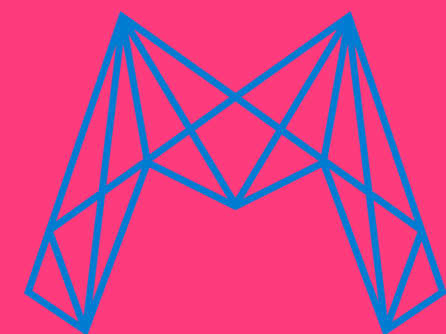




ESUG 2026 - Plovdiv

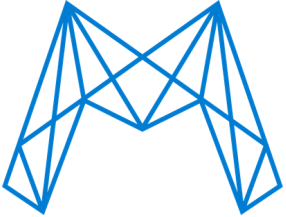
Building Moose importers with Tree Sitter

Cyril Ferlicot-Delbecque - cyril@ferlicot.fr



🌸 Hello, this is me!

Cyril Ferlicot-Delbecque

- Pharo dev 10 years+ 
- Improving Pharo
- Loves to refactor legacy code
- Doing software analysis 



 **Jecisc**

 **cyril@ferlicot.fr**

Context

Famix Integration

FAST Integration

Utilities

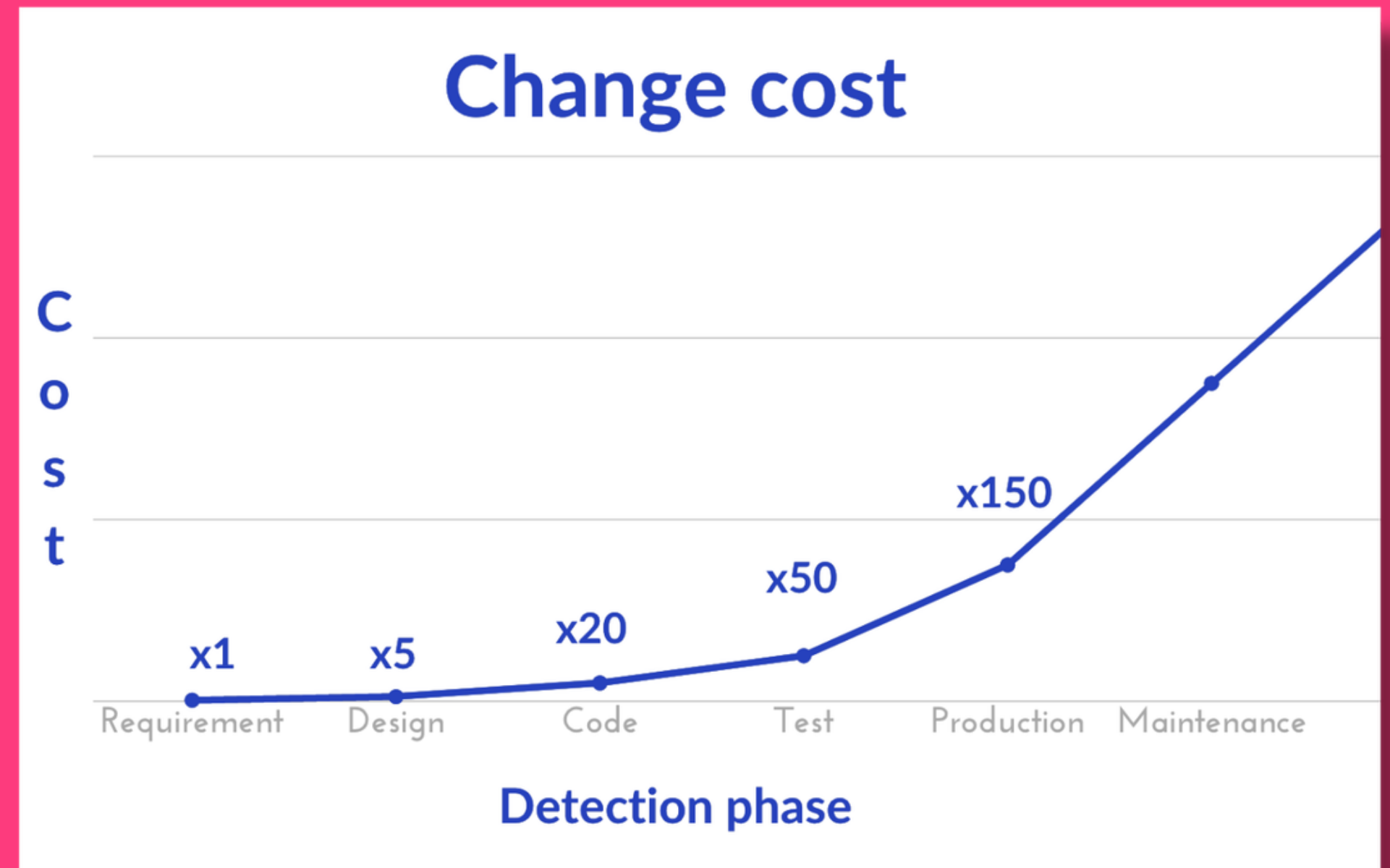
Conclusion

Context

+ ~80% of an IS cost is due to maintenance

+ The older the IS, the higher this percentage

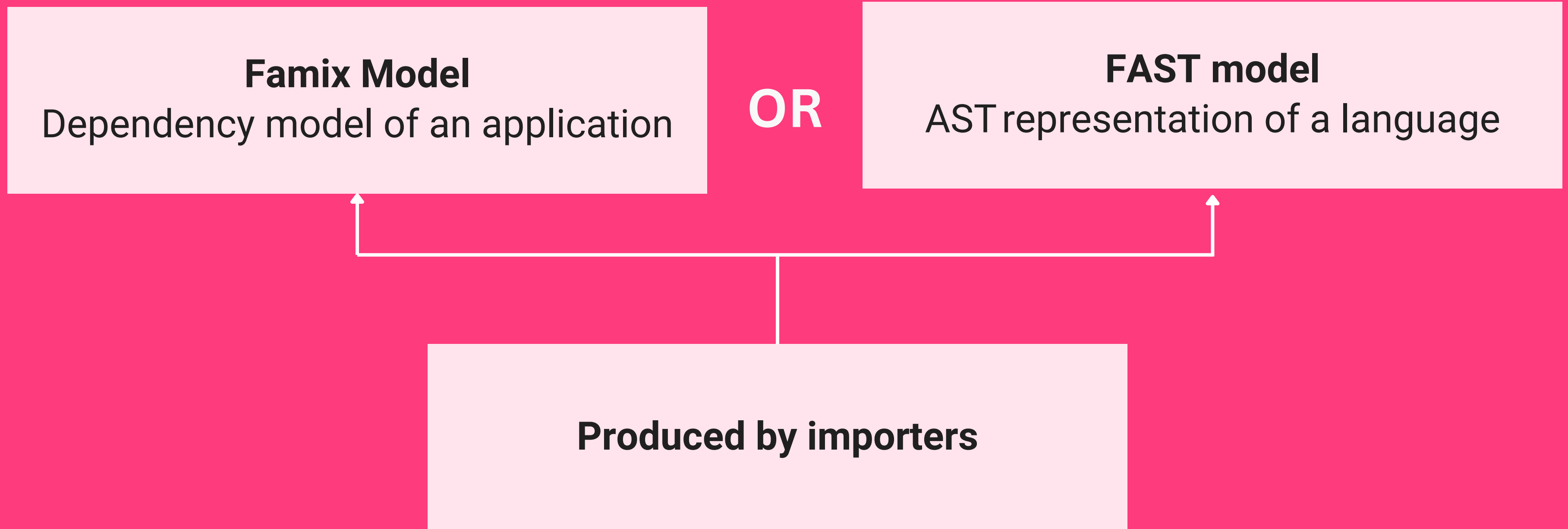
- +
- Large code base
 - More features
 - App in production



