

- Graph
- LargeTraces
- Multitraces
- Github

Graph

Trial and error, tried to get other libraries like graphviz/tried multiple things. Will adjust/fix later.

Check causal relationship

Node 0 is causal to Node 25.
Nodes 0 and 7 are not connected.

```
def print_causality_info(node_index1, node_index2, successors_dict):
    successors_of_node1 = successors_dict.get(node_index1, {}).get('successors', [])
    successors_of_node2 = successors_dict.get(node_index2, {}).get('successors', [])

    if node_index1 in successors_of_node2:
        print(f"Node {node_index2} is causal to Node {node_index1}.")
    elif node_index2 in successors_of_node1:
        print(f"Node {node_index1} is causal to Node {node_index2}.")
    else:
        print(f"Nodes {node_index1} and {node_index2} are not connected.")
```

See any node's relations/children.

```
# Print detailed path information for a specific node
node_index_to_query = 0 # Change this to the node index you're interested in
print_successor_info(node_index_to_query, successors_dict)
```

EXAMPLE: Initial Node 0 (0 : cpu0:cache0:wt:req) ✓

```
Node 0 (Source: cpu0, Destination: cache0) has successors:
  Successor: 8, Path: cache0 -> cache1
  Successor: 11, Path: cache0 -> cache1
  Successor: 13, Path: cache0 -> membus
  Successor: 17, Path: cache0 -> cache1
  Successor: 18, Path: cache0 -> membus
  Successor: 20, Path: cache0 -> cache1
  Successor: 21, Path: cache0 -> membus
  Successor: 27, Path: cache0 -> membus
  Successor: 25, Path: cache0 -> cpu0
  Successor: 26, Path: cache0 -> cpu0
```

EXAMPLE: Middle Node 7 : cache1:cache0:wt:req ✓

```

Node 7 (Source: cache1, Destination: cache0) has successors:
Successor: 8, Path: cache0 -> cache1
Successor: 11, Path: cache0 -> cache1
Successor: 13, Path: cache0 -> membus
Successor: 17, Path: cache0 -> cache1
Successor: 18, Path: cache0 -> membus
Successor: 20, Path: cache0 -> cache1
Successor: 21, Path: cache0 -> membus
Successor: 27, Path: cache0 -> membus
Successor: 25, Path: cache0 -> cpu0
Successor: 26, Path: cache0 -> cpu0

```

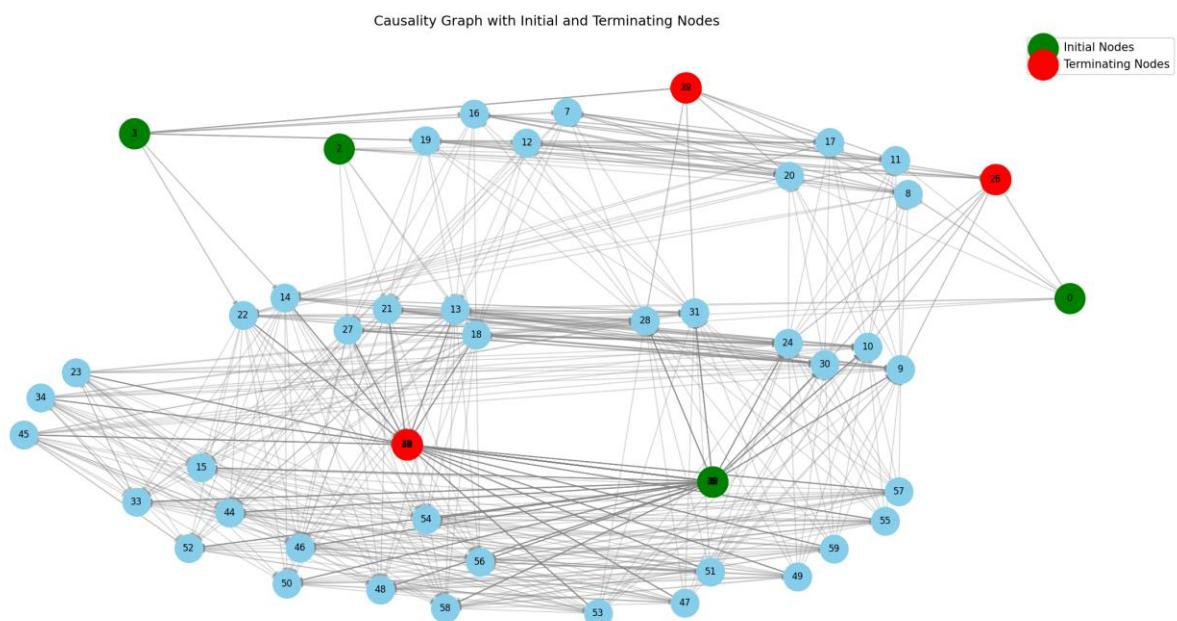
EXAMPLE: Terminating Node 25 : cache0:cpu0:wt:resp ✓

