STE||AR GROUP

a Scalable Toolkit for an Open Community Supporting Near Realtime High Resolution Coastal Modeling

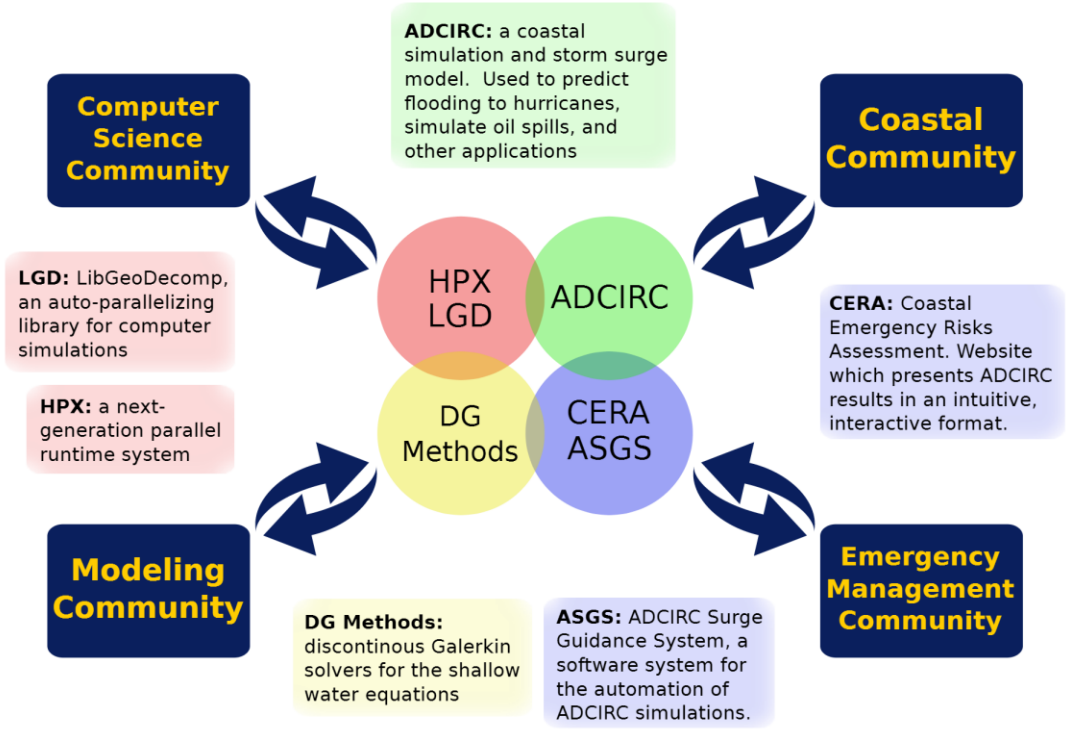
---

ADCIRC USERS GROUP MEETING 2015
MARCH 31, 2015

Zach Byerly (zbyerly@cct.lsu.edu)

Project Scope

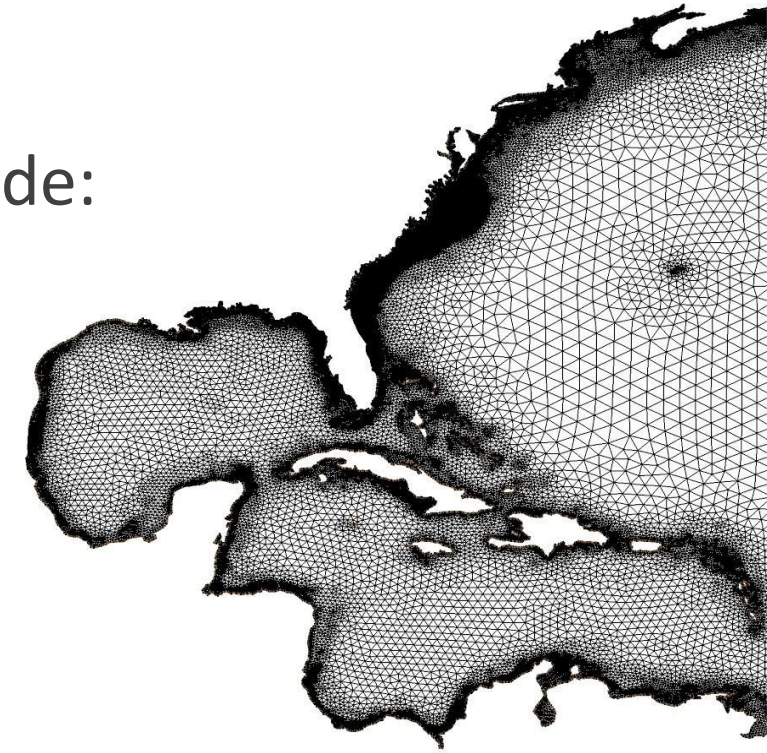
STE||AR GROUP



Our Goal

Build a next-generation ADCIRC code:

