Incremental Graph Code (IGC)

Bridging Incremental Programming and Complex Software Development in Modern Programming Environments

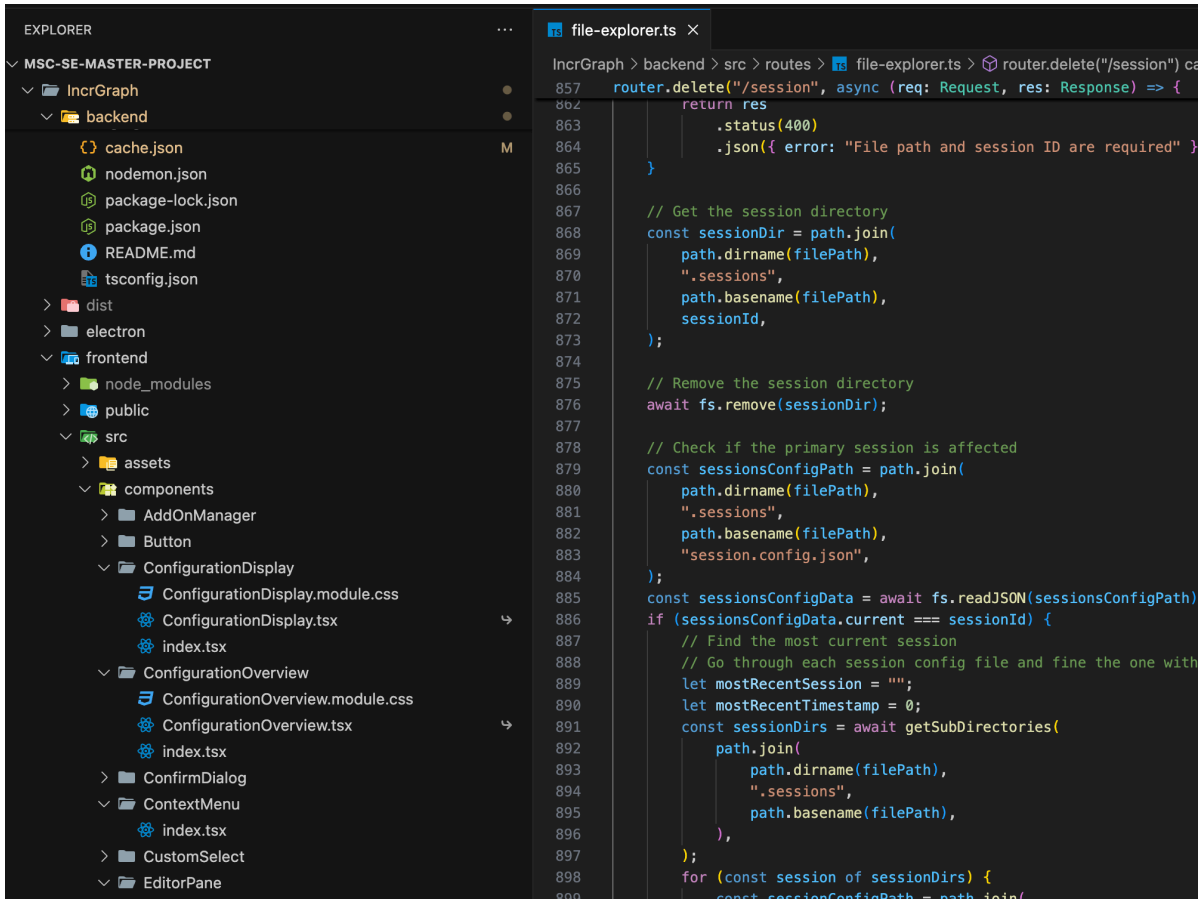
Max Boksem, Thomas van Binsbergen

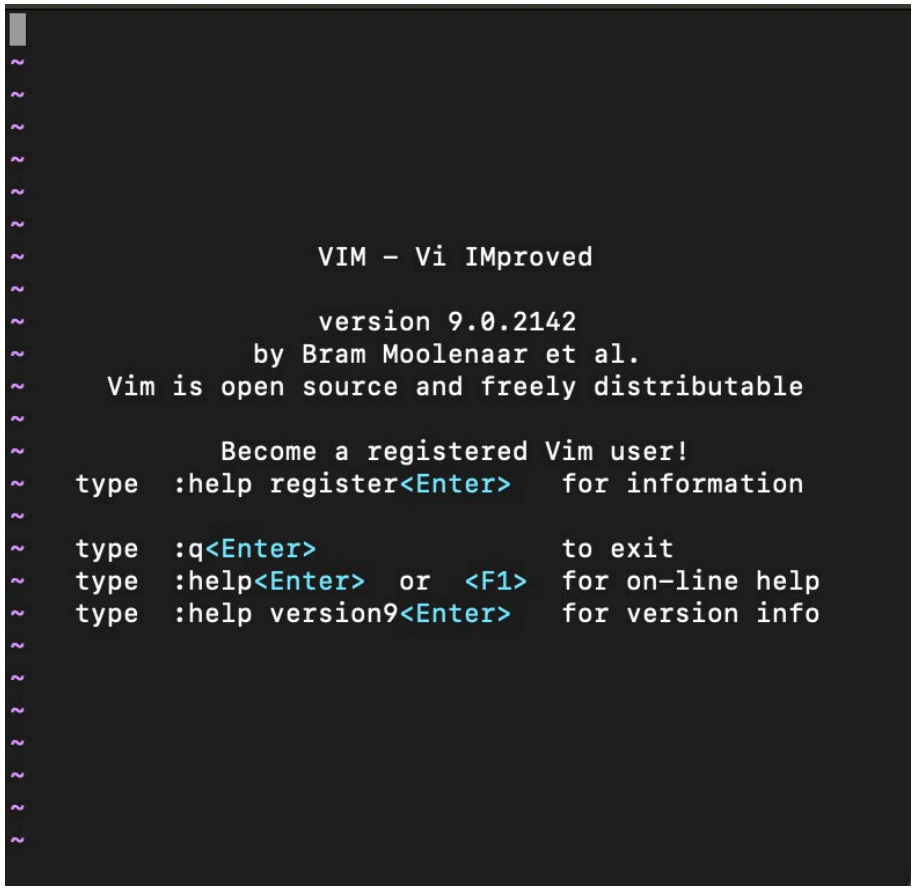
Background