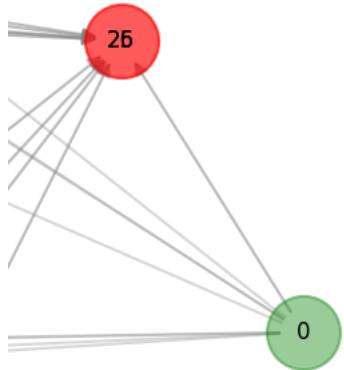
```

Node 25 (Source: cache0, Destination: cpu0) has successors:

```

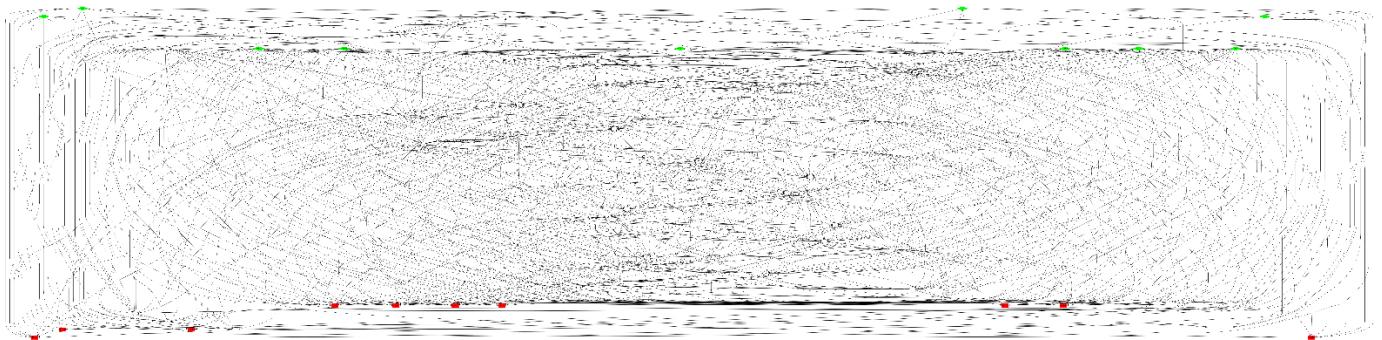
Graph

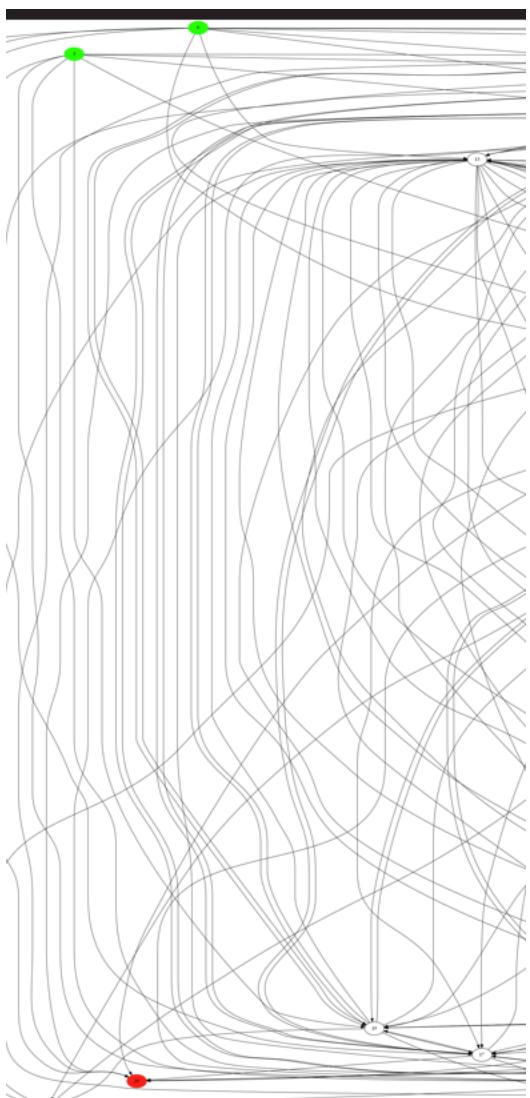




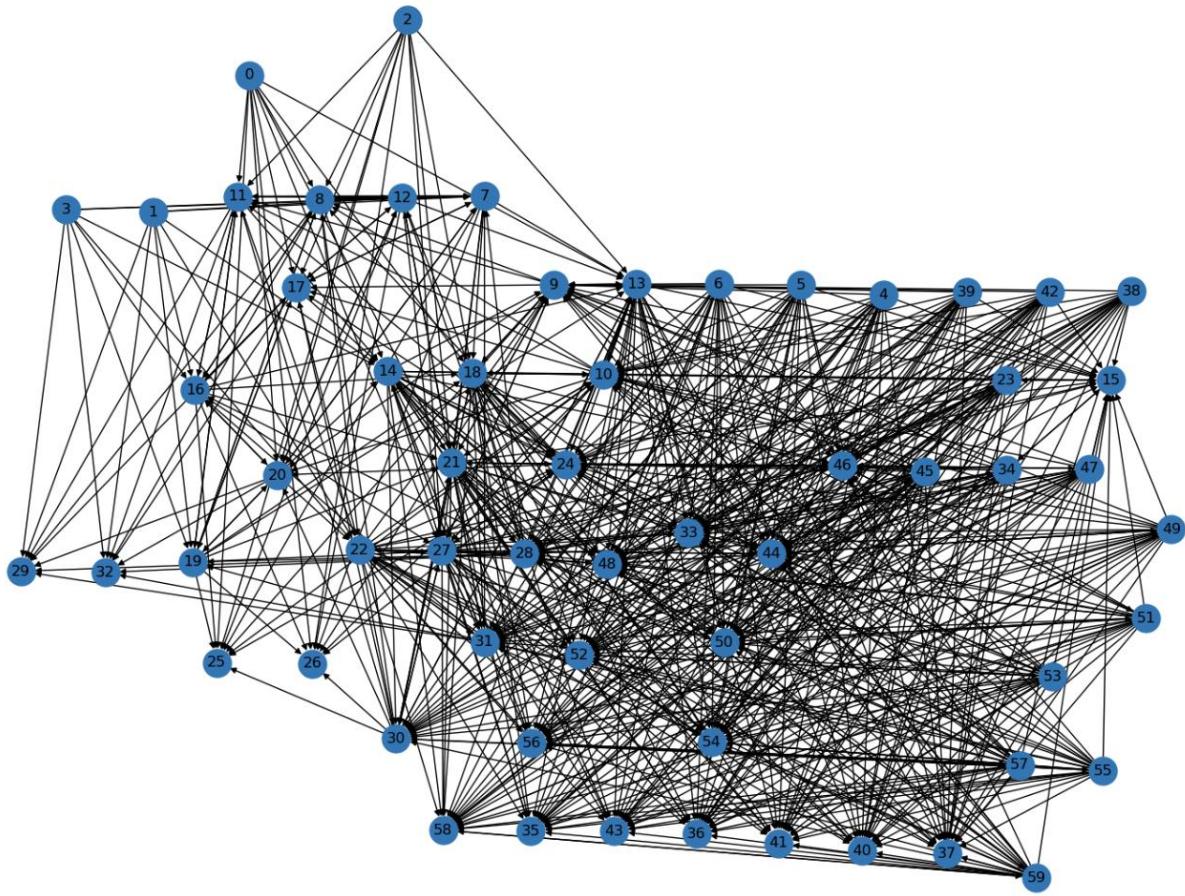
No incoming messages to initial nodes, no outgoing arrows from terminating nodes. ✓

Tried using graphviz: (too messy, but when zoomed in the image is clearer)





Or



Groups from message file:

```
('audio', 'membus', (5, 36, 39, 41, 48, 49, 58, 59))
('cache0', 'cache1', (7, 8, 11, 12, 16, 17, 19, 20))
