

Light Water Reactor Sustainability Program

Report for 2.2.1 Task 2: Provide Support for the Human Factors Studies Conducted for the LWRS Control Room Modernization Project for PVNGS

Michael Hildebrandt, Jens-Patrick Langstrand, and Hoa Nguyen

September 2018



U.S. Department of Energy

Office of Nuclear Energy

DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness, of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Light Water Reactor Sustainability Program

Report for 2.2.1 Task 2: Provide Support for the Human Factors Studies Conducted for the LWRS Control Room Modernization Project for PVNGS

Michael Hildebrandt, Jens-Patrick Langstrand, and Hoa Nguyen

September 2018

**Idaho National Laboratory
Idaho Falls, Idaho 83415**

**Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517**

IFE/INL-196543

SOW 14512

**Report for 2.2.1 Task 2: Provide
Execution Support for the Human
Factors Studies Conducted for the
LWRS Control Room Modernization
Project for PVNGS**



Address Telephone Telefax	KJELLER NO-2027 Kjeller, Norway +47 63 80 60 00 +47 63 81 63 56	HALDEN NO-1751 Halden, Norway +47 69 21 22 00 +47 69 21 22 01	
Report number IFE/INL-196543			Date 2018-09-07
Report title and subtitle Report for 2.2.1 Task 2: Provide Support for the Human Factors Studies Conducted for the LWRS Control Room Modernization Project for PVNGS			Number of pages 16
Project/Contract no. and name Contract no. 196543			
Client/Sponsor Organisation and reference Idaho National Lab, USA			
Abstract INL contracted IFE to develop custom software functionality to support INL's use of eye movement tracking in simulator studies. This includes the support of remote eye trackers, the support of dynamic areas of interest based on simulator events, and a general-purpose communication protocol between a simulator and the eye-tracking software. The report summarizes the task deliverables, which include a requirements identification workshop at INL, a software package, and instructional videos.			
Name		Date	Signature
Author(s)	Michael Hildebrandt Jens-Patrick Langstrand Hoa Nguyen	2018-09-07	
Approved by	Robert McDonald	2018-09-07	

1 Background

Institut for Energiteknikk (IFE) operates the OECD Halden Reactor Project and has extensive experience from more than 20 years of research in human system interface design and operation of nuclear power plant research simulators.

IFE also develops solutions for the efficient assessment of human performance and of cognitive-behavioral states of users. This includes the use of eye tracking, motion capture, and physiological measures.

Idaho National Laboratory (INL) has contracted IFE to develop tools for the efficient analysis of eye movement data during simulator studies. INL uses eye-tracking technology in its Human System Simulation Laboratory (HSSL) and for on-site studies at plants. INL currently owns two head-mounted eye trackers (eye tracking glasses; ETG) and one screen-mounted eye tracker (remote eye tracker; RET).

2 Introduction

Eye tracking can provide valuable insights into the behavior of control room operators by revealing which information they are and are not perceiving. Eye tracking can also support design evaluations by revealing how often and under what circumstances operators use certain information features. It can also reveal how much visual searching is needed to locate relevant information. Finally, eye metrics such as pupil dilation and blink rate can be used to calculate measures of cognitive workload, distraction, and fatigue.

The challenge with current commercial eye tracking systems is that the analysis of eye data can be very time-consuming. Therefore, researchers often cannot afford to extract all relevant information from the data. As an example, the manual coding of gaze points for a one-hour ETG recording takes about four to five hours. This effort is multiplied when two, three, or four operators need to be tracked. While eye tracking vendors attempt to use computer vision to automatically code gaze points, this approach is currently not accurate enough to provide good results in simulator studies.

Another problem with commercial eye tracking systems is the difficulty in linking them to simulator software. It is often desirable to record simulator events in the eye tracking data logs, for example, to analyze changes in workload indicators before and after an alarm. For this, a communication protocol between the simulator and the eye tracking software is needed, and the eye tracking software has to support the creation of markers in the log files.

INL requested IFE to develop customized eye tracking functionality for use in INL operator studies. This functionality is built on top of IFE's Synopticon software package. Synopticon can receive a wide range of data streams, including physiological data (gaze direction, head position, head orientation, heart rate, etc.), simulator data (events, operator actions, etc.), and external media (video and audio). Synopticon merges these

data streams, visualizes them in a single, unified 3D representation of the environment, and can produce real-time analysis of the data. It also supports playback and export of the recorded data.

