



Blame Chapel

Hui Zhang, Jeffrey K. Hollingsworth
{hzhang86, hollings}@cs.umd.edu

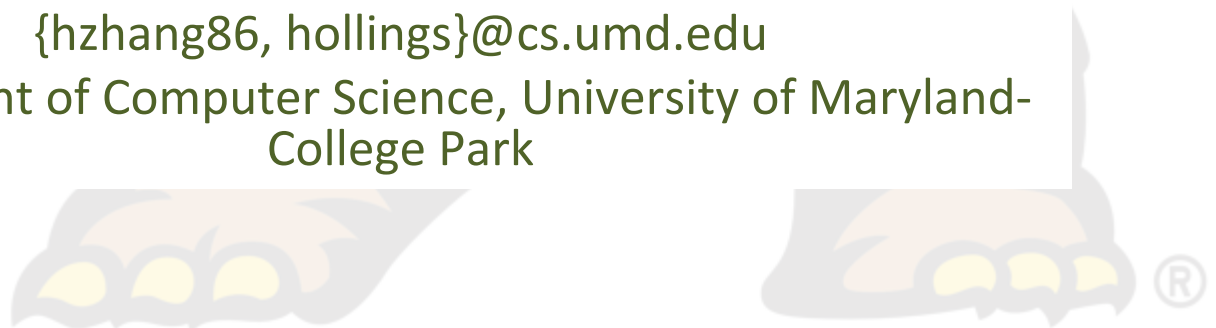
Department of Computer Science, University of Maryland-
College Park



