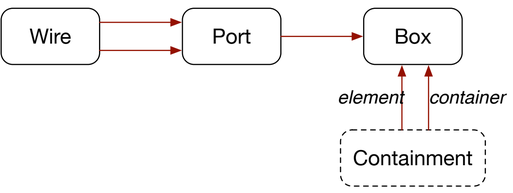
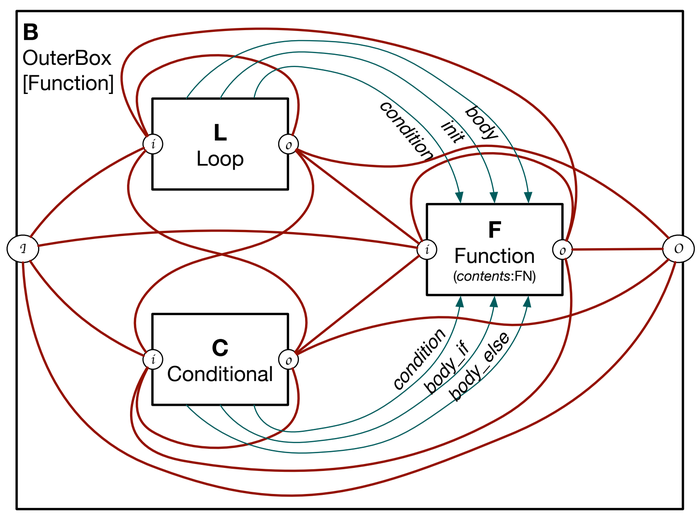
GroMEt Base Schema $\mathcal{G}$

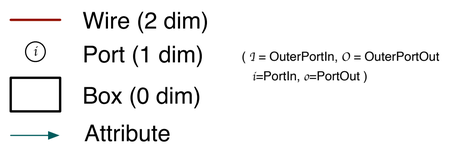


GroMEt FunctionNetwork $\mathcal{FN}$ v0.1.1

Instance of Base $\mathcal{FN} \in Set^{\mathcal{G}}$

(Instance of Base Schema as a category-theoretic Wiring Diagram)



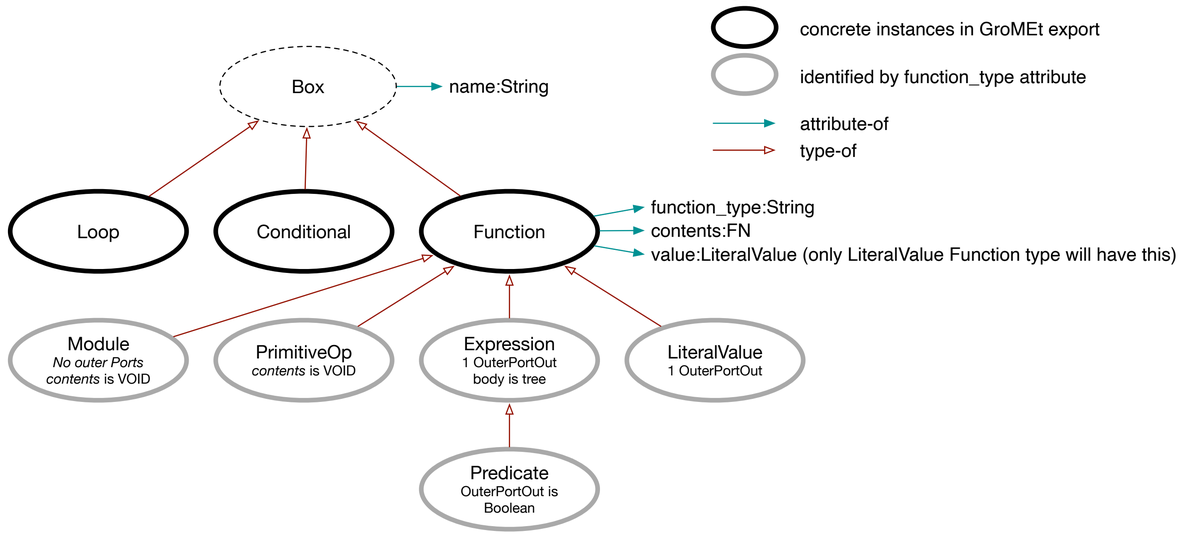


($\mathcal{I}$ = OuterPortIn, $\mathcal{O}$ = OuterPortOut
i=PortIn, o=PortOut)

Note (2022-07-13):
The `wl_cargs`, `wl_ioargs`, `wl_iiargs` and `wc_cargs`
Wires are NOT drawn here