3 Scope

While the core deliverable is a set of software functionalities, it was first necessary to identify INL's needs. This was achieved through a three-day workshop at INL, which was attended by two of IFE's eye tracking specialists. During that workshop, a set of functionalities was agreed upon between INL and IFE. IFE then developed the necessary functionality and delivered a software package to INL. To explain the functionality and usage of the system, instructional videos were developed. Finally, this report was generated to document the deliverables.

Summary of deliverables:

1. Requirements workshop at INL
2. Software package with custom eye tracking functionality
3. Instructional videos
4. Project report.

Each of the deliverables is described in more detail below.

4 Deliverable 1: Requirements Workshop at INL

A requirements identification workshop was conducted at INL in June 2018. The workshop was attended by various INL staff and by two IFE researchers.

4.1 Synopticon Demo

IFE staff gave a full demo of Synopticon. Because many of Synopticon's features rely on the fusion of eye tracking and position tracking data, half of the HSSL was outfitted with motion capture (MoCap) cameras. Then, a highly accurate 3D model of the screens in this area was generated using the MoCap cameras and the Synopticon software (Figure 1). Screenshots from these screens were used as textures in the 3D model. The team then demonstrated how MoCap markers are installed on the eye-tracking glasses (Figure 2), how these markers are registered in the software, and how they are calibrated.



Figure 1. This model of five HSSL bays was developed with Synopticon’s modelling function.



Figure 2. Synopticon combines gaze data from eye tracking glasses and position data from reflective markers (attached to the glasses) to automatically measure what an operator is looking at.

In addition to using the tracking glasses, the team also demonstrated how the operator’s hands can be tracked and how collision objects can be defined in order to log when and

where an operator touches the screen (Figure 3). Besides these functions, Synopticon also supports object tracking, for instance the tracking of a tablet computer.

The team also demonstrated how to track only the head position of a participant without eye tracking. For that purpose, a MoCap marker is attached to the head of the participant, for instance on top of a baseball cap. The general gaze direction of the participant can then be calibrated. The head orientation can give a rough indication of where the participant is looking, since in many cases people move their head toward relevant information. Such a “position-only” tracking system could be used for studies in which ultimate precision is not needed, for instance to measure how often in a scenario an operator will look up at a large overview display. This function is particularly useful, considering that not all participants can be eye-tracked accurately. In that case, “position-only” tracking offers a fall-back solution, albeit at much lower accuracy. For some studies, it might only be necessary to log the movement of the operators, for example when comparing two different control board layouts to determine how much the operators have to walk between boards to perform their actions.

After the demonstration, INL staff members were invited to a hands-on tutorial during which they went through the same functionality themselves. They completed the setup and calibration process without major problems.

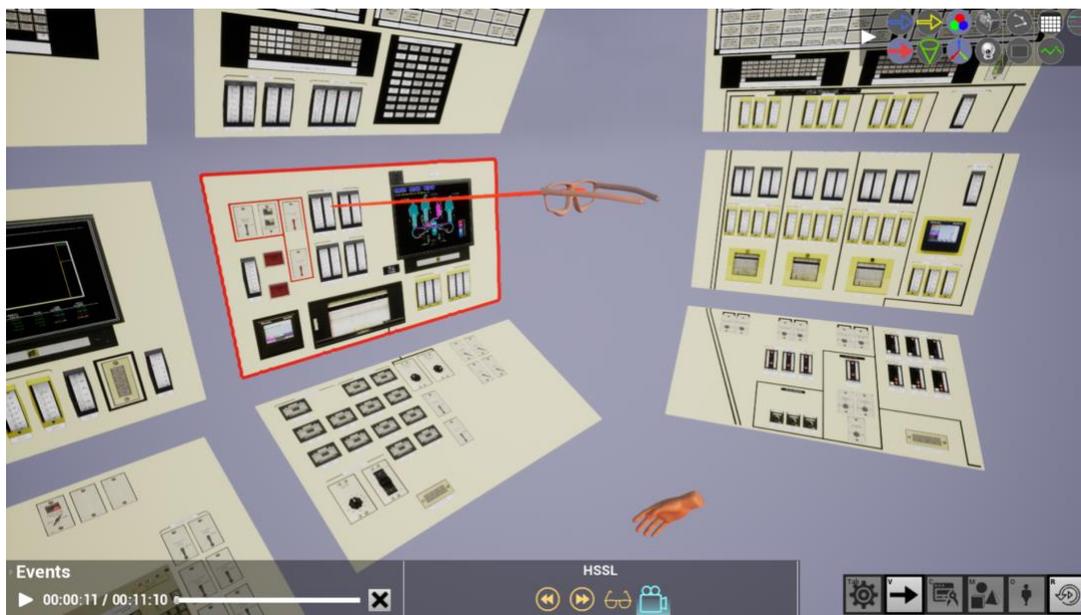


Figure 3. Virtual representation of the eye-tracking glasses based on the actual position of the glasses in the real world. The red vector is the gaze direction and the red rectangle highlights the gaze object. The hand symbol shows hand movement based on a MoCap marker attached to the participant’s hand.

