

# PIPS

Fabien Coelho  
CRI, M&S, MINES-ParisTech

September 2, 2024

## 1 Private data structures for FREIA -SPOC

WORK IN PROGRESS...

A special DAG for FREIA (SPOC) code generation.

The *Generic Oriented Graph* provided by PIPS is both too minimal and too generic for my purpose.

Lists are used instead of sets (for preds, succs, vertices) to help determinism.

```
Import entity from "ri.newgen"
```

```
Import statement from "ri.newgen"
```

```
pstatement = persistent statement + empty:unit
```

```
vtxcontent = optype:int x opid:int x source:pstatement x inputs:entity*  
x out:entity
```

The operation (*optype*, *opid*) may not be commutative, thus the order of input images (*inputs*) is significant, and some adjustment may be necessary when mapping onto the hardware.

There is a lot of information kept here, possibly redundant (?), because it is/may be needed for regenerating the code.

Somehow I need in/out effects from within the sequence, or possibly use-def chains?

```
dagvtx = content:vtxcontent x succs:dagvtx*
```

What about the first and last vertices? They can be deduced because they do not have predecessors or successors?

```
dag = inputs:dagvtx* x outputs:dagvtx* x vertices:dagvtx*
```

The order of *vertices* in the list is the inverse of their occurrence in the sequence, so that the first to produce an image is indeed to one that produced it if a variable is reused.

List *inputs* stores *external* inputs to the DAG. List *outputs* is for the nodes of the DAG which produce an image expected outside.