# A Process Language Runtime for the .NET platform

My proposal for a Masters project is to create a Process Language Runtime (PLR) that runs on top of the .NET Common Language Runtime (CLR). This is inspired by Microsoft's Dynamic Language Runtime (DLR) which runs on top of the CLR and provides features and services common to dynamic languages such as Python, Ruby and Javascript. Developers of new dynamic languages then have much less implementation work to do to get their languages to run on the .NET platform. Process languages (also known as Process Calculi) such as CCS, CSP and KLAIM have a number of common properties. These include action prefixing, a nil process, parallel composition and non-deterministic choice. The goal of the PLR would be to provide these for process languages and assist in compiling process languages to MSIL which is the bytecode format of the CLR.

The second main objective of the thesis would be to explore integration with other .NET languages. In process calculi it can often be beneficial to be able to use functions, for example for mathematical operations. These types of functions however are not at the core of what a process language is about and so a better choice would be to allow these process languages running on the PLR to call normal .NET functions, either from the standard library or custom functions written in any of the available .NET languages such as C# or F#.

If time permits I would also like to explore further the integration of these languages with existing .NET tools, for example IDE integration and using existing CLR debuggers for debugging process languages.

Einar Egilsson (070630)