4.2 Synopticon Recording at HSSL

Once setup and calibration were completed, the team conducted a data recording with an INL staff member. The purpose of the recording was to allow INL staff to assess the accuracy of the system, and also to document the different visualisation options in

Synopticon (heat-maps, sequence diagrams, gaze tracks, and pupil diameter charts; see Figures 4 and 5). A video of this test session was handed over to INL at the end of the workshop.



Figure 4. Sequence diagram is shown at the top of the picture. A color overlay shows heat map information to indicate which screens are used the most. Current gaze object highlighted in red.

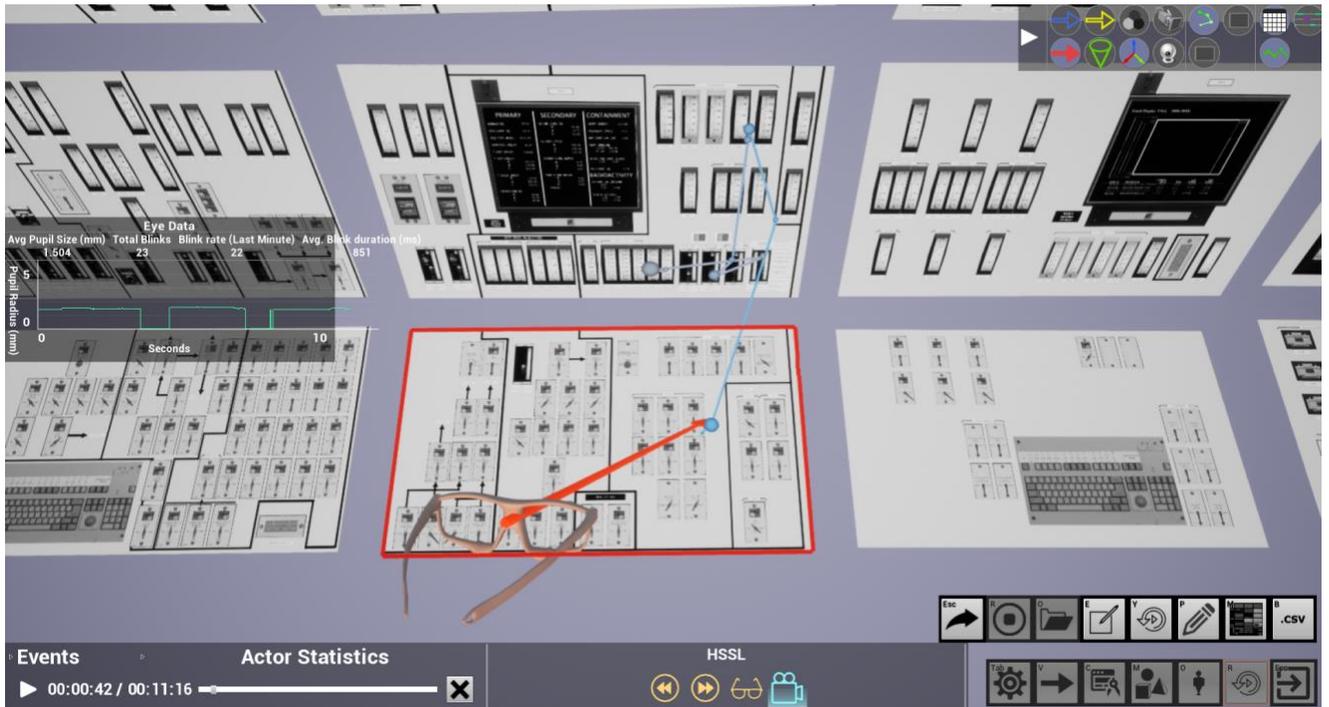


Figure 5. Gaze path (blue) overlaid onto 3D model. Red vector: Gaze direction. Red rectangle: Gaze object. Screen textures can be desaturated, as shown here, to make eye data stand out. Pupil data chart on the left.

4.3 Motion Capture (MoCap) Equipment Recommendation

INL requested advice on MoCap equipment that would allow full automatic gaze mapping at the HSSL. The HSSL is not currently equipped with MoCap cameras. IFE's advice is summarized in this section.

