



# Content

- Objectives
- DB Design Methodology
  - Readout network dataflow
  - Use cases
  - Entity relationship models
  - Table design
- Conf.DB & PVSS (JCOP)
- C routine example : use case 3 & 4
- Test Tool
- Conclusion & future works



# Objectives

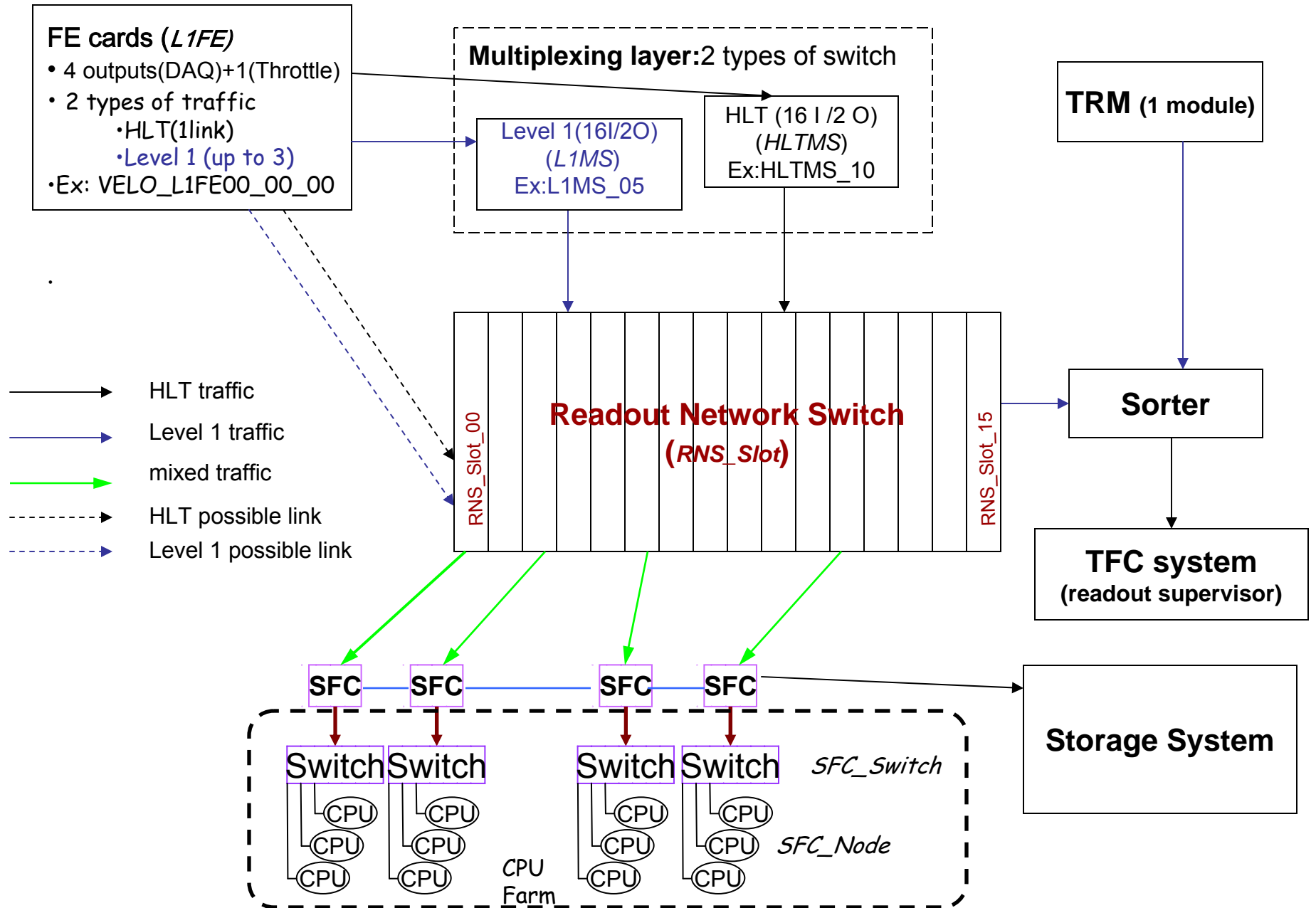
- Storage of all controllable devices from all sub-systems (electronics...)
  - static data (geographical position, links,...)
  - dynamic data (recipes, code to download...)
- Each device will be stored in the Conf.DB and linked to its connectivity counterpart.
- Implement tools: to populate, to configure, to maintain, to communicate with other existing tool such as PVSS.



