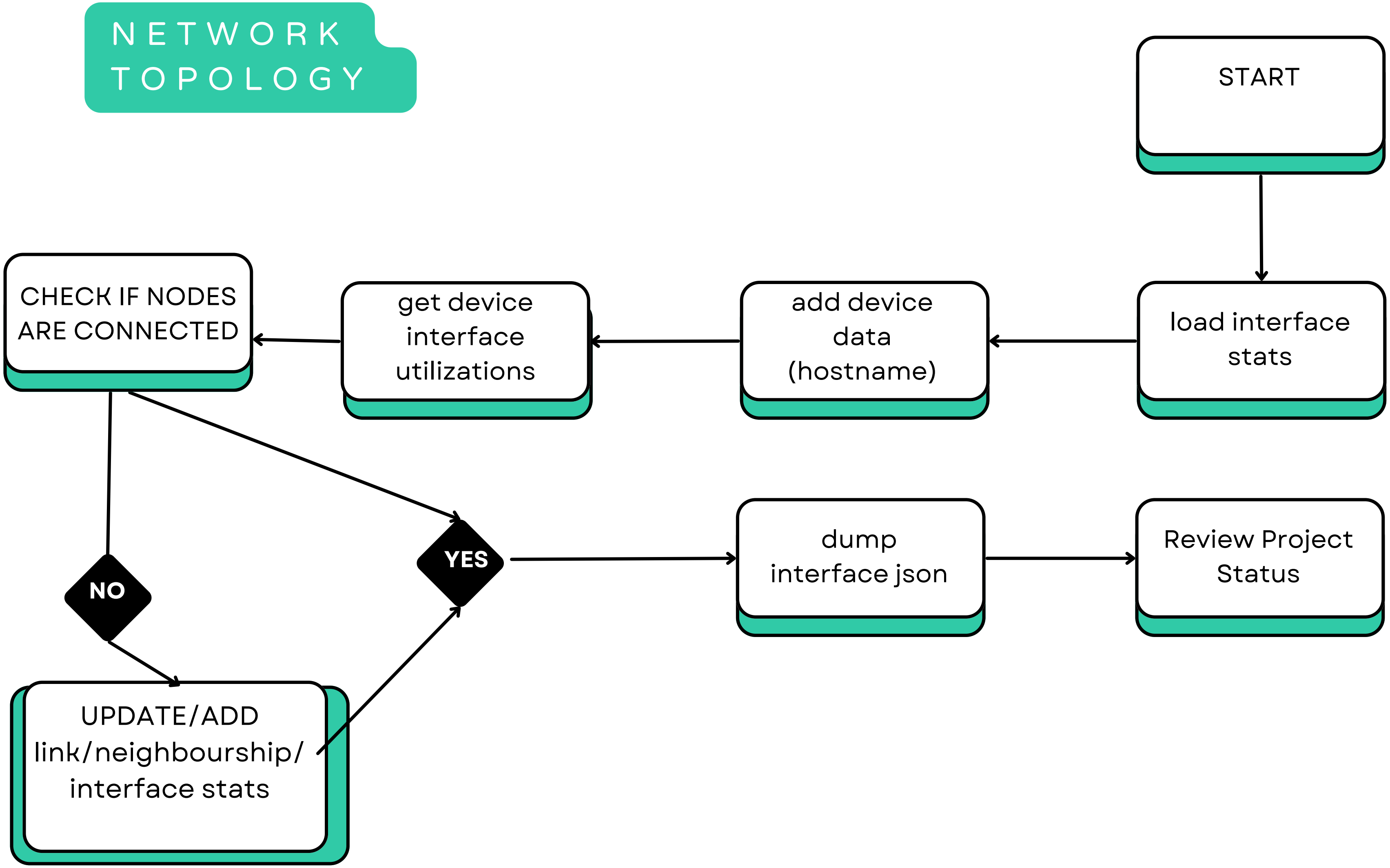
NETWORK TOPOLOGY



START

load interface stats

add device data (hostname)

