

Introduction to MPL Programming Language

May 15, 2017

MPL(Message Passing Language) is a concurrent, functional and strictly typed language with *message passing* as the concurrency primitive. Concurrent MPL programs comprise of *processes* with *channels* connecting them. Processes communicate by passing messages along the channels. MPL brings the convenience of type safety to the concurrent world by typing the channels. This is achieved by associating every channel with a protocol/coprotocol which determines permissible actions on a channel. Protocol/Coprotocol can be considered as the data types of the concurrent world. MPL also has a sequential side which resembles a strictly typed functional programming language like Haskell along with the additional facilities of defining and using *codata types* and writing disciplined recursive programs using *folds* and *unfolds*.

In the talk, the basic constructs used to develop concurrent programs in MPL will be described with examples that will be run on the MPL's compiler.