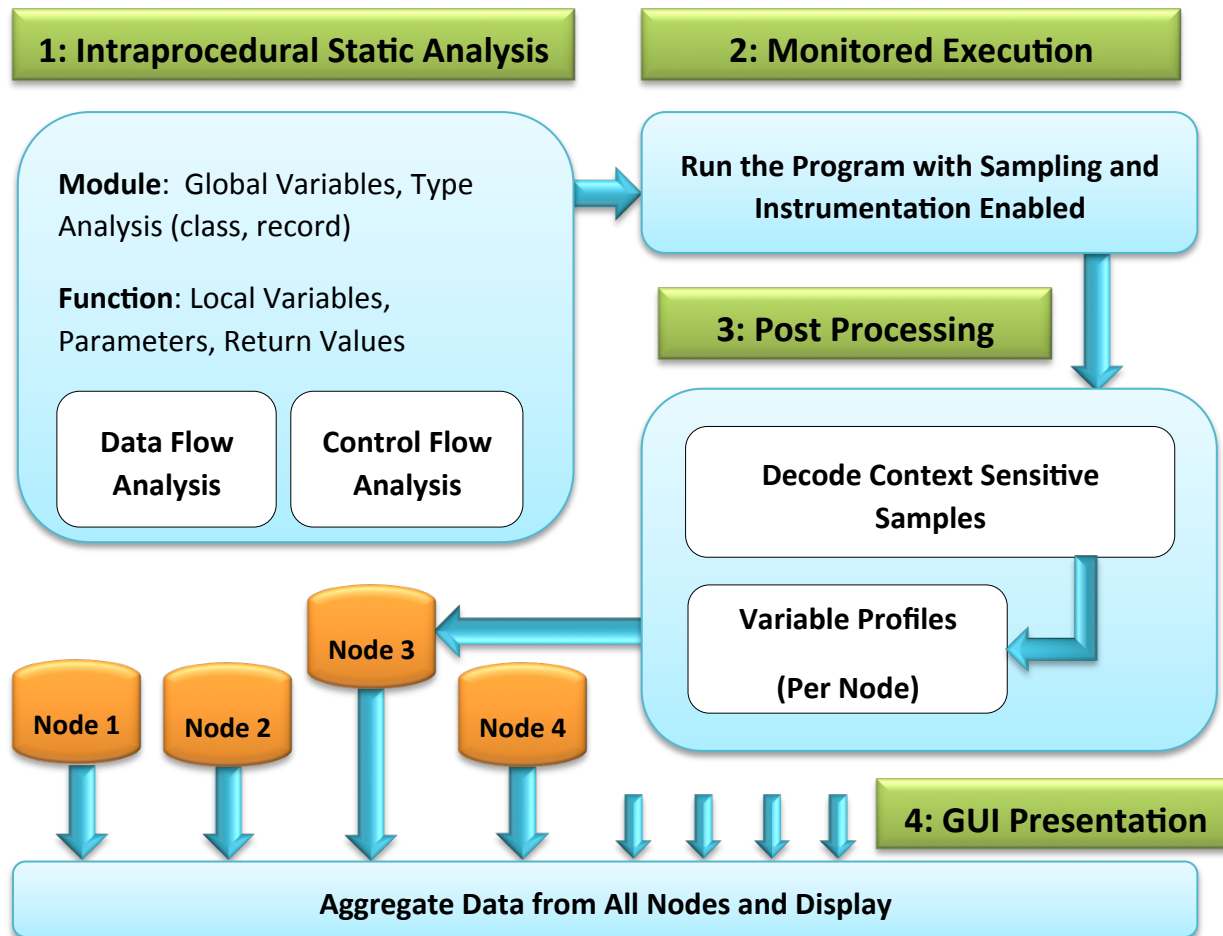
“Blame”** *for* **Chapel

Hui Zhang, Jeffrey K. Hollingsworth
{hzhang86, hollings}@cs.umd.edu

Department of Computer Science, University of Maryland-
College Park



Tool Framework



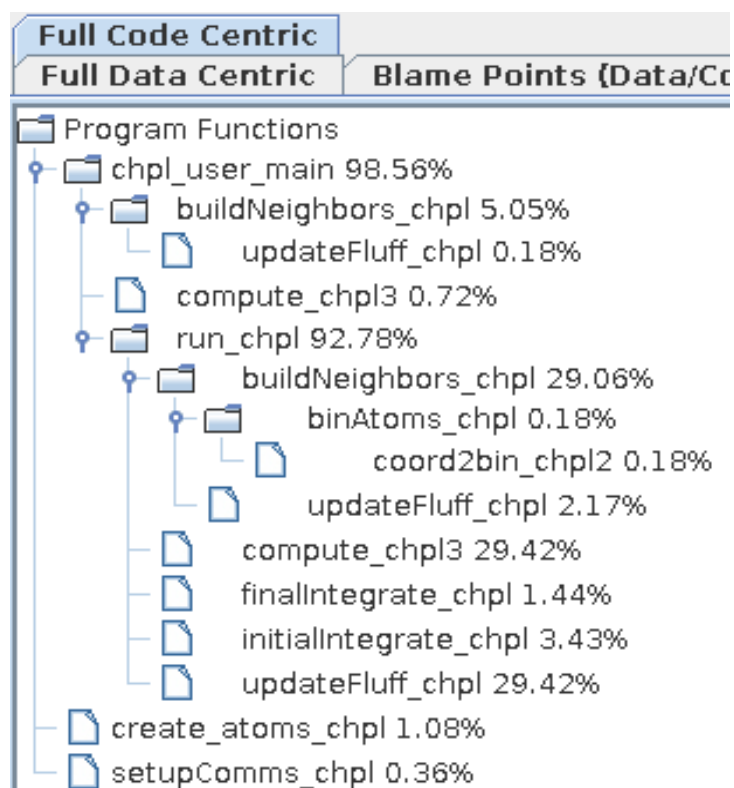
Blame Calculation Example

```
1      a=2;
2      b=3;           //Sample 1
3      if a<b         //Sample 2
4          a=b+1;      //Sample 3
5      c=a+b;          //Sample 4
```

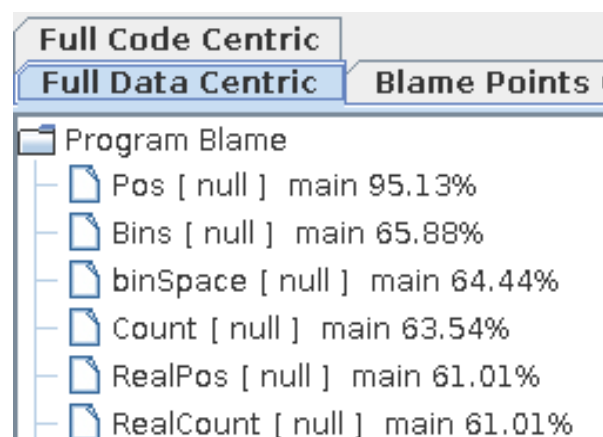
Variable Name	a	b	c
BlameSet	1, 3, 4	2	1, 2, 3, 4, 5
Blame Samples	S2, S3	S1	S1, S2, S3, S4
Blame	50%	25%	100%

