# Transport Tycoon - Part 2

It is the time to open up the black-boxes without imposing too much constraints on the implementation details. Let's modify the solution so that it would log important events in the following format:

* Log entries are in JSON, one JSON object per line
* Optional text comments could start with the #, they are ignored.

We need to **log an entry when the important domain events happen: transport departs and when it arrives**.   
A single line in the log might look like the one below. It is pretty-printed to look nice, normally it would be **one line**:

{  
 "event": "DEPART", # type of log entry: DEPART or ARRIVE  
 "time": 0, # time in hours  
 "transport\_id": 0, # unique transport id  
 "kind": "TRUCK", # transport kind  
 "location": "FACTORY", # current location  
 "destination": "PORT", # destination (only for DEPART events)  
 "cargo": [ # array of cargo being carried  
 {  
 "cargo\_id": 0, # unique cargo id  
 "destination": "A",# where should the cargo be delivered  
 "origin": "FACTORY"# where it is originally from  
 }  
 ]  
}

Given that file, we could do the following things with our event logs:

1. Compare the reasoning of our solution to the reasoning from the another solution (even though they could be in different languages).
2. Feed it to the <https://github.com/Softwarepark/exercises/tree/master/transport-tycoon/trace/> script, that will convert this log to Chrome Trace Viewer format file (also JSON, but a different format).   
   That file could be loaded in Chrome at [chrome://tracing](chrome://tracing/) to display the outline of our travel.
3. Visualize the journey with the visualization tool <https://github.com/Nagelfar/transport-tycoon/tree/enhancements/transport_visulazation>

## Task

* **Extend your solution** to print domain events.
* Run the domain event log through the trace.py converter and then **display in the Chrome Trace tool**.  
  Does the AABABBAB solution look right? Does it complete on the hour 29? What about ABBBABAAABBB?

Here is an example event log for the entire AB delivery:

# Deliver AB  
{"event": "DEPART", "time": 0, "transport\_id": 0, "kind": "TRUCK", "location": "FACTORY", "destination": "PORT", "cargo": [{"cargo\_id": 0, "destination": "A", "origin": "FACTORY"}]}  
{"event": "DEPART", "time": 0, "transport\_id": 1, "kind": "TRUCK", "location": "FACTORY", "destination": "B", "cargo": [{"cargo\_id": 1, "destination": "B", "origin": "FACTORY"}]}  
{"event": "ARRIVE", "time": 1, "transport\_id": 0, "kind": "TRUCK", "location": "PORT", "cargo": [{"cargo\_id": 0, "destination": "A", "origin": "FACTORY"}]}  
{"event": "DEPART", "time": 1, "transport\_id": 0, "kind": "TRUCK", "location": "PORT", "destination": "FACTORY"} {"event": "DEPART", "time": 1, "transport\_id": 2, "kind": "SHIP", "location": "PORT", "destination": "A", "cargo": [{"cargo\_id": 0, "destination": "A", "origin": "FACTORY"}]}  
{"event": "ARRIVE", "time": 2, "transport\_id": 0, "kind": "TRUCK", "location": "FACTORY"}  
{"event": "ARRIVE", "time": 5, "transport\_id": 1, "kind": "TRUCK", "location": "B", "cargo": [{"cargo\_id": 1, "destination": "B", "origin": "FACTORY"}]}  
{"event": "DEPART", "time": 5, "transport\_id": 1, "kind": "TRUCK", "location": "B", "destination": "FACTORY"} {"event": "ARRIVE", "time": 5, "transport\_id": 2, "kind": "SHIP", "location": "A", "cargo": [{"cargo\_id": 0, "destination": "A", "origin": "FACTORY"}]}  
{"event": "DEPART", "time": 5, "transport\_id": 2, "kind": "SHIP", "location": "A", "destination": "PORT"}

A screenshot of a cell phone

Description automatically generated