get device interface utilizations

CHECK IF NODES ARE CONNECTED

NO

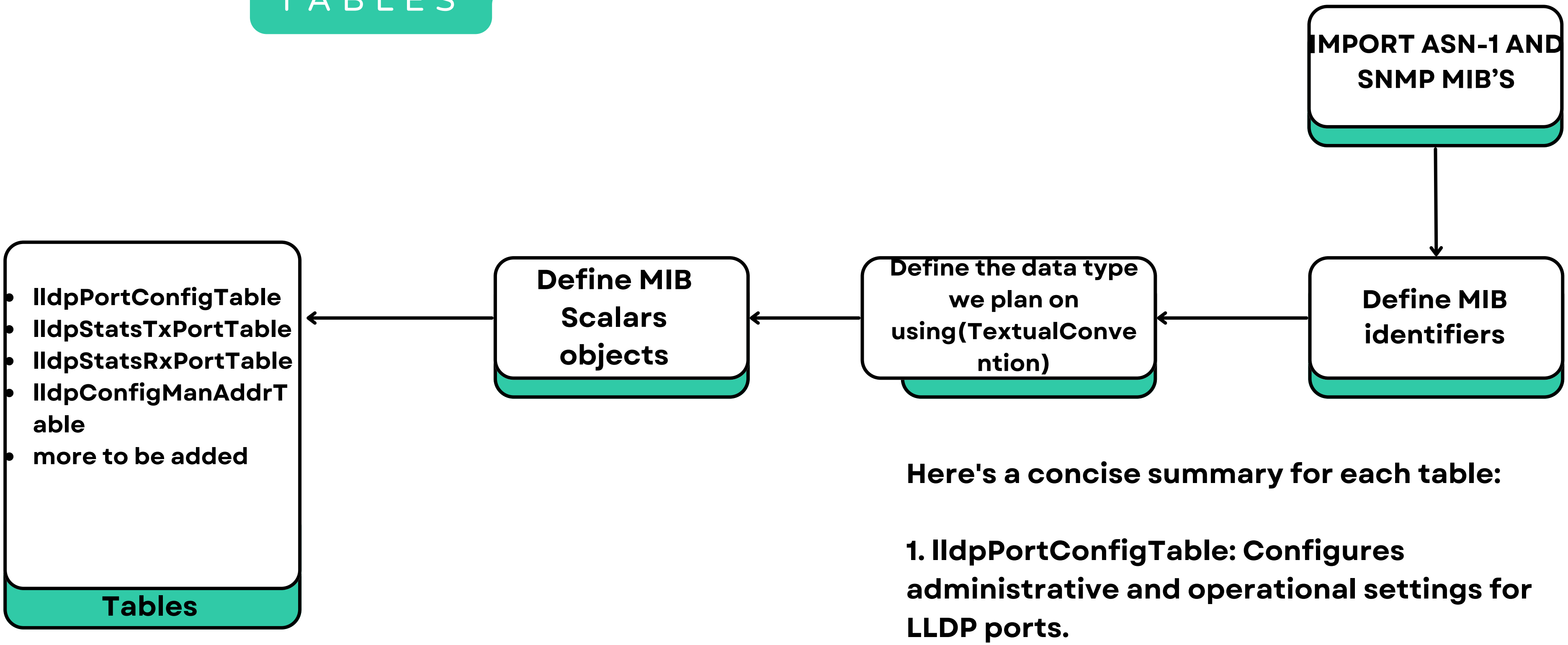
UPDATE/ADD link/neighbourship/ interface stats

YES

dump interface json

Review Project Status

LLDP-MIB TABLES



IMPORT ASN-1 AND SNMP MIB’S

Define MIB identifiers

Define the data type we plan on using(TextualConvention)

