Intelligent Verification/Validation for XR Based Systems

Research and Innovation Action

Grant agreement no.: 856716

D2.1 – 1st Prototype of the iv4XR Framework, Summary Document

iv4XR – WP2 – D2.1

Version 1.2

Dec. 2020

Project Reference	EU H2020-ICT-2018-3 - 856716
Due Date	31/12/2020
Actual Date	30/12/2020
Document Author/s	Wishnu Prasetya (UU)
Version	1.2
Dissemination level	Public
Status	Final

This project has received funding from the European Union's Horizon 2020 Research and innovation programme under grant agreement No 856716

Document Version Control			
Version	**Date**	**Change Made (and if appropriate reason for change)**	**Initials of Commentator(s) or Author(s)**
1.0	04/12/2020	Initial document structure and contents	WP
1.1	10/12/2020	Full draft	WP
1.2	24/12/2020	Revised draft after QA comments.	WP

Document Quality Control			
Version QA	**Date**	**Comments (and if appropriate reason for change)**	**Initials of QA Person**
1.1	17/12/2020	Review and minor suggestions	FK
1.1	17/12/2020	Confirming suggestions and edits	WP
1.1	18/12/2020	Review	IS
1.1	21/21/2020	Review and comments	RP

Document Authors and Quality Assurance Checks		
Author Initials	**Name of Author**	**Institution**
WP	Wishnu Prasetya	UU
FK	Fitsum Kifetew	FBK
IS	Ian Saunter	GWE
RP	Rui Prada	INESC-ID

TABLE OF CONTENTS

Acronyms and Abbreviations

BDI	Belief Desire Intent
SUT	System under Test

Executive Summary

This deliverable D2.1 presents the first prototype of the iv4XR Framework, which is an agent-based framework designed to enable automated testing of XR systems. This prototype is still incomplete, but it is available and operational. Small and large examples of its usage are also available. The purpose of this document is to give a summary of the Framework's architecture and its main concepts. The framework itself, along with examples of its usage, is provided as (open) software/source code in the project's Github repository[1].

1. Architecture

The iv4XR Framework is an agent-programming framework designed to enable automated testing of XR systems. The general architecture is shown in Figure 1 below. Given an XR system under test (SUT), one or more iv4XR test agents can be deployed to carry out one or more testing tasks, e.g. to verify that certain critical states in the SUT are indeed reachable, and that all interactions in those states lead to correct results. Iv4XR agents are not literally deployed inside the SUT, but instead they are used to control the SUT. The connection between the SUT and the agents are provided by the "Environment" in Figure 1 below. This Environment can be expected to be unique for each SUT, though we enforce a standard interface for it (in the form of a Java Interface). The Framework allows multiple agents to be deployed, which can also work collaboratively through one or more communication nodes (that will allow them to send messages to each other).

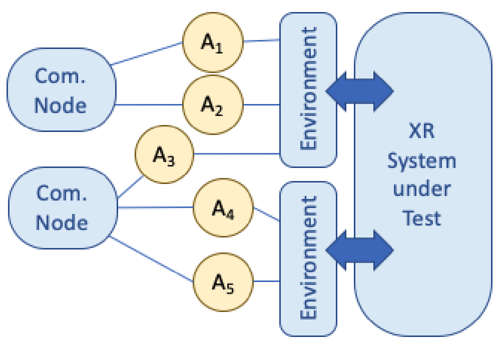

Figure 1: the architecture of a typical usage of iv4XR to test a system.

[1] https://github.com/iv4xr-project

2. Development Status

The first prototype of the iv4XR Framework is publicly available and fully functional. A summary of its features is provided in the next section. Mini examples of how to use agents to test small (non-XR) programs are included in the distribution. A full-scale demonstration of using the Framework to test a 3D computer game (a screenshot is shown in Figure 2 below) is also provided. The source code, documentation, tutorials etc can be found here:

- The **repository** of the framework: https://github.com/iv4xr-project/aplib
- Readme and basic **documentation** are provided in that distribution. They are also linked from the above URL.
- **Tutorials** for deploying simple test agents: also included in the distribution. Direct link: https://github.com/iv4xr-project/aplib/tree/master/docs/iv4xr
- Full-scale **demonstration** of iv4XR on a 3D game called Lab Recruits: https://github.com/iv4xr-project/iv4xrDemo

