

# Graph500

# Graph500とは

- エッジリストからグラフを作り、
- グラフを幅優先探索
- 処理の仕様のみが定められている
  - 実際のデータ構造は自由

# ベンチマークとしてのGraph500

- LINPACK
  - top500で使われている 浮動小数点数の演算
  - 演算器を増やしてノードあたりのメモリを十分確保すればOK
- SPEC
  - 実アプリのカーネルで測定 整数・FP
  - 各カーネルはシングルスレッド

# Graph500の使い道

- データ構造は自由なので  
測定される性能 = ハードの性能 × 最適化の努力
- リファレンス実装もある
  - OMP-CSR 隣接行列を圧縮して保持、OpenMP
  - SER-CSR
  - List リストによる実装
- ハードの性能を測定したいので OMP-CSR で

# Graph500の動かし方

- Graph500.orgからソースを取ってくる
- make.inc編集
  - デフォルトは謎のmallocラッパーを使うのでうまく動かない
  - 適当にコメントアウト
- make -C omp-csr
- omp-csr/omp-csr -s 26

# メモリ使用量

- Graph500はメモリ使用量が大きい
  - SCALEパラメータで決まる
- Toy [SCALE=26] 17GB
- Mini [SCALE=29] 140GB
- Small [SCALE=32] 1TB
- Medium [SCALE=36] 17TB
- Large [SCALE=39] 140TB

# 実際のメモリ使用量

- 実際のメモリ使用量をmassifで測定
- toyでも37,659,659,752/バイト使っていた
  - 17GBの倍以上

# とりあえずの結果

- 2xOpteron 6282SE 2.8GHz [32c], 256GB RAM
  - 6.69E+7 TEPS
- 2xXeon E5530 2.4GHz [8c], 96GB RAM
  - 7.78E+07 TEPS
- Opteronはメモリが4-nodeに分散し、2-nodeのE5530より不利か?
  - メモリコントローラが本当にダメという説も