Text using $ syntax

```
defpart TOP {
  inputs: "start"
  outputs: "result"
  parts: ${box1} : hello, ${box2}: hello, ${box3} : string-join
  wires:
    ${0}: ${self}.start -> ${box1}.start, ${box2}.start
    ${1}: ${box1}.s -> ${box3}.a
    ${2}: ${box2}.s -> ${box2}.b
    ${3}: ${box3}.c -> ${self}.result
}

defpart hello {
  inputs: "start"
  outputs: "s"
  }

defpart world {
  inputs: "start"
  outputs: "s"
}

def Leaf string-join {
  inputs: "a", "b"
  outputs: "c"
}
```

Differences Between Diagram and Text

- Text needs a Name for every Part. This is implicit in the diagram (the “position” of a box is like a name)
- Wires are explicit.
- Later, both, Parts and Wires need to be instantiated (in the Loading Phase)