

Collaborative Modeling across language workbenches

- a case based on JetBrains MPS and Eclipse Spoofax

Authors: Samuel Noah Voogd*, Kousar Aslam*, Louis van Gool**, Bart Theelen**, Ivano Malavolta*

* Vrije Universiteit Amsterdam, The Netherlands

** Canon Production Printing, The Netherlands

Problem



Problem



- Language workbenches → design and define DSLs



Problem



- Language workbenches → design and define DSLs
- Model engineering → *complex* tasks



Problem



- Language workbenches → design and define DSLs
- Model engineering → *complex* task
 - ↳ **real-time collaboration**



Problem





- Language workbenches → design and define DSLs
- Model engineering → *complex* task
 - ↳ real-time collaboration
- Current limitation: real-time collaboration *single-platform support*

What if model engineers use different language workbenches?

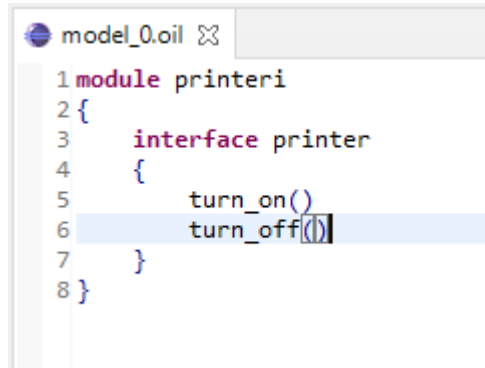


Proposed Solution: Parsafix

- **Real-time collaboration**
 - different language workbenches
 - same DSL
- Eclipse Spoofax 
- JetBrains MPS 
- **OIL** (Open Interaction Language)

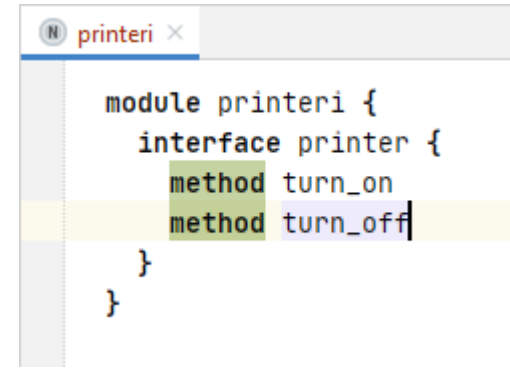
Open source: <https://github.com/blended-modeling/parsafix>

Proposed Solution: Parsafix



```
model_0.oil
1 module printeri
2 {
3     interface printer
4     {
5         turn_on()
6         turn_off()
7     }
8 }
```

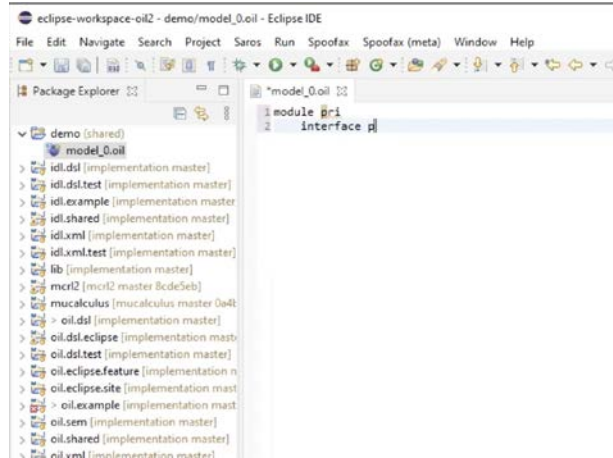
(a) Model inside of Eclipse



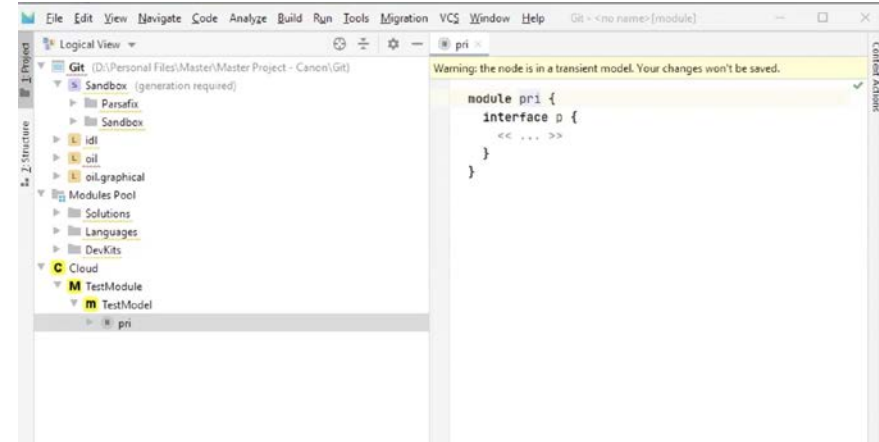
```
printer
module printeri {
    interface printer {
        method turn_on
        method turn_off
    }
}
```