GUI screenshots of MiniMD

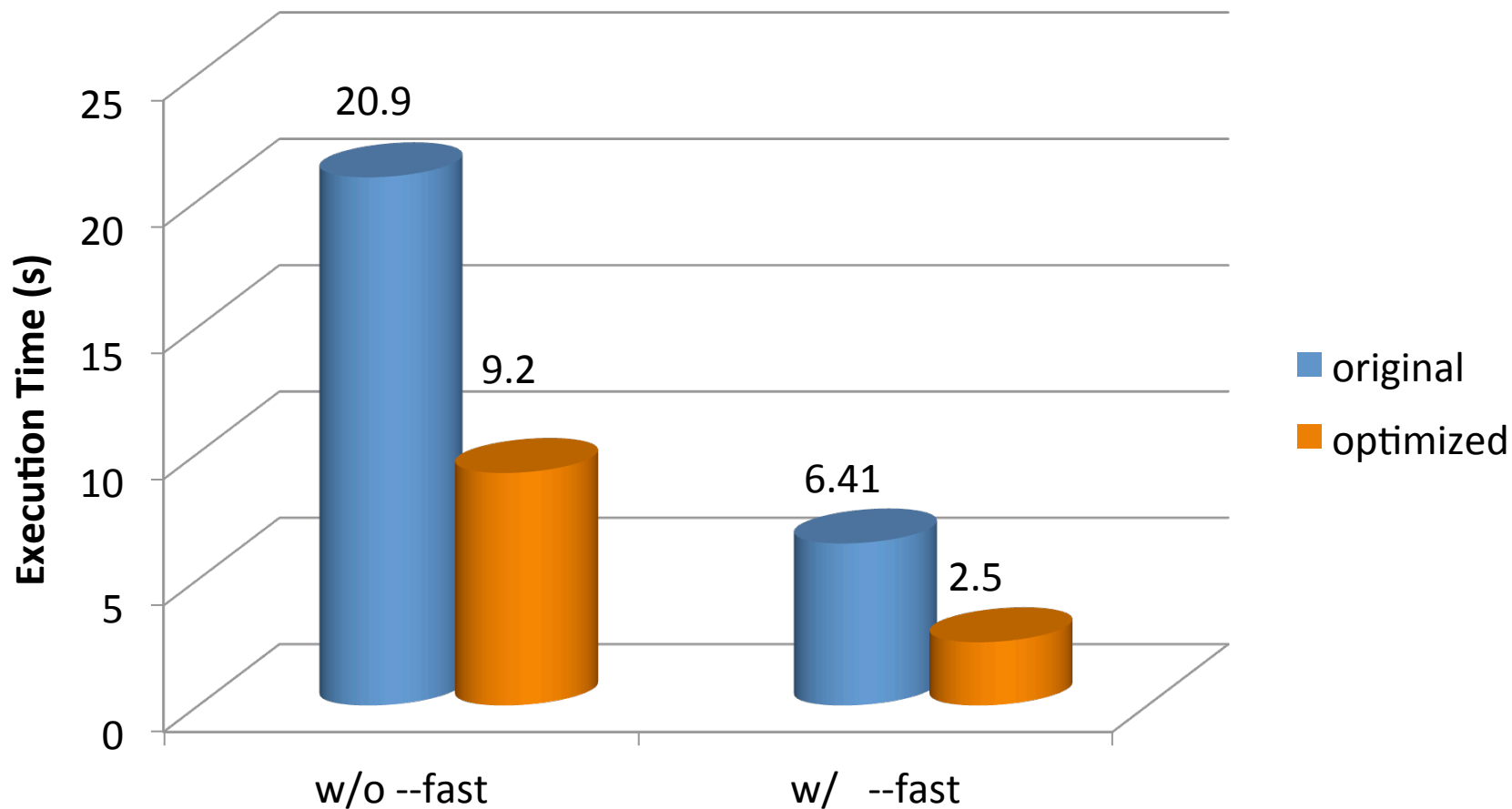
Code-centric



Data-centric



Optimization Result - MiniMD



Experiment – LULESH

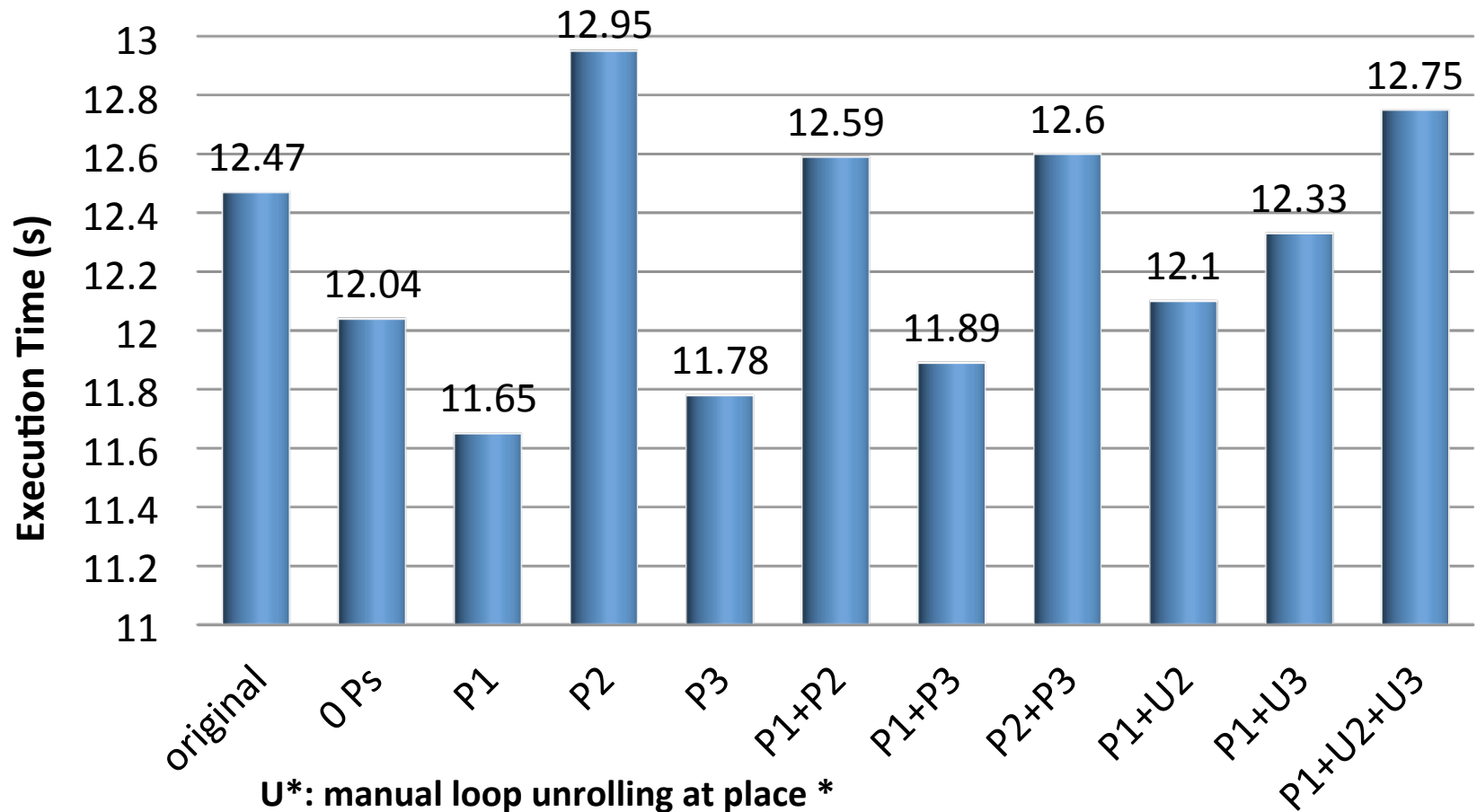
Name	Type	Blame	Context
hgfbz	8*real	30.8%	CalcFBHourglassForceForElems
hgfbx	8*real	29.5%	CalcFBHourglassForceForElems
hgfbz	8*real	29.2%	CalcFBHourglassForceForElems
shz	real	27.9%	CalcElemFBHourglassForce
hz	4*real	27.6%	CalcElemFBHourglassForce
shx	real	26.9%	CalcElemFBHourglassForce
shy	real	26.6%	CalcElemFBHourglassForce
hx	4*real	26.6%	CalcElemFBHourglassForce
hy	4*real	26.6%	CalcElemFBHourglassForce
hourgam	8*(4*real)	25.0%	CalcFBHourglassForceForElems
determ	[Elems] real	15.7%	CalcVolumeForceForElems
b_x	8*real	9.7%	IntegrateStressForElems
b_z	8*real	9.7%	IntegrateStressForElems
b_y	8*real	8.7%	IntegrateStressForElems
dvdz(y/z)	[Elems] 8*real	8.3%	CalcHourglassControlForElems
hourmodx	real	5.8%	CalcFBHourglassForceForElems
hourmodity	real	5.1%	CalcFBHourglassForceForElems
hourmodz	real	4.8%	CalcFBHourglassForceForElems

