GOOGLE SUMMER OF CODE 25’

IMPLEMENTING DEBUGGING SUPPORT FOR XEUS-CPP

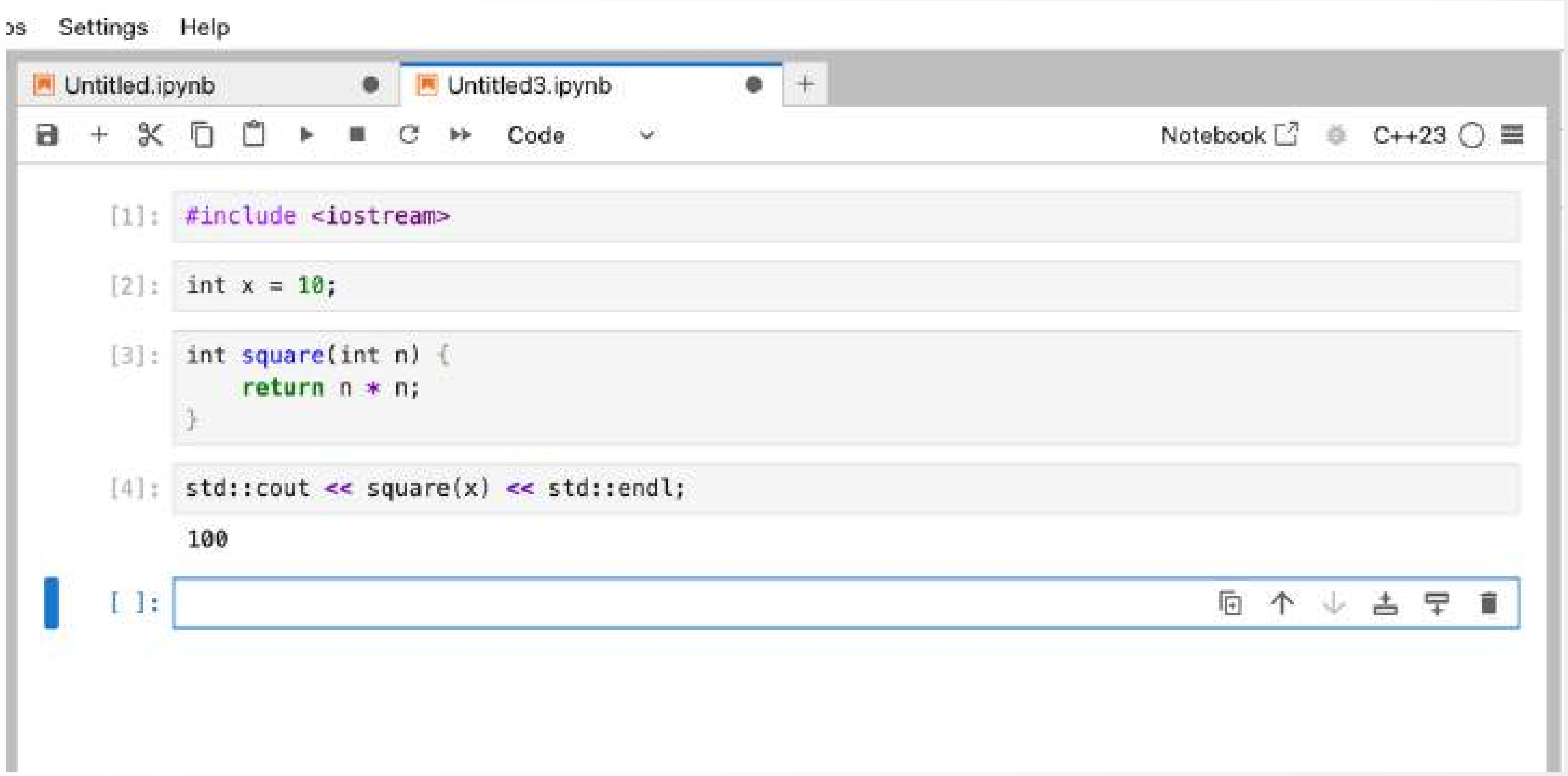
MIDTERM PROGRESS UPDATE

Author: Abhinav Kumar

Mentors: Anutosh Bhat, Vipul Cariappa, Aaron Jomy, Vassil Vassilev

WHAT IS XEUS-CPP?