(b) Model inside of MPS

Proposed Solution: Parsafix



(a) Model inside of Eclipse



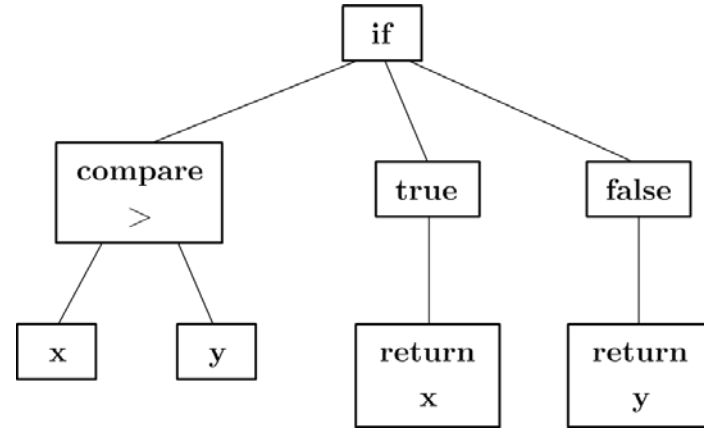
(b) Model inside of MPS

Abstract Syntax Trees

- Representation of the *abstract syntax structure* of source code in the form of a tree graph.

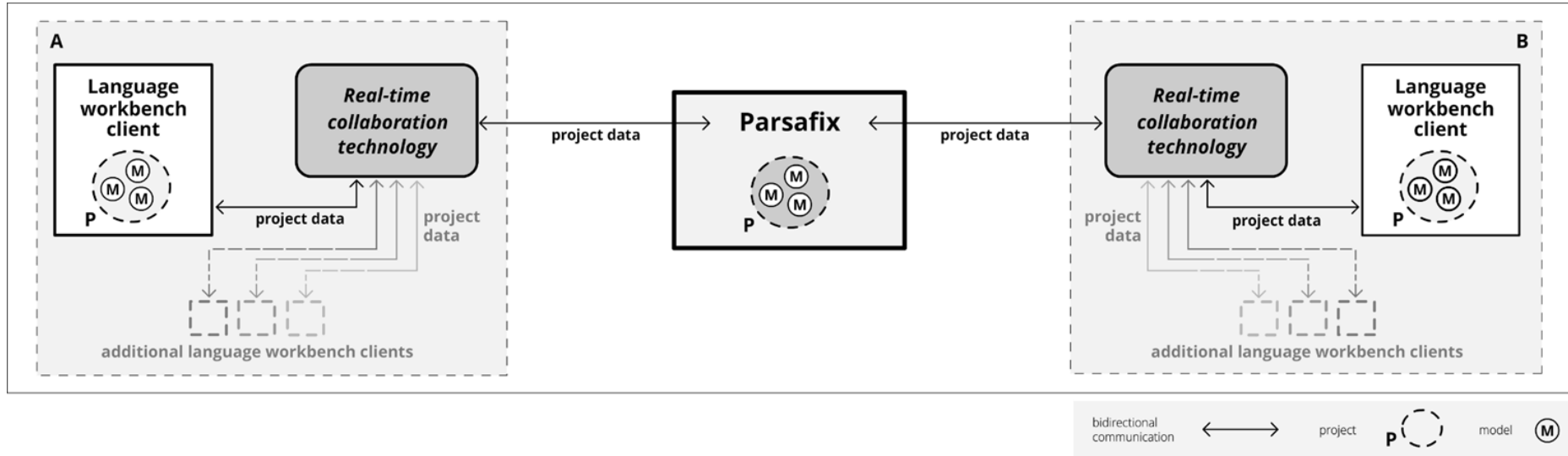
```
if (x > y)
{
    return x
}
else
{
    return y
}
```

