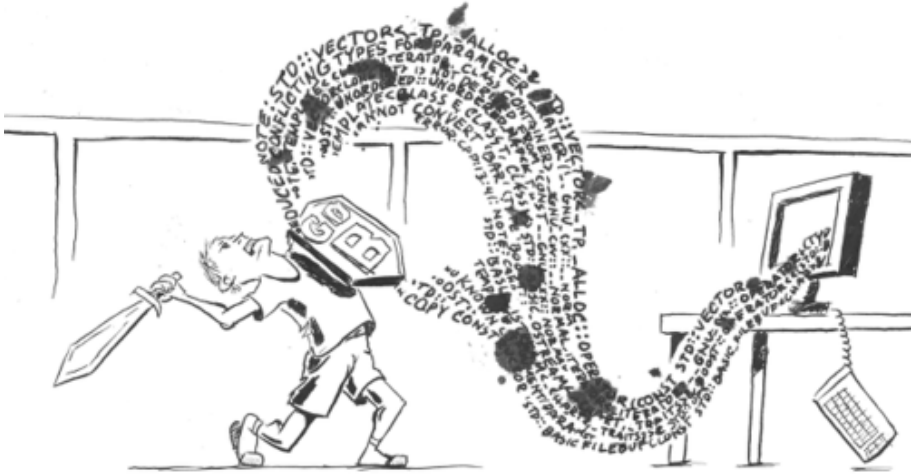
UDELHACK16
Day 2

5DSpeedsters: Zhihua Dong, Chulwoo Jung, Chris Kelly, Meifeng Lin
Mentors: Mat Colgove, Tristan Vanderbruggen, Mathias Wagner



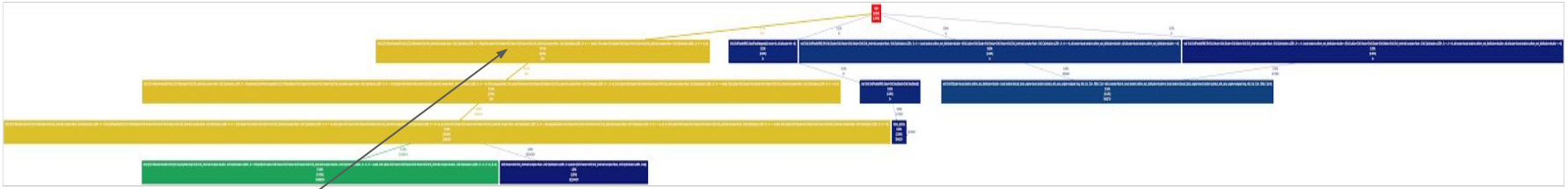
Goals from Day 1:

- Track down the compile error with “-acc” turned on.
- Add trial “pragma acc” to the compute kernel
- Use -ta=tesla to compile to see what happens
- Generate a call graph to understand the dependencies

Progress since last scrum:

- Installed PGI 16.5 on BNL development cluster
- Generated a call graph for the kernel benchmark code : Benchmark_dwf.
- Localized the code that caused the “memmove” error.
- Fixed a “memcpy” error
- Implemented a non-SIMD basic data type that may be more suitable for GPU offloading.
- Attempting to implement alternative to std::vector…..

Callgraph (Not suitable for naked eyes)



- Grid::QCD::WilsonFermion5D is the main compute kernel (~80% of compute)
- Trial OpenACC pragmas in WilsonFermion5D.cc, WilsonKernels.h, WilsonKernels.cc

```
136 int maxiters = 1000;
137 #pragma acc kernels default(present)
138 PARALLEL_FOR_LOOP
139
140 //  for(int ss=0;ss<Umu._grid->oSites();ss++){
141 for(int ss=0;ss<maxiters;ss++){
142     for(int s=0;s<Ls;s++){
143       int sU=ss;
144       int sF = s+Ls*sU;
145       Kernels::DiracOptDhopDir(Stencil,Umu,Stencil.comm_buf,sF,sU,in,out,dirdisp,gamma);
146     }
147   }
```

Temporary hack

```
56 #pragma acc routine seq
57      void DiracOptDhopDir(StencilImpl &st,DoubledGaugeField &U,
58                           Vector<SiteHalfSpinor>  &buf,
59                           int sF,int sU,const FermionField &in, FermionField &out,int dirdisp,int gamma);
```