- Sustainable
- Extensible
- Scalable
- Flexible



STE||AR GROUP

The 4 Horsemen of the Apocalypse: SLOW

Starvation

- Insufficient concurrent work to maintain high utilization of resources

Latencies

- Time-distance delay of remote resource access and services

Overheads

- Work for management of parallel actions and resources on critical path which are not necessary in sequential variant

Waiting for Contention resolution

- Delays due to lack of availability of oversubscribed shared resources

courtesy of www.albrecht-durer.org

The Challenges

Ways to defeat the four horsemen:

- Use fine grain parallelism to reduce Starvation
- Focus on data dependencies rather than communication
- Latency hiding rather than latency avoiding
- Adaptive locality control vs. static data distribution

General Purpose Runtime System

Allows an application to be parallelized on shared memory, distributed memory, and heterogeneous architectures (GPUs, Intel Xeon Phi coprocessor)

Solidly based on a theoretical foundation - ParalleX

- A general purpose runtime system for applications of any scale
  - http://stellar.cct.lsu.edu/
  - https://github.com/STEllAR-GROUP/hpx/

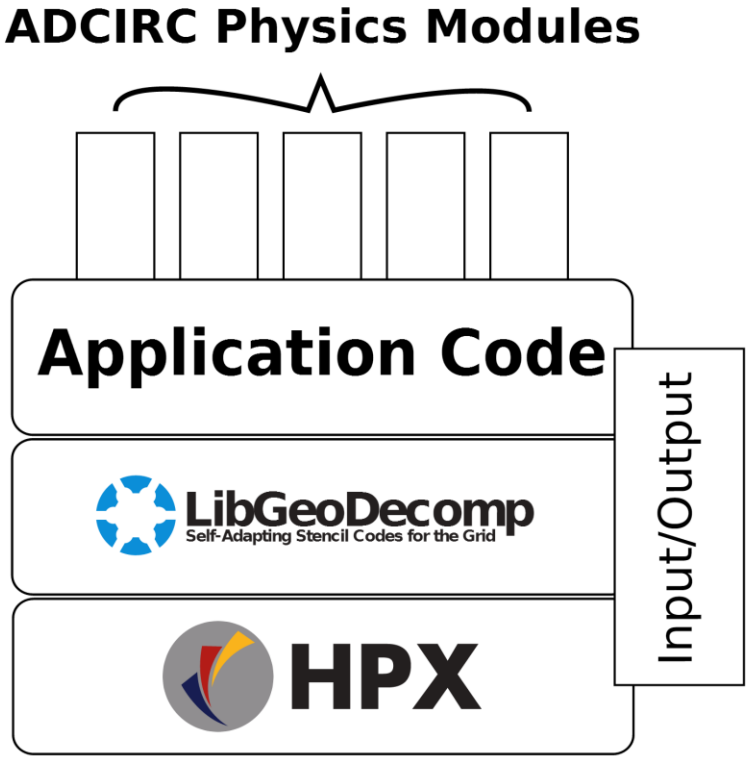
Exposes an uniform, standards-oriented API for ease of programming parallel and distributed applications.

- Enables to write fully asynchronous code using hundreds of millions of threads.
- Provides unified syntax and semantics for local and remote operations.

Enables writing applications which outperform and out-scale existing ones

Published under Boost license and has an open, active, and thriving developer community.

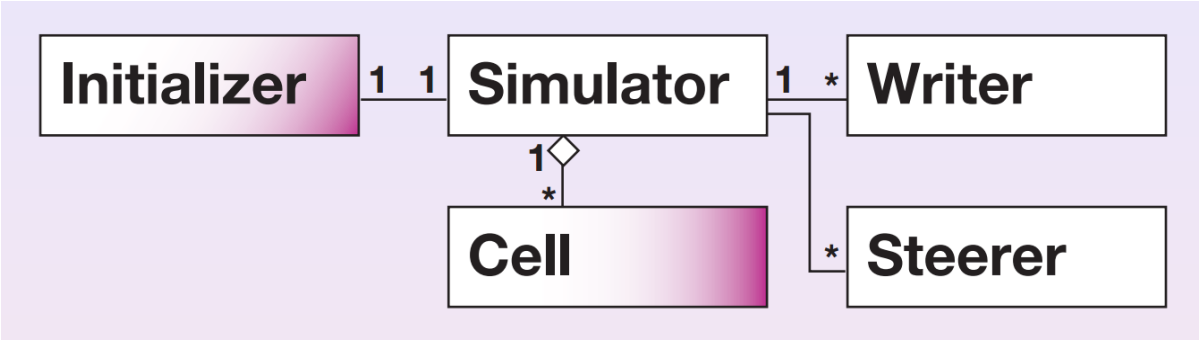
Software Stack:



