Debugging and expanding Meta Casanova

Louis van der Burg

June 27, 2016

Introduction

- Motive
- Research questions
- Results
- Conclusion
- Questions

Motive

- Video game industry
- Video game development
- Casanova
- Meta Casanova

Main research question

How can the programming language Meta Casanova be improved for the user within the timeframe of the internship?

Sub research questions

- What is a good programming language to the user?
- What is MC and how does it work?
- How can the current syntax be improved to serve the user?
- How can the standard library be improved to serve the user?

What is a good programming language to the user?

Criteria:

- Read- & Writability
- Simplicity
- Definiteness
- Predictability
- Expressiveness
- Implementability
- Efficiency
- Custom Libraries
- Time
- Hackability
- Succinctness
- Redesign
- External Factors

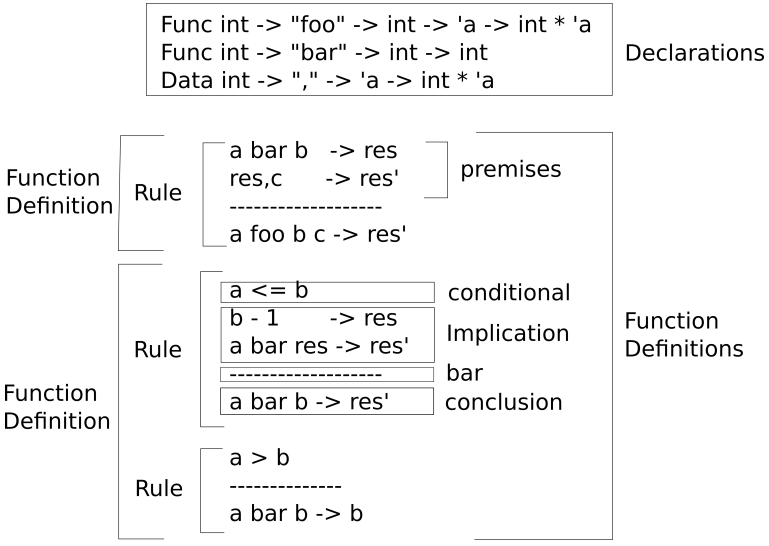
Criteria are used as guidelines

What is MC and how does it work?
Meta Casanova

- Functional
- Declarative
- Pure

What is MC and how does it work?

Basics: part one



What is MC and how does it work?
Basics: part two

- TypeFunc
- TypeAlias
- Module

How can the current syntax be improved to serve the user?
Expanding Modules

- Old

```
TypeFunc "expanding" => Module => Module
expanding M => M{
  ......
}
```

- New

```
TypeFunc "expanding" => Module => Module
expanding M => Module{
  inherit M
  ......
}
```

How can the current syntax be improved to serve the user?
Syntax additions

- Priority

```
Func "bar" -> Int  #> 12
TypeFunc "foo" => Float => 'a => 'b  #> 9 R
```

- .NET

```
import System

Func "dotNetTest" -> String
dotNetTest -> DateTime.Now.ToString()
```

How can the standard library be improved to serve the user?

- Prelude
- Number
- Record
- Monads

How can the standard library be improved to serve the user?

Prelude

```
TypeAlias "Then" => Type
Data "then" -> Then

TypeAlias "Else" => Type
Data "else" -> Else

Func "if" -> Boolean^System -> Then -> 'a -> Else -> 'a -> '
    ↪ a
if True^builtin then f else g -> f
if False^builtin then f else g -> g
```

How can the standard library be improved to serve the user?
Number & Record

- Number
  - Generic
- Record
  - Compile time

How can the standard library be improved to serve the user?
Monads

- Monad
- Monad transformers
- TryableMonad

How can the standard library be improved to serve the user?
Implemented monads & monad transformers

- List
- Either
- Result
- State
- Id

Advantages of Meta Casanova

- Predictability
- Definiteness
- Succinctness
- Safety
- Choice of execution time

Conclusion

- Sub research questions
  - What is a good programming language to the user?
  - What is MC and how does it work?
  - How can the current syntax be improved to serve the user?
  - How can the standard library be improved to serve the user?
- Main research question
  - How can the programming language Meta Casanova be improved for the user within the timeframe of the internship?
- Video game industry

Conclusion

Conclusion

Conclusion

Questions

Any questions?

Recommendations

- Monads
  - IO monad
  - Coroutine monad
- The compiler

Monads in other languages

Language	Monad	Number of lines
Python	State	75
Haskell	State transformer	232
Meta Casanova	State transformer	23