number of specific monologues/dialogues have more customised learning activities attached to them, in which learners can focus on preparing for that particular interpreting assignment and then reflect on specific aspects and interpreting challenges that arose in that monologue/dialogue (see Deliverable 5.3).

Fig. 7 The HUD's address bar with browser-like controls (native SL interface)

The learning activities are PDF files, stored in the application server and viewed in the HUD by means of an Embeddable Google Document Viewer. Each document's URL (assembled from each scenario's unique ID and file path to the PDF directory) is used as reference, in order to display the appropriate exercises.

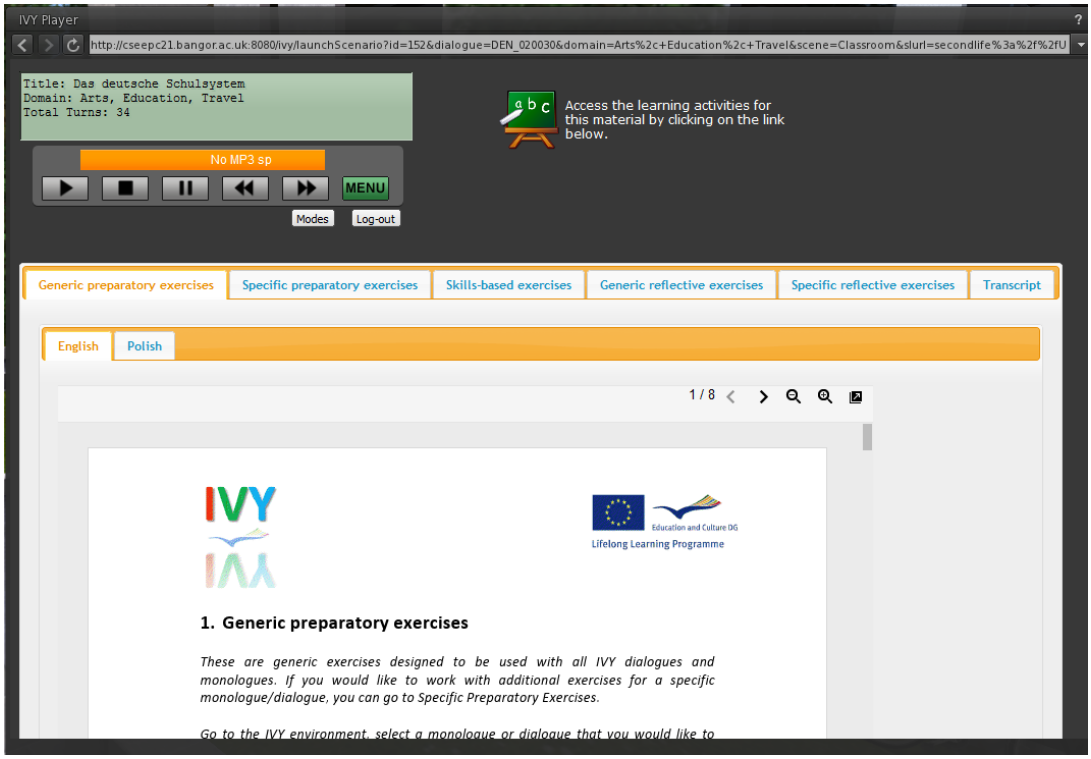


Fig. 8 Expanded HUD, displaying preparatory and reflective learning activities

4.6 Administration Panel

The IVY-VE is managed by a custom user and scenario management system with strong internationalisation features to support the multilingual corpus. Content administrators can create, edit and inspect users and scenarios, through a form-based interface (Fig.9). For scenarios they provide each scenario's textual information, accompanied by an ordered list of the audio tracks associated with it. Textual information is stored in the database and the audio file listing in XSPF format, in the playlist inventory. Audio tracks are manually uploaded to the system through FTP. User information, including name/surname, contact email and security level (user, admin) are stored in the database as well.

A scenario list view allows content administrators to monitor the usage of each scenario, by showing which user is currently working on it and which other scenarios, sharing the same location, are unavailable at a given time. Administrators can initiate teleport events and pass them to their local SL client by clicking on *slurl*-based links.