STE||AR GROUP

- **Lib**rary for **Geo**metry **Decomp**osition codes
- Auto-parallelization for computer simulations
- Developed since 2006 by Andreas Schaefer @ FAU Erlangen
- Open Source development
- Library written in C++, supports custom kernels in C, C++, FORTRAN
- Design Goals:
  - Separate **Domain Science** from **Performance Engineering**
  - Support Legacy Codes
  - Scalable, Portable, Efficient

LGD Building Blocks



User Defines:

- Initializer (Initial Configuration and Input, Mesh, etc.)
- Cell Update (Kernel)

Library Provides:

- Simulator (Handles spatial and temporal loops)
- Writer (Output)
- Steerer (Input)

Current Work

Developing proxy application to simulate workload and communication patterns

`gameoflife_adcirc`:

- Uses decomposed ADCIRC computational mesh
- Simple “game of life” kernel implemented in FORTRAN
- Application code runs on top of LibGeoDecomp which uses HPX for parallelization

gameoflife_adcirc

main.cpp

```
class ADCIRCInitializer : public SimpleInitializer<ContainerCellType>
{
    /*
    Read in fort.18, fort.80, fort.14 files
    Construct ADCIRC mesh and connectivity information
    Initialize Grid
    */
}

void DomainCell::update(const NEIGHBORHOOD& hood, int nanoStep)
{
      //Call FORTRAN subroutine
      kernel_(
              &numnodes,
              alive,
              numneighbors,
              neighbors
              );

}
```

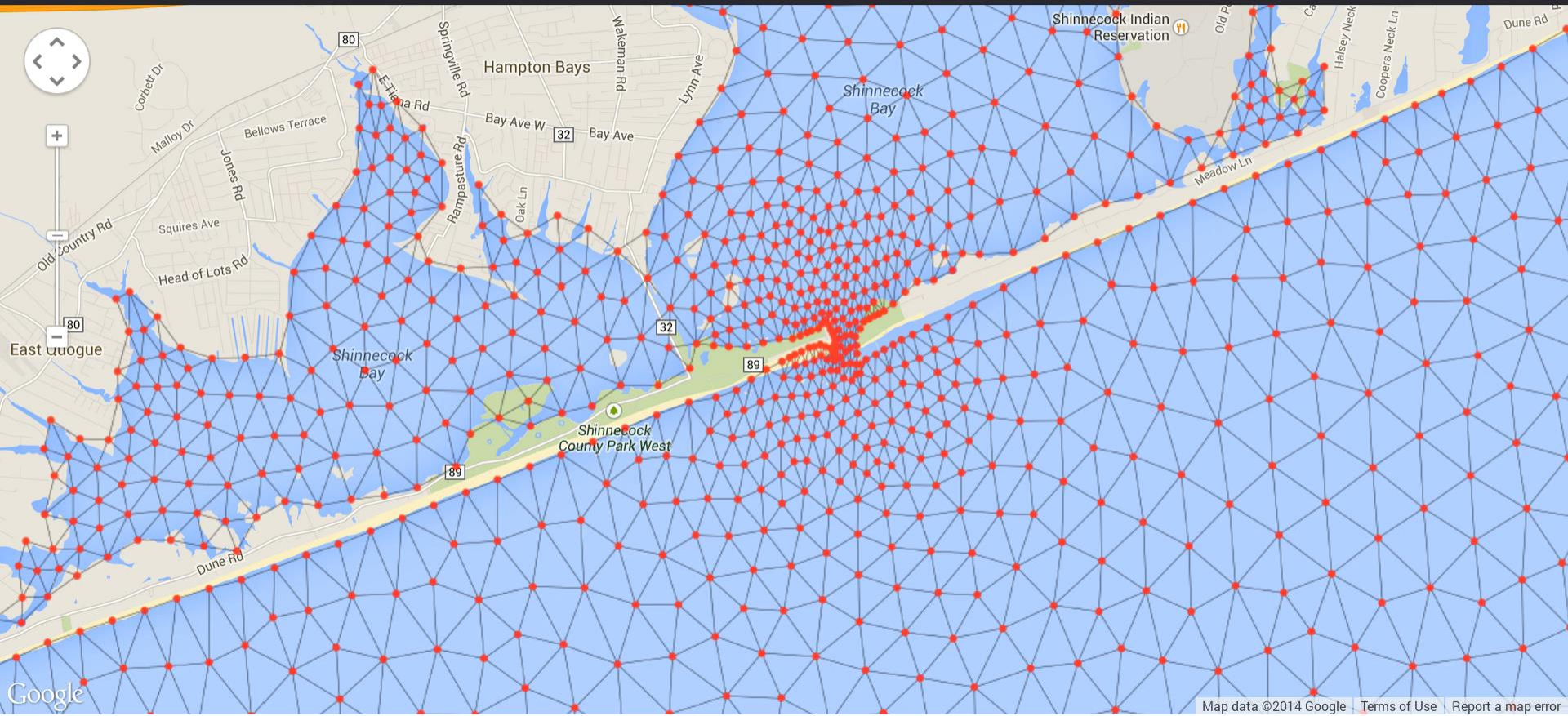
kernel.f

```
      SUBROUTINE KERNEL(N, ALIVE, NUMNEIGHBORS, NEIGHBORS)

      !LOOP OVER NODES
      DO I=1,N
         NEW_ALIVE(I)=0
         IF(ALIVE(I).eq.1) THEN
            NEW_ALIVE(I)=0
         ELSE
            SUM=0
            DO J=1,NUMNEIGHBORS(I)
               SUM=SUM+ALIVE(NEIGHBORS(J,I))
            ENDDO
            IF(SUM.gt.0)THEN
               NEW_ALIVE(I)=1
            ENDIF
         ENDIF
      ENDDO

      END SUBROUTINE KERNEL
```