Minimise the **costs and **risks** of a
project maintenance**



Requirements



Parsing Library

- Fast
- Many languages
- Robust
- Dependency free



Pharo FFI Integration

<https://github.com/Evref-BL/Pharo-Tree-Sitter>



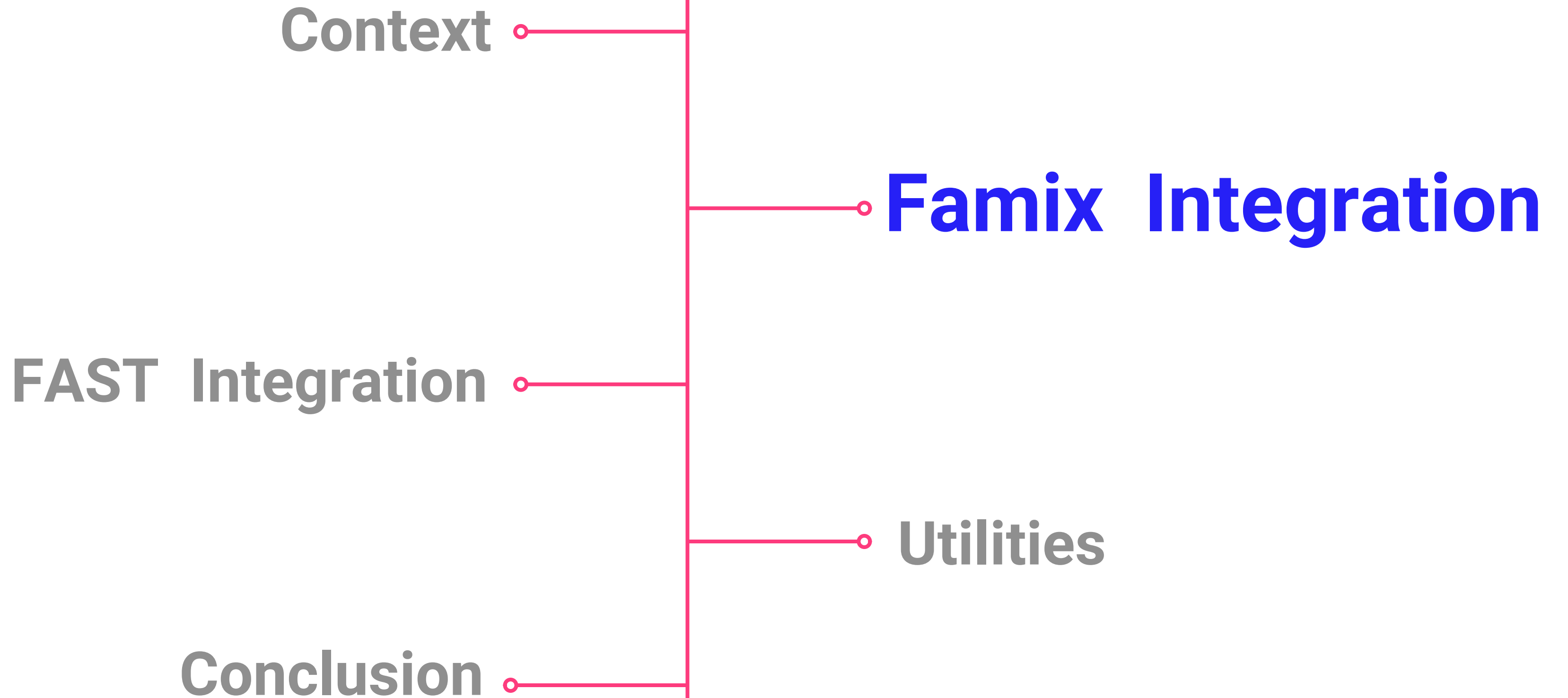
V1

Benoit Verhaeghe

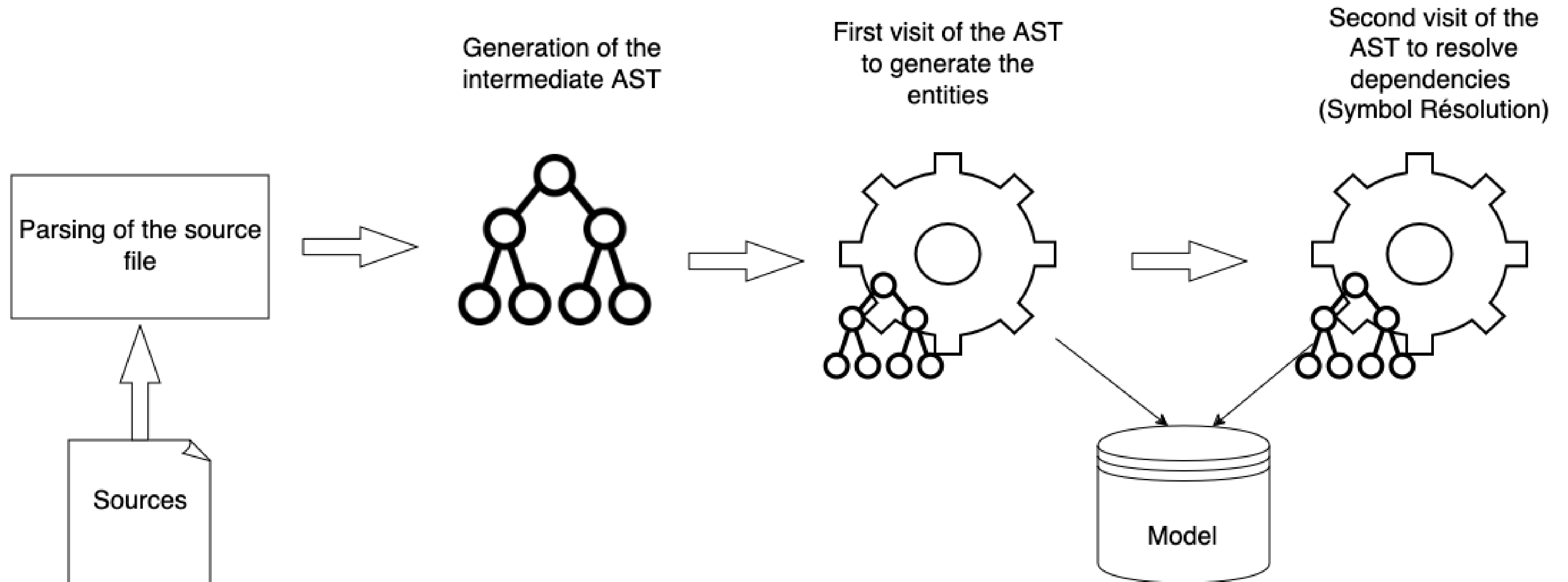


Main maintainer

Aless Hosry



Classic import process



Lifted limitations - <https://github.com/moosetechnology/TreeSitterFamixIntegration>



