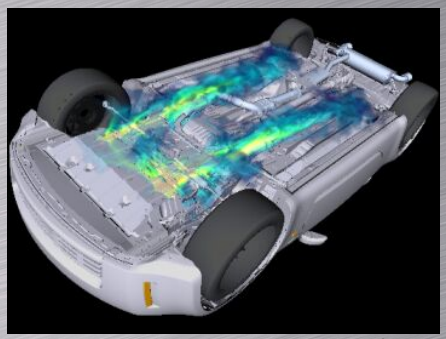
Parallel Interactive Volume Rendering

Kenji ONO
Intelligent **M**odeling **L**aboratory, Dept. of Mech. Eng.
University of Tokyo
keno@{iml.u-tokyo.ac.jp, riken.jp}

Engineering and Vis.

- 車両設計のシミュレーション
  - 研究から設計へ
  - 適用範囲の拡大
  - 大規模化
  - 非定常データ
  -



©Nissan

- 設計情報の抽出
  - イメージの重要性
    - 設計者とデザイナーの意志疎通
    - 迅速なイメージの作成

Visualization for Engineering

- 流れ場の理解
  - 可視化の過程で思考を中断しないこと
  - 高いインタラクティブ性，レスポンス
  - 優れた操作性
  - 必要な情報を抽出し，提示すること
  -
- 大規模データ
  - 数千万点オーダー
  - レスポンスの維持

Importance of Interactivity

- 現象の理解
  - 流動構造の把握
  - 時間軸（非定常な現象）
  - 多方向からの観察
  - 5fpsがひとつの目安
  -
- 頭の中で再構築
  - イメージ化（写実的・非言語的）
  - 抽象化（簡略化）

Real-time and Interactive Visualization

Real-time	Interactive
計算と同時に可視化画像を生成	計算終了後，可視化処理
ステアリング	十分なスピードで対話処理
低ストレージ	データをディスクに保存（大容量）
パラメータ決定後，バッチ処理	パラメータを試行錯誤で決定
ネットワーク透過性	処理の柔軟性

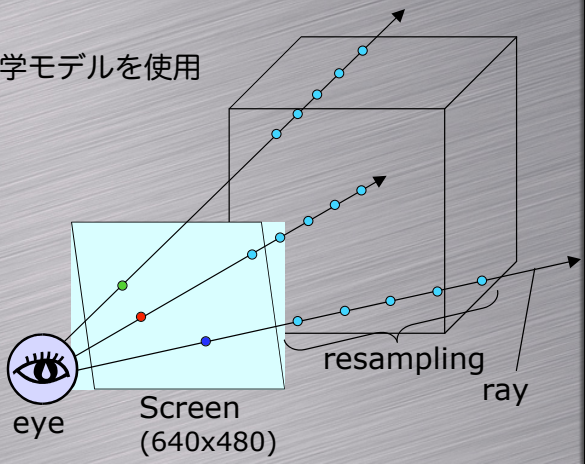
Volume Rendering

- ボリュームデータのトランスルーセントな表現
  - 直感的な流れ現象の表現
  - 光線の追跡を行い、光の減衰などを考慮して画像を生成
  - 内部構造の透過性を有し，全体像の把握に優れる
  - 複雑な組織や構造・挙動を理解する
- レイトレーシングアルゴリズム
  - 時間がかかる演算
- 伝達関数の設計が重要
  - 試行錯誤，支援環境

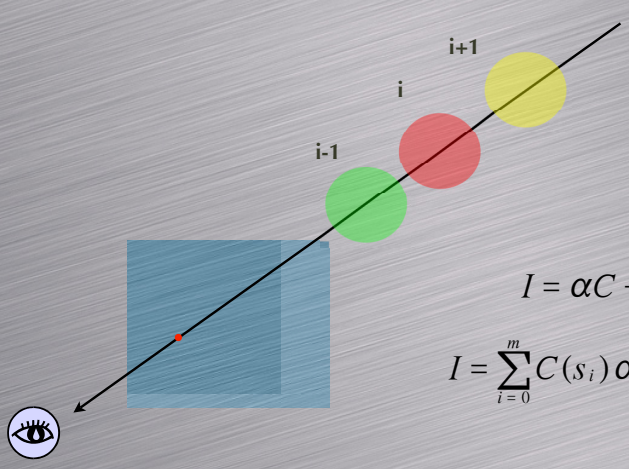
http://www.cs.sunysb.edu/~vislab/gallery_list.html