Coastal Emergency Risks Assessment

ADCIRC Coastal Circulation and Storm Surge Model + SWAN Wave Model

CERA Viewer



Contact us: email

CERA Web Interface

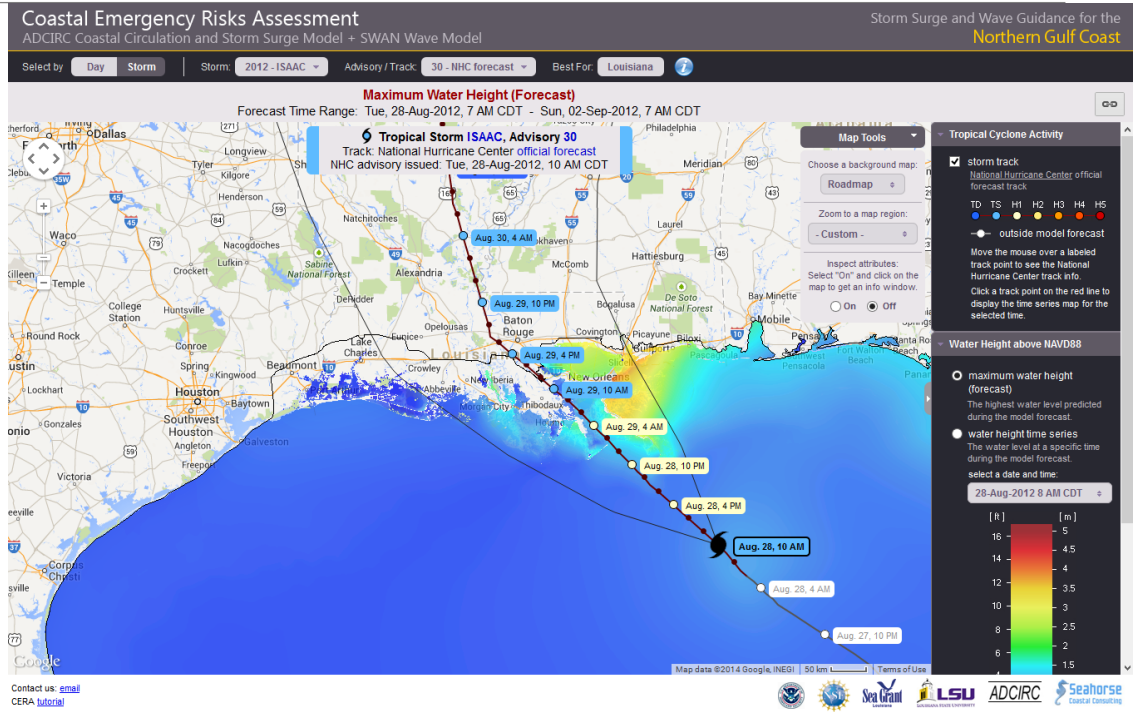
Model output:

