

# From GF RGL to Universal Dependencies

Prasanth Kolachina

Chalmers University

October 20<sup>th</sup>, 2015

# Universal Dependencies

- A community-based effort to uniformly annotate multilingual corpora with syntactic structure i.e. dependency parse tree
- Their vision is closely related to philosophy of GF
  - a shared representation for many languages with extensions for language specificities
- That is GF's goal during RGL design
  - we have an “abstract syntax” shared across languages (30 currently)
  - extension modules for language specific constructions
- How mutually beneficial are these two?

- How to construct dependency trees from RGL trees?
- How to construct *universal* dependency trees from RGL?
- What can UD<sub>s</sub> get from GF ?
  - Bootstrapping parse trees for new languages in the RGL
  - Consistency checking in multilingual annotations
- What can GF get from UD<sub>s</sub> ?
  - Dependency annotated corpora allow for data-driven approaches
  - The probabilistic model can be improved using the annotated corpora

- ① How to construct *universal* dependency trees from RGL?
- ② Comparison of the abstract syntax in RGL and UDs
  - We find a few point of difference between the two approaches
  - But, they are mostly similar, and can be uniformly converted from one to another
- ③ Insights into constructing an “enhanced” dependency representation

# Dependency-decorated abstract syntax tree