Optimization Example - Loop

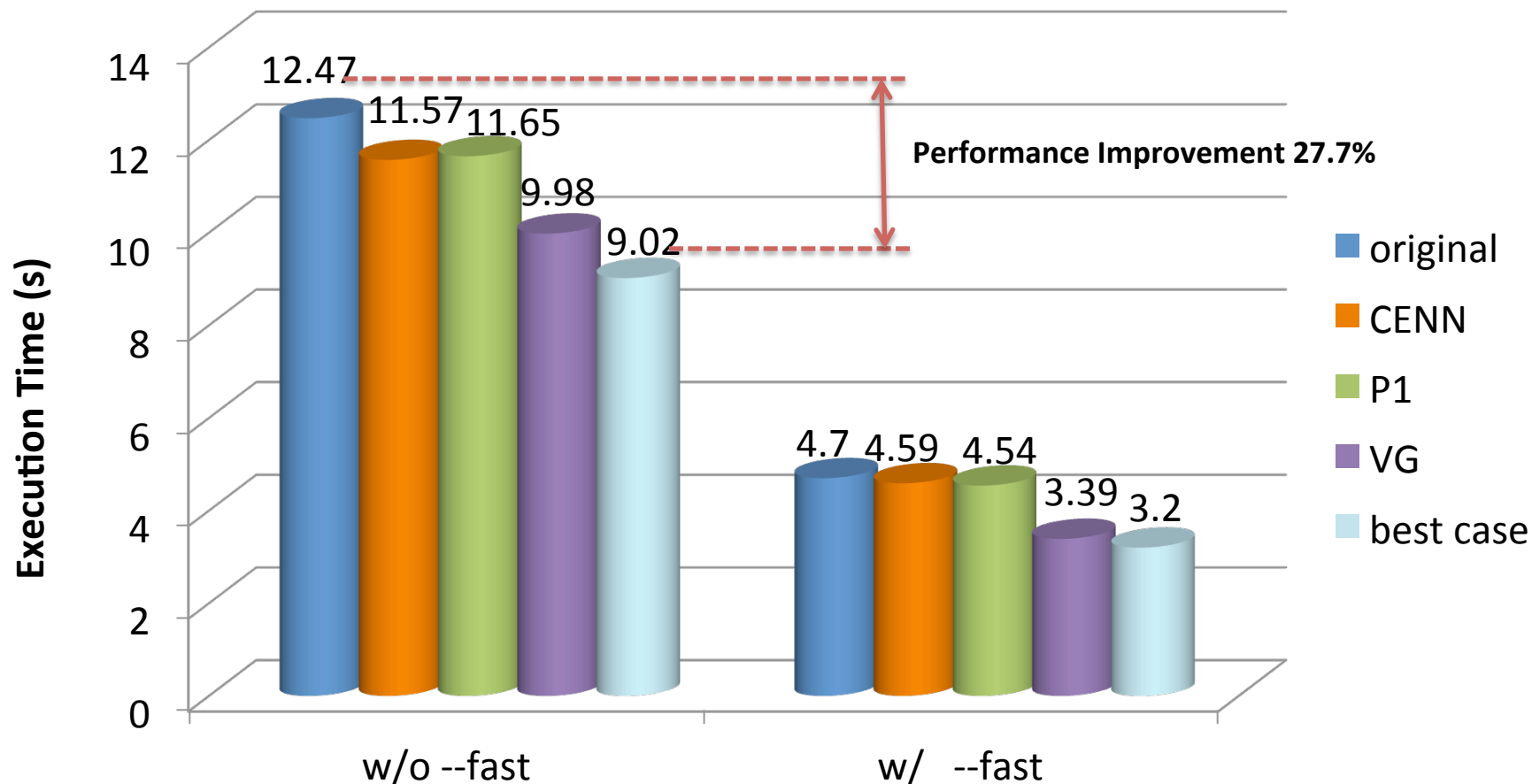
```
for param i in 1..4 {      //P1
  var hourmodx, hourmody, hourmodz: real;
  // reduction
  for param j in 1..8 {    //P2
    hourmodx += x8n[eli][j] * gammaCoef[i][j];
    hourmody += y8n[eli][j] * gammaCoef[i][j];
    hourmodz += z8n[eli][j] * gammaCoef[i][j];
  }
  for param j in 1..8 {    //P3
    hourgam[j][i] = gammaCoef[i][j] - volinv *
      (dvdxdx[eli][j] * hourmodx +
       dvdy[eli][j] * hourmody +
       dvdz[eli][j] * hourmodz);
  }
}
```

Code Snapshot of LULESH Hot Spot

Results for different loop optimizations



Optimization Result – LULESH





Updates & Future Work

- **Updates:**
 - Built a prototype for multi-node Chapel
 - Optimized runtime instrumentation
 - Improved Graphic-User-Interface
- **Future work:**
 - Large-size problems on distributed systems
 - Further application of “Blame” in other fields

Conclusion

- “Blame” application on PGAS programs
- First Chapel-specific profiler
- Benchmark optimization