# DB Design Methodology

1. Dataflow of the system
2. Define the **use cases** (scenarios)
3. Build **Entity relationship models**
4. Design tables

# ReadOutNetwork Data Flow





## Use cases

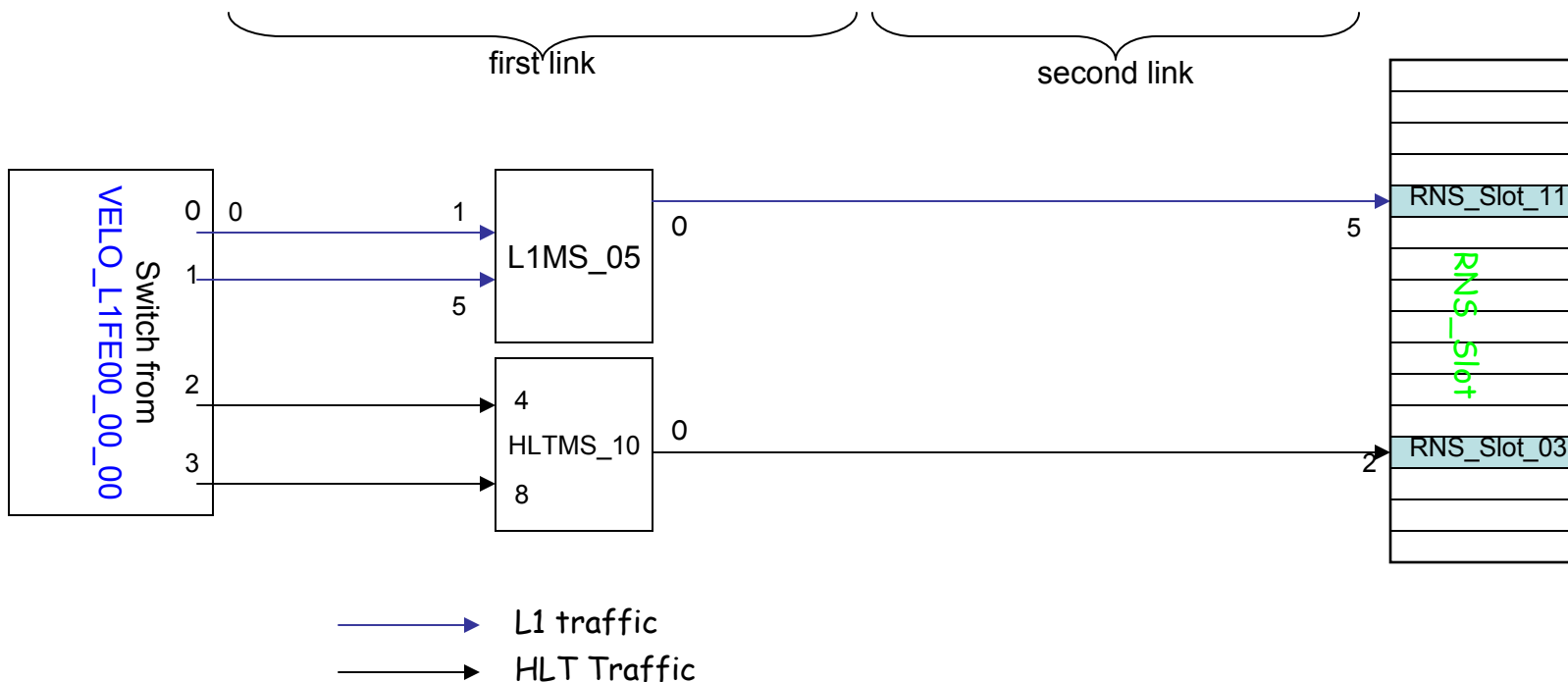
1. Define the properties of a switch  
A switch has a type (L1MS for example) and a name (L1MS\_05) and a hostname.  
Each switch type is characterized by a number of input and output ports.
2. Define the properties of a port
3. Find the adjacent switches (links) up or down connected to a given switch
4. Find all the connection paths between 2 devices
5. Check that a path exists between 2 extreme nodes (ie L1FE and SFC\_node)
6. Retrieve all the links belonging to a given band link number and vice versa.



