Very High Level Languages (VHLL) for No Pain Scalable Computing on High Performance Systems (Linux Clusters, MS HPC 2008 clusters, GPUs, SGI Altix, Cray XT5)

Siddharth Samsi, Harrison Ben Smith, Laura Humphrey, Stanley Ahalt, Alan Chalker
Ohio Supercomputer Center,

Bracy H. Elton,
High Performance Technologies, Inc.,

Roope Astala, Niraj Srivastava,
Interactive Supercomputing Inc.
VHHL Overview

Use Very High Level Languages
- MATLAB or Python

Star-P runtime handles memory decomposition & management

Star-P runtime is abstract HPC resource: SMP or Cluster
- M language:
  ```
  >> n = 50000*p;
  >> x = rand(n);
  >> y = rand(n);
  >> z = x * y;
  >> [q r] = qr(x);
  ```
- x, y distributed objects

- Python:
  ```
  >>> n = 50000
  >>> x = starp.numpy.random.rand(n,n)
  >>> y = starp.numpy.random.rand(n,n)
  >>> z = starp.numpy.dot(x,y)
  ```
- qr performed in parallel
Scalability

Data parallel computation
\[ \text{id}x = 200000 \]
\[ x = \text{rand}(\text{id}x, \text{id}x*p); \]
\[ y = \text{rand}(\text{id}x*p, 1); \]
\[ \text{tic}; \ z=x\textbackslash y; \ \text{toc}; \]

Task parallel computation
\[ x = \text{rand}(1025, 1024, \text{id}x*p); \]
\[ y = \text{ppeval('fft2filter',x)}; \]

Over 2.7 T\text{flop/s sustained}
**Real world Problem**

**Application**
- Radio frequency (RF) tomography imaging

**Method**
- Illuminate area of interest with transmitting antenna
- Measure scattered field with receiving antennas
- Determine reflectivity function from scattered field ("inverse scattering")

**Issues**
- Inversion algorithms quite computationally intensive
- Signal processing algorithms frequently modified & written in Very High Level Languages, e.g., M language of MATLAB®
- RF tomography sensor systems have mission requirements for timeliness in deployment scenarios

**Solution**
- Visit our poster for more details!!!