- **Xeus-Cpp** is a Jupyter kernel that enables interactive C++ programming within the Jupyter environment.
- It is built on the Xeus library—a C++ implementation of the Jupyter kernel protocol.
- Powered by the ***Clang-Repl*** interpreter from the ***CppInterOp*** library, Xeus-Cpp allows you to write, execute in real-time, much like you would with Python.



```
[1]: #include <iostream>

[2]: int x = 10;

[3]: int square(int n) {
         return n * n;
     }

[4]: std::cout << square(x) << std::endl;

     100

[ ]:
```

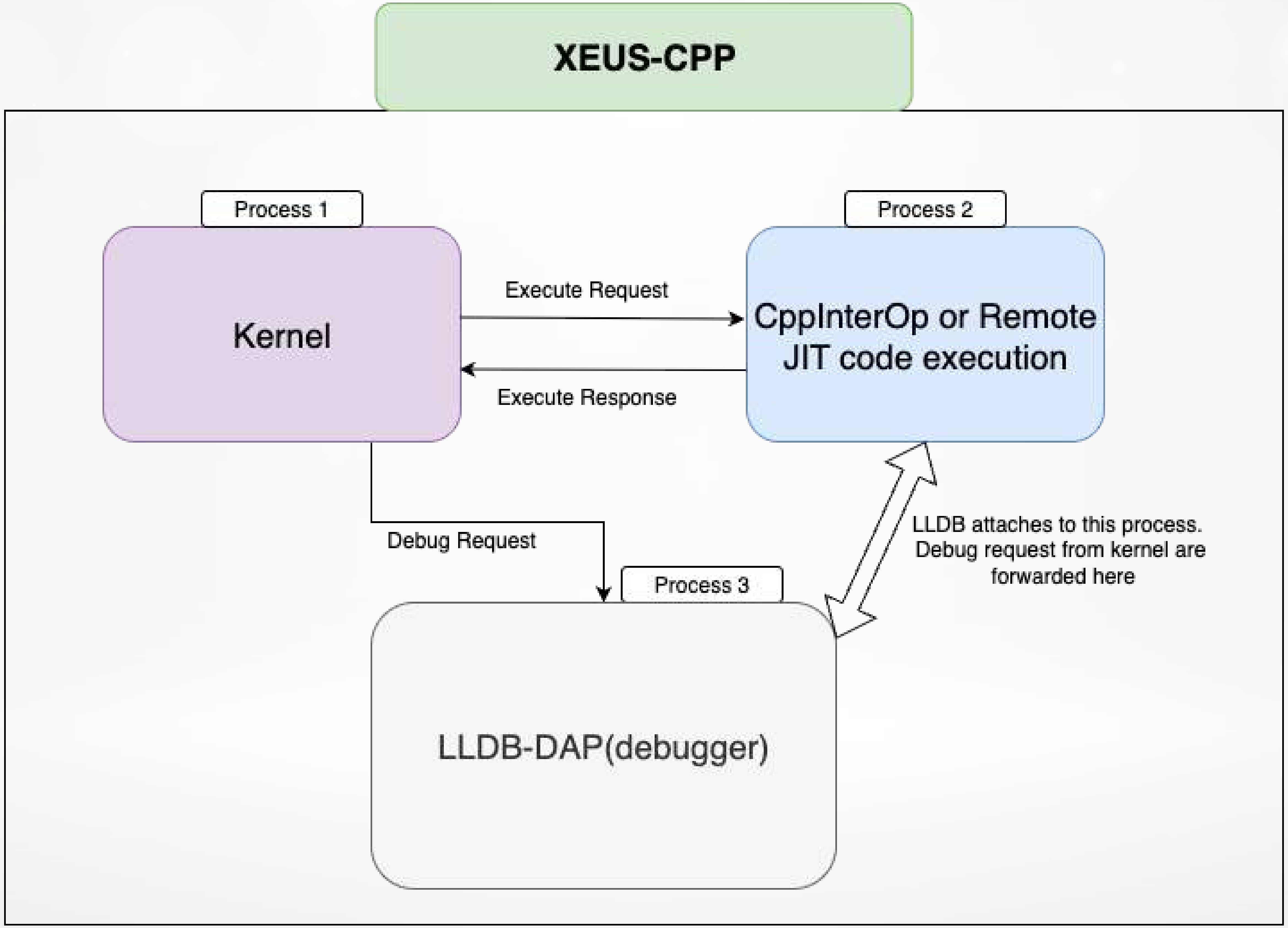
ATTACHING LLDB-DAP TO
JIT CODE EXECUTION