Typical Software Development Environments

Background

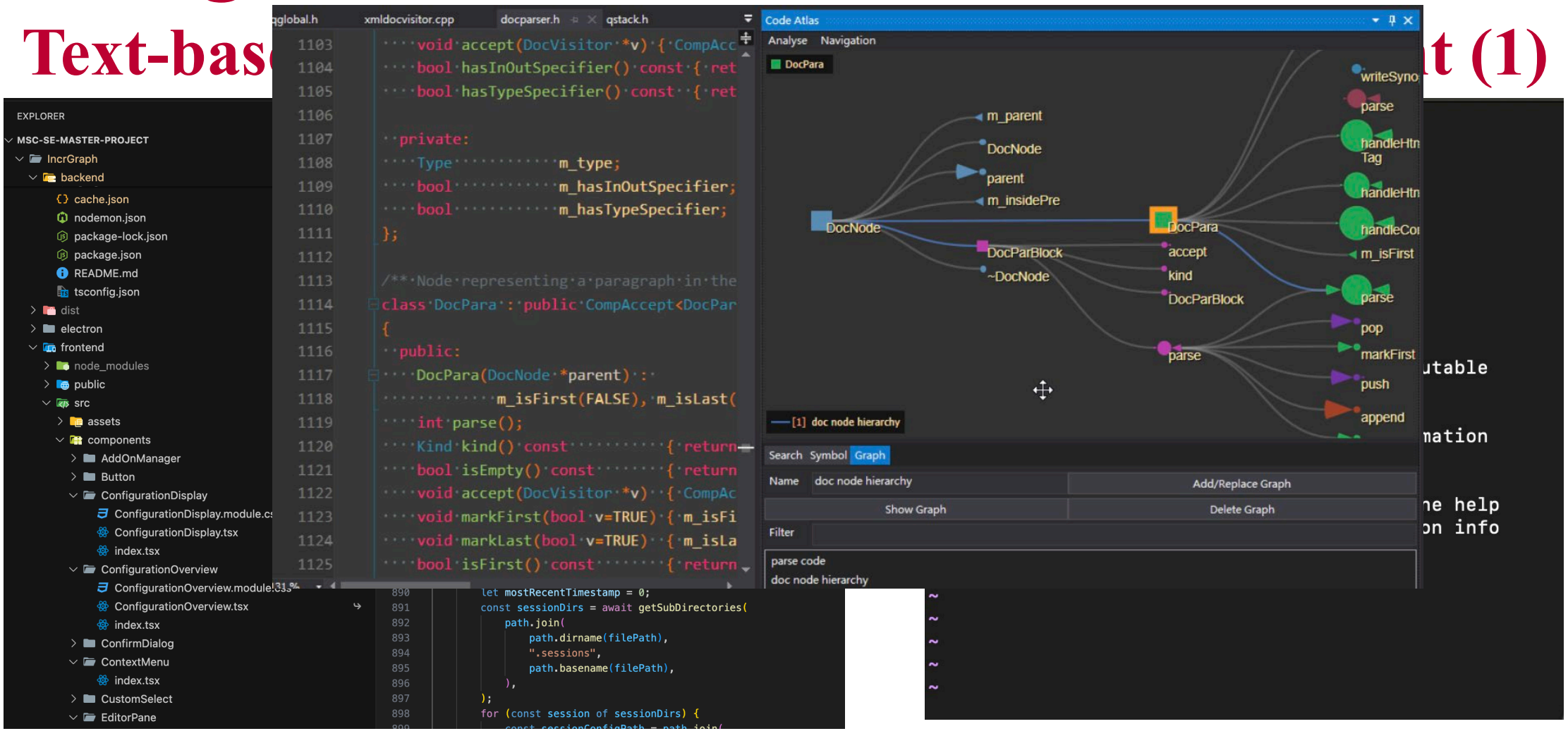
Text-based Integrated Development Environment (1)





