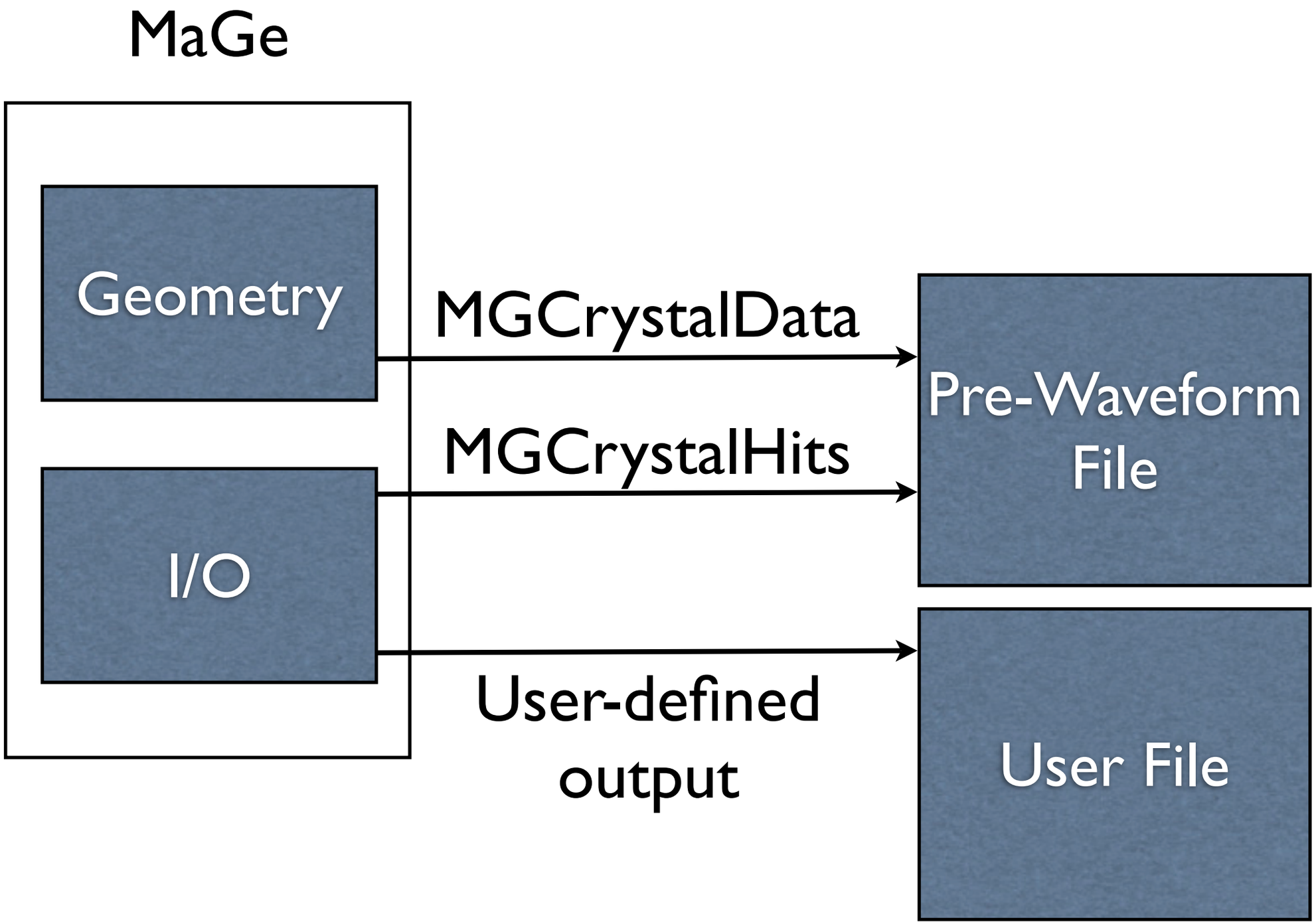
Pulse Shape Simulation Infrastructure

Jason Detwiler

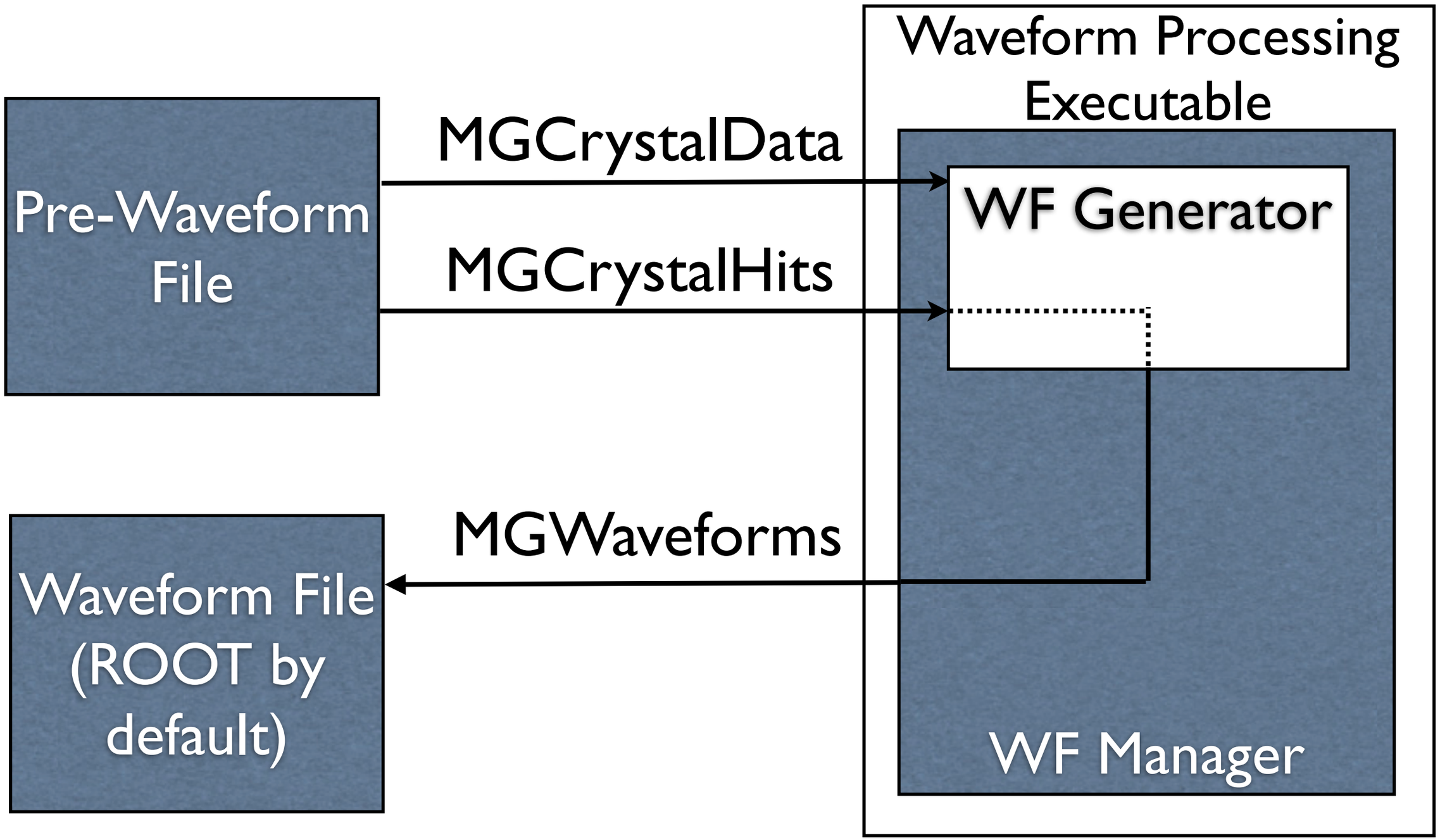
MaGe Workshop 2010

18-20 January, Munich

Processing Overview



Processing Overview



MGVWaveformGenerator

```
MGVWaveformGenerator(MGCrystalData* crystalData, size_t wfLen = 512, double
sampFreq = 1.0*CLHEP::GHz, MGWaveform::EWFType type = MGWaveform::kCharge,
double preTriggerTime = 0.0*CLHEP::ns);

virtual void Initialize() = 0;
virtual const char* GetName() = 0;
virtual void GenerateWaveformsForHit(double x, double y, double z, double t, double E,
MGWaveformCollection* cumulativeOutput) = 0;

protected:
  MGCrystalData* fCrystalData;
  size_t fWFLength;
  double fSamplingFrequency;
  MGWaveform::EWFType fWFType;
  double fPreTriggerTime;
```

Example Generator: MGWFGenGaussResponse

```
MGWFGenGaussResponse::MGWFGenGaussResponse(MGCrystalData* crystalData, double
preTriggerTime, double sigma, double ePair) :
MGVWaveformGenerator(crystalData, 512, 1.0*GHz, MGWaveform::kCharge, preTriggerTime)
{
  fSigma = sigma;
  fEPair = ePair;

  if(!fCrystalData->IsCentralContact(0)) {
    MGLog(warning) << "Bogus iCentralContact" << endlog;
  }
}

void Initialize() {}
inline const char* GetName() { return "WFGenGaussResponse"; }
