SLEBoK Use Cases

Language Designer

- ⚠️ Disambiguation in language definitions (PM)
  - Research? Or does best practice exist?
  - 👍 Scannerless parsers ...
  - 👍 Reengineering

- ⚠️ Coming up with the right language definition (BC)
  - To ensure that it supports the right abstractions for application building
  - 👍 Language definition process (Jordi Cabot's Colaboro)

- ⚠️ To ensure the right services to manipulate the right abstractions (BC)
  - E.g., You have a nice DSL; you want to step through the domain abstractions, not the generated code
  - 👍 Moldable debugger, inspector etc

- ⚠️ Help domain experts to encode their domain
  - E.g., give a financial expert a high level DSL
  - 👍 DSL design

- ⚠️ Language / tool / application instance co-evolution
  - 👍 Co-evolution research

- ⚠️ Defining software quality metrics for language definitions
  - Ease-of-use, expressiveness, abstraction
  - ❓ Language design guidelines and heuristics

Tool designer

- ⚠️ How to architect a language tool suite?
  - 👍 Use a meta-language to define the language and support the tools
  - 👍 Software generation
  - 👍 Plugin architectures etc
    - 😐 Not specific to SLE, though
  - 👍 LSP — Language Server Protocol to enable communication between IDE and services

- ⚠️ Is my type checker sound?
  - 👍 Use Coq
    - 😐 Not specific to SLE

- ⚠️ Are my compiler and interpreter consistent?
  - 👍😐 Randomized testing
  - 👍😐 Bisimulation

- ⚠️ Does my compiler respect the language definition?
  - 👍 Verified compilers
  - 👍 Test generation

- ⚠️ Building a performant text editor
  - 👍 Data structures, standard architecture

Forward engineer (joe programmer)

- ⚠️ Develop code at a suitably high level of abstraction
  - 👍 DSL technology

Domain expert

- ⚠️ Express domain knowledge

Reverse engineer

- ⚠️ Dealing with large legacy systems (TS)
  - 👍 Techniques to parse, model and visualize old languages
    - Rascal, Moose ...

- ⚠️ I want to understand software written in a legacy language (no tools)
  - 👍 Reverse engineering practices

Re-engineer

- ⚠️ Migrate from one language to another
- ⚠️ Clean up code

http://langserver.org/