Pycbc signal processing graph for very early test prototype

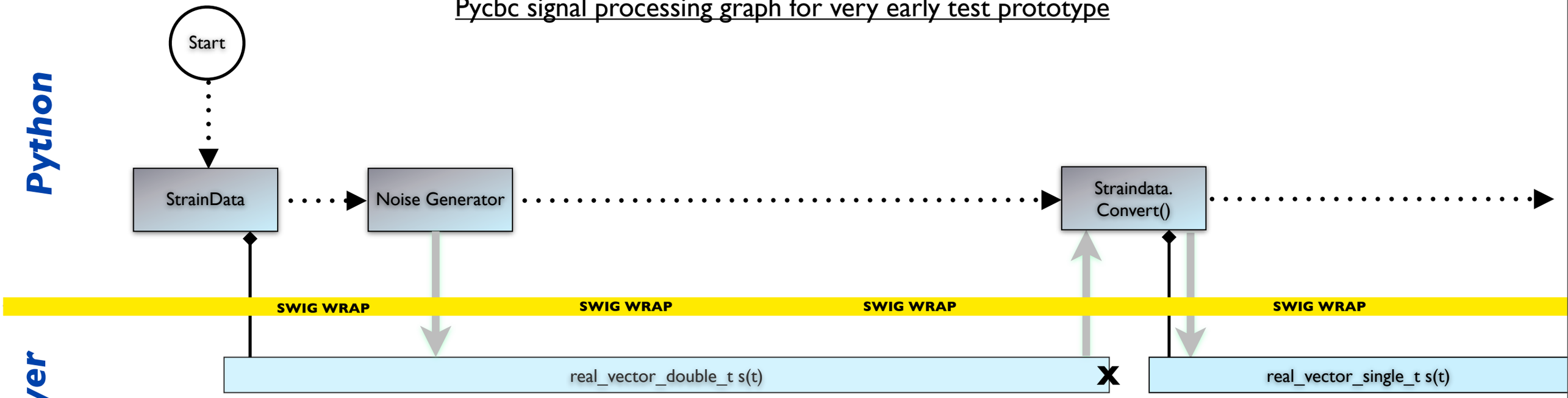
Python

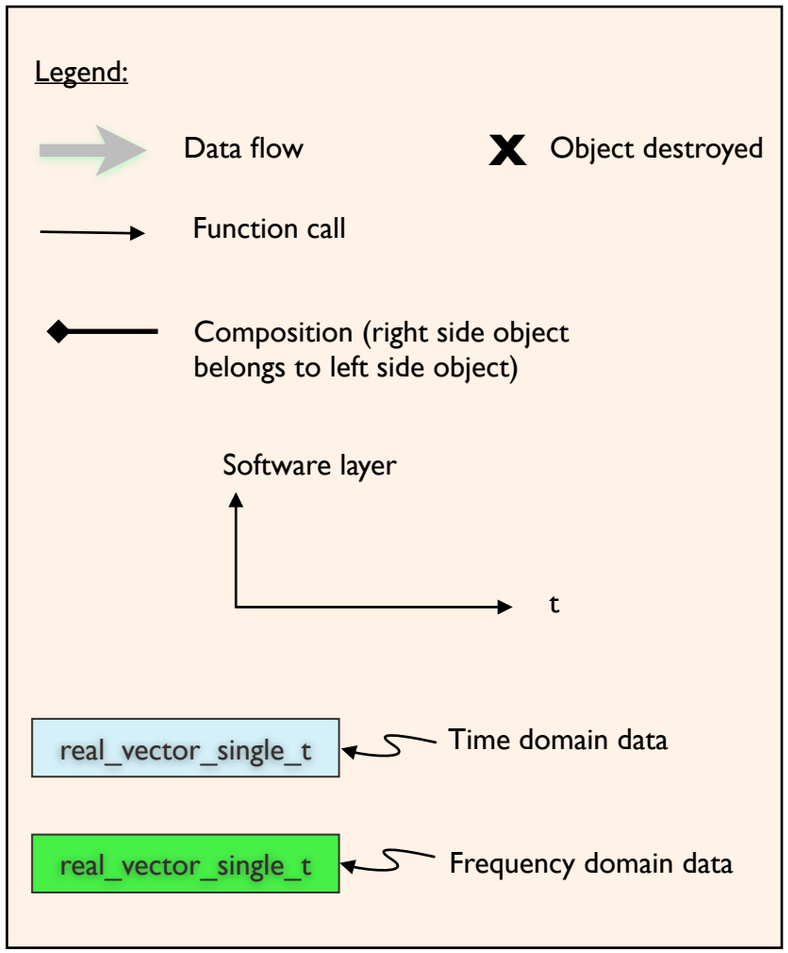
clayer

GPU

Lal

files





Pycbc signal processing graph for very early test prototype

Python

one fixed equal mass system 2 * 1.4 M

Straindata.
Segmenting()

TemplateBank

Straindata.
DataTransfer()

SWIG WRAP SWIG WRAP SWIG WRAP SWIG WRAP

_vector_single_t s(t)

[typ. 15 segments freq data from typ 256s timedomain data -> (N=2^20)] complex_vector_single_t stilde(f) (Python list of datavector_cpu)

FFTs of overlapping input data

render

[ ] complex_vector_single_t stilde(f)

clay

GPU

GLUE/
Lal

files

Pycbc signal processing graph for very early test prototype

Python

clayer

GPU

```
for 5 templates in bank:
```

one waveform

TemplateBank

```
for 15 Segments in data:
```

MatchedFilter

If(max>thresh)

SWIG WRAP

SWIG WRAP

SWIG WRAP

SWIG WRAP

[] complex_vector_single_t stilde(f) (Python list of datavector_cuda)

complex_vector_single_t htilde

Generate_snr

Find_maximum

complex_vector_single_t qtilde (snr tilde)

FFT

real_vector_single_t q (snr)

```
# filter the data against the template bank
for template in bank:
    htilde = bank.perform_generate_waveform(template)
    for stilde in strain_data:
        matched_filter.perform_generate_snr(snr, stilde, htilde)
        matched_filter.perform_find_max(max, snr)
```