Define MIB Scalars objects

- lldpPortConfigTable
- lldpStatsTxPortTable
- lldpStatsRxPortTable
- lldpConfigManAddrTable
- more to be added

Tables

Here's a concise summary for each table:

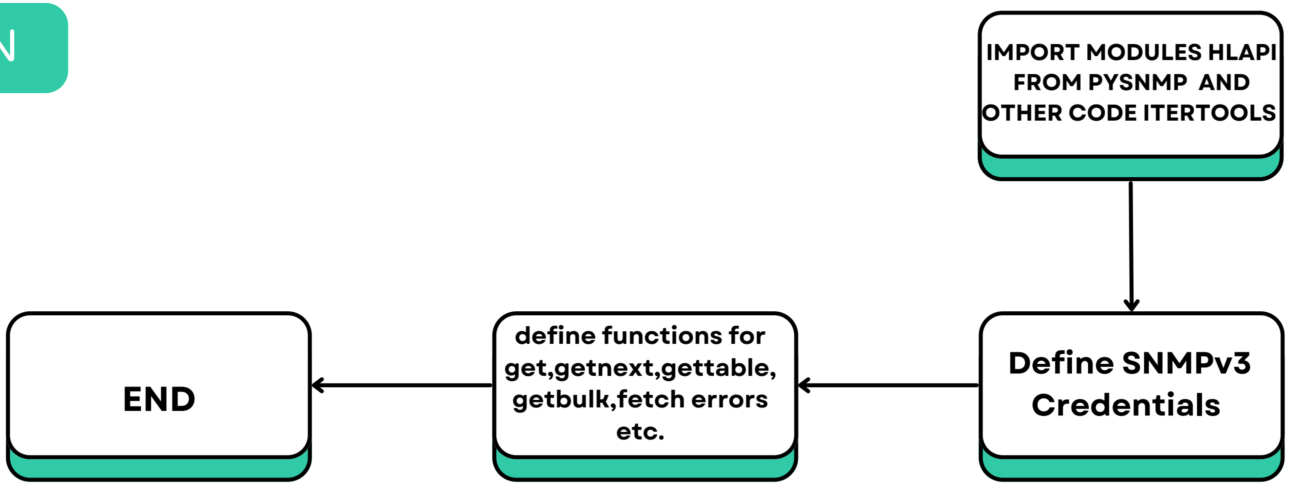
1. lldpPortConfigTable: Configures administrative and operational settings for LLDP ports.

2. lldpConfigManAddrTable: Specifies management addresses and their associated transmitting ports.

3. lldpStatsTxPortTable: Tracks statistics for LLDP frames transmitted by each port.

4. lldpStatsRxPortTable: Tracks statistics for LLDP frames received by each port.

