Minutes meeting 2016-04-25

Agenda

- Statusupdate
- Discuss Hendrik's feedback
- Other points

Statusupdate

- We are on the right track, but we feel as if we are a bit behind due to another essay for the IT & Values course.
- It's a bit difficult to find up to date and/or complete documentation for Spoofax and its components. However, we do feel as if we have a good global overview.
  - spoofax.readthedocs.org
  - Gabriel has a short Java program that uses all the stages of the Spoofax API needed to go from source code to execution, given an existing language. Contact Gabriel for pointers and more information.
  - A compiler/IDE consists of three pieces: syntax, analysis and interpreter/compiler. Which pieces of Spoofax (SDF, NABL, TS and DynSem) does our REPL touch and how does it influence these pieces?
  - We should make a model of this and perhaps show this in the research report.

Discuss Hendrik's feedback

- High-level requirements have not been filled in yet.
  - We agreed (see minutes 2016-04-19) to make a list of requirements following our research that we are still conducting.
  - However, we could define global requirements to get a gist of what we want (*"What should we be able to do at the end?"*)
  - It shouldn't be final (write down global requirements now, refine it later). The purpose is that the faster we have an outline, the faster we will see what is possible and what is not.
- Verifying the results: possible to do this once we have requirements. How do you suggest we verify?
  - We did not have any concrete ideas written down, just something about unit tests and that's it. It was more meant as a stimulus to think about it: do we use unit tests (specify a certain coverage percentage?), do we use code reviews, et cetera? That is for code quality. For product verification, we could for example have one or two people from the research group try to develop a REPL.
    So: we need to specify how we achieve code quality and what a viable product is.
- What is meant with his comments on the research questions?
  - The topics of the research questions is oke, however the parts about Spoofax could be made more explicit. The first question (*"What is Spoofax?"*) should have a subquestion *"How does a*

  user create a language in Spoofax", for example.
  - Split up research questions into what Spoofax offers now and what we need to develop to meet the requirements for a REPL.
- Repository isn't private so we need to send a link to it.
  - Tag Hendrik in issues/PRs if we need him.
  - Be proactive in asking questions about Spoofax's documentation.

Other points

- How were dynamic semantics specified using Stratego before DynSem was around? What was the AST transformed to?
  - Several options; Java bytecode or a Stratego interpreter/rewriter.
- To what extent should we know DynSem's notation? Can we find documentation somewhere?
  - Again, think about the three pieces of a compiler/interpreter and which pieces our REPL touches. Get a high level overview and work (topdown) from there.
  - Don't get lost in details of specific components.
  - Send Vlad a PM on Slack to request more information or perhaps make an appointments.
- Do not decide our solution/product yet, but do give global guidelines as to how we envision it.
- Create diagrams/images for ourselves (and perhaps the report) so assist in getting an understanding of how certain components cooperate et cetera. Do not lose ourselves in UML details or whatever.
- Literate programming: focus on the interactive parts it defines.