Node wrappers

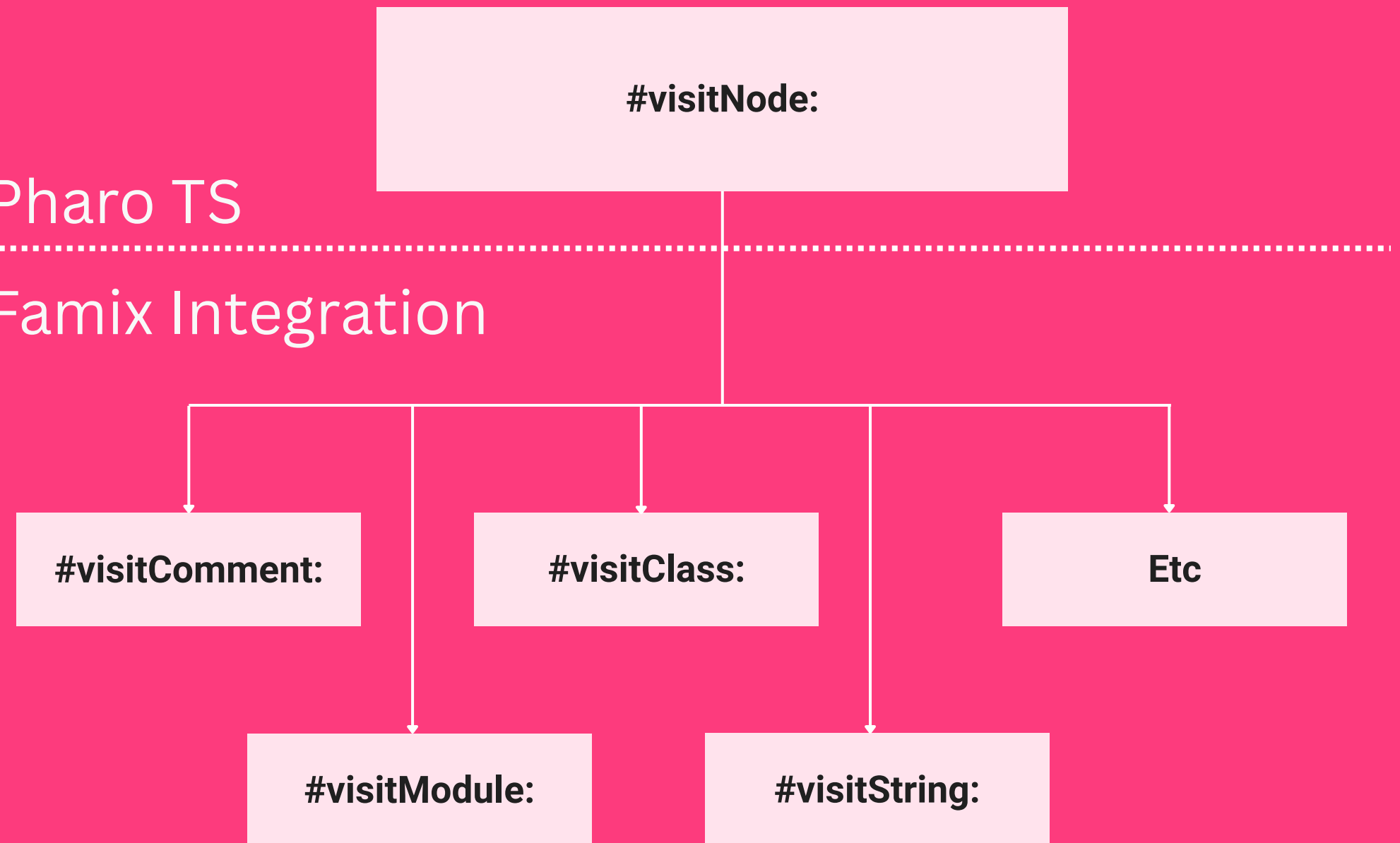
- Improve references
- Utilities (source, positions...)



Visitor based on type

Pharo TS

Famix Integration



Provided bases

Importer

- Common API for Moose
- Error management

Visitor

- Automatic source anchors
- Encoding management
- Utilities

Comments - FamixTSAbstractCommentsVisitor

- Out of the main visit
- Merge single line comments
- Automatic association to entities

