

Presentation Abstract

A Shell-like Model for General Purpose Programming

JEANINE MILLER ADKISSON^{1,a)} HIDEHIKO MASUHARA^{1,b)} JOHANNES WESTLUND^{1,2,c)}

Presented: January 17, 2019

Shell languages such as bash are designed to integrate with an OS, which mainly involves managing processes with implicit input and output streams. They also attempt to do this in a compact way that could be reasonably typed on a command-line interface. However, existing shell languages are not sufficient to serve as general-purpose languages - values are not observable except in raw streams of bytes, and they lack modern language features such as lexical scope and higher-order functions. By way of a new programming language, *Magritte*, we propose a general-purpose programming language with semantics similar to bash. In this presentation, we discuss the early design of such a system, in which the primary unit of composition, like bash, is processes with asynchronous inputs and outputs, which can be read from or written to at any time, and which can be chained together via a pipe operator. We also explore concurrency semantics for such a language. We explore the implementation and design challenges that come with such a model, and also show that this model is able to easily create and compose concurrent algorithms.

This is the abstract of an unrefereed presentation, and it should not preclude subsequent publication.

¹ Department of Mathematical and Computing Science, Tokyo Institute of Technology, Meguro, Tokyo 152-8552, Japan

² Kungliga Tekniska Högskolan, KTH Royal Institute of Technology, Stockholm, Sweden, 10044

^{a)} jneen@prg.is.titech.ac.jp

^{b)} masuhara@acm.org

^{c)} jwestlun@kth.se