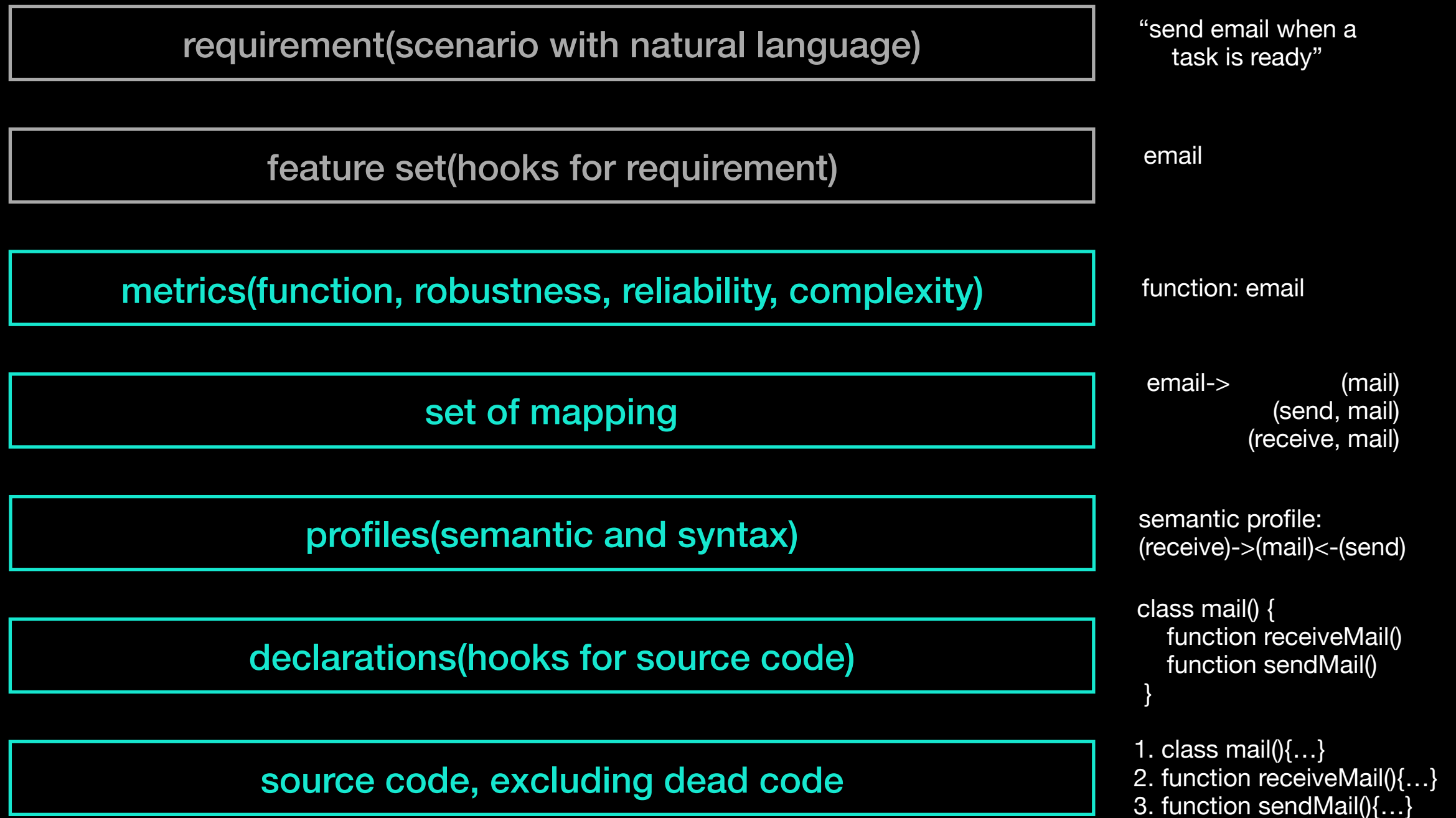


Toward Knowledge Systems in Software Engineering

Knowledge of programming

- Objectives
 - to formulate ideas(abstract)
 - to modify the source code according to the idea
 - to verify the change meeting the idea
 - to make sure the change does not break other parts

Whole Picture

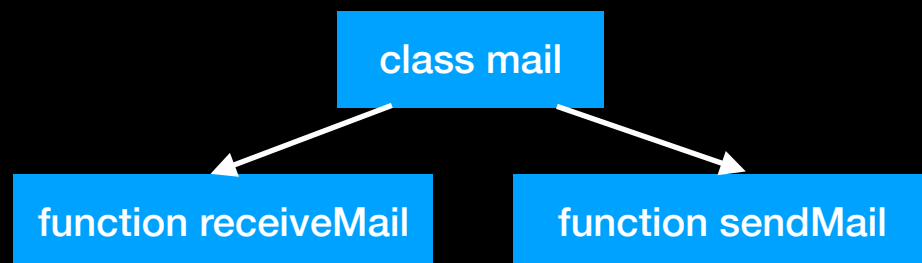


syntax and semantic profile

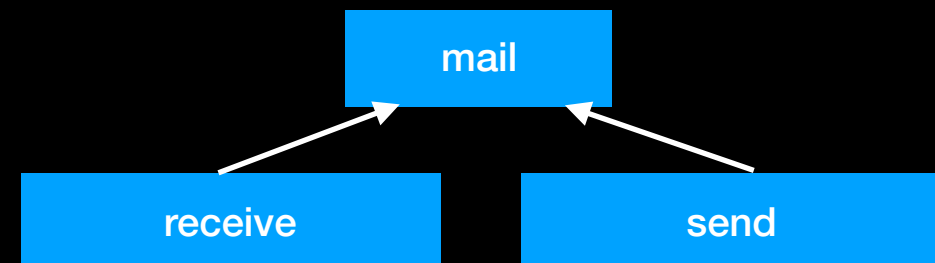
example:

```
class mail() {  
  function receiveMail() {  
    ...  
  }  
  function sendMail() {  
    ...  
  }  
}
```

for this code, syntax profile is:



for this code, semantic profile is:



Metrics and Implementing Profiles

- Metrics
 - function, robustness, reliability, complexity
- Profiles
 - source code, developer, bug
- Relationships between profiles and code section
 - source code -> semantic profile -> declaration -> code section
 - source code -> syntax profile -> declaration -> code section
 - developer -> commit history -> code piece -> semantic profile -> declaration -> code section
 - developer -> commit history -> code piece -> syntax profile -> declaration -> code section
 - bug -> code piece -> semantic profile -> declaration -> code section
 - bug -> natural language description -> syntax profile -> declaration -> code section

Metrics and Implementing Profiles

- Relationships between profiles and metrics
 - source code -> complexity and function
 - developer -> reliability
 - bug -> robustness

Previous

- Declaration -> code section
- Developer -> commit history -> code piece -> declaration

TODO (zimeng)

- Mapping: function \leftrightarrow semantic profile
- Discussion:
 - defining representations of semantic profiles (set, tree, graph)
 - implementing the mappings

TODO (zhenghui)

- Mapping: function \leftrightarrow syntax profile
- Discussion:
 - defining representations of syntax profiles (graph)
 - the representation only considers high-level modules, like function, class...
 - considering how to work with zimeng's part, like introducing the distance among modules
 - implementing the mappings