

OpenFOAMとParaViewの並列実行

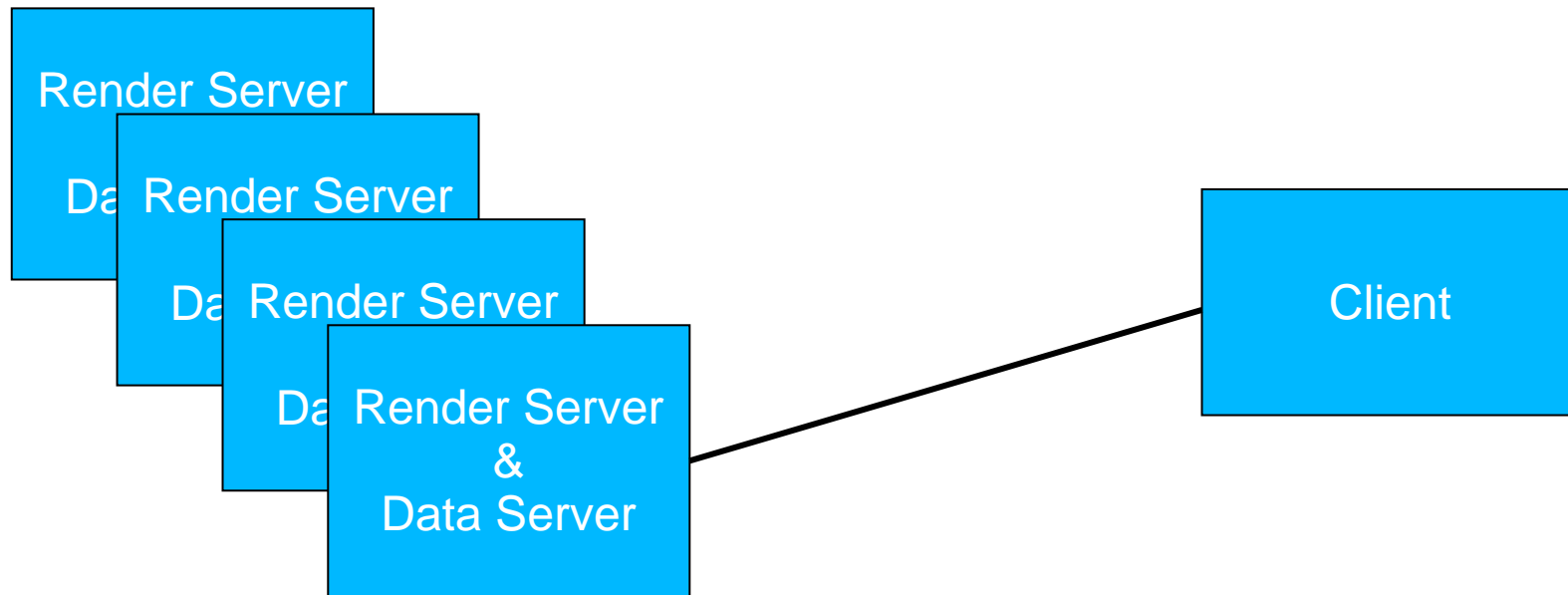
2009.11.07

株式会社爆発研究所 / 東京工業大学

吉田正典

並列版 paraview のインストール

- 並列機を構築、並列環境設定
- 並列化ライブラリをインストール
- Qt をコンパイル&インストール
- paraviewをコンパイル&インストール



Xのセキュリティをはずす
各ノードにXログインし、`xhost +` を実行
(各ノードの画面にレンダリング画像が現れる)

Pvserver(または `pvdataserver & pvrenderserver`) をマスターノードで並列起動

`paraview` をクライアントで起動、Masterノードに connect

使用したマシン

Low Spec です

- Core2Duo E8500 (3.16GHz)
- DDR2-800 2GB × 4 = 8GB
- GeForce 9600GT

- 4ノードクラスター
 - 8 CPU cores
 - 32 GB
 - 4 GPU
 - Gbit Network

- CentOS 5.3 x86_64
- Qt 4.3.4
- ParaView 3.4

(コンパイル時に最適化フラグなし、数十%遅い)

ParaView での並列処理のポイント

- FILE I/O
高速 DISK, 分散ファイル,
File Format
高速ネットワーク
- Filters → CPU 処理
コア数が多いほうが有利
- Rendering
GPUによる hardware rendering