(a) Source code of an if-statement

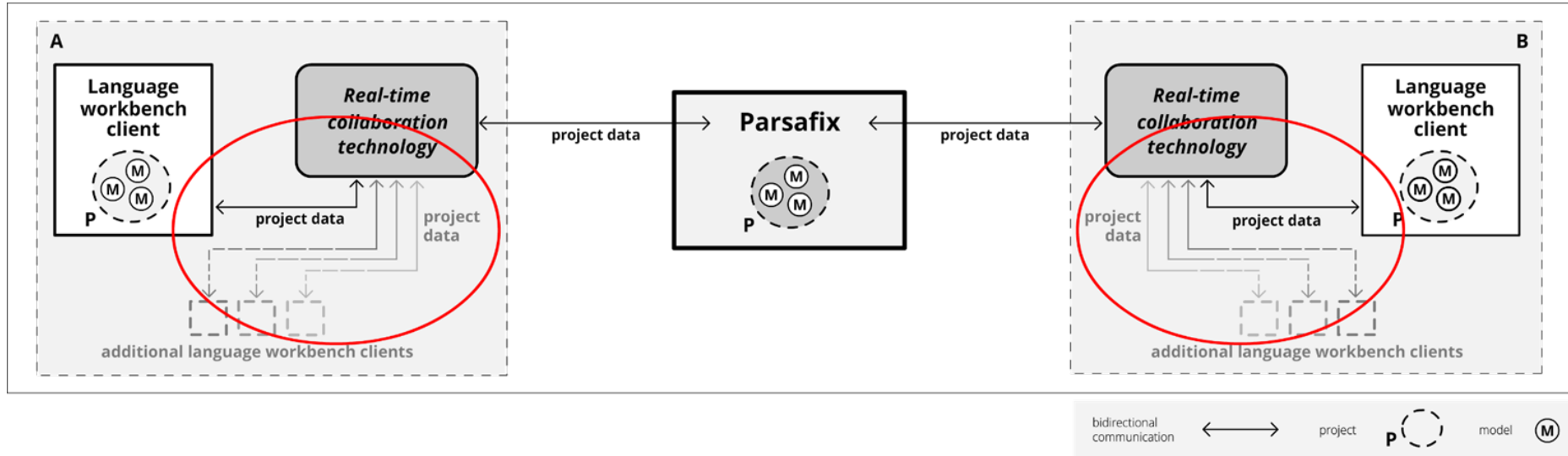


(b) AST of an if-statement

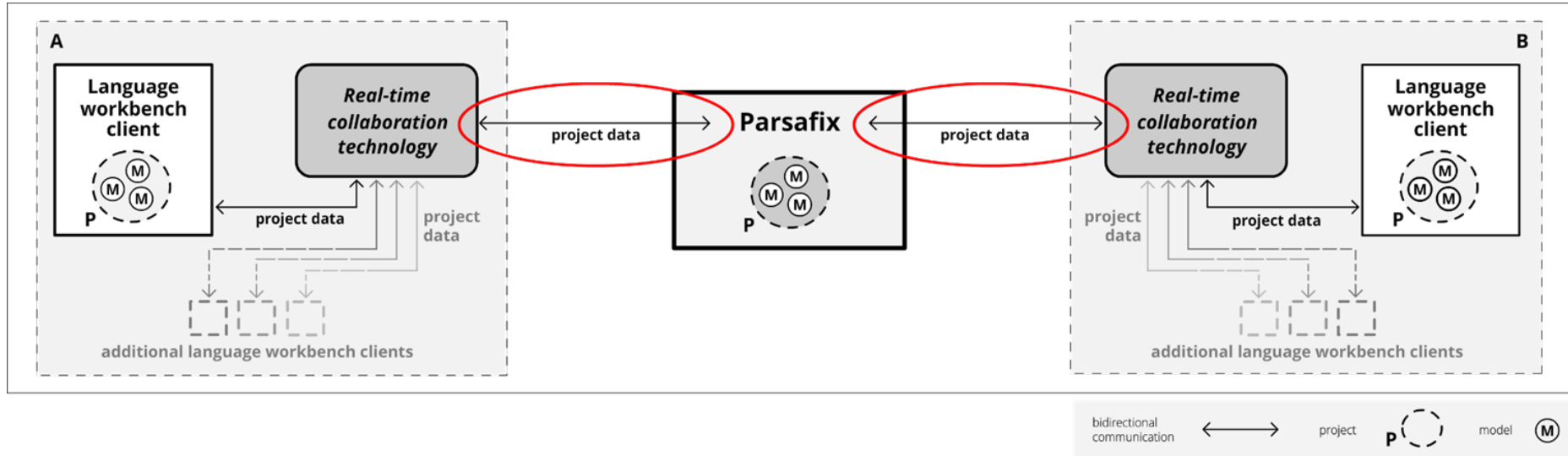
Collaboration Architecture



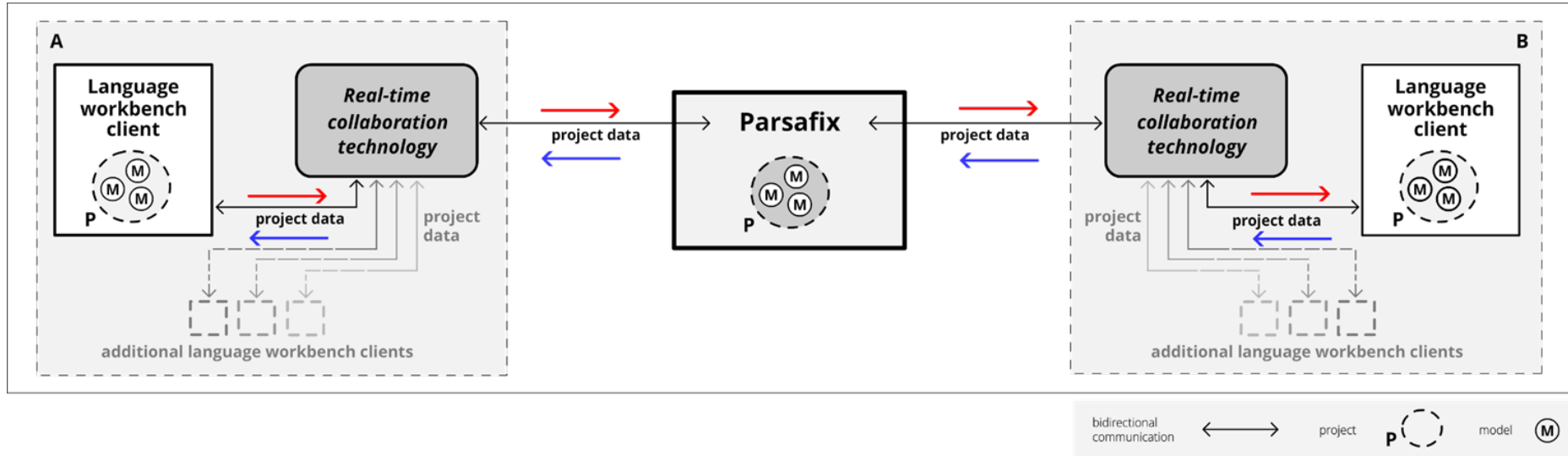
Collaboration Architecture



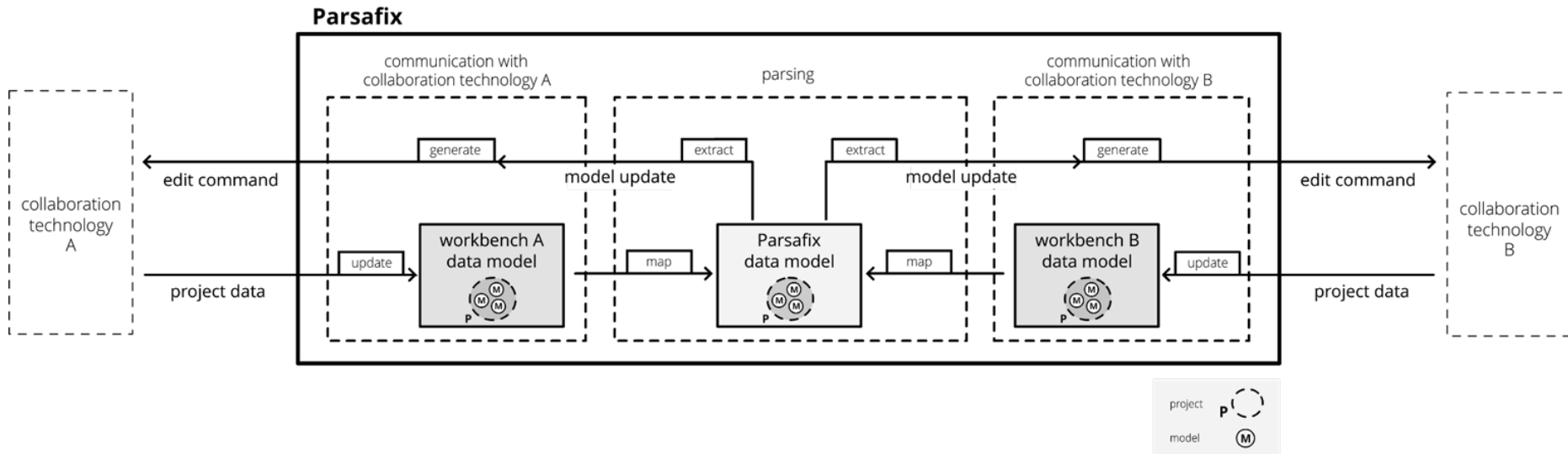
Collaboration Architecture



