Customisation of Rational Rose

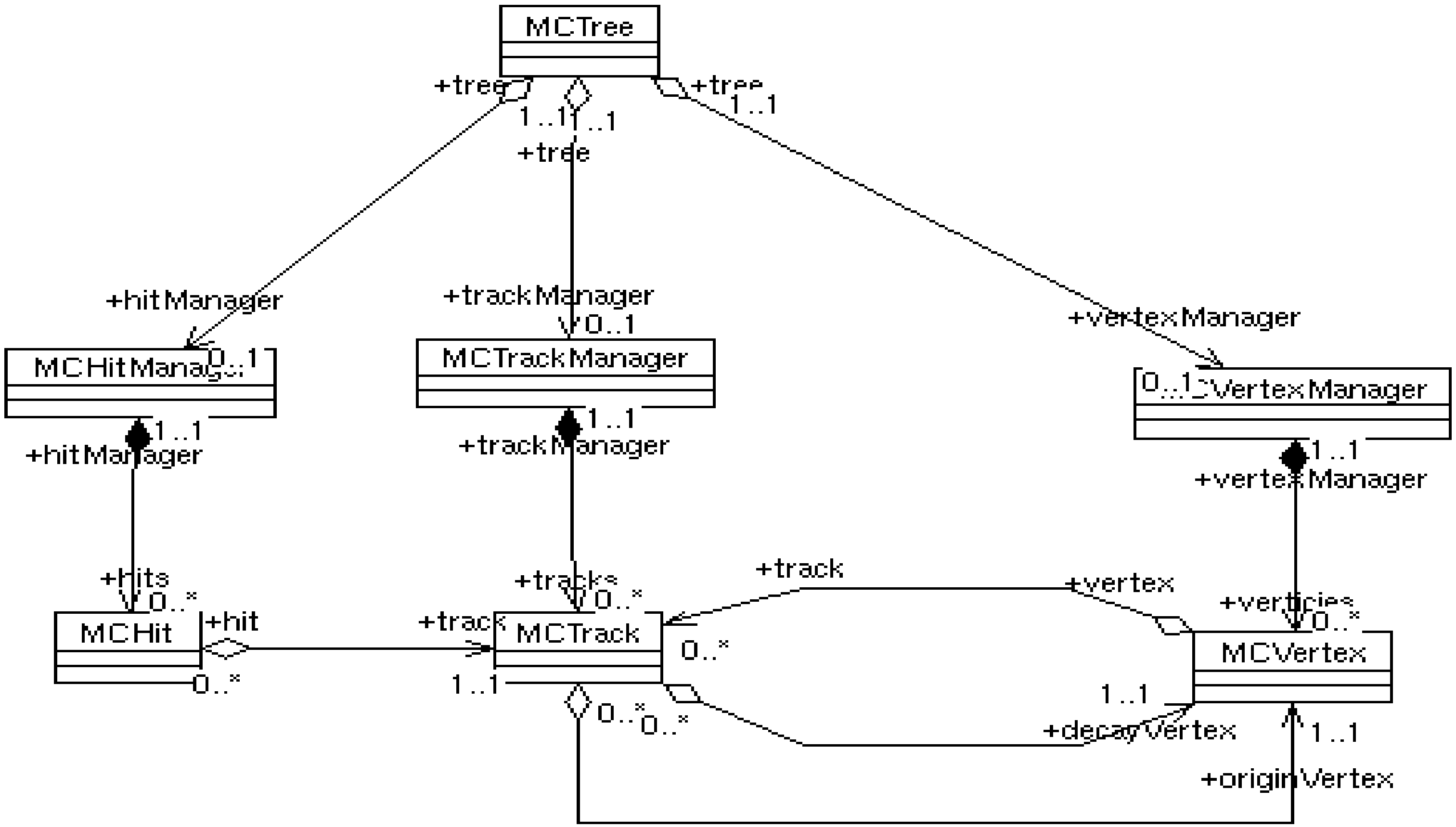
Why Customise Rose?

- **Nico Neufeld's results on rich detector design using rose (presented last week)**
  - **1 he found rose useful**
  - **2 he said it took time to get used to Rose**
- **Rose customisation is necessary to generate code from the model**
- **If a customisation file for Rose could be created:**
  - **Anyone could use Rose to produce code**
  - **All code produced will conform to the LHCb C++ conventions**

What Have I Done

- **Starting on Friday with**
  - **Nico's properties file and**
  - **MonteCarloTree class diagram (by Andrei Tsaregorodtsev and Pavel Binko)**
- **I have tried to customise the properties file for LHCb and produce code as close as possible to the LHCb coding conventions**
- **Results**
  - **I have managed to produce working code that is close to the C++ coding conventions**
  - **However to do this I needed two properties files**

The Montecarlotree UML Diagram



What Is the Code Like

- **Full of strange comments**
  - **I have not yet succeeded in removing the unnecessary comments**
- **Difficult to get to exact LHCb coding conventions**
  - **Examples**
    - **So far can not define get functions for data members as getData. I can only manage getdata**
    - **To accomplish forward declarations you need to use two property files in your code generation**
- **Apart from that the code is fine...**

Conclusions

- It is possible to make a master property file (or files)
- My most recent effort is at ~/binko/event/etc/RoseLHCbproperties.Prp
- Reverse engineering returns the correct UML model for the code
  - But it takes a little work to generate new code from this model (round trip engineering)
- The work is continuing
  - The generated code keeps improving
  - ~/Users/oreilly/public for all the generated files