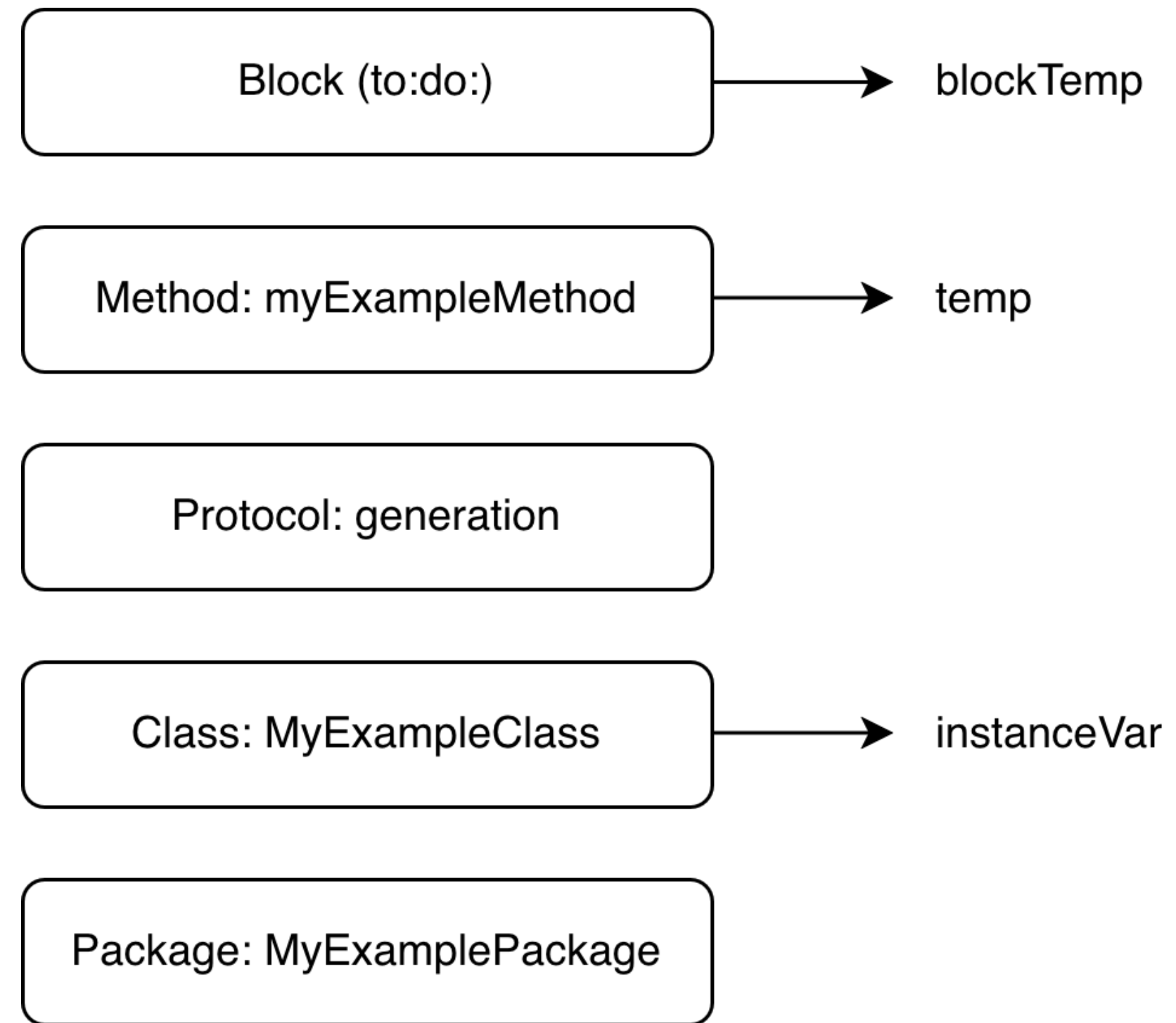
All the code needed for Python:

```
1 visitNode: aTSNode
2
3     aTSNode type = #comment ifTrue: [ self addSingleLineCommentNode: aTSNode ].
4
5     (aTSNode type = #string and: [ aTSNode sourceText beginsWith: "'''" ])
6         ifTrue: [ self addMultilineCommentNode: aTSNode ].
7
8     super visitNode: aTSNode
```

🔗 Symbol Resolution - <https://github.com/jecisc/SymbolResolver/>

```
Object << #MyExampleClass  
  slots: { #instanceVar };  
  package: 'MyExamplePackage'
```

```
1 myExampleMethod  
2   | temp |  
3   temp := 10.  
4  
5   1 to: 10 do: [ :int |  
6     | blockTemp |  
7     blockTemp := temp + instanceVar.  
8     temp := blockTemp + int ].  
9  
10  ^ temp
```



Default resolvables

- Method invocations
- Variables / Basic identifiers
- Local entities in a class

Tip:

The project also provides utilities for custom resolvers like imports that vary a lot between languages

Context

Famix Integration

FAST Integration

Utilities

Conclusion

Generate your base - <https://github.com/Evref-BL/Pharo-Tree-Sitter>

TSFASTBuilder

- Basic Metamodel
- Import infrastructure
- Use fields to associate relations

For Python:

```
TSFASTBuilder new  
  languageName: 'Python';  
  tsLanguage: TSLanguage python;  
  build
```

Generate your base

Your importer already works!