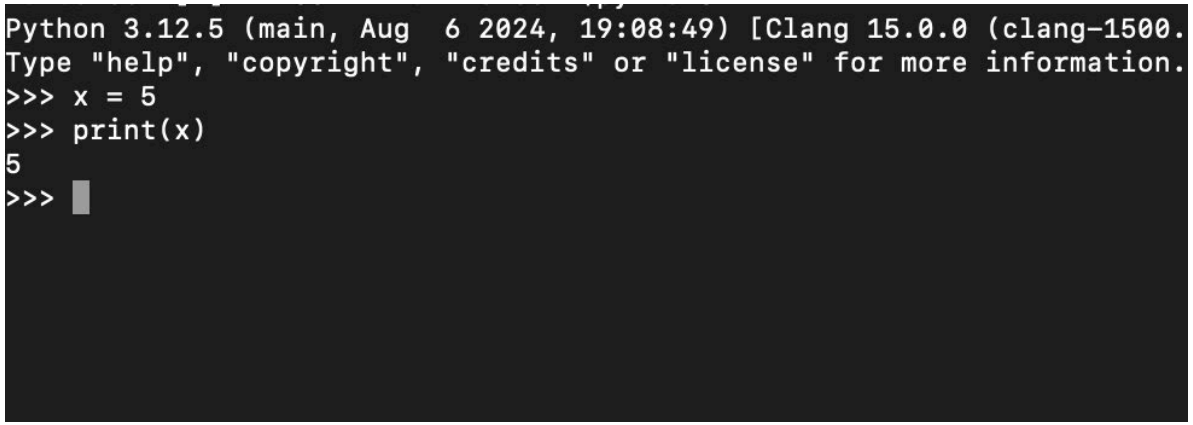
Background

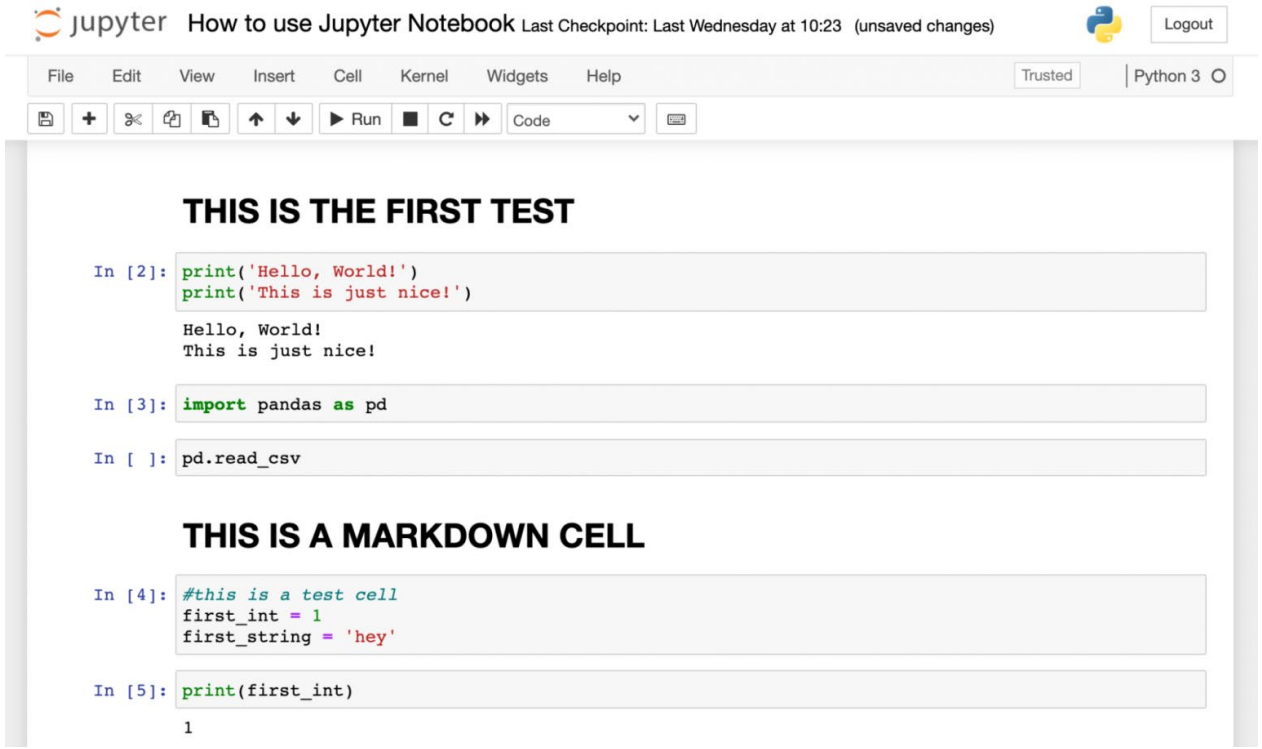
Text-bas t (1)



Background

Incremental Programming Environment (IPE)





Design

How can we bridge these two environments?

Complexity management tools from typical IDEs

Incremental Programming techniques from IPEs

Design

How can we bridge these two environments?

Integrated Development Environments

Incremental Programming Environments

Design

How can we bridge these two environments?

Integrated Development Environments

- Manage large, multi-file codebases
- Strong project structure (files, packages, classes)
- Readability and code comprehension
- Extensibility
- Integration with version control

Incremental Programming Environments

Design

How can we bridge these two environments?

