

DynamicRules

ABSTRACT Tangible User Interfaces (TUIs) have recently shown a high potential for enhancing gaming experience, playful learning and for supporting social interaction and collaboration. TUIs are however only one way to build more general mixed reality games. Mixed reality games try to take the best of purely physical games and the best of purely digital games to create an augmented gaming experience. Our intuition is that mixed reality games should go further and aim at creating novel paradigms of interaction that are superior to just the sum of the advantages of each world. The DynamicRules project focuses on rules dynamic reprogramming, a feature that exists intrinsically in physical games and that has disappeared from current mixed reality tabletop games.

INTRODUCTION

Physical Games Properties

In standard classical games, rules can be treated as fully flexible objects, house rules can be defined before a game session, properties of object can be redefined, and on-the-fly winning conditions can be dynamically added during the game by means of the social agreement between players. On the other hand, it is difficult to create new objects, or to redefine the game world. Finally, a drawback of the physical world is the huge amount of pieces and cards that have to be managed, leading to a large amount of mundane tasks.

Digital Games properties

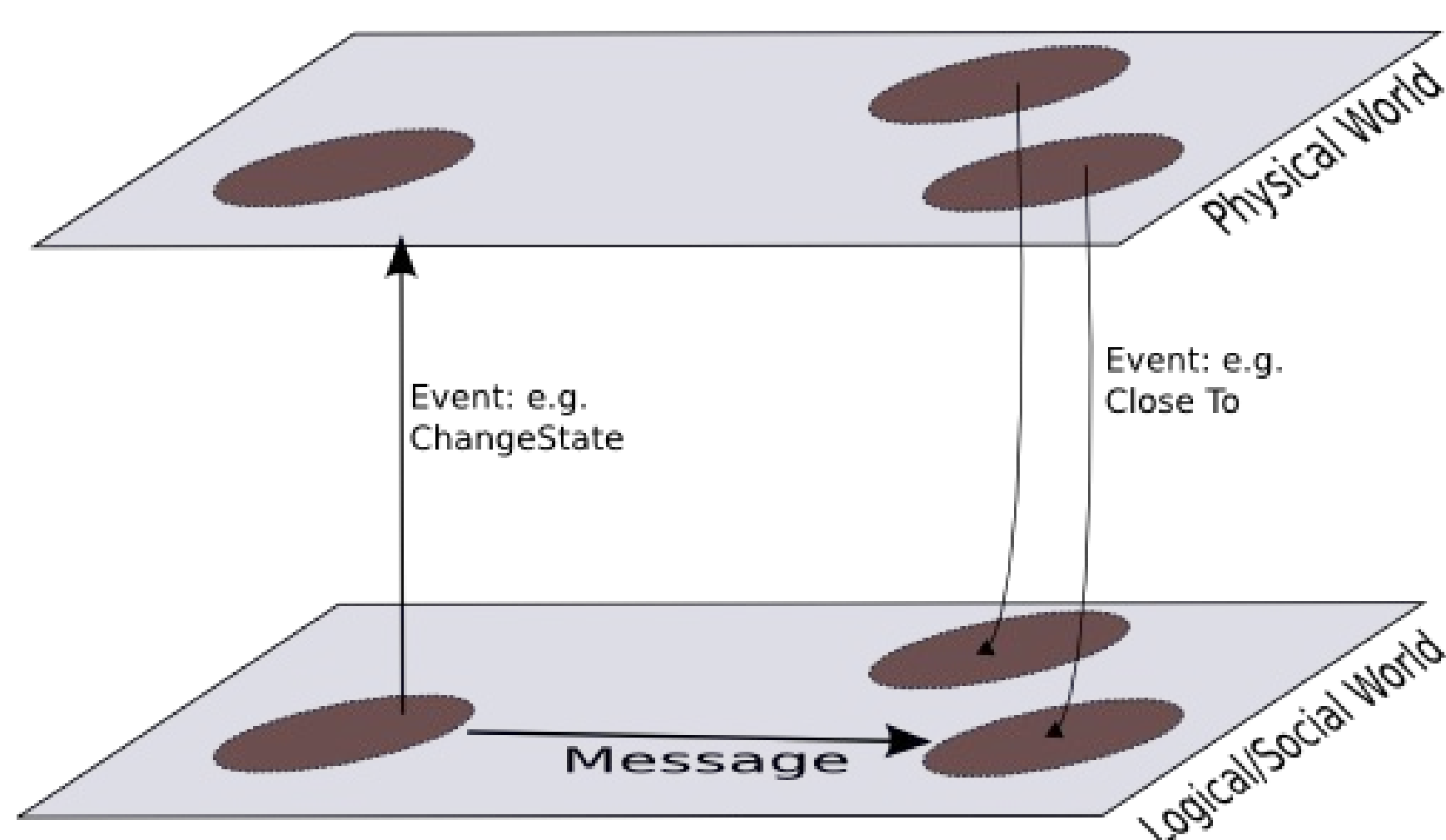
In digital games the degree of flexibility in game rules is fixed by the developer of the game. Most of the time, parameters can be tuned at the beginning of the game session, and they remain fixed until the end. This drawback on the game rules side is compensated by the flexibility of the game board and objects in the digital world; the game board can be more detailed and can dynamically change according to the game evolution.