Synopticon supports MoCap cameras by Optitrack (optitrack.com). The team's experience with this equipment has been positive. During the HSSL workshop, the team used seven cameras that were attached with clamps to the rails above the screen bays (Figure 6). To track a MoCap marker, the marker must be visible by at least two cameras. The greater the number of cameras used, the better the system will handle issues like object occlusion. Occlusion is less of a problem when an operator is expected to work in a smaller, pre-defined area, but can be significant when the operator turns too far away from the cameras, or two or more operators occlude each other. To avoid these problems, good all-round camera coverage is desirable, and cameras should be positioned as high up as possible, pointing downward. The rails above the screen bays at the HSSL offer good mounting options to optimize camera coverage.

The team estimates that 8–10 cameras are needed to achieve good coverage of the front half of the HSSL (the area where most of the reactor operator equipment is located). For complete coverage of the HSSL, 16–20 cameras would be needed. Note that the cameras emit infrared light, which interferes with the touch feature of the HSSL screens. That means the cameras and the touch feature cannot be used simultaneously.

IFE uses the Prime13 and Prime13w (wide-angle) range of Optitrack cameras (Figure 7). These cameras are fairly small and can therefore be used for on-site studies at training simulators. These cameras are suitable for HSSL. If portability is not a requirement and cameras are permanently installed at HSSL, the team recommends the Prime41 and Prime17w (wide-angle) range of cameras. These cameras are significantly larger, but provide higher resolution and more range.



Figure 6. Motion capture camera setup during a 2016 demo at HSSL. In the 2018 workshop, cameras were attached to the rails above the screen bays in the background.

All Prime cameras communicate via ethernet and are also powered under the PoE+ standard. This makes for very easy installation, provided that the network switch supports PoE+. IFE uses a UniFi Switch 8-150W (unifi-sdn.ubnt.com). Several other UniFi switches with PoE+ capability would also be suitable.



Figure 7. Prime13w (left) and Prime41 (right) MoCap cameras from optitrack.com. Pictures not to scale.

4.4 Definition of Scope for Custom Eye Tracking Functionality

On the last day of the workshop, the INL and IFE teams discussed which functionality has priority for INL's upcoming studies.

INL conducts user workshops and human performance experiments in its HSSL facility. For these types of studies, operators usually move freely around the simulator, and head-mounted eye tracking glasses (ETGs) are needed. INL also conducts part-task simulator studies and micro-task studies, in which participants are usually seated in front of one or two screens. For such studies, a remote eye tracker (RET), attached below the screen, is used. The INL team determined that RET functionality is the priority for the current work task. More specifically, the following development needs were identified:

RET integration. INL wishes to use just one unified software platform for both RETs and ETGs. Previously, Synopticon only supported ETGs, and IFE provided a separate software for RETs. IFE will now integrate the RET used by INL (REDn by SensoMotoric Instruments) into Synopticon. This integration will make all of Synopticon's real-time visualization and analysis functions available for RET studies. IFE will develop a simplified process for modelling user screens in Synopticon that does not rely on MoCap equipment. The new functionality will enable studies in which some participants are tracked through ETGs and some (for instance, supervisors who are seated in fixed locations) are tracked through RETs. It will also allow the simultaneous use of multiple RETs.

Communication between simulator and Synopticon. For the use of eye tracking with INL's Radwaste simulator, INL requests functionality whereby event data from the simulator can be broadcast and logged in Synopticon. Synopticon already supports markers, but IFE has to develop a messaging protocol to enable communication between the simulator and Synopticon. INL will implement the protocol for the Radwaste simulator (or other simulators, such as an HSSL full-scope simulator). The teams agreed that the message exchange will be conducted over the Web Application Messaging Protocol (WAMP).

Dynamic areas of interest. Areas of interest (AOIs) are particularly important for INL. An example of a dynamic AOI is a screen pop-up or set of pop-ups. INL requests functionality whereby the simulator can send information about dynamic AOIs to Synopticon, and Synopticon will create a corresponding AOI. The simulator has to send another message when the AOI no longer exists (e.g., the pop-up is closed), and Synopticon will remove the AOI. The solution will work both for single AOIs and AOI groups.

AOI editor. IFE will supply a separate tool for creating static AOI maps. Screenshots of process displays can be loaded into the tool, and AOIs can be defined either as rectangles or polygons. When the process display on the simulator changes, the new image ID and AOI map can be sent to Synopticon using the same protocol as for dynamic AOIs.