Ray Casting (Levoy1988)

- ボリュームを半透明のゲル状物質と考える
  - 光線が通過するときの輝度変化を表す光学モデルを使用
- スクリーンの配置
  - ビューイング設定
- リサンプリング
  - 視線ベクトル方向の標本化
- 伝達関数による変換
  - フィールド値→カラー輝度と不透明度
- 画素の値の決定
  - リサンプリング点に対してカラー輝度と不透明度の積和をとる



Algorithm of Volume Ray Tracing



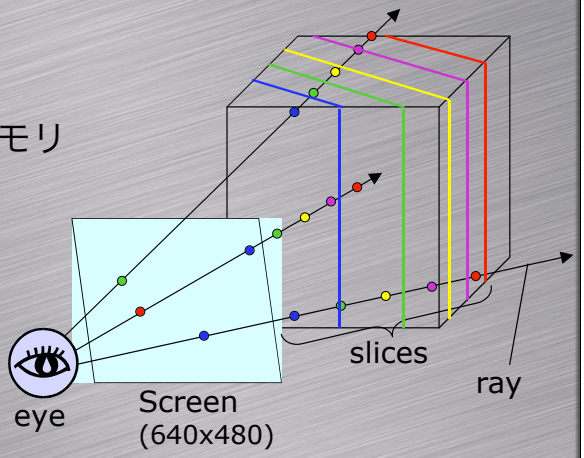
$$I = \alpha C + (1 - \alpha) I_0$$

$$I = \sum_{i=0}^{m} C(s_i)\,\alpha(s_i) \prod_{j=0}^{i-1} (1 - \alpha(s_j))$$

Fujishiro,
情報の可視化，岩波

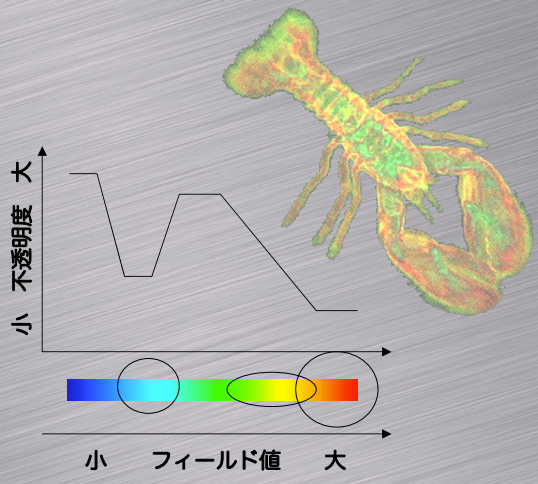
Texture Mapping

- Ray castingの計算順序を変更
  - 各スライス面毎にリサンプリングとレンダリングを行い，テクスチャを計算し，各サーフェスにマッピング
  - Ray castingと等価な画像
- テクスチャ面の半透明処理
  - Zバッファを用いた重ね合わせ
- ボリュームデータをテクスチャメモリ上に展開する
  - 非常に高速
  - GPU利用のHWレンダリング