Since, LLDB-DAP can't be directly attached to xeus-cpp kernel(due to kernel-dead issue), we came up with a solution that LLDB-dap should attach to JIT code execution.

So, the JIT code execution part has to be processed on different port or as a different process than kernel.

In clang-repl, this can be achieved via **Out-Of-Process Remote JIT execution**.

WORKFLOW


XEUS-CPP
Process 1
Kernel
Execute Request
Execute Response
Process 2
CppInterOp or Remote JIT code execution
Debug Request
Process 3
LLDB-DAP(debugger)
LLDB attaches to this process. Debug request from kernel are forwarded here

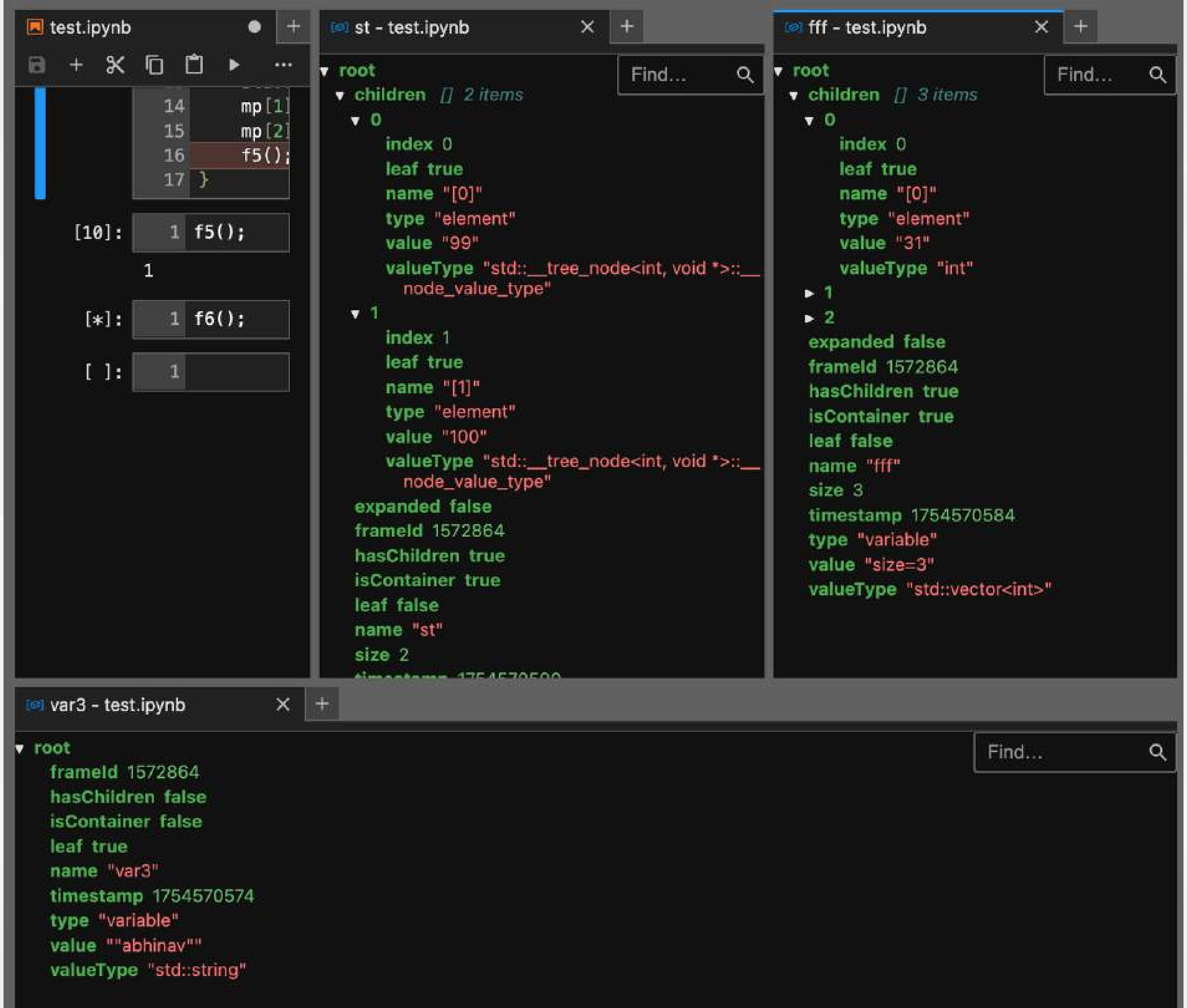
DEBUGGER FOR XEUS-CPP

WHAT’S WORKING?

- LLDB-DAP attaches to remote JIT code execution process.
- Setting breakpoints.
- Hitting breakpoints.
- Variable inspection(primitive datatypes).
- Continue.
- Step In(Although requires multiple step-ins due to JIT intermediate code).

PROTOTYPE DEMO

Link: Demo



```
    mp[1]
    mp[2]
    f5();
}
```

```
[10]: f5();
1
[*]: f6();
[ ]:
```

```
▼ root
  ▼ children  [] 2 items
    ▼ 0
        index 0
        leaf true
        name "[0]"
        type "element"
        value "99"
        valueType "std::__tree_node<int, void *>::__
          node_value_type"
    ▼ 1
        index 1
        leaf true
        name "[1]"
        type "element"
        value "100"
        valueType "std::__tree_node<int, void *>::__
          node_value_type"
    expanded false
    frameId 1572864
    hasChildren true
    isContainer true
    leaf false
    name "st"
    size 2
    timestamp 1754570580
```

```
▼ root
  ▼ children  [] 3 items
    ▼ 0
        index 0
        leaf true
        name "[0]"
        type "element"