5 Deliverable 2: Software Package with Custom Eye Tracking Functionality

To deliver the functionalities described in Section 4.4, IFE extended the capabilities of its Synopticon platform by developing several new software modules. These modules

are described in Sections 5.1 (RET support), 5.2 (WAMP message protocol and dynamic AOIs), and 5.3 (AOI editor).

5.1 Remote Eye Tracker (RET) Support

Synopticon now supports the REDn eye tracker from SensoMotoric Instruments. Multiple RETs, as well as mixed RET/ETG studies, are supported. All of Synopticon's gaze visualisation features now also work with RETs.

To use an RET, attach it below the screen that will be eye tracked. Then follow these instructions to establish a data connection between the RET and Synopticon.

Crossbar server:

1. Start the crossbar server (refer to crossbar.io for installation instructions).

RemoteTracker Server:

1. Start SMI iViewRED and connect to eye tracker.
2. Start the RemoteTracker Server using the RemoteEyetrackingServer.bat.
 - a. You can change the address of the crossbar server or the remote eye tracker ID (should be unique).

Synopticon:

1. Create screens and set up their IDs using the "Modeling" menu. The IDs are used in the AOI editor and for updating AOIs and pop-ups using WAMP messages.
2. Create the "Texture" folder inside the "Resources" folder. Copy all required images into the "Texture" folder.
3. Create an actor for the remote tracker using the "Actor" menu.
 - a. Add the RemoteTracker component.
 - i. Select the screen it is attached to from the dropdown menu.
 - ii. Press the "Auto Position" button to position it in the center of the screen.
 - b. Press the "Create" button.

Simulator:

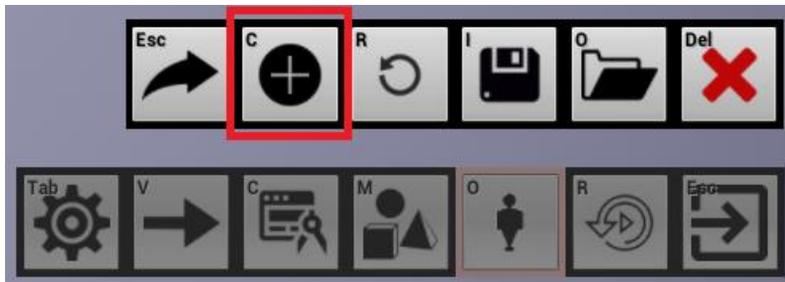
- Connect to the crossbar server.
- Broadcast the desired WAMP messages to start recording, update screens, and stop recording.

Table 1. System setup instructions for RET support in Synopticon.

The "Actor" menu is accessed through this icon:



and the "Create Actor" icon is shown here:



An actor in Synopticon refers to a collection of sensor modules. To make a digital representation of a participant, one would create an actor named, for instance, “ShiftSupervisor.” Next, one would add any sensors (also referred to as components) that provide data about the participant, such as a position-tracking data stream, an eye-tracking glasses data stream, a heart rate data stream, or a hand marker data stream.

In the case of an RET, it is assigned to a screen using the dropdown list “Attached Screen Name,” as shown in Figure 8. The IDs of the RET and of the screen will appear in the dropdown menus “Remote Tracker ID” and “Attached Screen Name,” respectively. For further setup details for RETs, refer to the instruction video described in Section 6.



Figure 8. “Actor” menu in Synopticon, with RET component selected. The component types in the “Add Component” section are, from left to right, RET, ETG, Myo armband (myo.com), Shimmer physiological sensors (shimmersensing.com), position tracking, and hand tracking.

Once these steps are completed, a participant can be seated in front of the RET. The RET is calibrated through the SMI iView REDn software. The stimulus presentation on the participant’s screen is handled by an external application (e.g., INL’s Radwaste

simulator) and not by Synopticon. The screen content in Synopticon can be updated through the process described in Section 5.2.

The participant's eye movements can be observed in real time as shown in Figure 9. All Synopticon gaze and pupil tools are available for RET studies. The sequence diagram will be updated with the appropriate AOI information, based on either dynamic AOIs (see Section 5.2) or AOI maps created by the AOI editor (see Section 5.3).