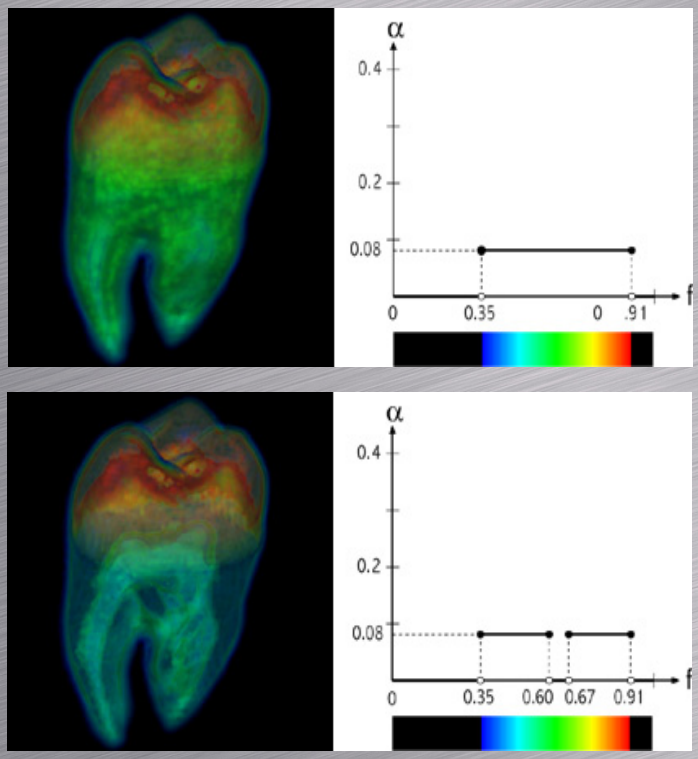


Transfer Function

- フィールド値をカラー情報と不透明度に変換
  - カラー情報任意
  - 不透明度
  - 各ボクセル密度（フィールド値）に応じて遮光効果が増加する
  - 密度勾配を考慮する
  - 画像の品質に大きく影響
  -
- TFの設計
  - 試行錯誤が必要



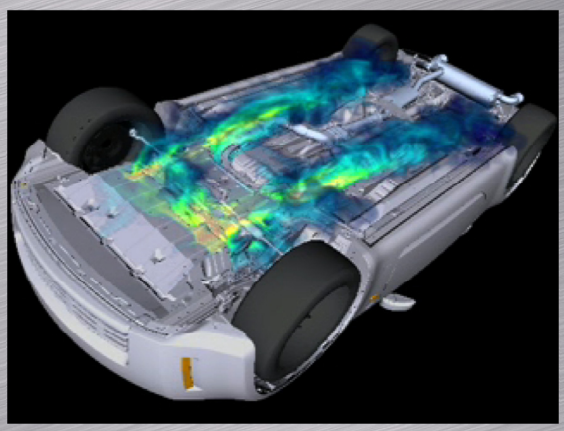
Transfer Function



Fujishiro,
情報の可視化，岩波

Miscellaneous to make a Piece

- 可視化する物理量の値の範囲を調べる
  - スカラ値： 静圧，動圧，総圧，温度，密度
- 投影法と画角を設定
  - 平行投影，透視投影，ビューポート，クリッピング
- 光源の設定
  - 光源の種類，強度
- Transfer Function
  - 使用する色，分布，透過率
- サンプリングレート
  - 結果に大きく影響