        value "31"
        valueType "int"
    ► 1
    ► 2
    expanded false
    frameId 1572864
    hasChildren true
    isContainer true
    leaf false
    name "fff"
    size 3
    timestamp 1754570584
    type "variable"
    value "size=3"
    valueType "std::vector<int>"
```

```
▼ root
    frameId 1572864
    hasChildren false
    isContainer false
    leaf true
    name "var3"
    timestamp 1754570574
    type "variable"
    value ""abhinav""
    valueType "std::string"
```

CURRENT
PROGRESS

- *jupyterlab*: Fix threadId being passed to the debugger (**Merged**)
- *CppInterOp*: Documentation for debugging CppInterOp using LLDB (**Merged**)
- *llvm-project*: [Clang-Repl] Add custom function as lambda in launchExecutor and fetch PID of launched executor (**LGTMed. Soon to be merged**)
- *CppInterOp*: [GSoC 2025] Enabling Out-Of-Process JIT Execution in CppInterOp (**Soon to be merged**)
- *Xeus-Cpp*: [wip] Implementing debugger in xeus-cpp(**Prototype Implementation**)

CURRENT
PROGRESS

- *llvm-project*: [Clang-Repl] Using shared memory in Out-Of-Process JIT execution gives Permission Denied error on MacOS - **ISSUE**

- *xeus-cpp*: [GSoC 2025] Tracking Issue: Implementing Debugger Support for xeus-cpp - **ISSUE**

- *jupyterlab*: Bug: Multiple configurationDone Requests Sent by JupyterLab - **ISSUE**

- *jupyterlite*: Debugger Support for Jupyterlite - **FUTURE WORK**

FUTURE WORKS

- Merging out-of-process JIT execution PR in CppInterOp.
- Planning out step-by-step small PRs for xeus-cpp debugger implementation(currently a big prototype implementation is there).
- Enabling exhaustive tests for xeus-cpp debugger.
- Think about debugger in wasm(xeus-cpp-lite)
- Fix some bugs.

THANK YOU