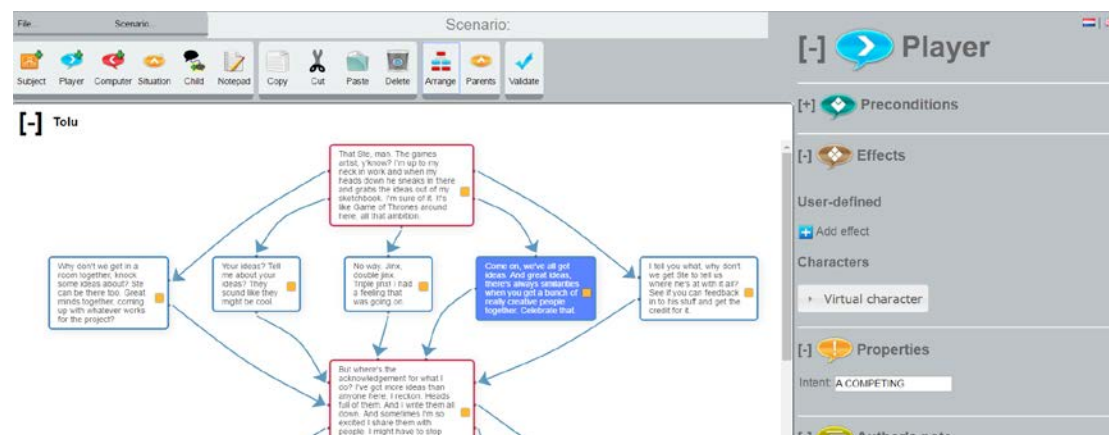


Communication scenario editor

The Communication scenario editor (called scenario editor from now on) is a tool in which a communication expert iteratively develops a communication scenario as a directed acyclic graph of steps. A scenario consists of statements between a player and a virtual character, in which a player chooses between various pre-specified options. The editor balances usability for a non-programming (communications) expert and expressiveness of the constructs in which a scenario can be expressed.



With the latest release of the asset, an instance of this editor can be created catering to the specific needs of a game developer. A game developer can specify virtual character(s), properties (e.g. name of a character) and parameters (e.g. score).

What does the scenario editor do?

Usability of authoring environments often comes at the expense of expressiveness. Our assets tries to combine the two. Besides standard sequence, choice, and conditional options, two unique aspects we offer in our scenarios are interleaving and premature endings. A subject may be marked as interleaving with another subject(s). During the simulation a player is presented with statement choices from within these interleaving subjects without a predetermined order. This is useful when a player should communicate with a virtual character on multiple subjects, but the order in which statements within these subjects are followed is not important. A statement marked as a premature ending enables ending a subject at that statement and thereby skipping the succeeding statements in the sequence within the subject. This is useful when a player makes a 'bad' choice within a subject and conversation of the subject should end.

Other important features of the editor are the possibility to specify parameters for example learning goals for a scenario, emotional effects and the score on (some of) the learning goals of the choice of a player for a particular statement. None of the existing editors on for example the Unity store, such as Dialoguer, or Simple Dialogue Engine Asset, combines all these features of this scenario editor.

When should a game developer use the scenario editor?

One key advantage of using this assets is that the authoring of a dialogue/scenario is decoupled from the programming. At Utrecht University, we have communication teachers from diverse faculties: Psychology, Pharmacy, Medicine, Veterinary science etc. who create complex scenarios in the editor which are later played by their respective

students. Teachers develop new scenarios/dialogues without any direct programming support. Moreover, we have implemented feature requests from communication experts for the past few years.

Use the scenario editor when you want teachers to develop (a lot of different) scenarios for your game.

Technologies

The editor is implemented in JavaScript and runs in a web-browser, which makes it easily accessible to communication experts.

The output of the editor, a dialogue, is stored as an XML file that follows the schema:

<https://github.com/UUDSL/scenario/blob/v4.0.0/scenarioLanguage.xsd>

The output of the editor can be 'parsed' and 'reasoned' by another asset: Step-based competency asset. We have designed the 'Editor' and the 'Assessment' assets to be loosely coupled using a REST architecture. The reasoner asset processes any valid scenario produced according to this schema, using any valid configuration (single VC /multiple VC's; configurable scores, parameters, properties etc)

When can I start using the scenario editor?

The beta version of this component is available in GitHub:

(<https://github.com/UURAGE/ScenarioEditor>).

For more information please contact Johan Jeuring, Utrecht University, the Netherlands <J.T.Jeuring (AT) uu.nl>.