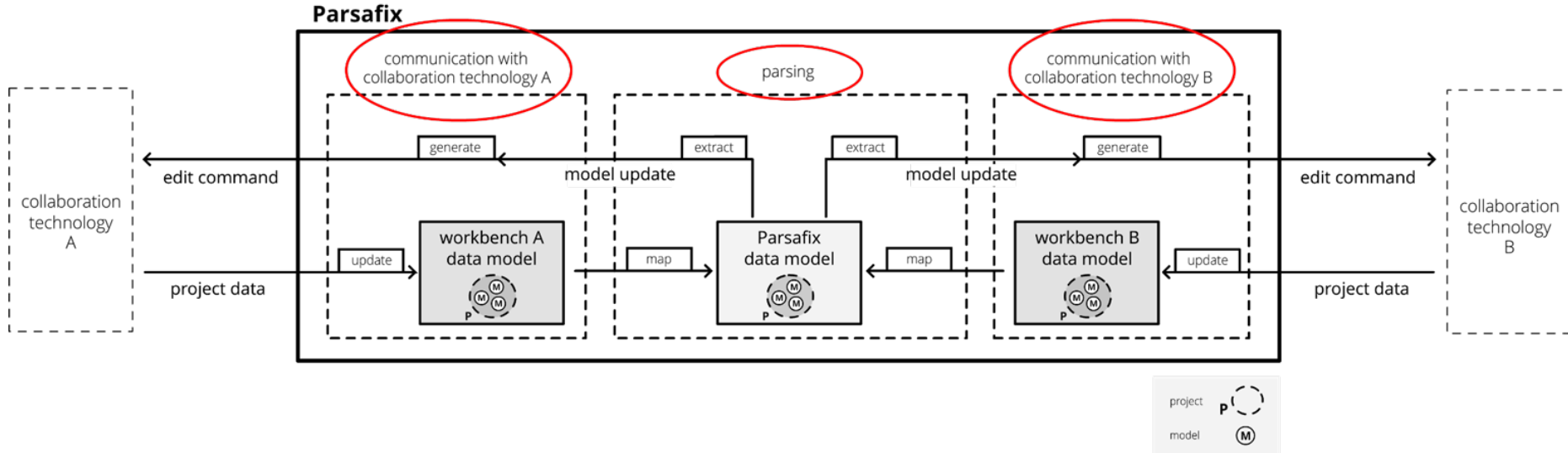
Collaboration Architecture



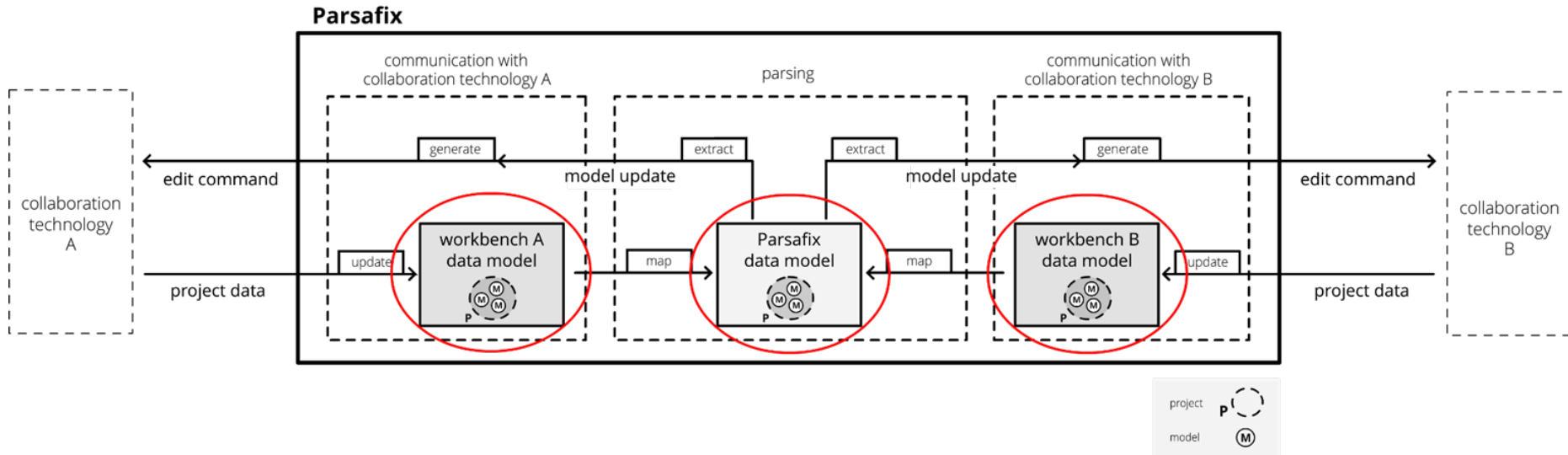
Internal Structure



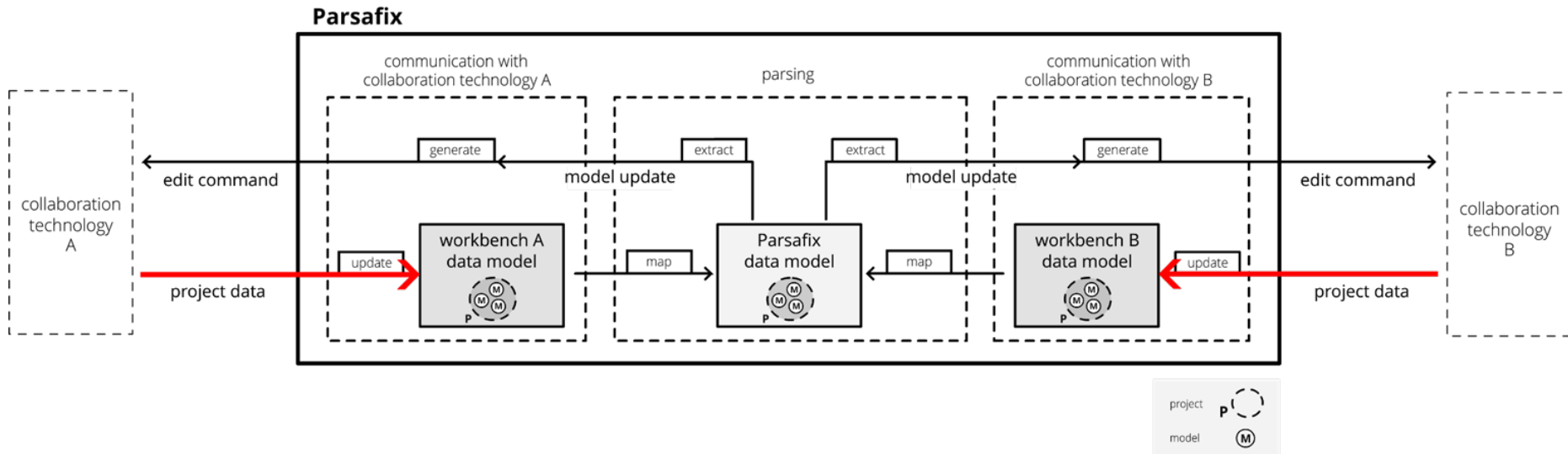
Internal Structure



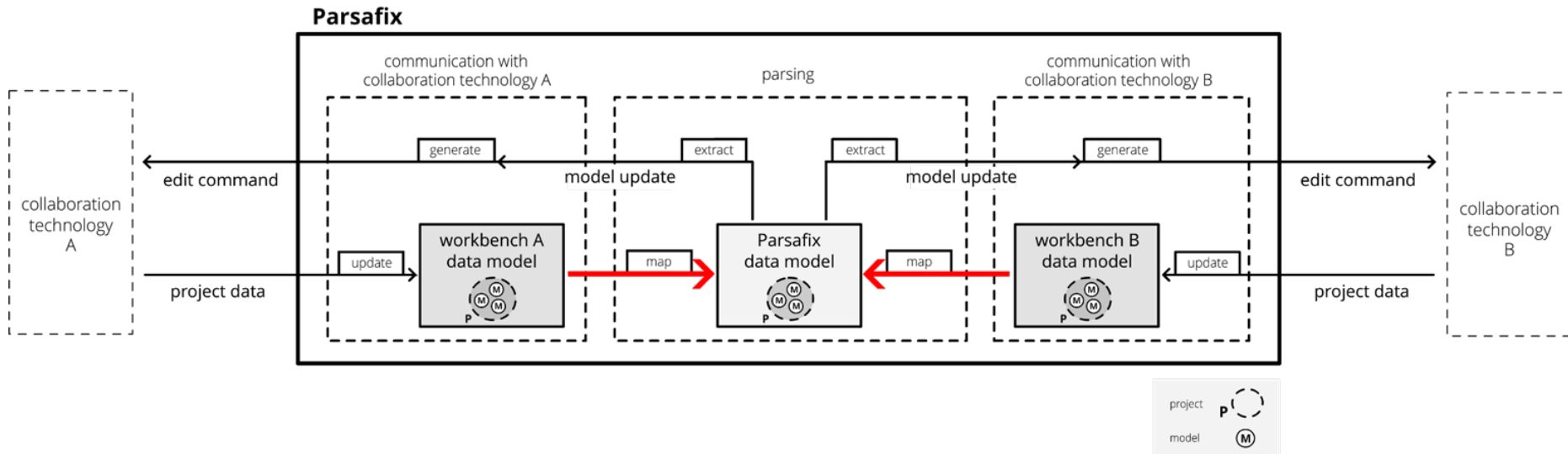
Internal Structure



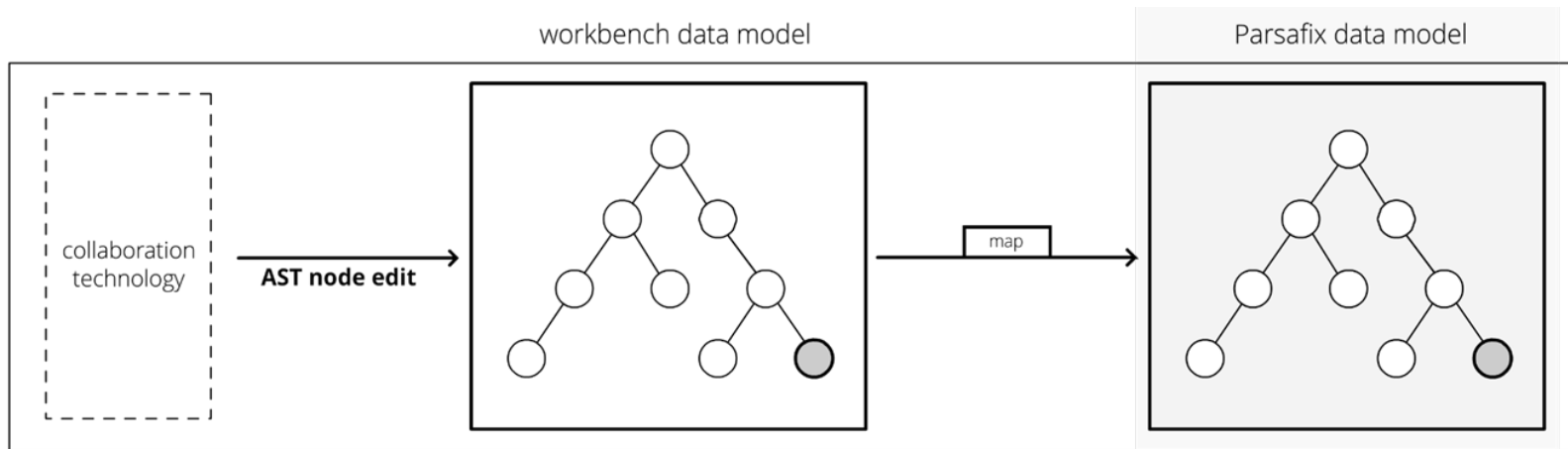