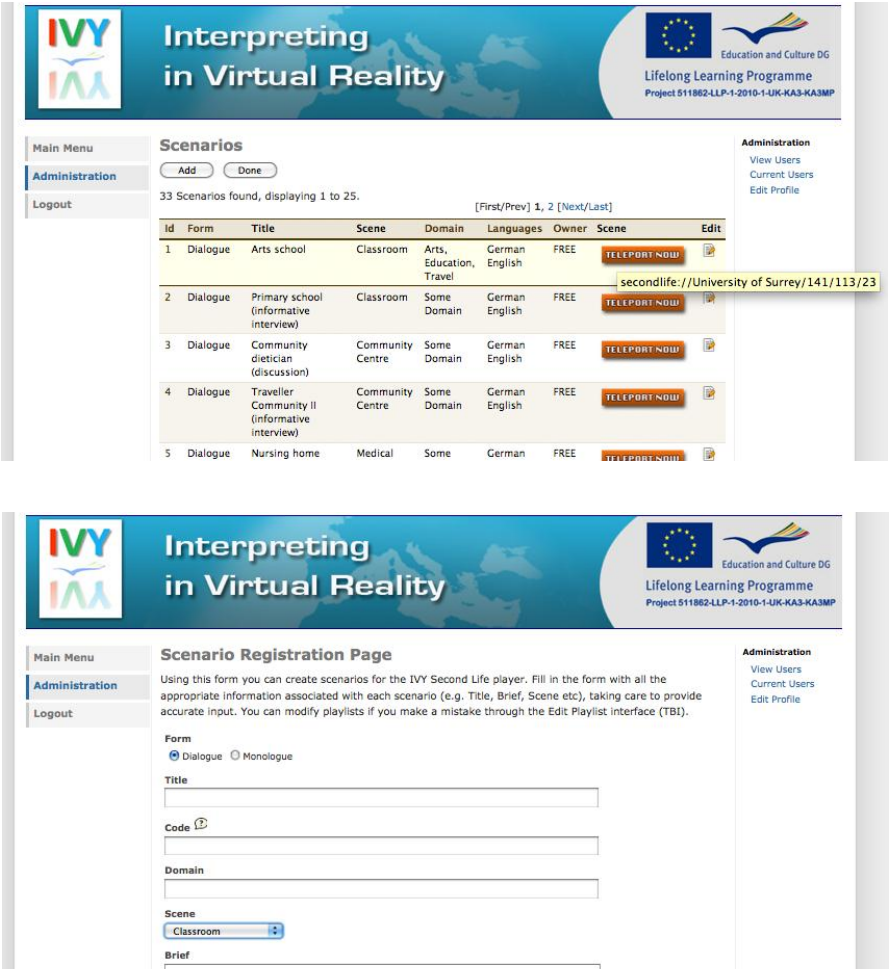


Fig. 9 Views of the Administration panels

Appendix A: Dialogue Examples

Example 1:

Imagine the following dialogue has questions in Polish and answers in English. It is made up of 8 sections: 4 questions (A) and 4 answers (B) (i.e. ABABABAB). The audio files for this dialogue are therefore as follows:

A_DPL_000010_0001 (= turn 1 in the dialogue)
A_DPL_000010_0002 (= turn 3 in the dialogue, because interspersed by B1)
A_DPL_000010_0003 (= turn 5 in the dialogue, because interspersed by B2)
A_DPL_000010_0004 (= turn 7 in the dialogue)

AND

B_DEN_000010_0001 (= turn 2 in the dialogue)
B_DEN_000010_0002 (= turn 4 in the dialogue, because interspersed by A2)
B_DEN_000010_0003 (= turn 6 in the dialogue, because interspersed by A3)
B_DEN_000010_0004 (= turn 8 in the dialogue)

Example 2:

Imagine the following dialogue (dialogue 1 in the list of dialogues with German questions and Polish answers). It is made up of 7 sections: 4 questions (A) and 3 answers (B) (e.g. ABBAABA). The files for this dialogue are therefore as follows:

A_DGE_030010_0001
A_DGE_030010_0002
A_DGE_030010_0003
A_DGE_030010_0004

AND

B_DPL_030010_0001
B_DPL_030010_0002
B_DPL_030010_0003

Appendix B: IVY XML Playlist

```
<?xml version="1.0" encoding="UTF-16" standalone="no"?>
<playlist xmlns="http://xspf.org/ns/0/" version="1">
<title>Bookshop</title>
      <trackList>
            <track>
            <location>/Inventory/mp3/A_DGR_000020_0001.mp3</location>
            <title>Participant A Turn: 1/2</title>
            </track>
            <track>
            <location>/Inventory/mp3/B_DEN_000020_0001.mp3</location>
            <title>Participant B Turn: 1/2</title>
            </track>
            <track>
            <location>/Inventory/mp3/A_DGR_000020_0002.mp3</location>
            <title>Participant A Turn: 2/2</title>
            </track>
            <track>
            <location>/Inventory/mp3/B_DEN_000020_0002.mp3</location>
            <title>Participant B Turn: 2/2</title>
            </track>
</trackList>
</playlist>
```