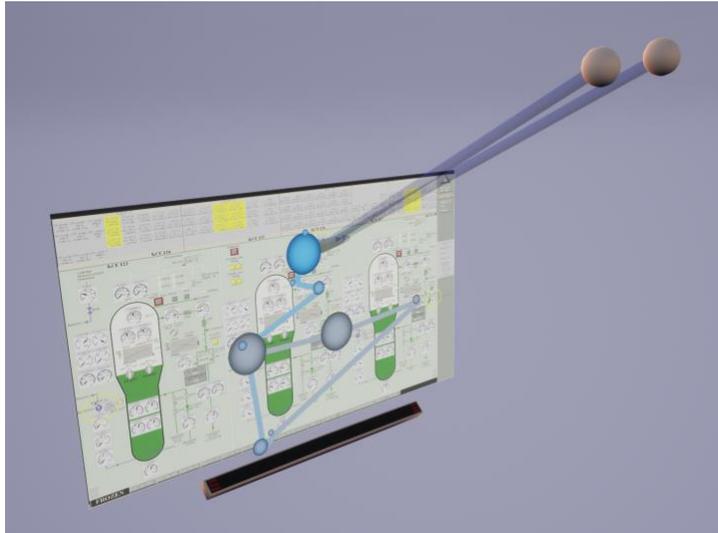


Figure 9. RET support in Synopticon. A representation of the RET can be seen below the screen. Eyeball representations are shown in the top-right corner. The two dark blue vectors are the gaze vectors for each eye. Overlaid on the screen are the gaze path (blue line) and fixations (blue circles).

5.2 WAMP Message Protocol and Dynamic AOIs

IFE developed functionality to receive messages from simulators via the WAMP (Web Application Messaging) protocol. In WAMP, the term “topic” is used to describe separate communication channels for different message types. Applications can subscribe to and broadcast on specific topics instead of listening to all content. INL needs to develop the necessary programming code on the side of the simulator to broadcast messages to Synopticon via WAMP.

The different message types currently supported in Synopticon are listed in Table 2. A WAMP message has to be constructed with two parts: (1) the name of the topic to broadcast to and (2) a list of fields. In JavaScript, a WAMP message with two fields, a string and an integer could look like this: `session.publish('topicname', ['Hello, world!', 2018]);`.

Note that the team did not include further code samples, as the code for sending WAMP messages depends on the programming language used (C#, C++, Python, JavaScript, etc.). WAMP libraries for the different program languages come with their own documentation for message formatting. Refer to the documentation for details.

NewImageAndAOIsTopic: This topic is used to set an image and its areas of interest for a specific screen.

WAMP data format:

- **ImageName:** string (the name of the image in the texture folder)
- **ScenarioLabel:** string (a label that can be freely chosen to tag scenario phases or sections within the experiment run, e.g. “training”, “sc1”, “sc2”. The tags will appear in the exported data file. Can be left empty, but must be included in the message)
- **ScreenID:** string (the unique ID of the screen where the image is displayed)
- **AOIs:** list of AOIs
 - **Name:** string (the name of the area of interest)
 - **BoundingBox:** float x, float y (list of normalized coordinates of points outlining the area of interest. Should be in order)

PopupTopic: This topic is used to control a popup and its areas of interest for a specific screen. There are three actions: Open, Update, and Close.

Open: When the popup appears, the message must contain a list of all AOIs.

Update: When the popup is moving, the AOI list cannot be empty. It contains the new AOI locations.

Close: When the popup is closed, the AOI can be empty.

Note: If a new popup appears without the closing message of the previous one, Synopticon will close the previous one before opening the new one.

WAMP data format:

- **ScenarioLabel:** string (see variable definition under **NewImageAndAOIsTopic**)
- **ScreenID:** string (the name of the screen containing the popup)
- **Action:** string (Open, Update, Close)
- **AOIs:** list of AOIs
 - **Name:** string (the name of the area of interest)
 - **BoundingBox:** float x, float y (list of normalized coordinates of points outlining the AOI. Should be in order)

StartStopTopic: This topic is used to start and stop recordings in the Synopticon system.

WAMP data format:

- **Start:** Boolean (“true” to start the data recording in Synopticon and “false” to stop the data recording in Synopticon)

EventMarkerTopic: This topic is used to send any event information from the simulator to Synopticon. That could include information that a certain scenario phase has started, that an event has occurred (e.g., “Radiation alarms”), or a process variable exists (e.g., “RCS level”). Markers will be logged in Synopticon and displayed in the event list. The event list is available both during recording and when replaying a recording later on.

WAMP data format:

- **Name:** string (the name of the event)
- **Description:** string (description of the event)

- `ScenarioLabel: string` (see variable definition under `NewImageAndAOIsTopic`)

Table 2: WAMP message format for communication between a simulator and Synopticon.