Internal Structure



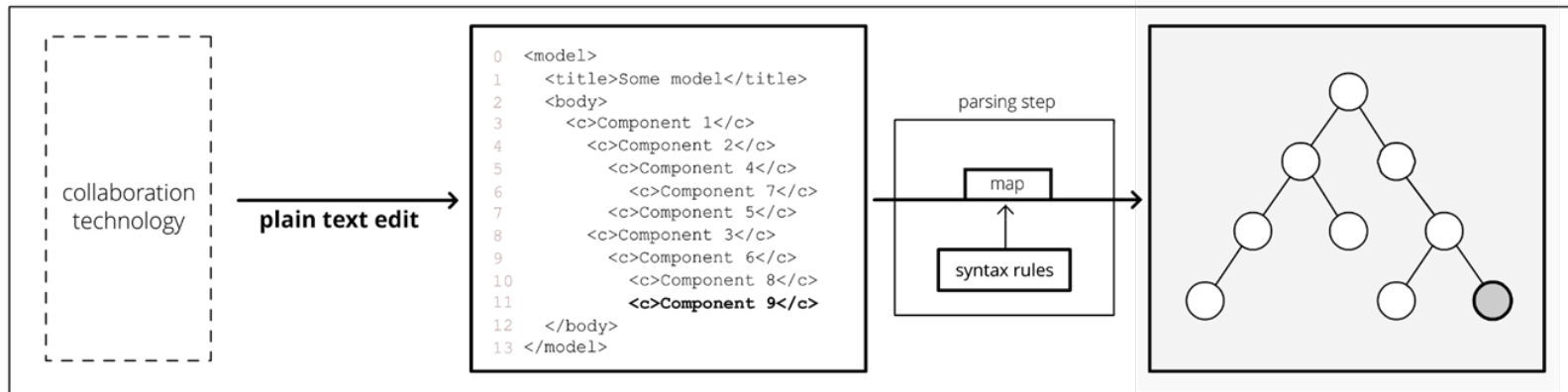
Internal Structure



(a) Parsing AST data

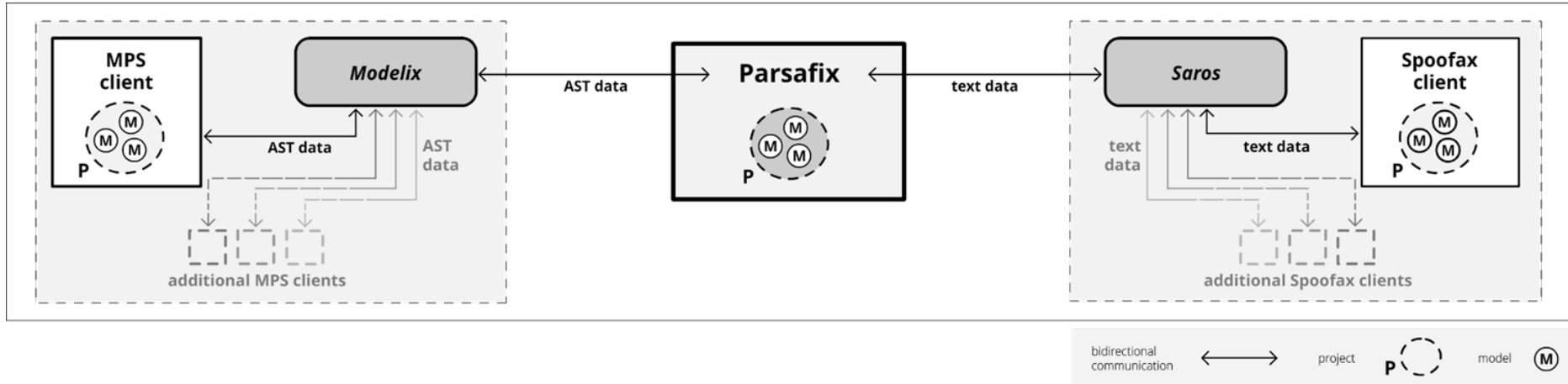


(b) Parsing text edits



Data parsing in Parsafix

Parsafix prototype



Collaboration architecture of Parsafix

Conclusion