('cache0', 'cpu0', (0, 2, 25, 26))
('cache0', 'membus', (9, 10, 13, 18, 21, 24, 27, 30))
('cache1', 'cpu1', (1, 3, 29, 32))
('cache1', 'membus', (14, 22, 28, 31))
('gfx', 'membus', (4, 35, 38, 40, 54, 55, 56, 57))
('mem', 'membus', (15, 23, 33, 34))
('membus', 'uart', (42, 43, 46, 47, 50, 51))
('membus', 'usb', (6, 37, 44, 45, 52, 53))
```

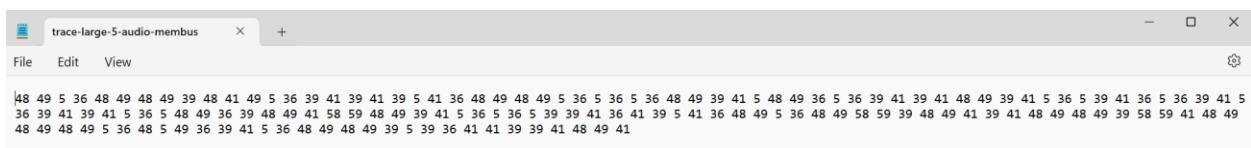
Reading in Large Traces

File	Date	Type
trace-large-5	2/26/2024 5:09 AM	File folder
trace-large-10	2/26/2024 5:09 AM	File folder
trace-large-20	2/26/2024 5:09 AM	File folder

