

---

Subject: Re: Usage of exceptions in CINT  
Posted by [Philipp Bielefeldt](#) on Wed, 24 Feb 2016 17:46:13 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Maybe I do not get the point in your question accurately... But the object thrown is a GException, and the referenced line RKTrackRep.cxx:1215 reads

```
GException exc(ssstream.str(), __LINE__, __FILE__);  
exc.setFatal();  
throw exc;
```

It is a bit unclear to me what you mean with "outside of GenFit". I am still working with CINT linked against Genfit... so it is not really "outside", but maybe I am misled here. I cannot see how this could be a default/empty/... exception. As far as I can see some thing bewildering happens with the thrown exception. And by the way: I could reproduce the same weird behaviour in a "neatly compiled" class, where I wrapped up the ->processTrack like so:

```
bool GExecution::cwProcessTrack(GFAbsFitter* aFitter, GFTrack* aTrack){  
    fStatus = true;  
    std::cout << "GExecution::cwProcessTrack... Status is " << fStatus << std::endl;  
  
    try{  
        aFitter->processTrack(aTrack);  
    }catch(const GException& e){  
        fStatus = false;  
        std::cout << "[WARN  ] Exception in track procession -- ProcessTrack set to false" <<  
std::endl;  
        std::cerr << e.what() << std::endl;  
    }  
  
    std::cout << " return status is " << fStatus << std::endl;  
    return fStatus;  
}
```

but again, in case an exception is thrown, the exception's content appears on the cli, whilst the catch block itself seems to be not executed, it looks like so:

```
GExecution::cwProcessTrack... Status is 1  
GException thrown with excString:  
RKTrackRep::RKutta ==> momentum too low: 3.53823 MeV  
in line: 1215 in file:  
/home/bielefeldt/g2fopiroot/GenfitTools/trackrep/RKTrackRep/RKTrackRep.cxx  
with fatal flag 0  
return status is 1
```

Most likely, I am doing some thing very incorrect in the way I implement exception handling,

but I just cannot find my flaw.

---