DynamicRules: Toward a mixed gaming approach

The aim of this project is to develop an architecture for augmented tabletop games. It aims at combining the advantage of computer technology with the social impact of board games. Players should be allowed to experiment game variants by simply redefining the rules before or during a game session, or even invent a new variant defining a new rule without prior knowledge of the underlying programming language but using a high level user interface.

RULE MODEL

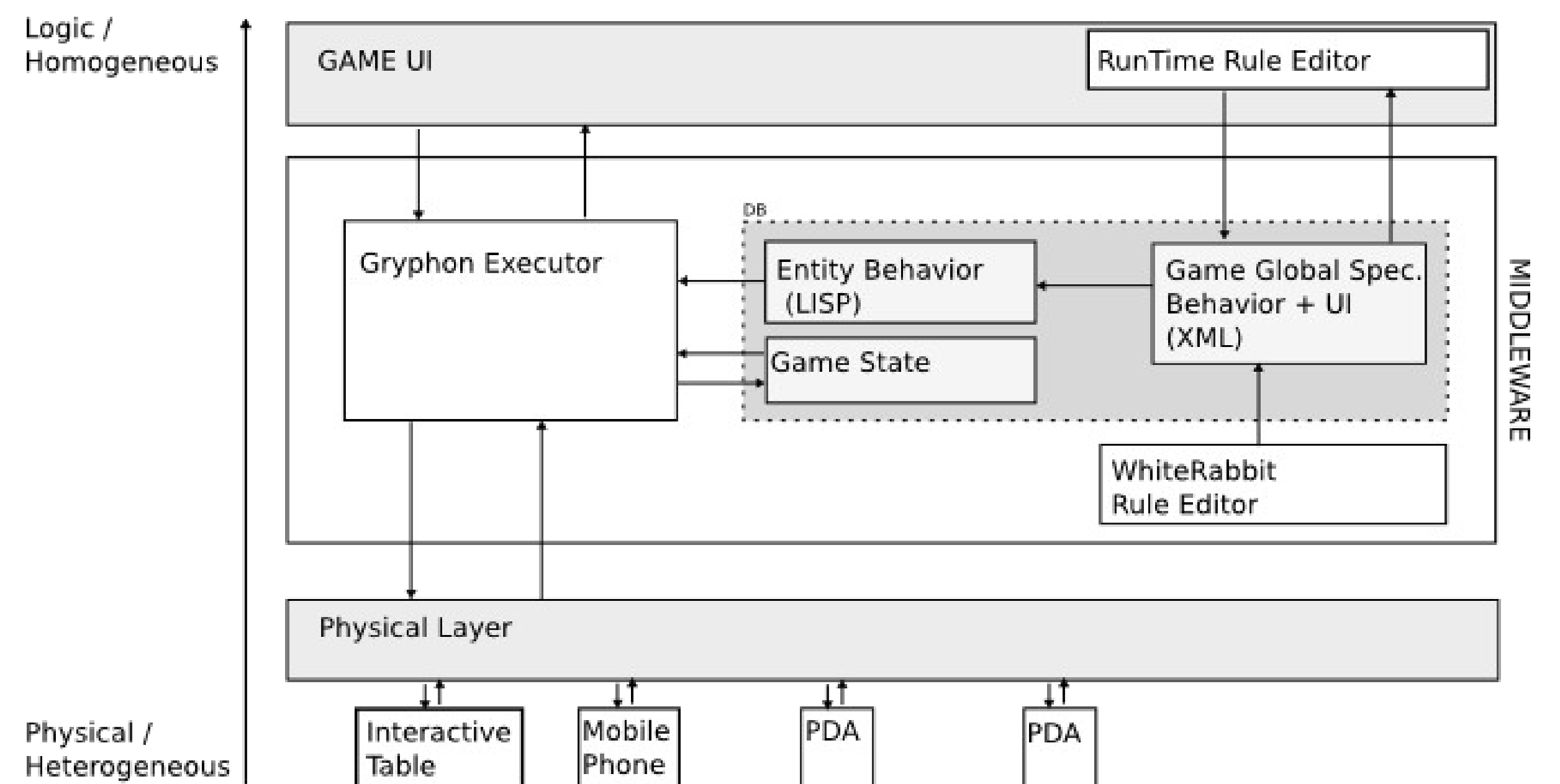
The main idea is to distribute the rules on the game entities. Every entity has only a relative knowledge of the game rules and topology. The game rules are decomposed in three parts: behavior (reaction to game event), laws (allowed and disallowed behaviors) and winning conditions. Each entity is represented in a physical world (where the physical representation of the entity resides) and in a social world (where the behavioral representation of the entity is found).