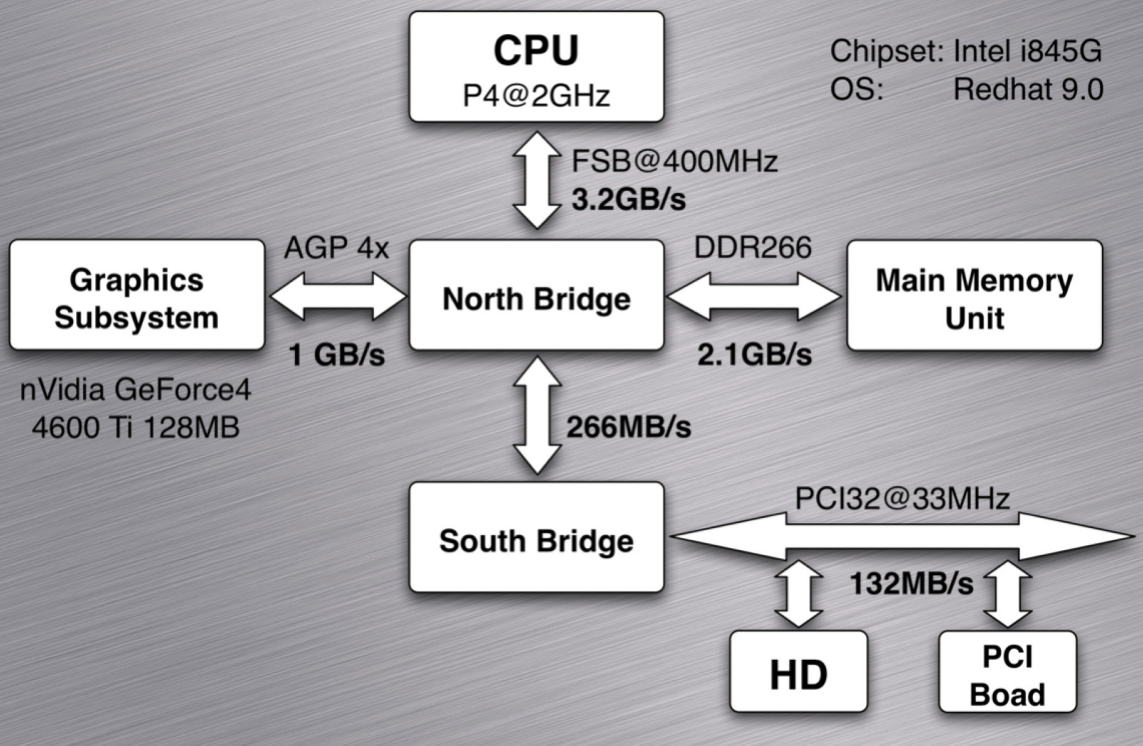


©Nissan

Issues for Large-scale VR on PC

- 大きなデータをどうハンドリングするか
  - 階層的なPCの記憶構造を利用
  - Tex. Mem. > MMU > HDD
  -
- インタラクティブ性の確保 (Ma, UC Davis)
  - 帯域の有効利用
  - In core / Out of core
  - Pre-compression / Decode on the fly by GPU
  -
- 物理的な制限
  - 2GB (IA-32)