Trace-large-5 example

- trace-large-5-audio-membus
- trace-large-5-cache0-cache1
- trace-large-5-cache0-cpu0
- trace-large-5-cache0-membus
- trace-large-5-cache1-cpu1
- trace-large-5-cache1-membus
- trace-large-5-gfx-membus
- trace-large-5-membus-uart
- trace-large-5-membus-usb
- trace-large-5-mem-membus

Audio-membus Example: should have these numbers ('audio', 'membus', (5, 36, 39, 41, 48, 49, 58, 59))



Verification (ran a python function to check all unique numbers) ✓

36, 5, 39, 41, 48, 49, 58, 59

Reading in Multiple Traces

multipleTraces-synthetic (has 6 traces)

📁 multipleTraces-synthetictrace-1	2/26/2024 5:21 AM	File folder
📁 multipleTraces-synthetictrace-2	2/26/2024 5:21 AM	File folder
📁 multipleTraces-synthetictrace-3	2/26/2024 5:21 AM	File folder
📁 multipleTraces-synthetictrace-4	2/26/2024 5:21 AM	File folder
📁 multipleTraces-synthetictrace-5	2/26/2024 5:21 AM	File folder
📁 multipleTraces-synthetictrace-6	2/26/2024 5:21 AM	File folder

And each trace has

📄 multipleTraces-synthetic-audio-membus
📄 multipleTraces-synthetic-cache0-cache1
📄 multipleTraces-synthetic-cache0-cpu0
📄 multipleTraces-synthetic-cache0-membus
📄 multipleTraces-synthetic-cache1-cpu1
📄 multipleTraces-synthetic-cache1-membus
📄 multipleTraces-synthetic-gfx-membus
📄 multipleTraces-synthetic-membus-uart
📄 multipleTraces-synthetic-membus-usb
📄 multipleTraces-synthetic-mem-membus

multipleTraces-syntheticSmall (has 3)

📁 multipleTraces-syntheticSmalltrace-1	2/26/2024 5:21 AM	File folder
📁 multipleTraces-syntheticSmalltrace-2	2/26/2024 5:21 AM	File folder
📁 multipleTraces-syntheticSmalltrace-3	2/26/2024 5:21 AM	File folder

multipleTraces-syntheticLarge (has 3)

📁 multipleTraces-syntheticLargetrace-1	2/26/2024 5:21 AM
📁 multipleTraces-syntheticLargetrace-2	2/26/2024 5:21 AM
📁 multipleTraces-syntheticLargetrace-3	2/26/2024 5:21 AM

multipleTraces-RubelMultiTrace (has 100)

RubelMultiTracetrace-1	2/26/20
RubelMultiTracetrace-2	2/26/20
RubelMultiTracetrace-3	2/26/20
RubelMultiTracetrace-4	2/26/20
RubelMultiTracetrace-5	2/26/20
RubelMultiTracetrace-6	2/26/20
RubelMultiTracetrace-7	2/26/20
RubelMultiTracetrace-8	2/26/20
RubelMultiTracetrace-9	2/26/20
RubelMultiTracetrace-10	2/26/20
RubelMultiTracetrace-11	2/26/20
RubelMultiTracetrace-12	2/26/20
RubelMultiTracetrace-13	2/26/20
RubelMultiTracetrace-14	2/26/20
RubelMultiTracetrace-15	2/26/20
RubelMultiTracetrace-16	2/26/20
RubelMultiTracetrace-17	2/26/20
RubelMultiTracetrace-18	2/26/20
RubelMultiTracetrace-19	2/26/20
RubelMultiTracetrace-20	2/26/20
RubelMultiTracetrace-21	2/26/20
RubelMultiTracetrace-22	2/26/20
RubelMultiTracetrace-23	2/26/20

Example for trace 63

- rubelmultitrace-audio-membus
- rubelmultitrace-cache0-cache1
- rubelmultitrace-cache0-cpu0
- rubelmultitrace-cache0-membus
- rubelmultitrace-cache1-cpu1
- rubelmultitrace-cache1-membus
- rubelmultitrace-gfx-membus
- rubelmultitrace-membus-uart
- rubelmultitrace-membus-usb
- rubelmultitrace-mem-membus