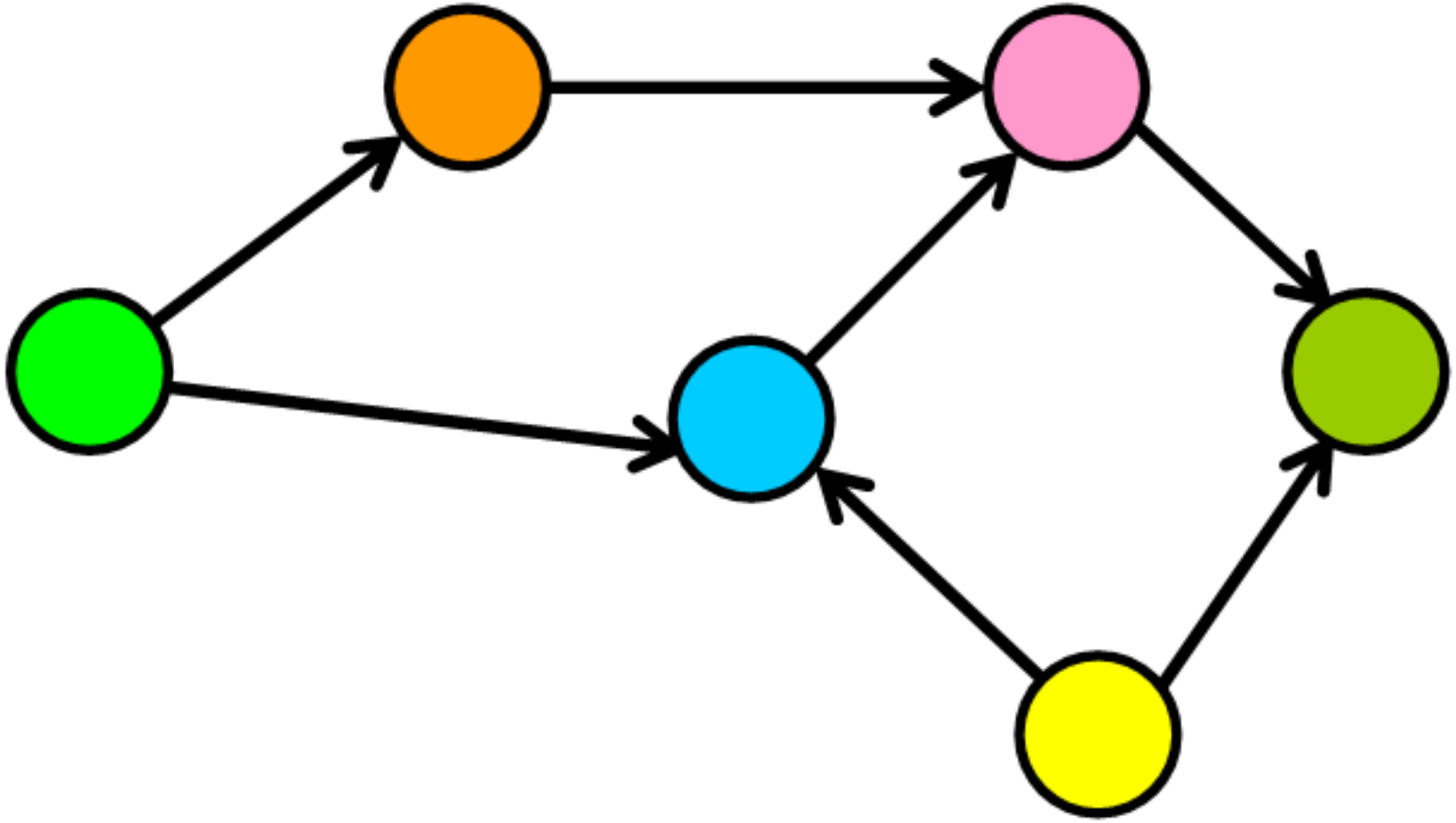
Integrated Development Environments

- Manage large, multi-file codebases
- Strong project structure (files, packages, classes)
- Readability and code comprehension
- Extensibility
- Integration with version control