```

Example Generator: MGWFGenGaussResponse

```
void MGWFGenGaussResponse::GenerateWaveformsForHit(double x, double y,
double z, double t, double E, MGWaveformCollection* partialOutput)
{
  double chargeDep = E/fEPair*eplus;
  size_t iContact = fCrystalData->GetIContact(x, y, z);
  MGWaveform* outerContactWF = partialOutput->GetWaveform(iContact);
  MGWaveform* centralContactWF = partialOutput->GetWaveform(0);

  for(size_t iSample = 0; iSample < fWFLength; iSample++) {
    double tSample = partialOutput->GetWaveform(iContact)->GetT(iSample);
    double cSample = chargeDep * TMath::Gaus(tSample, t, fSigma, kTRUE);
    (*outerContactWF)[iSample] += cSample;
    (*centralContactWF)[iSample] -= cSample;
  }
}
```

A few notes

- We call the waveform trace a “waveform”, and the pulses inside it are called “pulses”.
- Sampling frequency doesn’t have to be an ADC sampling frequency, it should just be significantly fast enough so that later the WF can be “digitized”
- Remember to add to the WFs passed into GenerateWaveformForHit(), do NOT “set” them.

Issues

- Large number of MGVWaveformGenerators = memory issues
- Currently storing all waveforms, even when they are empty
- Currently storing waveforms as vector<double>
- Let me know if you have any problems.