- water height (storm surge)
- water inundation above ground
- wind speed
- significant wave height
- relative peak wave

ADCIRC node attributes

Overlay features:

- barriers & levees
- shoreline
- rainfall data (radar, real-time stations)
- real-time water stations
- hurricane track info

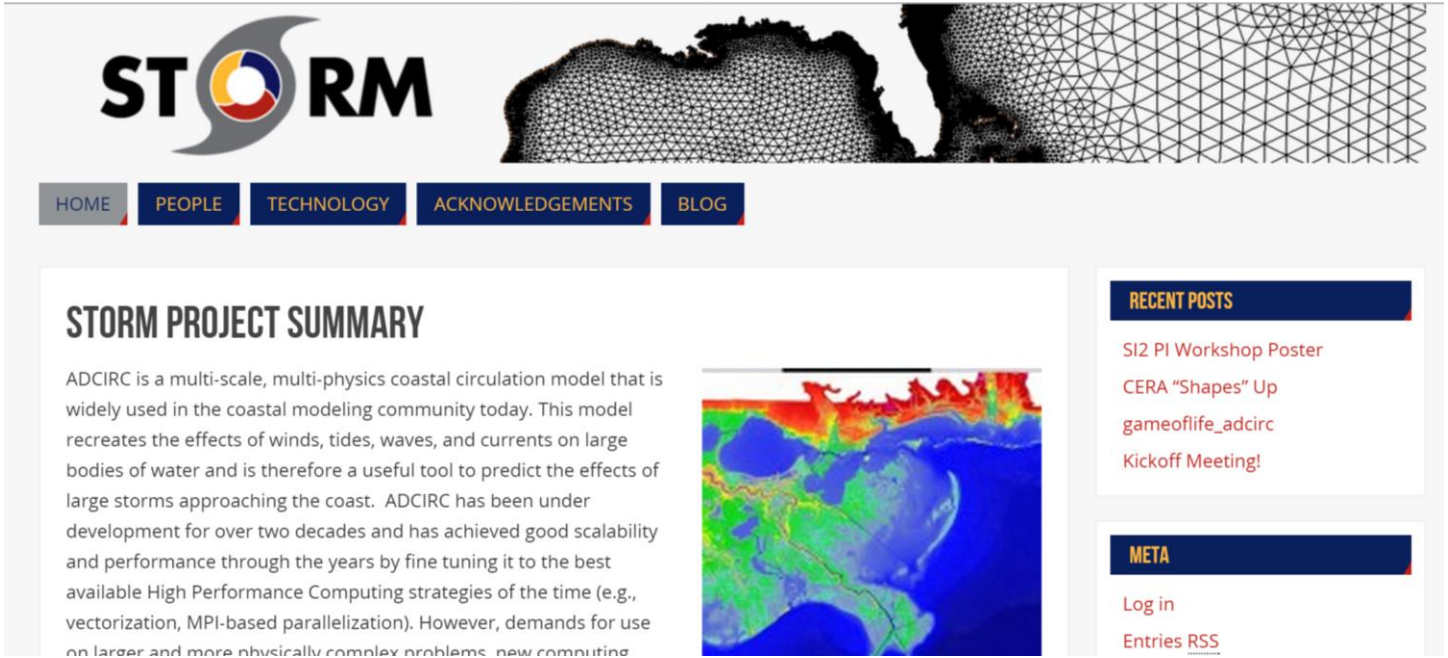


Approach

- Results from ADCIRC/LGD/HPX **must** match ADCIRC/MPI results exactly
- Current ADCIRC/MPI source code left (mostly) intact
- Development of ADCIRC/MPI will continue
- Physics kernels automatically pulled from ADCIRC/MPI source code
- Frequent automatic testing

Our Collaborators

- Joannes Westerink (University of Notre Dame)
- Rick Luettich (University of North Carolina)
- Clint Dawson (University of Texas at Austin)
- Robert Twilley (LSU, Louisiana Sea Grant)
- Jason Fleming (Seahorse Coastal Consulting)
- Hartmut Kaiser (Louisiana State University, STE||AR Group)

Supported by: NSF SI2-SSI: Collaborative Research

storm.stellar-group.org