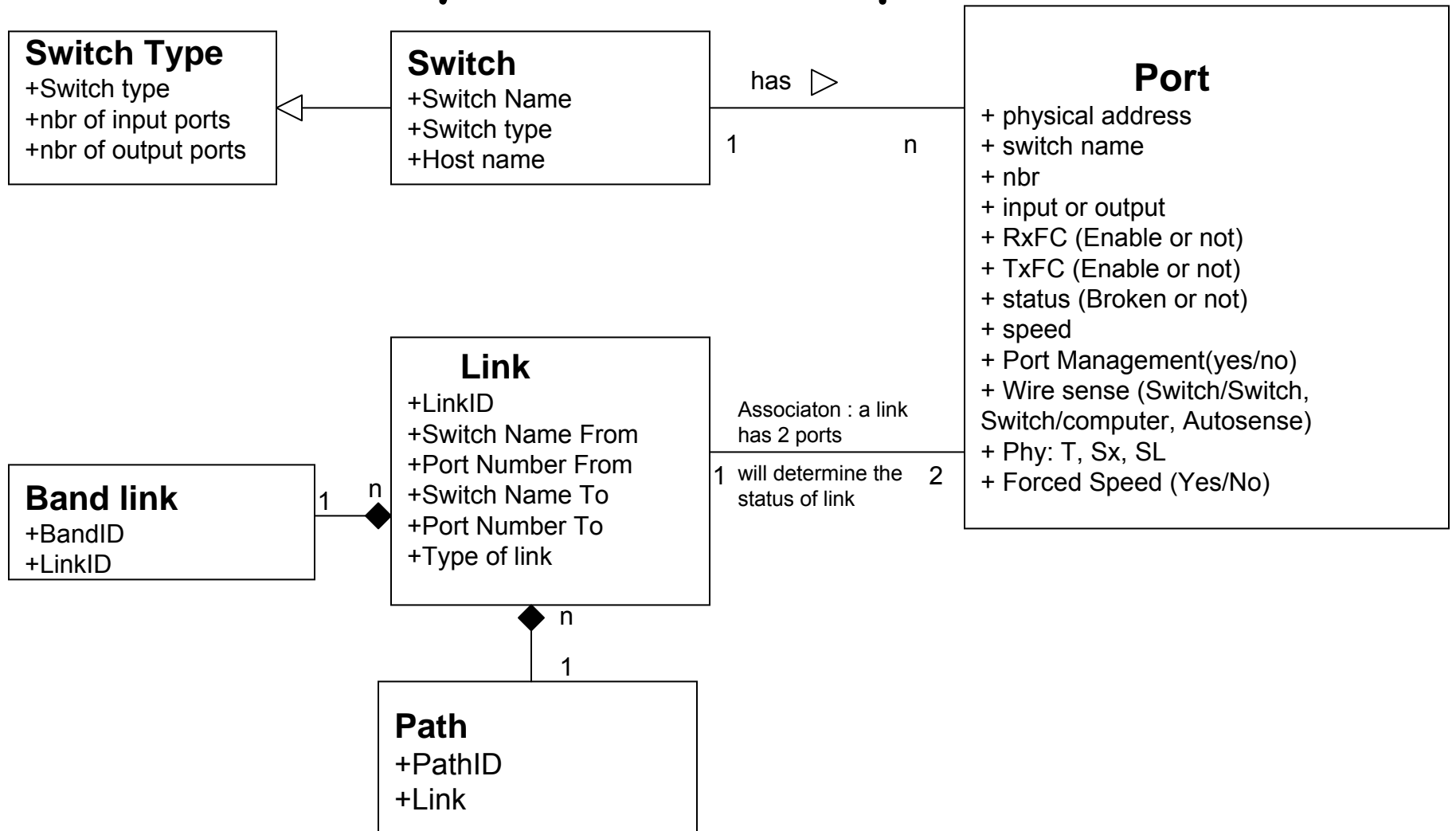
# Path concept

Example: I want to know **all** the possible connexion paths from **VELO\_L1FE00\_00\_00** to **RNS\_Slot** which carries **HLT Traffic**.

- path 1: [(VELO\_L1FE00\_00\_00, 2), (HLTMS\_10, 4), **HLT**] [(HLTMS\_10,0),(RNS\_Slot\_03, 2), **HLT**]→ Selected
- path 2 : [(VELO\_L1FE00\_00\_00, 3), (HLTMS\_10, 8), **HLT**] [(HLTMS\_10,0),(RNS\_Slot\_03, 2), **HLT**]→Selected
- path 3: [(VELO\_L1FE00\_00\_00, 1), (L1MS\_05, 5),**L1**] [L1MS\_05,0),(RNS\_Slot\_11, 5), **L1**]→ **Not Selected**
- path 4: [(VELO\_L1FE00\_00\_00, 0), (L1MS\_05, 1),**L1**] [L1MS\_05,1),(RNS\_Slot\_11, 5), **L1**]→**Not Selected**