5.3 Static AOI Editor

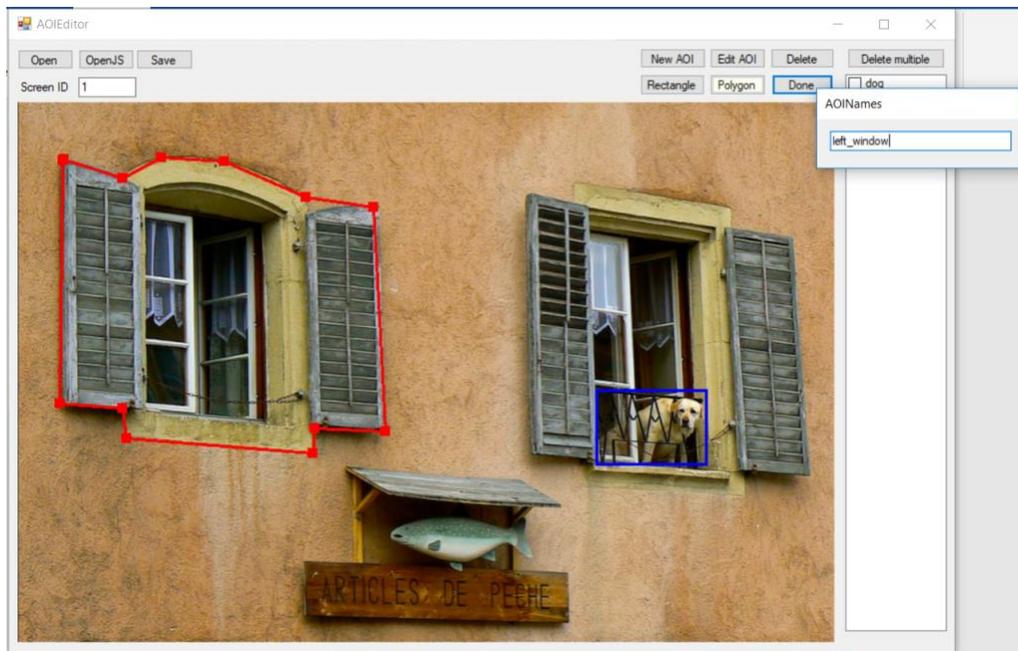
In addition to the dynamic AOIs described above, IFE delivered a stand-alone tool for defining static AOIs (open `AOIEditor.exe` in the `AOIEditor` folder of the software package delivered by IFE).

Instructions for using the AOI editor can be found in Table 3 below.

The editor will generate a json file. This file needs to be copied to a location where it is accessible to the simulator. When a new process display is selected on the simulator, the simulator needs to send a message to Synopticon containing the AOI information (from the json file) and the image name (ID of the new process display). The format of this message is described in the `NewImageAndAOIsTopic` section in Table 2.

Screenshots of the process display must be copied to the “Texture” folder in Synopticon, as described in Table 1. IFE provided an instructional video to show the use of the AOI editor.

1. **Open:** Load the desired image. The editor will load AOIs from the json file with the same name in the same directory path (if any).
2. **OpenJS:** Load the annotated json file. This is useful when the annotated json file is saved in a different directory or with a different name.
3. **Save:** Save the annotated json file. The screen ID must not be empty, as it is important information for WAMP messages.
4. **New AOI:** Create new areas of interest. There are two options for an AOI: rectangle and polygon. Pick either of them for drawing.
 - **Rectangle:** Click and drag to draw the rectangle bounding box. After finishing, enter a new name for the created AOI and hit “Enter.” The new AOI will be added to the right column. Note that AOI names are unique and cannot be duplicated.
 - **Polygon:** Click to pick points on the outline of the polygon. The last point will be linked to the first point automatically. Right-clicking removes the last point of the polygon. After finishing, click the “Done” button. Enter the new name into the popup and hit “Enter.” The new AOI will be added to the list on the right.



5. **Edit AOI:** Check an AOI on the right column and click “Edit AOI.” The “AOI Names” popup will appear, allowing you to enter a new name.
6. **Delete:** Check an AOI on the right and click “Delete.” The checked AOI will be removed.
7. **Delete multiple:** Check multiple AOIs on the right column and click “Delete multiple.” The checked AOIs will be removed.

Table 3. Instructions for use of AOI editor.

6 Deliverable 3: Instructional Videos

Two instructional videos were created:

- The video “Synopticon-RET-Howto” shows the process of adding a remote eye tracker in Synopticon via the “Actor” menu, as described in Section 5.1.
- The video “AOI-Editor-Howto” shows the process of creating static Areas of Interest for screen images, as described in Section 5.2.

Both videos can be found in the “Documentation” folder of the Synopticon software package.

7 Deliverable 4: Project Report

The final deliverable is this report. It describes the project scope and results.

