# Generating Julia: Finding Commonalities between OO and Procedural Languages

**Brandon Bosman**, Dr. Jacques Carette, Dr. Spencer Smith
Department of Computer and Software, McMaster University

McMaster University | Computing & Software

## Introduction

### The Task

We want to add Julia to the list of programming languages supported by GOOL. Julia is a relatively new language that is primarily used for scientific computing (SC) and mathematical visualizations. Adding Julia to GOOL will align well with GOOL's primary use case – the Drasil project, which focuses on SC.

### What is GOOL?

GOOL is our Generic Object-Oriented Language, which we use to generate code in a variety of Object-Oriented (OO) languages. It consists of a series of Haskell typeclasses which together represent an abstract OO language. At the start of the summer, we had renderers from GOOL to five different OO languages.

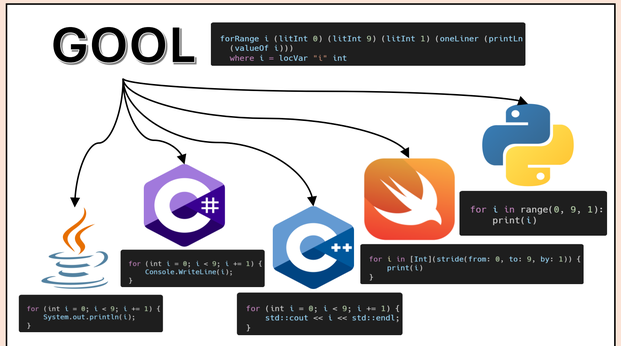


*Figure 1: GOOL code and its renders in different languages.*

## The Problem

### Julia is not OO

Julia is not OO, so not everything we can express in GOOL can be expressed in Julia.

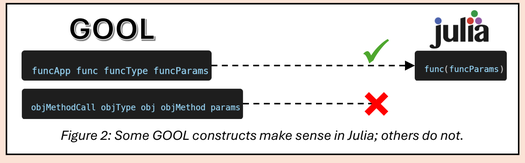


*Figure 2: Some GOOL constructs make sense in Julia; others do not.*

### Solution: A New Generic Language!

Since GOOL is too flexible, we need a new set of typeclasses to express what can be expressed in procedural languages. We created GProc, the GOOL of procedural programs.

## Designing GProc

**Goal:** minimize code duplication between GOOL and GProc.

- Since most of GOOL and most of GProc hold the same features, both can inherit from a 'shared' generic language with typeclasses that both can use.
- Problem: many of GOOL's typeclasses are a mix of features that can be shared and features that are OO-Only. For example, types:

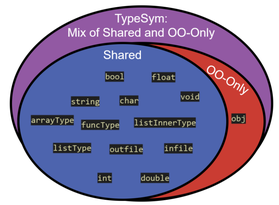


*Figure 3: Most types are can be shared between different types of languages; objects are OO-only.*

- Solution: split up Mixed typeclasses into Shared and OO-Only components.

**Result:**

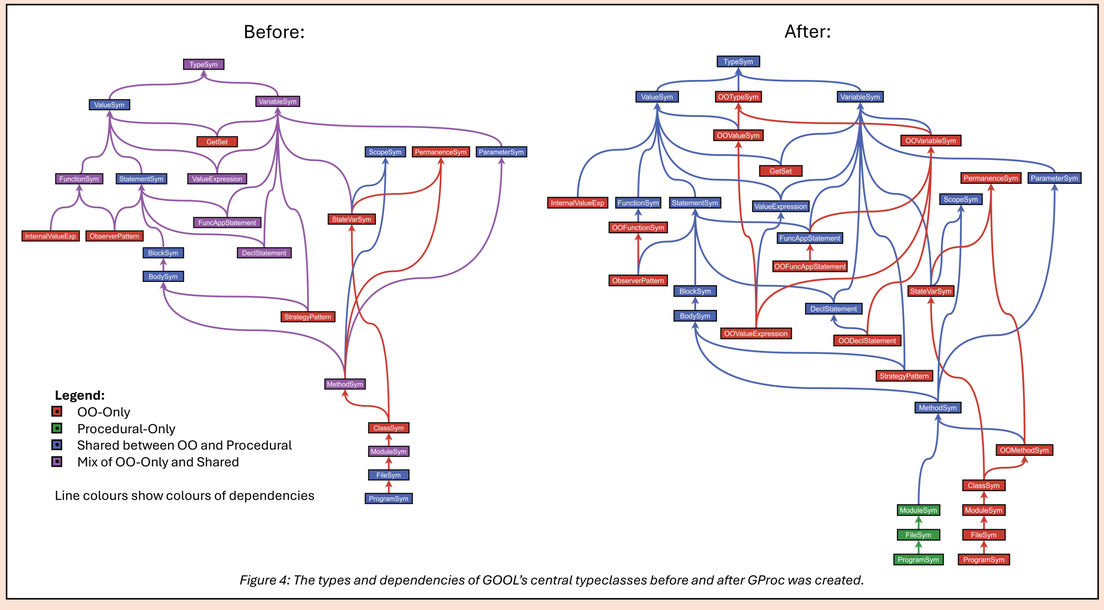


*Figure 4: The types and dependencies of GOOL's central typeclasses before and after GProc was created.*

With these changes in place, we were able to integrate Julia into the Drasil project and generate **545** lines of Julia code.

## Conclusions and Future Work

- OO and procedural languages have a lot in common:
  - 37 typeclasses are shared.
  - 19 typeclasses are OO-Only, making GOOL 66% shared.
  - 3 typeclasses are Procedural-Only, making GProc 93% shared.

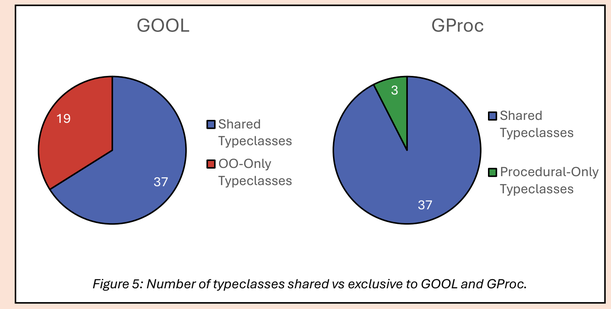


*Figure 5: Number of typeclasses shared vs exclusive to GOOL and GProc.*

- Next steps:
  - Add struct support to GProc. This will allow for better data-bundling techniques, which are currently lacking in GProc.

## I *Guess* I'll Cite My Sources

- Just not yet ;)

NSERC CRSNG