VAST 2025 MC3 Data Description

Graph Description

- File name: MC3_graph.json
- Directed Graph
- 1159 nodes
- 3226 edges
- The graph format is a JSON format generated by Python's network.node_link_data() function. It can be loaded to as a NetworkX object using the corresponding node_link_graph() function.
    - `G = json_graph.node_link_graph(json_data, edges="edges")`
- The root-level JSON object consists of graph-level properties specifying that it is directed graph, a "nodes" key which holds the list of nodes, and a "edges" key which holds the list of edges.
- Nodes are one of three types:
    - Entity
    - Event
    - Relationship

Node Attributes

A full description of node attributes is described within the corresponding MC_schema.json file. Each Entity, Event and Relationship node shares a common set of *common_attributes*, each of the possible subtypes outlined within Table 1 below also possess a unique set of *specific_attributes* both of which are outlined in full within the MC3_schema.json file.

Entity Subtypes	**Event Subtypes**	**Relationship Subtypes**
Person	Monitoring	AccessPermission
Organization	Assessment	Operates
Vessel	VesselMovement	Colleagues
Group	Enforcement	Suspicious
Location	TourActivity	Reports
	Collaborate	Jurisdiction
	TransponderPing	Unfriendly
	HarborReport	Friends
	Criticize	

Table 1: Node Subtypes

Edge Attributes

Edge attributes are additionally described within the MC_schema.json file along with their counts, with the matrix of possible edges outlined below within Table 2.

		Target Node		
		Entity	**Event**	**Relationship**
Source Node	**Entity**		Entity initiates or participates in an Event	Entity is a source of Relationship
	Event	Event targets or involves an Entity	**Communication Event** providing evidence for an Event	**Communication Event** providing evidence for a Relationship
	Relationship	Entity is the target of a relationship		

Table 2: Node-Edge-Node Matrix

In general, Entities are connected to other Entities via an Event or Relationship node. The one exception to this is the Communication Event subtype, which is additionally linked to either an Event or Relationship node. This is fully described within the following Graph Generation Process section.

Graph Generation Process

Clepper diligently recorded all intercepted radio communications over the last two weeks. With the help of his intern, they have analyzed their content to identify important events and relationships between key players. The result is a knowledge graph describing the last two weeks on Oceanus.

He demonstrated how to process two messages to his intern as an example, which is provided below:

Message 1

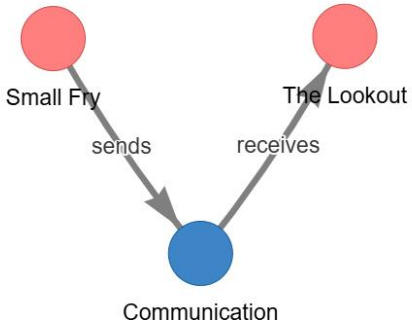
Clepper notes the following message:

> *"Hey Lookout, Small Fry here! I'd love to join for wildlife watching this afternoon. What time were you thinking? I can bring my new spotting scope that The Accountant helped me get for my internship."*

Step 1.

- He determines that it is sent from Small Fry to Lookout
- And adds a Communication node between with its content and the time it was sent

Entity is red
Event is blue
Relationship is yellow



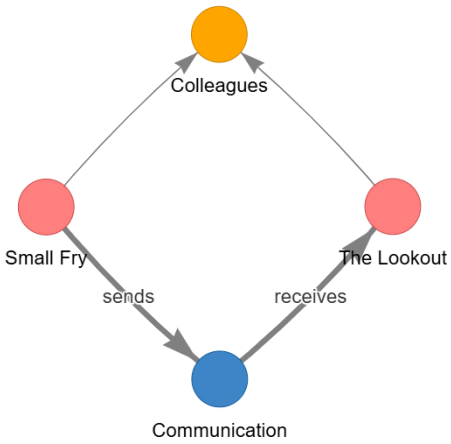
Supertype is entity..Person is red
Entity to Comm Event
Comm Event to Entity.

Communication node is the Comm Event

Colleague node is the Comm Relationship

Step 2.

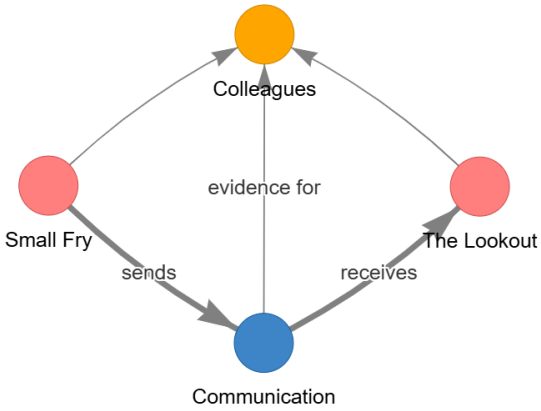
- After some thought, he determines that based on the message Small Fry and Lookout are likely colleagues
- A Colleague relationship node is then added between them



Step 3.