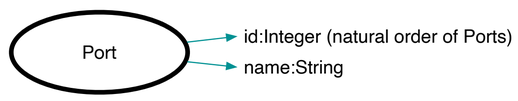
Box Types

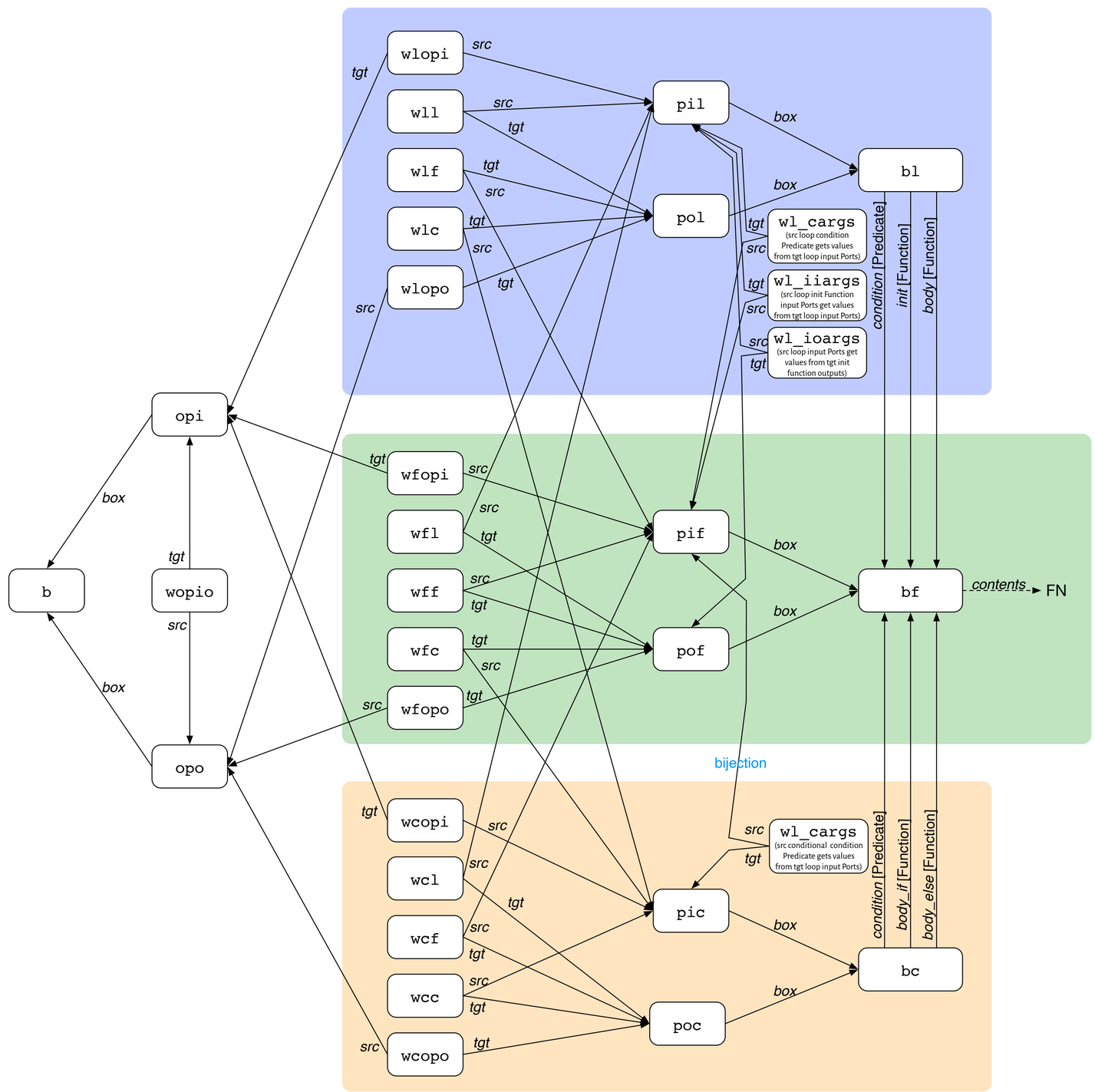
Functions have their own hierarchy that will not be explicitly represented in the GroMEt export,
but instead represented by a *function_type* (String) attribute associated with the Function Box.
The most general type is Function, which includes no constraints on number of input and output Ports.
When a box has FN contents, the Box type will be the type of the outer box of the FN.
These structural constraints are not enforced by the schema but would need to be validated
These Box types will also have different attributes, as represented by the green arrows pointing at the attribute names:types.
The smaller text within the ovals describe some additional constraints imposed by a given type.
All child Box types inherit attributes of parents (even if hey don't use/fill them) as well as structural/contents constraints.
Module is special and only appears that the "top" level FN.



$\int_{\mathcal{G}} \mathcal{FN}$ Derived schema for $Set^{\mathcal{G}/\mathcal{FN}}$

(Can be directly interpreted as an ER (Entity-Relation) database schema)





Notes:

Naming Conventions: All type names start with their schema syntactic type:
- op = OuterPort
  - opi = OuterPort Input Ports, opo = OuterPort Output Ports
- p = Port
  - the next Port name letters:
    - i = in, o = out ; followed by l = Loop, f = Function, c = Conditional
- w = Wire
  - For 3-lettered Wire names, the last two letters are:
    - src -> tgt, with l = Loop, f = Function, c = Conditional
- b = Box
  - followed by l = Loop, f = Function, c = Conditional

Special Wire cases:
(1) **wopio**: Wire from OuterPortIn to OuterPortOut (i.e., passthrough)
(2) **wl_cargs** and **wc_cargs** represent a map (as Wires) from the arguments of the condition (**cargs**) Predicate (src) of the **L**oop or **C**onditional to the PortIn's (tgt) of the Loop or Conditional Box.

Similarly, **wl_iiargs** and **wl_ioargs** represent a map from the input and output (respectively Ports of the loop init Function to the loop input Ports.

The background color patches group Ports, Wires and Boxes that are present depending on whether Function (green), Loop (blue) or Conditional (orange) is present.