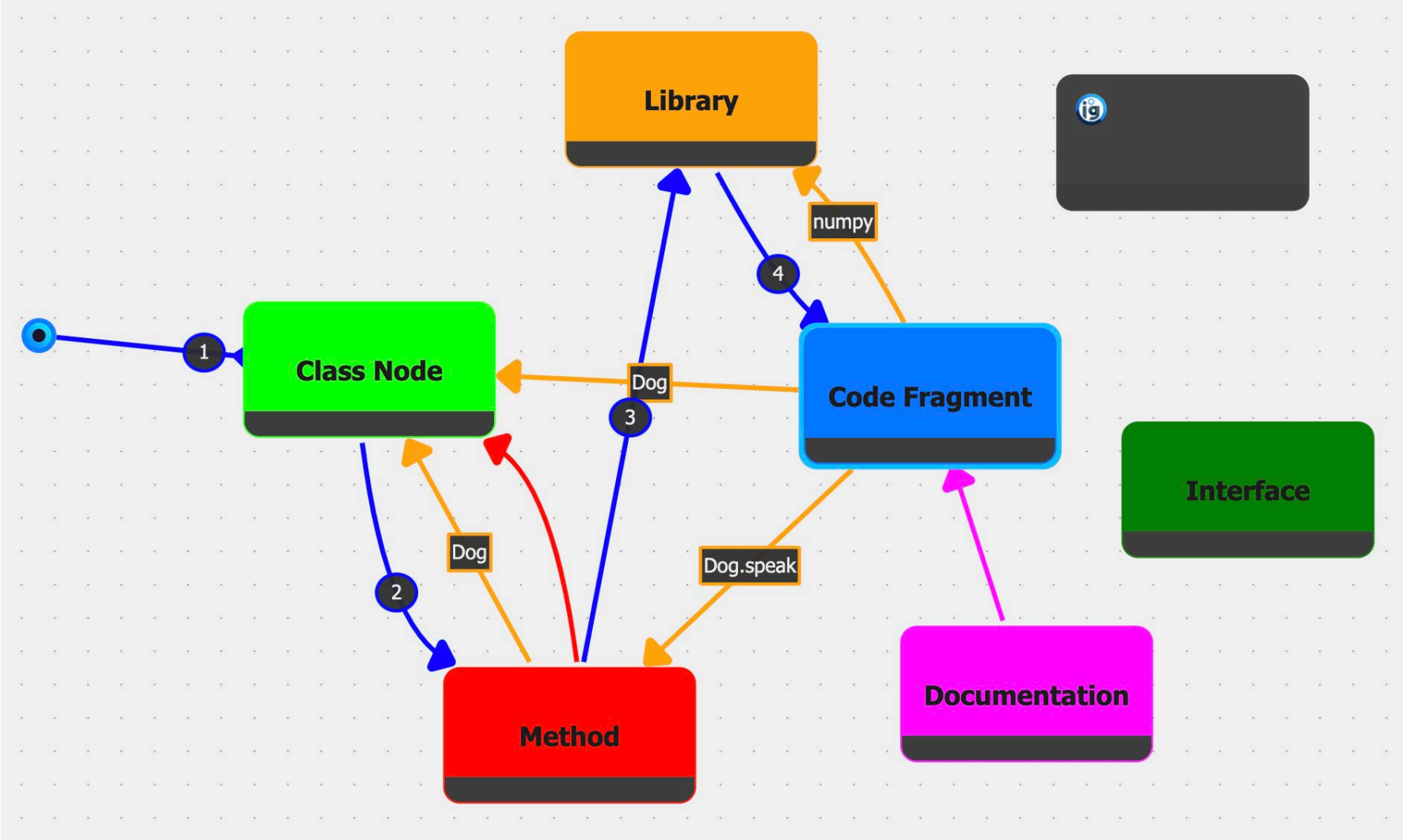
Incremental Programming Environments

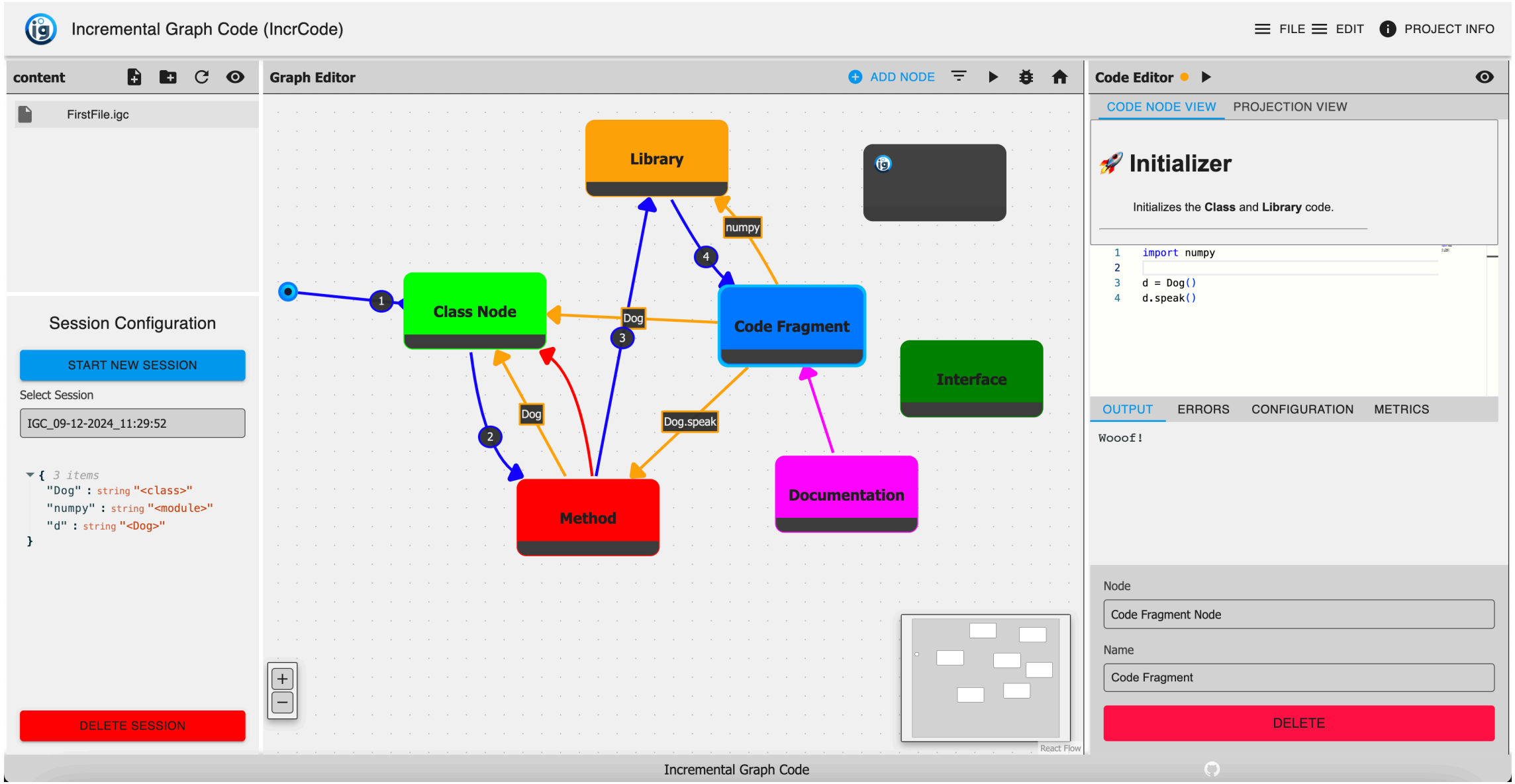
- Run and test small code snippets quickly
- Immediate feedback and rapid iteration
- Exploratory coding approach
- Documentation
- Incremental recomputation