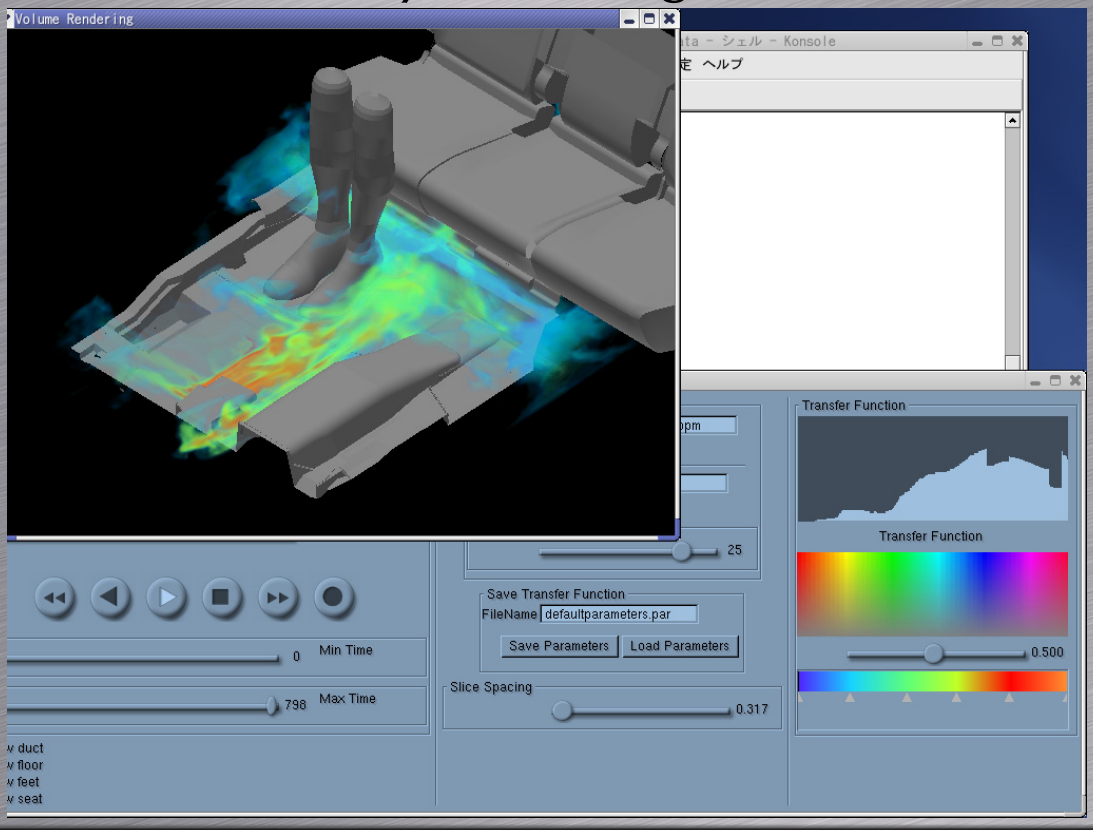
PC Architecture



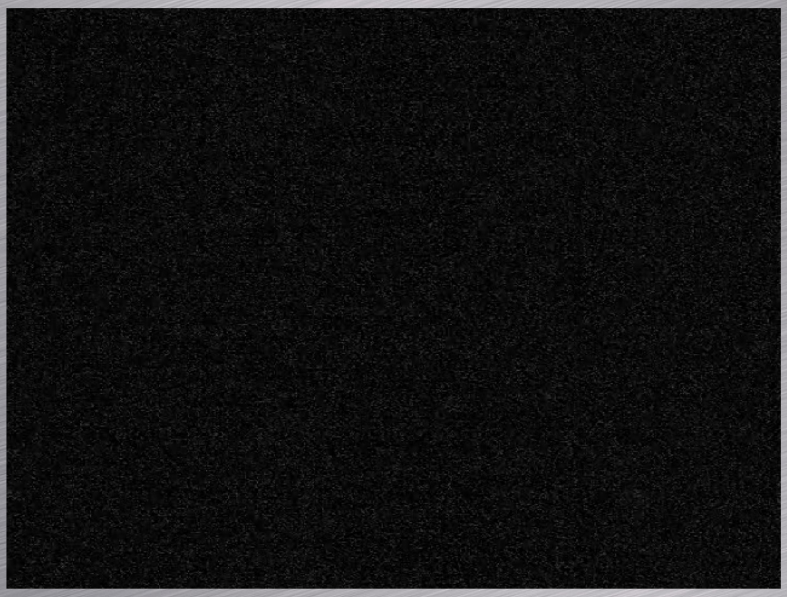
Basic Volume Renderer

- 流体解析のデータ
  - 浮動小数点の結果ファイル（単精度，倍精度）
- 可視化の前処理
  - 量子化：　4(8)bytes > 1byte
  - 更に，DCTによるエンコード（x2, x4, x8）
- 可視化
  - 前処理したデータをMMU上に展開
  - MMU→Tex.memにロード
  - GPUを使ってデコード，レンダリング