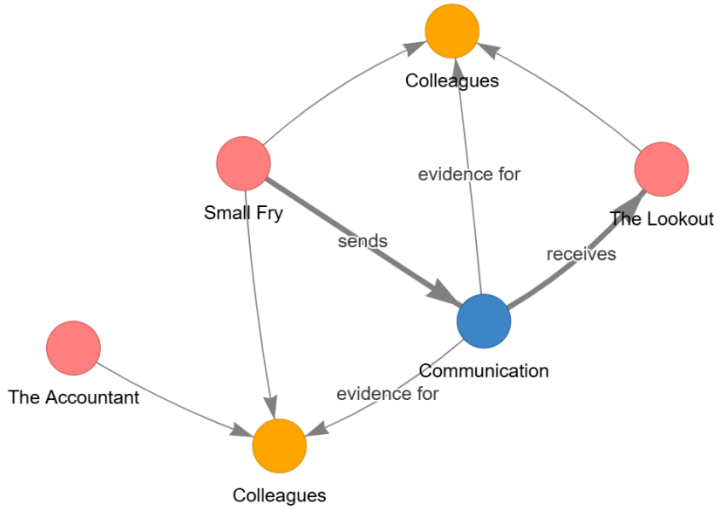
- The Communication node is linked to the Colleague relationship, labeled as evidence for

Comm Event additionally linked to Relationship node



Step 4

- Finally, from the same message he determines that Small Fry and The Accountant are also likely colleagues
- A Colleague relationship is then added between them (as per step 2)
- Along with an edge to the Communication node with the label evidence for (as per step 3)



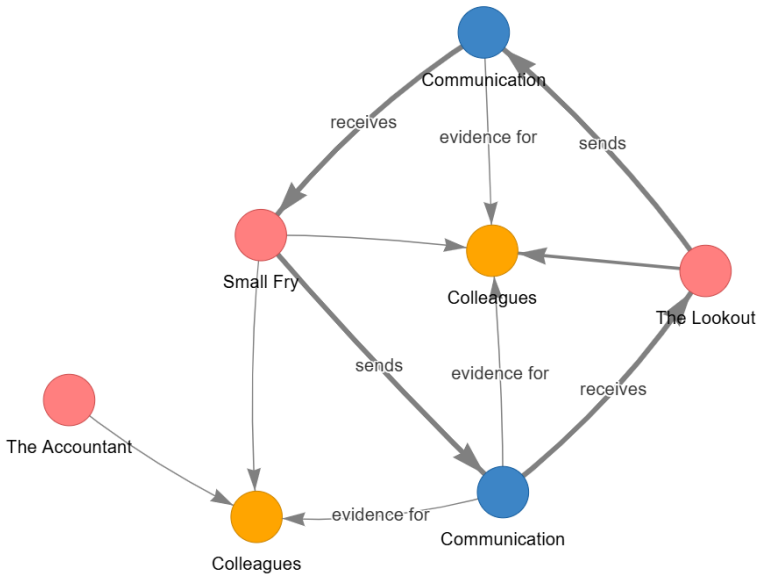
Message 2:

Clepper now records a new message:

> *"Hey Small Fry, The Lookout here! I'm thinking around 3pm at the eastern boundary. Perfect tide for spotting those herons. Bring that scope!"*

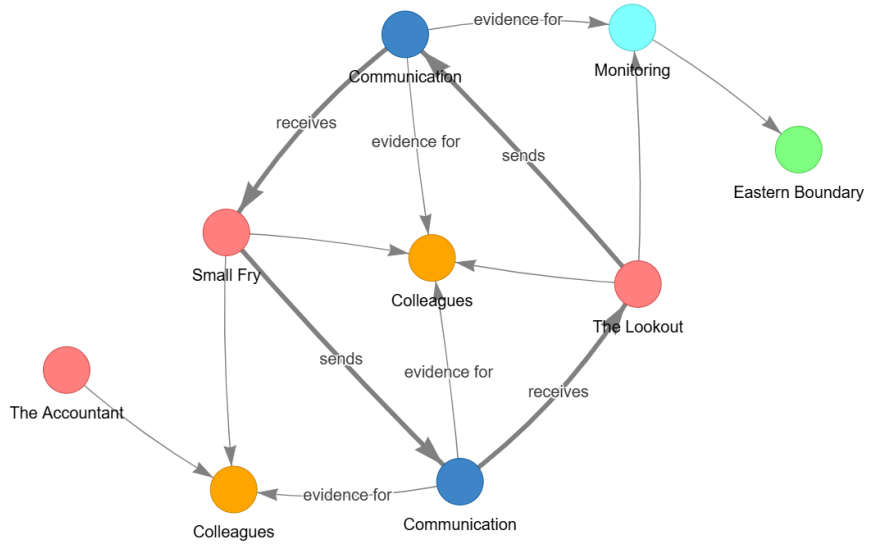
Step 5

- He determines that it is sent from Lookout to Small Fry, and adds a new Communication node between them
    - Note the direction of the edges is based on who sends or receives the message
- The message again indicates that Lookout and Small Fry are colleagues
- The new Communication node is linked to the existing Colleague relationship between Lookout and Small Fry
    - This reinforces the evidence for this relationship



Step 6

- Clepper notes that The Lookout seems to be monitoring the Eastern Boundary based on the second message.
- As such, he adds a Monitoring relationship between The Lookout and Eastern Boundary
- Along with an edge to the Communication node, with the label evidence for



Clepper and his intern continued this workflow until all 584 messages were processed, and the output knowledge graph was complete.

A note regarding directional edges

- For two specific relationships (Colleagues and Friends), edges point from both entities towards the relationship node as the relationship is not directional
    - e.g. both The Lookout and Small Fry are colleagues of each other (see step 2 and 5)
- For all other relationships and events, the direction of the edge describes their ordering
    - e.g. the Monitoring node has an edge from The Lookout, and an edge towards Eastern Boundary
    - meaning The Lookout is monitoring the Eastern Boundary, not the other way around

Disclaimer

- Events and Relationships have been inferred by Clepper and his Intern using the recorded communications to the best of their ability, yet there may be errors or contradictory information
- Use only data from this mini challenge to answer the questions. Data may be inconsistent between challenges. No external sources are required to answer the questions posed.
- All the provided data is synthetic. Resemblance to real people or situations is incidental.