

# PROP – Assignment 3

Ulf Noring & David Rogers

December 2015

## 1 Introduction

Functions in Clojure evaluate their arguments before they are passed to the function, while arguments are passed to macros without being evaluated. This allows macros to accept undefined symbols, while functions can only accept defined symbols. (Rich Hickey 2014)

When Clojure tries to resolve a symbol, several checks are done before throwing an error. Clojure checks if the symbol is considered a special form (i.e. built-in primitives serving as core functions in Clojure), if the symbol matches a Java class name in the current namespace, if the symbol matches something in the local namespace and finally if the symbol matches something in the global namespace. If none of these checks are true, an error is thrown. (Rich Hickey 2014)

## 2 Safe Macro

In the safe macro, it should be possible to make the call

```
(safe [s (FileReader. (File. "file.txt"))] (.read s))
```

If the safe operator was implemented as a function, Clojure would try to resolve `s` by looking for it in the local namespace as described above. If it would not be defined, an error would be thrown. However, when implemented as a macro, there is no need to have predefined `s` in order for the above call to work, as `s` is not resolved before it is passed to the macro.

## 3 SQL-like Macro

In the SQL-like macro, it should be possible to make the call

```
(select [columns] from #{table} where [:column op value] orderby :column)
```

If the select operator was implemented as a function, Clojure would try to evaluate the symbols (`from`, `where`, `orderby`) before the arguments are passed to the function. As these symbols are most likely undefined, an error would be thrown. If implemented as a macro, the macro would be able to parse these arguments as the keywords they are intended to be, without Clojure attempting to evaluate them.

It may be possible to implement this macro as a function if the *from*, *where*, *orderby* symbols were implemented as keywords (`:from`, `:where`, and `:orderby`) instead of symbols.

## References

Rich Hickey (2014). *Clojure - evaluation*. [Electronic] Retrived on 2015-12-11. URL: <http://clojure.org/evaluation>.