8 Conclusion and Future Possibilities

IFE provided INL with software functionality that should significantly improve the usability and efficiency of eye movement recording in simulator studies. Higher efficiency can lead to better utilization of data and of equipment. The functionality is based on an analysis of INL's study requirements and should therefore closely fit INL's needs. Particularly significant steps forward were the introduction of a general-purpose communication protocol between Synopticon and external sources such as simulators, and the introduction of dynamic areas of interest based in simulator events. The functionality can be used for the Radwaste simulator, for INL's full-scope simulator, or for any other system that can send WAMP messages.

To further support INL's data collection capabilities, several opportunities for future work tasks can be identified:

Support for other eye tracking equipment. While INL's current eye tracking hardware from SensoMotoric Instruments (SMI) is still state of the art, SMI no longer exists as an eye tracking vendor. Therefore in the long term, a change to other eye tracking vendors is inevitable. The current market leader for professional eye tracking glasses is Tobii (tobii.com). In a future work task, IFE could develop Tobii support in Synopticon. In addition to Tobii, there are smaller vendors, such as Pupil Labs (pupil-labs.com) that offer innovative and competitively priced alternatives to Tobii equipment.

Gaze-based interaction. While, in this task, the team developed support for the transmission of event information from the simulator to Synopticon, transmission in the other direction is equally possible. Functionality could be developed whereby Synopticon could send information about the current gaze object to the simulator, and the simulator could either log this data or respond dynamically to where the operator is looking.

Advanced gaze metrics. Synopticon currently supports a number of gaze visualizations such as heat maps (Figure 4) and gaze paths (Figure 5). In the future, advanced functionality could be developed, such as weighting of importance for each AOI (including fixation-to-importance ratio), scan path duration and density, dwell duration per AOI, or definition of expected gaze objects and paths.

Advanced pupil metrics. Synopticon currently provides a simple pupil chart that shows pupil diameter, average blink rate, and average blink duration (Figure 5), while full pupil data are stored for export and further analysis. In the future, advanced preprocessing, visualization, and real-time analysis for pupil data could be developed. This could include the generation of AOI-specific pupil statistics or the detection of cognitive workload changes through changes in pupil diameter.

Recording of simulator screen content in Synopticon. Currently there are two options for displaying the content of simulator screens in Synopticon. One is to send a message with the current screen content ID to Synopticon. Synopticon will then display the corresponding screen texture in the model. This system uses pre-defined screenshots, which means that while the correct process display is shown, the values and equipment positions do not correspond to what the operator sees, and do not

update. The second option is to record the screen content with an external recording device (e.g., screen capture card) or with a screen capture program (e.g., OBS Studio). These screen capture videos can then be imported and synchronized in Synopticon. In the future, functionality could be developed whereby screen content is streamed to Synopticon in real time. In that case, the current screen content would be available both during the recording and during later analysis.

Motion capture with computer vision. The current system for motion capture in Synopticon relies on specialized MoCap cameras and reflective markers. With advances in computer vision and machine learning, it will be possible to extract such data from a normal video feed as provided by a webcam. While such a system is unlikely to match the accuracy of a full MoCap system, it would significantly reduce the equipment setup effort for studies in which ultimate precision is not needed. Such systems could not only track the position of the participants, but provide automatic object recognition, posture analysis, and head movement analysis.

Facial analysis and eye tracking with computer vision. There are currently a number of research groups and commercial enterprises working on facial analysis. Such systems can automatically detect signs of distraction, fatigue, high cognitive workload, emotion, and engagement. Similar technology is being developed to replace the current generation of remote eye trackers with much simpler, webcam-based eye trackers. IFE has explored these approaches and believes that such functionality can be integrated into Synopticon.

Speech analysis with machine learning. IFE is currently investigating how automated speech analysis could be used for simulator studies. Initial applications could include simple speaker detection and speech duration recording. This would provide a simple log of who in the crew spoke and for how long. This would, for instance, support the analysis of teamwork during emergency scenarios. Combined with gaze direction data, this could provide new opportunities for analysis of team interactions and collaboration. The next level of speech analysis could involve the detection of stress or fatigue. Finally, automatic voice recognition technology could be used to transcribe the content of speech.

9 Abbreviations

AOI	area of interest
ETG	eye tracking glasses
HSSL	Human System Simulation Laboratory
IFE	Institutt for Energiteknikk
INL	Idaho National Laboratory
JSON	JavaScript Object Notation
MoCap	Motion Capture

OECD	Organisation for Economic Co-operation and Development
REDn	Remote eye tracker from SensoMotoric Instruments
RET	remote eye tracker
SMI	SensoMotoric Instruments
WAMP	Web Application Messaging Protocol