FRAMEWORK

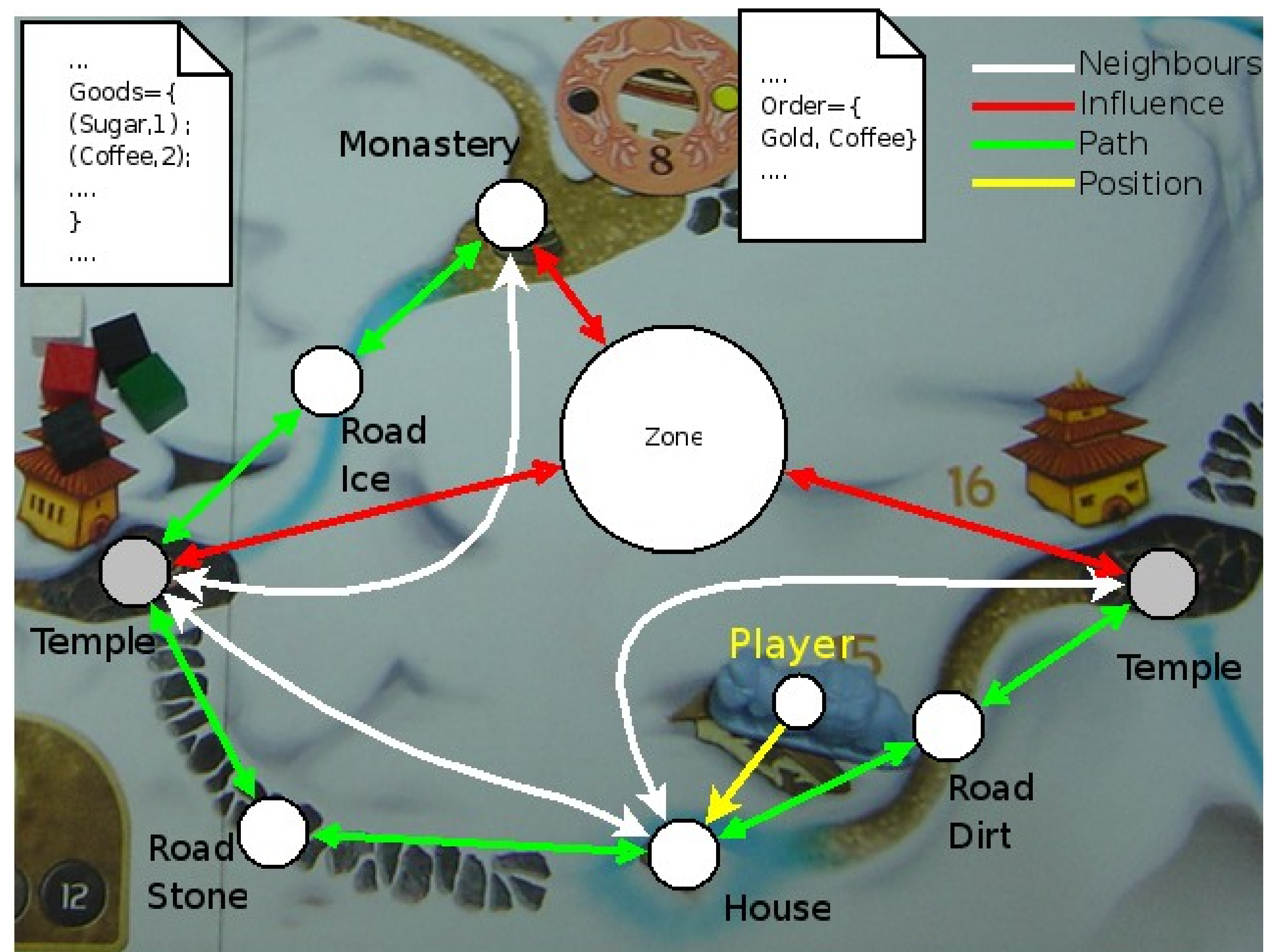
In the context of this project we have defined a framework aiming at:

- managing dynamic rules
- providing physical/social world specification



WhiteRabbit EDITOR

Programming a game is a complex task. In order to facilitate this task we are developing a specific editor, which allows to naturally define relations, rules, behaviors, etc. using a GUI (Graphical User Interface). Our interface has been conceived to assist game creators during the design phase.



CONTACT

Responsible:
Fulvio Frapolli, Department of Informatics,
University of Fribourg, Switzerland
Email: fulvio.frapolli@unifr.ch
WWW: <http://diuf.unifr.ch/people/frapolli/>



Bachelor Projects:
dynaMaze: <http://diuf.unifr.ch/pai/dynamicRules/dynaMaze>
visualEditor: <http://diuf.unifr.ch/pai/dynamicRules/visualEditor>