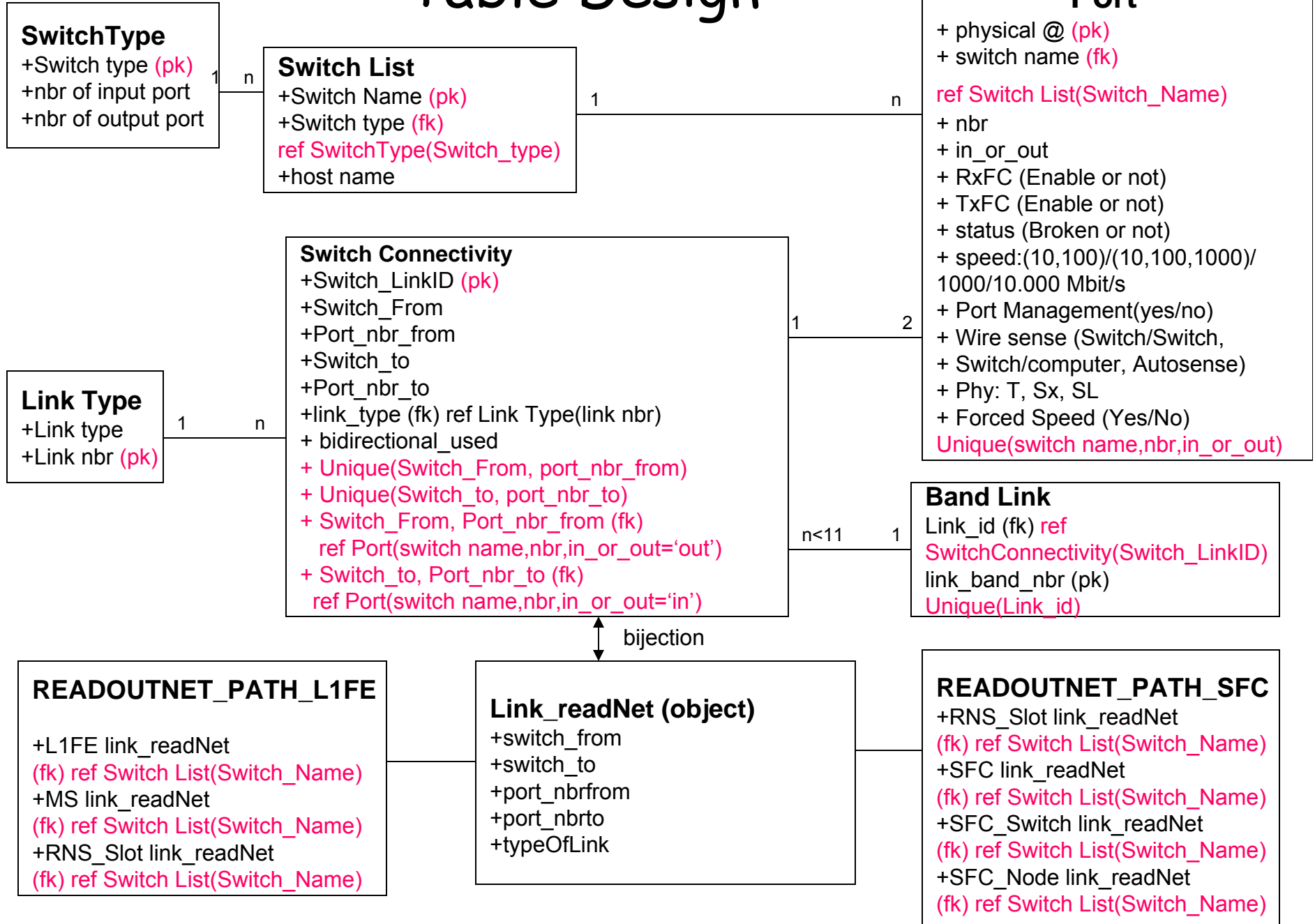


# Entity relationship models

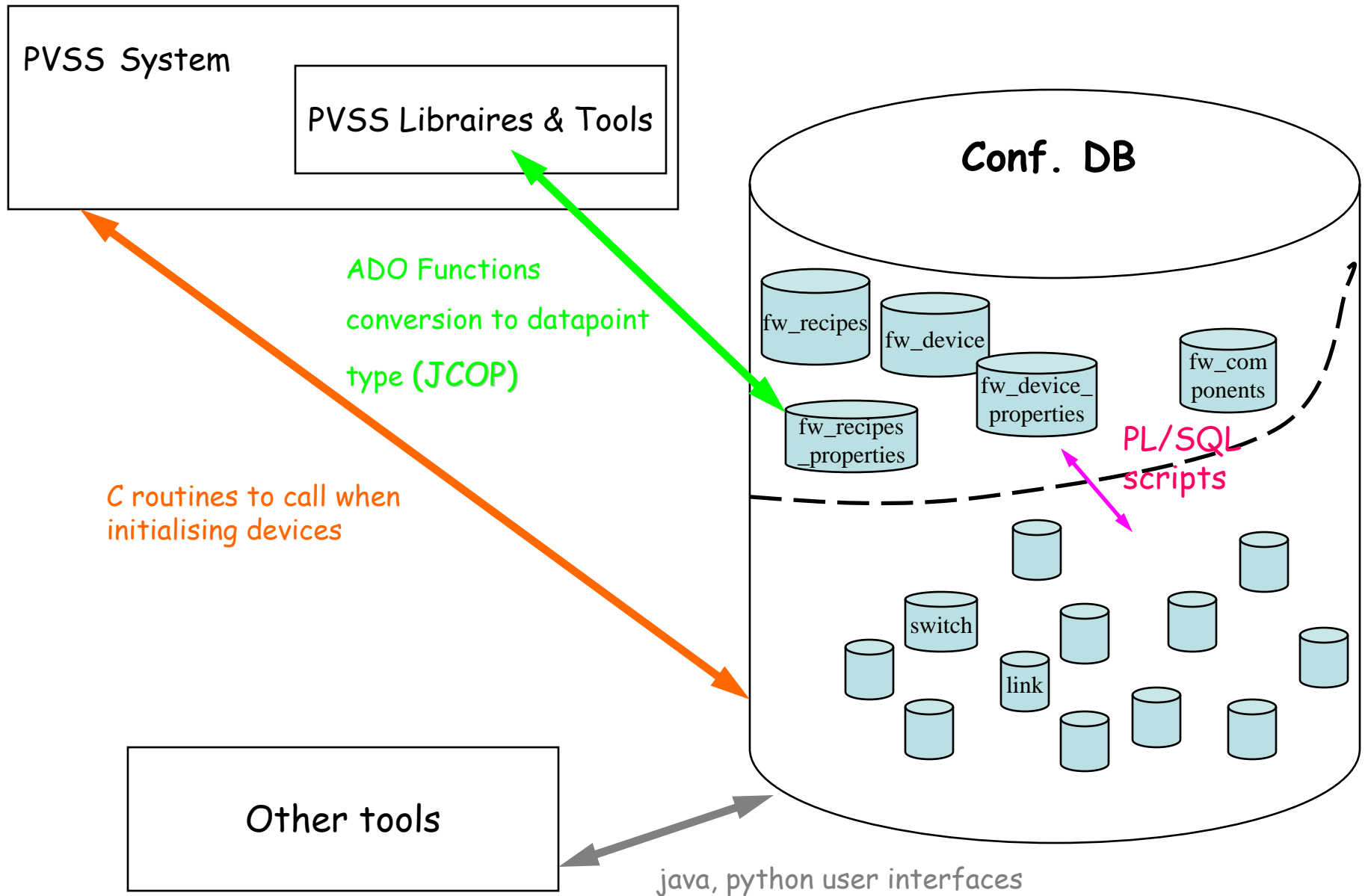




# Table Design



# Conf.DB & PVSS





## C routine example: use case 3 & 4

- **ProC/C++ precompiler**
- **Microsoft Visual C++**
- **struct Port** {char SwitchName\_FROM[21];int PortNb\_FROM;  
char SwitchName\_TO[21];int PortNb\_TO;struct Port \*next;};
- **Port\*\* test1** (char switch\_name[21], int type\_port, int link\_nature,  
char switch\_type[21], int adjacent\_link)
- Example: I want to know **all** the possible connexion paths  