“Parsafix enables real-time collaboration between MPS and Spoofox for model engineers working on a model in the same DSL”

- Extrapolate to other language workbenches and DSLs

Future Work

- Technical improvements
- Generalization → plugin system



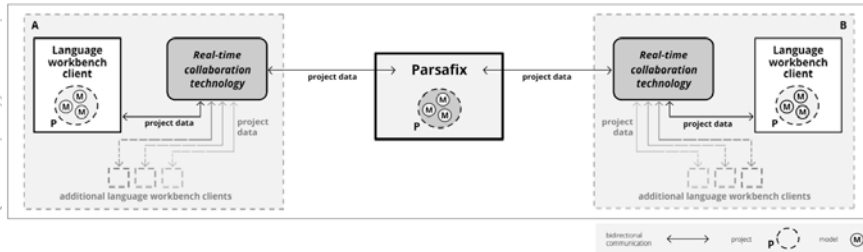
Problem



- Current limitation: real-time collaboration **single-platform support**




What if model engineers use different language workbenches?

Collaboration Architecture



Proposed Solution

open source: <https://github.com/blended-modeling/parsafix>

- **Model-Driven Engineering**
 - Domain Specific Language (DSL)
- **Spoofax**  → plugin for Eclipse 
- **OIL** (Open Interaction Language)
- **JetBrains MPS** 

Conclusion

"Parsafix enables real-time collaboration between MPS and Spoofax for model engineers working on a model in the same DSL."

- Extrapolate to other language workbenches and DSLs

Future Work

- Technical improvements
- Generalization → plugin system