Design



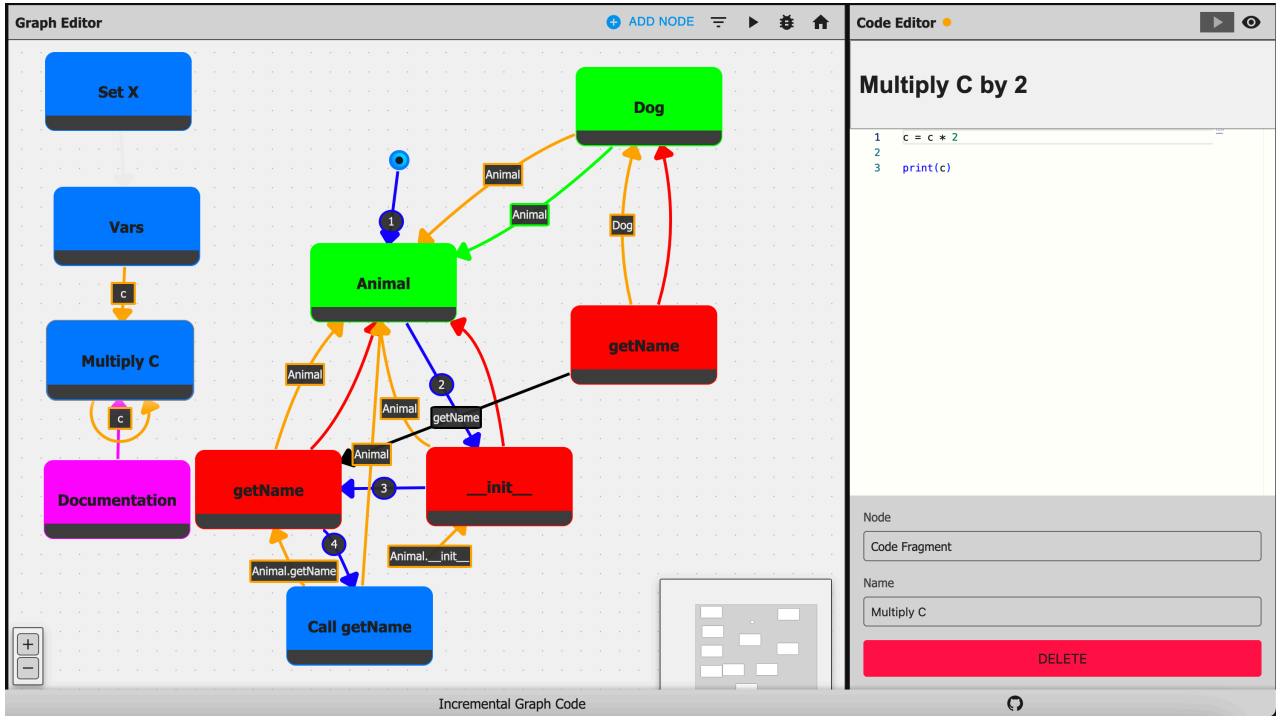
Design


Library
numpy
4
1
Class Node
Dog
3
Code Fragment
Interface
Dog
2
Dog.speak
Method
Documentation


Incremental Graph Code (IncrCode)
FILE
EDIT
PROJECT INFO
content
FirstFile.igc
Session Configuration
START NEW SESSION
Select Session
IGC_09-12-2024_11:29:52
{ 3 items
"Dog" : string "<class>"
"numpy" : string "<module>"
"d" : string "<Dog>"
}
DELETE SESSION
Graph Editor
ADD NODE
Library
Class Node
Code Fragment
Interface
Method
Documentation
numpy
Dog
Dog
Dog.speak
1
2
3
4
React Flow
Code Editor
CODE NODE VIEW
PROJECTION VIEW
Initializer
Initializes the Class and Library code.
1 import numpy
2
3 d = Dog()
4 d.speak()
OUTPUT
ERRORS
CONFIGURATION
METRICS
Wooof!
Node
Code Fragment Node
Name
Code Fragment
DELETE
Incremental Graph Code

Incremental Graph Code (IGC)
Overview

- Electron Webapp
  - Frontend – React (Typescript) using ReactFlow
    - Responsible for visuals, logic, and overall operation
  - Backend – Node.js (Typescript)
    - Handles file operations and code analysis



Currently Supports Python

Open-Source and Community focused

Case Studies

- Computational Notebook
- Projectional Display (PescaJ *Paint 2023*)
- Exploratory Programming View
- Composition / Architectural View

Final Thoughts

Max Boksem – max.boksem.nl@gmail.com

Thomas van Binsbergen – l.t.vanbinsbergen@uva.nl

Github