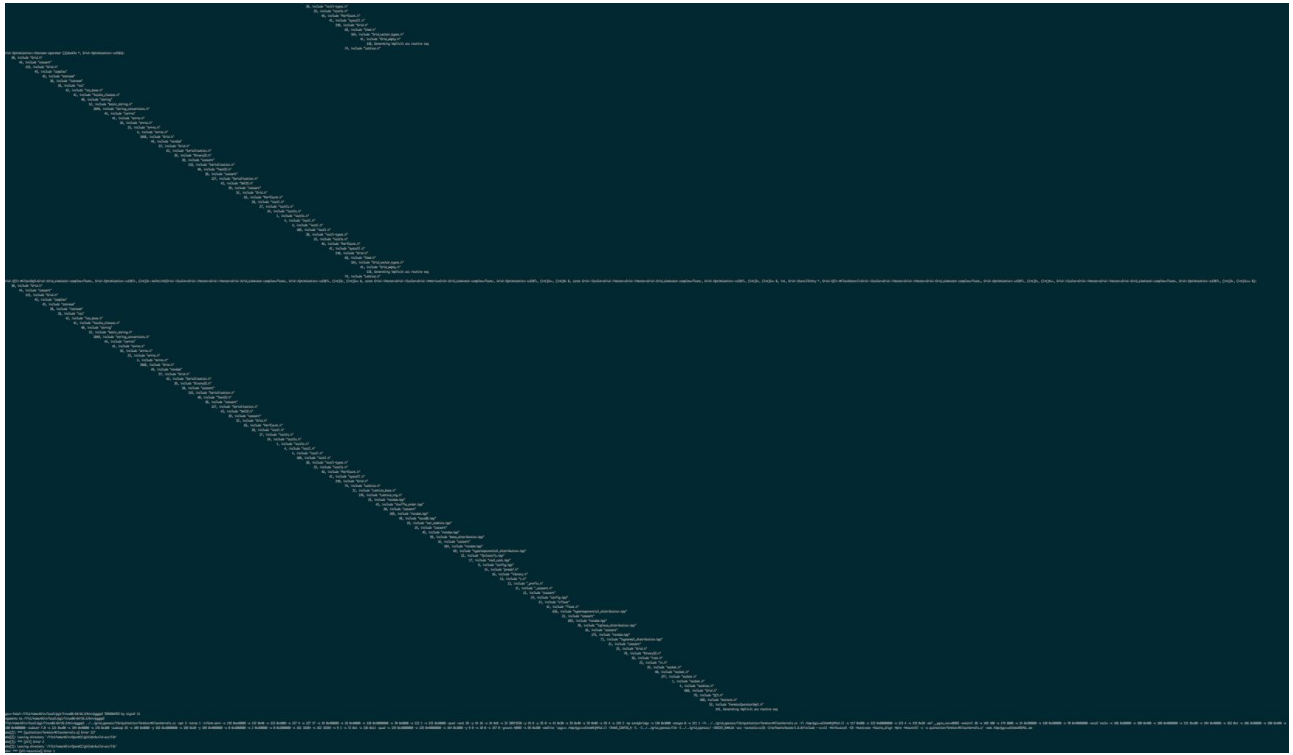
```
commit 4046e7341aad0a902d2c03c4a553f864332baa70
Author: Meifeng Lin <mflin228@gmail.com>
Date:   Tue May 3 13:29:23 2016 -0400

    fixes for memmove and memcpy pgc++ -acc error

commit dcd35ef762f4664c341df3c7966a3cce26f71d75
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Tue May 3 11:16:16 2016 -0400

    -vector fixes

commit e649b75c84e33bd7028208da6b92408be19fab18
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Tue May 3 10:40:10 2016 -0400

    test

commit 65d96ea357c27a1e5ff49b16db35a40229e1a283
Merge: f5e1fd5 59b3df9
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Tue May 3 09:22:53 2016 -0400

    Merge remote-tracking branch 'origin/meifeng-devel' into ckelly

commit f5e1fd59bb7afcf397730942a740e53ab5742c2e
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Tue May 3 09:12:29 2016 -0400

    -Vector size for real types is now half that of complex, representing single logical node

commit 59b3df92db654f45d7bdbe854e09cb8ae8525910
Author: Meifeng Lin <mflin228@gmail.com>
Date:   Mon May 2 20:35:43 2016 -0400

    initial acc implementation

commit 487463fea0e0cfc668aa172cf15aa4ace06db9ff
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Mon May 2 20:28:55 2016 -0400

    -Added SIMD wrapper for std::complex, meant to be used for cases when we don't want any vectorization at this layer

commit bfadd8b56196108978414294d2709f90f70e513d
Merge: d196bce ee25ceb
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Mon May 2 19:46:26 2016 -0400

    Merge branch 'meifeng-devel' of github.com:chulwoo1/Grid into ckelly

commit ee25ceb9e340e1b475ab567def5f3f9c07d3217a
Author: Meifeng Lin <mflin228@gmail.com>
Date:   Mon May 2 19:44:51 2016 -0400

    Put std::random back in Lattice_rng.h

commit d196bcec6723ec5a9d70fed6696e909faaf157fb
Author: Christopher Kelly <ckelly@phys.columbia.edu>
Date:   Mon May 2 19:44:16 2016 -0400

    -Removed unnecessary unions from Grid_empty
```

- 10 git commits since last scrum.
- Two new features
  - New SIMD class using std::complex
  - user-defined vector class (DumbVector)

```
37  template<typename T, typename ignore = std::allocator<T> >
38  class DumbVector{
39  private:
40    T* data;
41    size_t len;
42  public:
43    DumbVector(const size_t size = 0): len(0), data(NULL){
44      resize(size);
45    }
46    void resize(const size_t nlen){
47      T* ndata = new T[nlen];
48      if(data != NULL){
49        size_t cplen = nlen < len ? nlen : len;
50        memcpy(ndata, data, cplen*sizeof(T));
51        delete[] data;
52      }
53      data = ndata;
54      len = nlen;
55    }
56    T& operator[](const size_t i){ return data[i]; }
57    const T& operator[](const size_t i) const{ return data[i]; }
58
59    size_t size() const{ return len; }
60
61    ~DumbVector(){
62      if(data!=NULL) delete[] data;
63    }
64  };
```

- **Several bug fixes and compiler bug reports**
- Compile still fails at generating acc routine

Where we are currently stuck



- Memmove, memcpy not supported on device (probably coming from C++11 move constructor)
- Compiler gets lost in long list of includes: unsupported statement in line 0
- Command and conquer strategy to pin down errors
- Benchmarks: Seriously, Fernanda?

- NEW GOAL: Get Michael Wolfe on a plane (THANK YOU!)
- NEW GOAL: 15 compiler bugs reported for C++11 until Friday (got it: pit of despair)