Includes error management

```
FASTPythonImporter parse: 'if x > 0:  
    if x < 10:  
        1  
    else:  
        2  
else:  
    3'.
```

```
FASTPythonImporter parseFile: 'myFile.py'
```

But we can make it better

TSFASTCustomizableVisitor

Customize relations

Use aliases if a TS field does not match a FAST relation

Skip useless nodes

For example, in Python a String contains a start, end and content nodes

Update Entities names

If the TreeSitter node name is not the one of the FAST entity

Add properties

Such as names, operands, aliases...

Custom parents

In case the Tree Sitter tree does not match the FAST one

Regression tests

```
FASTPythonImporterTest
  generateTestNamed: 'ifElif'
  fromCode: 'if x > 0:
    True
  elif x < 10:
    False'
  protocol: 'tests - ifs'
```

- FASTPython: **907 tests** in a few days
- Tests are **REALLY** important in importers
- Ability to regenerate tests if the model change (To be used wisely!)

Context

Famix Integration

FAST Integration

Utilities

Conclusion

TSSymbolsBuilderVisitor

Inspector

a TSSymbolsBuilderVisitor

Symbols explorer

Raw

Meta

Symbols

escape_interpolation

escape_sequence

except_clause

expression_list

false

finally_clause

float

for_in_clause

for_statement

format_expression

format_specifier

function_definition

future import statement

Filter...

Possible children by fields

none

not_operator

parenthesized_expression

string

subscript

true

▼ Field: consequence (Size: 1)

block

▼ Field: alternative (Size: 0..24)

elif_clause

else_clause

▼ Field: <unnamedChild> (Size: 0..29)

comment

Possible parents

block

module

Filter...

Copy filed items

Copy All fields

Copy name

Inspect field items

Example of code with a else_clause inside a if_statement

```
if hasattr(ns, k2):
    setattr(ns, k2, value)
elif hasattr(ns, k1):
    setattr(ns, k1, value)
else:
    raise AttributeError("no argument named '{}".format(k1))
```

TSNodeFinderVisitor

TSNodeFinderVisitor

language: TSLanguage python

extensions: #('py')

selection: [:node |

node type = #assignment and: [

node collectFieldNameOfNamedChild at: #right ifPresent: [:node | false] ifAbsent: [true]]

buildOn: '/Users/cyril/testPython/cpython-main' asFileReference.



a TSNodeFinderVisitor

SourceText

Full sourceText

Raw

Meta

```
36 from . import terminfo
37 from .console import Console, Event
38 from .fancy_termios import tcgetattr, tcsetattr, TermState
39 from .trace import trace
40 from .unix_eventqueue import EventQueue
41 from .utils import wlen
42
43 # declare posix optional to allow None assignment on other platforms
44 posix: types.ModuleType | None
45 try:
46     import posix
47 except ImportError:
48     posix = None
```

Context

Famix Integration

FAST Integration

Utilities

Conclusion

Current importers

Famix:

- Python
- C (Work in progress)

[*https://github.com/Evref-BL*](https://github.com/Evref-BL)

[*https://github.com/moosetechnology*](https://github.com/moosetechnology)

FAST:

- Python (Full)
- Typescript
- HTML
- CSS
- YAML
- XML

 Thank you for your attention

TS Pharo binding: <https://github.com/Evref-BL/Pharo-Tree-Sitter>

Famix integration <https://github.com/moosetechnology/TreeSitterFamixIntegration>

Symbol Resolver <https://github.com/jecisc/SymbolResolver>

Cyril Ferlicot-Delbecque
cyril@ferlicot.fr