System Image

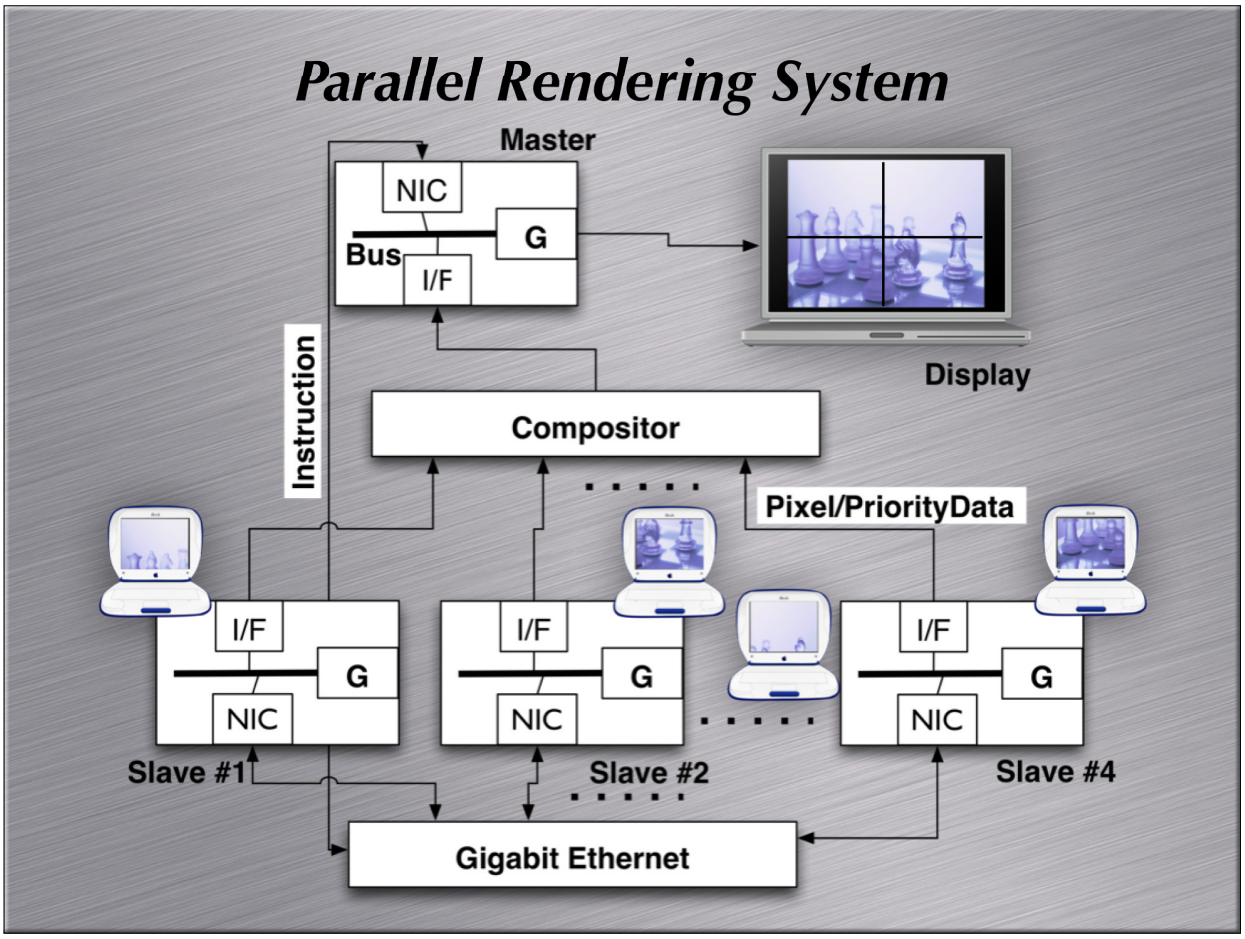


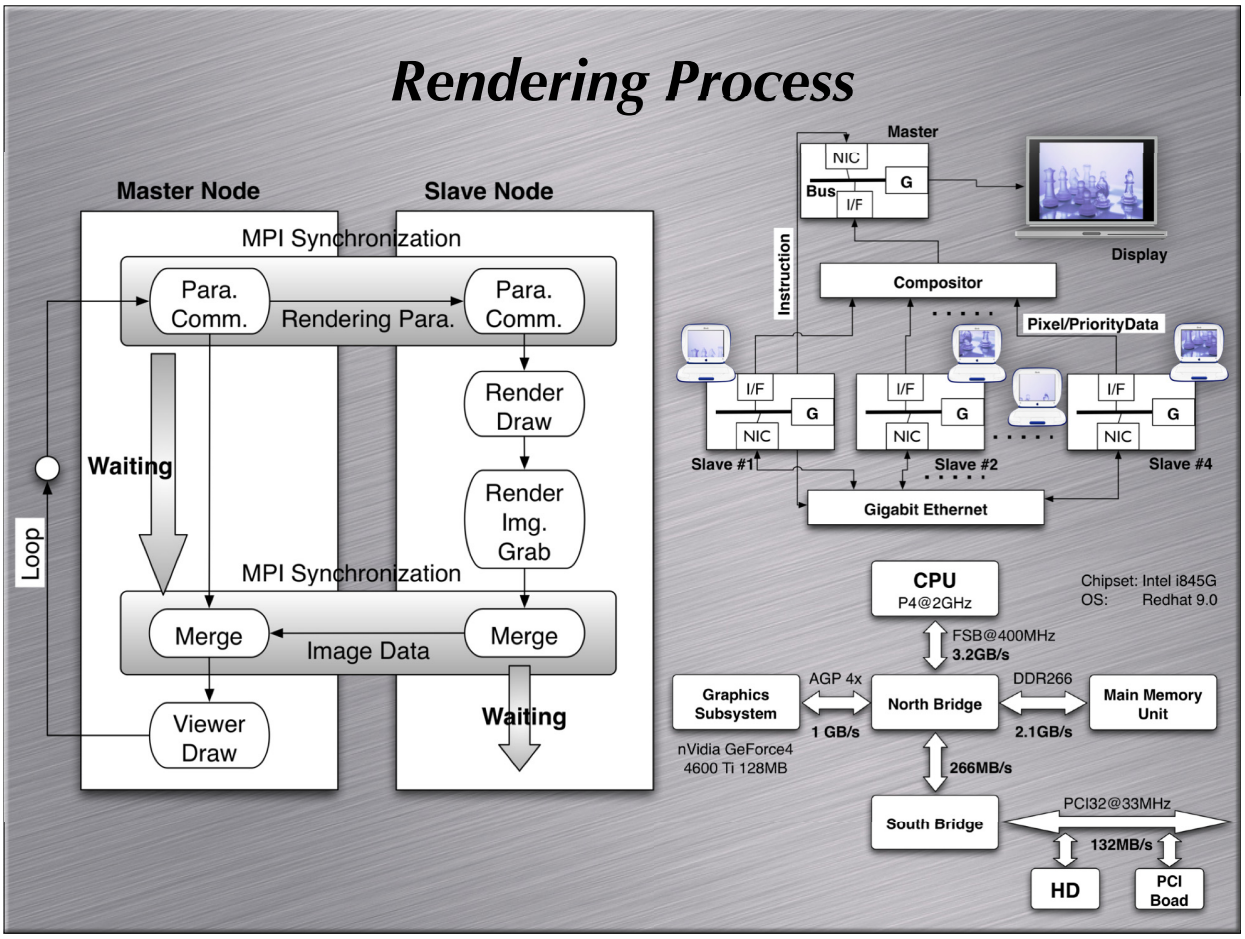
Interactive Demo.