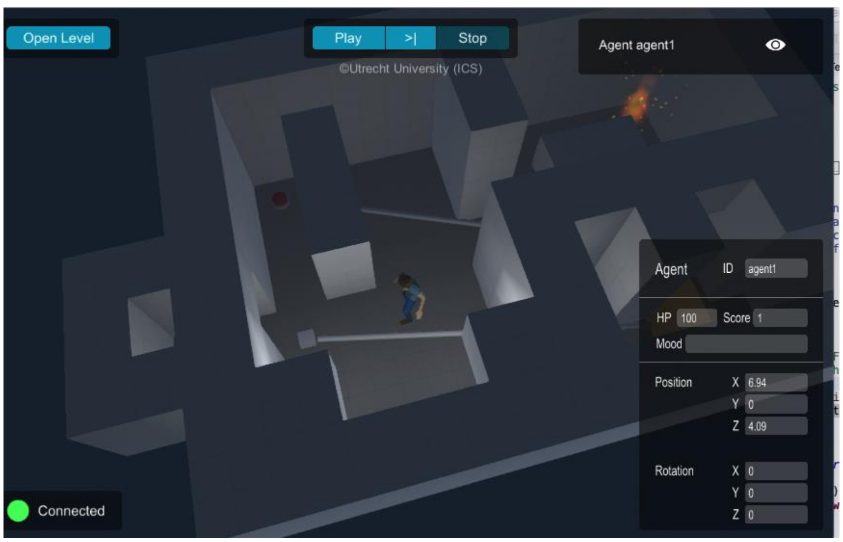

Figure 2: a screenshot of a small game-level in a game called Lab Recruits. We used iv4XR to automate the verification that this level can indeed be completed.

3. Summary of the Prototype's Main Features

- **BDI agents**. Iv4XR agents implement the BDI (Belief-Desire-Intent) concept of agency [see references below: HL+17, MBH15], which is a popular concept for defining intelligent agents. BDI agents maintain a set of human-inspired mental states, such as belief, desire, and intention, and are able to reason over these states when deciding their actions. Although the use of agents to automate testing is not new, previous studies have never fully explored the exploitation of agent-based intelligence. With the current pilots that we internally run (e.g. with the 3D game Lab Recruits mentioned above), the iv4XR's BDI layer was found to be very effective in providing automation. More on the BDI concept of iv4XR is explained in [reference PD+20].

- **Tactical and goal-based agent programming.** While BDI provides a high level concept of how to program intelligent behavior, at the execution level agents are reactive programs that react to incoming events, which can come in rapid succession; some might signal detrimental situations that may need to be handled with higher urgency. Bridging low level reactive programming and the high level BDI concept is non-trivial. To achieve this objective the iv4XR Framework offers the concepts of tactics and goals. Tactics are used to program how agents can choose and prioritize their short term actions and plans, whereas goals (or "desires", in the jargon of BDI) are used to express longer term strategies. More on iv4XR tactical and goal-based programming is explained in [reference PD+20].
- **World Object Model.** The Framework offers a definition of the so-called World Object Model (WOM) as a generic representation for the state of the virtual world of an XR system. The idea is similar to the way that web browsers provide Domain Object Models(DOM) as a generic representation of in-browser documents, hence allowing languages like Javascript to process and manipulate these documents irrespective of the underlying ontology of the documents. The WOM will allow us to define libraries of standard tactics and goals that are reusable across different virtual worlds.
- **Spatial navigation and exploration**. To reach various states in a target XR system, a test agent will inevitably need to navigate the virtual world that the XR system operates upon. As virtual worlds are typically made to simulate physical worlds, limitations that apply to physical worlds often apply in virtual worlds as well (e.g. a virtual character cannot walk through a solid wall, or jump over a 2m fence). The iv4XR framework includes path planning modules to enable a test agent to efficiently travel from place to place in a virtual world. Since the characteristics of the destination place may not be precisely known in advance (e.g. they might be abstractly specified), the modules also include automated exploration capabilities to search for such destinations. More on iv4XR automated spatial navigation and exploration is explained in [reference PV+20].

4. RELEVANT PUBLICATIONS

- The Framework BDI, tactical and goal-based features were presented at a recent International Workshop on Engineering Multi-Agent Systems (EMAS):

  [PD+20] *Tactical Agents for Testing Computer Games*, by Prasetya, Dastani, Prada, et al., in the proceedings of the 8th International Workshop on Engineering Multi-Agent Systems (EMAS) 2020. Pdf: https://emas2020.in.tu-clausthal.de/files/emas/papers-h/EMAS2020_paper_6.pdf

- Automated spatial navigation and exploration capability of the Framework was presented at a recent International Workshop on Automating Test Case Design, Selection, and Evaluation:

  [PV+20] *Navigation and Exploration in 3D-Game Automated Play Testing*, by Prasetya, Volhol, Tanis, et al., in the proceedings of the 11th ACM SIGSOFT International Workshop on Automating TEST Case Design, Selection, and Evaluation, co-located at the ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE), 2020. Pdf: https://arxiv.org/pdf/2009.07015

5. Conclusions

The first prototype of the iv4XR Framework is functional and publicly available, along with a demonstration of its capabilities. Concepts behind the Framework have also been presented in scientific venues and disseminated as published papers. In the 2nd year, we plan to integrate the work from WP3 (functional test agent, model based testing, algorithms to improve test coverage, reinforcement learning) and WP4 (socio emotional test agents) to the iv4XR Framework. We will also collaborate with WP5 to support them in integrating the Framework to the WP5's pilots.

References

[HL+17] Herzig, A., Lorini, E., Perrussel, L., and Xiao, Z. (2017). BDI logics for BDI architectures: old problems, new perspectives. KI-Ku ̈nstliche Intelligenz, 31(1).

[MBH15] Meyer, J.-J., Broersen, J., and Herzig, A. (2015). Handbook of Logics for Knowledge and Belief, chapter BDI Log- ics, pages 453–498. College Publications.