from VELO\_L1FE00\_00\_00 to RNS\_Slot which carries  
**HLT Traffic**.  
test1('VELO\_L1FE00\_00\_00', 2 , 2 , 'RNS\_Slot' ,0)

# Test Tool

Enter the switch name  Check the type of link  L1 Traffic  HLT traffic  Not specified Select a switch type

Check 1 if you want to get the upstream connectivity, 2 to get the down stream.  1  2

Direct connectivities: check if yes and don't check a switch type

Switch Name From Port Nbr Switch To Port Nbr

Switch Name From	Port Nbr	Switch To	Port Nbr
VELO_L1FE00_07	1	RNS_Slot_00	7
RICHT_L1FE01_01	1	RNS_Slot_00	8
RICHT_L1FE01_02	1	RNS_Slot_00	8
RICHT_L1FE01_04	1	RNS_Slot_00	9
RICHT_L1FE01_05	1	RNS_Slot_00	9
VELO_L1FE00_02	1	RNS_Slot_00	2
VELO_L1FE00_03	1	RNS_Slot_00	3
VELO_L1FE00_04	1	RNS_Slot_00	4
VELO_L1FE00_05	1	RNS_Slot_00	5
VELO_L1FE00_05	1	RNS_Slot_00	7

**Dialog**

To view the full path

VELO_L1FE00_02_02 port nbr:1	L1M5_02 port nbr:2
L1M5_02 port nbr:0	RNS_Slot_00 port nbr:2



## Conclusion & future works

- First Table design for TFC & Readout system
- Integrate this design with JCOP tools
- Take into account versioning, history
- Debug C routines & check their compactness.