SNMP EXTRACTION



IMPORT MODULES HLAPI FROM PYSNMP AND OTHER CODE ITERTOOLS

Define SNMPv3 Credentials

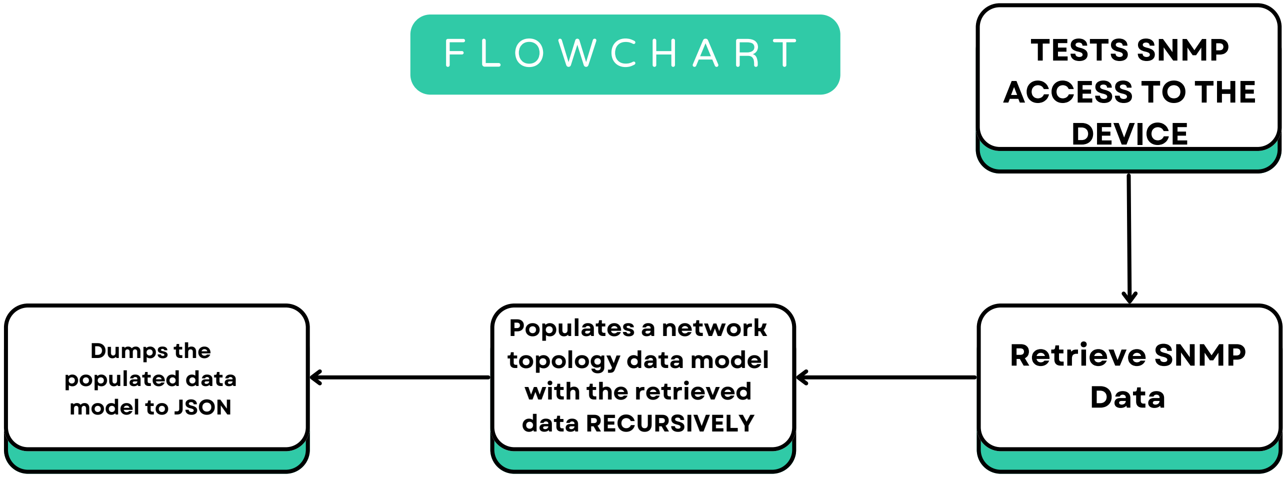
define functions for get,getnext,gettable, getbulk,fetch errors etc.

END

PARSING THROUGH THE DATA

1. PARSE ARGUMENTS : USE ARGPARSE TO HANDLE COMMAND-LINE ARGUMENTS, INCLUDING --VERSION AND --REPEAT.
2. CHECK REPEAT ARGUMENT:
3. IF REPEAT IS 0, CALL THE MAIN_WITH_ARGS() FUNCTION ONCE.
4. IF REPEAT IS GREATER THAN 0, ENTER A LOOP TO CALL MAIN_WITH_ARGS() AND THEN SLEEP FOR THE SPECIFIED NUMBER OF SECONDS BEFORE REPEATING.
5. INITIALIZE A NETWORKTOPOLOGY MODEL INSTANCE.
6. READ AND PARSE THE CONFIGURATION FILE WHICH WE WILL DEFINE
7. IF REQUIRED SECTIONS (DEVICES AND DEFAULT) ARE MISSING, LOG AN ERROR AND EXIT.
8. IF SECTIONS ARE PRESENT, PROCEED TO ITERATE OVER THE LISTED DEVICES.
9. FOR EACH DEVICE, LOG THE START OF ACCESS TESTING.
10. ATTEMPT TO GET THE DEVICE'S HOSTNAME USING SNMP.
11. ON SUCCESS, LOG THE DATA.
12. ON ERROR, LOG THE ERROR AND CONTINUE TO THE NEXT DEVICE.
13. IF SNMP ACCESS IS SUCCESSFUL, LOG THE CONNECTION SUCCESS AND PROCEED TO:
14. RETRIEVE THE INTERFACES TABLE USING SNMP.
15. ON SUCCESS, LOG THE INTERFACES DATA.
16. ON ERROR, LOG THE ERROR AND CONTINUE.
17. RETRIEVE THE LLDP TABLE USING SNMP.
18. ON SUCCESS, LOG THE LLDP DATA.
19. ON ERROR, LOG THE ERROR AND CONTINUE.
20. ADD THE RETRIEVED INTERFACES AND LLDP DATA TO THE NETWORKTOPOLOGY MODEL.
21. ONCE ALL DEVICES ARE PROCESSED, DUMP THE POPULATED TOPOLOGY MODEL TO JSON AND DELETE THE MODEL INSTANCE.
22. PRINT RETURN CODE: PRINT THE RETURN CODE AND EXIT THE SCRIPT. IF REPEAT WAS SPECIFIED, THIS PROCESS LOOPS.

JSON


FLOWCHART
TESTS SNMP ACCESS TO THE DEVICE
Retrieve SNMP Data
Populates a network topology data model with the retrieved data RECURSIVELY
Dumps the populated data model to JSON

- We will be using a queue to add node data and will be popping out edges recursively for the creation of the whole topology