- FILE I/O

分散ファイル → 今回は1ファイルを分散配置
→ 並列化分散ファイルへ移行

File Format 今回は xdmf を採用 (HDF5)

- Filters → CPU 処理

コア数が多いほうが有利

8コアは単一機程度 → 32~64 コアへ

- Rendering

GPUによる hardware rendering

GeForce は OpenGL に不向き → Quadro に

検証に使用したファイル

- 直交等間隔構造格子
- 球対称プロファイル、 $r \rightarrow$ 大で減衰
- 10M, 20M, 40M, 80M, 160M, 320M の格子数を xdmf で作成
320Mで 1.2GB 程度の hdf5 ファイル
+ .xdmf ファイル
- すべてのノードに格納 (NFSは使用しない)
- 読み込み & 等値面作成
- Tools \rightarrow Timer Log で時間を記録
- Single Process、並列処理の場合の時間を比較


```
--- Disable triangle strips.  
Still Render, 0.04905 seconds  
--- Disable display lists.  
--- Disable triangle strips.  
Still Render, 0.011582 seconds  
Still Render, 0.023573 seconds  
    Execute vtkMPIMoveData id: 398, 0.019074 seconds  
Still Render, 0.018928 seconds  
Still Render, 0.568505 seconds  
Still Render, 0.314522 seconds
```

Server, Process 0

```
Execute vtkPVDRReader id: 161, 9.95879 seconds  
  
Execute vtkContourFilter id: 633, 1.6884 seconds  
Execute vtkPVGeometryFilter id: 651, 0.010377 seconds
```

Server, Process 1

```
Execute vtkPVDRReader id: 161, 10.2519 seconds  
  
Execute vtkContourFilter id: 633, 1.79274 seconds  
Execute vtkPVGeometryFilter id: 651, 0.084135 seconds
```

Server, Process 2

```
Execute vtkPVDRReader id: 161, 9.76198 seconds  
  
Execute vtkContourFilter id: 633, 1.78003 seconds  
Execute vtkPVGeometryFilter id: 651, 0.083415 seconds
```

Server, Process 3

```
Execute vtkPVDRReader id: 161, 10.1984 seconds
```

Refresh

Clear

Time Threshold:

0.01

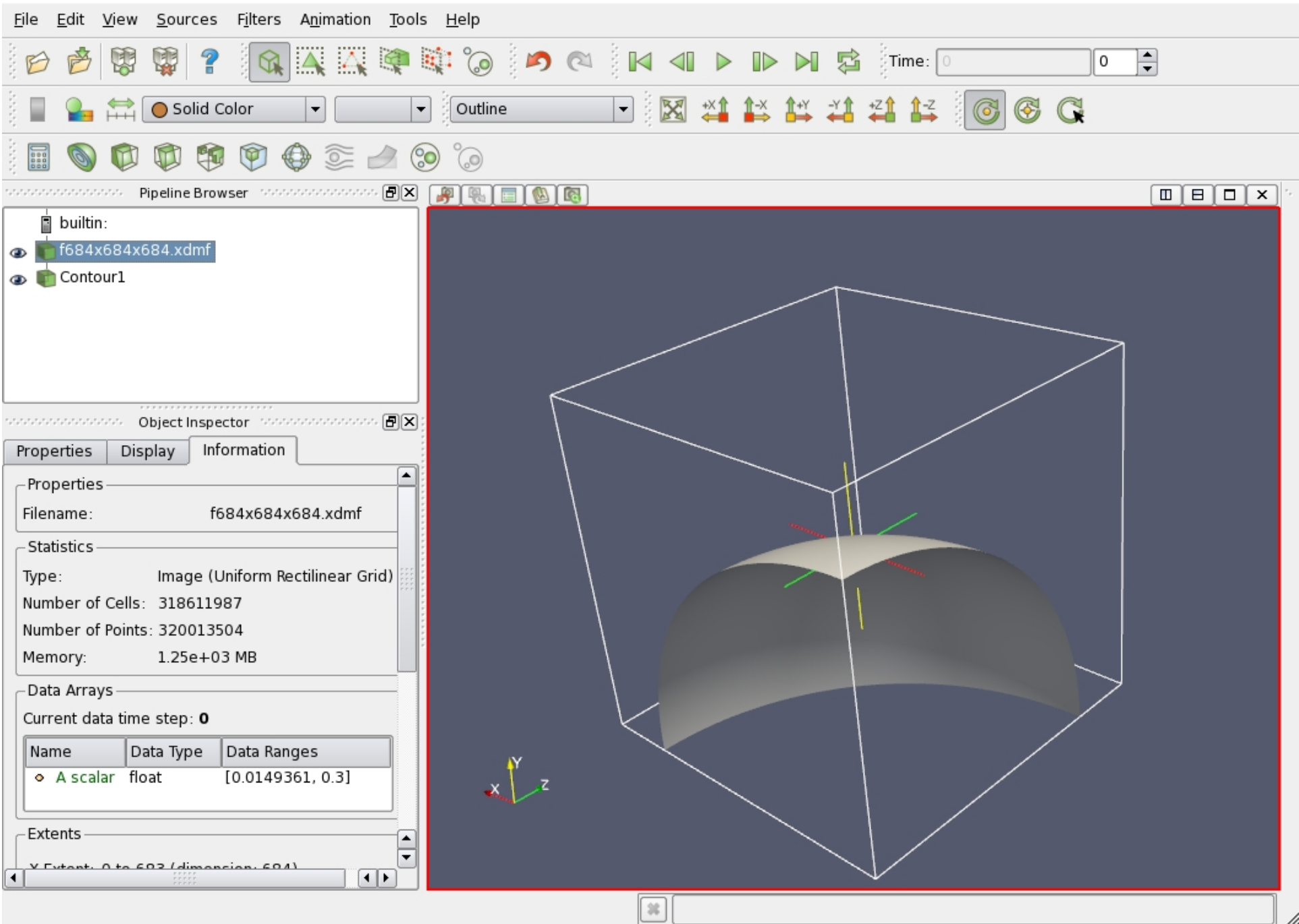
Buffer Length:

500

Enable

Save

Close



single process (seconds)

| | 10M | 20M | 40M | 80M | 160M | 320M |
|----------------|----------|----------|----------|----------|----------|----------|
| Reader | 7.82E-02 | 1.50E-01 | 3.00E-01 | 6.01E-01 | 1.20E+00 | 2.90E+00 |
| ContourFilter | 4.47E-01 | 8.80E-01 | 1.75E+00 | 3.48E+00 | 6.99E+00 | 1.84E+01 |
| GeometryFilter | 2.91E-02 | 4.61E-02 | 7.36E-02 | 1.17E-01 | 2.89E-01 | 3.01E-01 |
| Render | 2.94E-02 | 5.98E-02 | 1.14E-01 | 2.01E-01 | 3.99E-01 | 2.62E-01 |

160M 並列 (node1,2,3,4 8processes)

| | 1 | 2 | 4 | 8 |
|----------------|----------|----------|----------|----------|
| Reader | 1.20E+00 | 1.24E+00 | 1.23E+00 | 1.55E+00 |
| ContourFilter | 6.99E+00 | 3.55E+00 | 1.74E+00 | 1.01E+00 |
| GeometryFilter | 2.89E-01 | 1.17E-01 | 6.23E-02 | 5.30E-02 |
| Render | | | | |

320M 並列 (node1,2,3,4 8processes)

| | 1 | 2 | 4 | 8 | 8 分散ファイル |
|----------------|----------|----------|----------|----------|----------|
| Reader | 2.90E+00 | 2.51E+00 | 1.66E+01 | 1.01E+02 | 1.02E+01 |
| ContourFilter | 1.84E+01 | 7.33E+00 | 4.11E+00 | 7.36E+00 | 1.81E+00 |
| GeometryFilter | 3.01E-01 | 1.88E-01 | 1.23E-01 | 7.26E-01 | 1.03E-01 |
| Render | | | | | |

8分散ファイル:8並列でファイルを Open した後、Save Data → 分散格納
(ただしエラーを吐き出す。Infoが不適正? Contourは正常)

結論

- Core2Duo マシン 4ノードで ParaView並列実行のパフォーマンスを検証した
- 320 Million 格子の等値面作成は 10秒程度で可能である。
- 実機としてはDual Xeon (8cores) 4ノードマシン + Quadro のハードウェア程度で100M クラスの格子ならストレスなく可視化が可能である。
- 並列ファイルシステムに関しては更なる検証・実装が必要と思われる。