More Large-scale Vis. on PC

- データ規模が大きくなると，処理時間が多大になる
- 並列化が処理時間の短縮の一方法
  - 様々な並列化（Sort Fast/Middle/Last）
  - ここでは，領域分割によるデータ並列
  - 個々の領域でレンダリングし，そのイメージを重畳
  - 重畳にはBSP木による高速化方法
  - 専用ハードウェアによる遅延の無い画像重畳


Parallel Rendering System
Master
NIC
Bus
G
I/F
Display
Instruction
Compositor
Pixel/PriorityData
I/F
G
NIC
Slave #1
Slave #2
Slave #4
Gigabit Ethernet


Rendering Process
Master Node
Slave Node
MPI Synchronization
Para. Comm.
Rendering Para.
Para. Comm.
Render Draw
Render Img. Grab
Waiting
Loop
MPI Synchronization
Merge
Image Data
Merge
Viewer Draw
Waiting
Master
Display
Instruction
Compositor
Pixel/PriorityData
Slave #1
Slave #2
Slave #4
Gigabit Ethernet
CPU
P4@2GHz
Chipset: Intel i845G
OS: Redhat 9.0
FSB@400MHz
3.2GB/s
Graphics Subsystem
AGP 4x
1 GB/s
North Bridge
DDR266
2.1GB/s
Main Memory Unit
nVidia GeForce4
4600 Ti 128MB
266MB/s
South Bridge
PCI32@33MHz
132MB/s
HD
PCI Boad

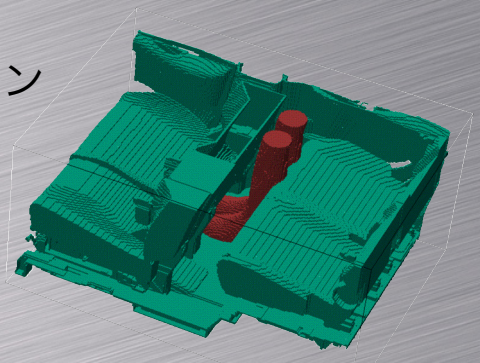
Hardware Specification

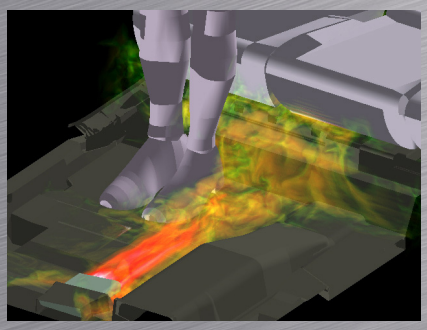
	Master Node	Slave Node
Chipset	Intel i860	Intel i845
CPU	Xeon 2.0GHz	P4 2.6GHz
MMU	1GB	1GB
GPU, nVidia,AGP4x	Quadro FX2000 (256MB)	GeForce4 4600 Ti (128MB)
Ethernet	1000 base	
OS	Redhat 9, kernel 2.4.20-8	
Disk	SCSI UW160	E-IDE

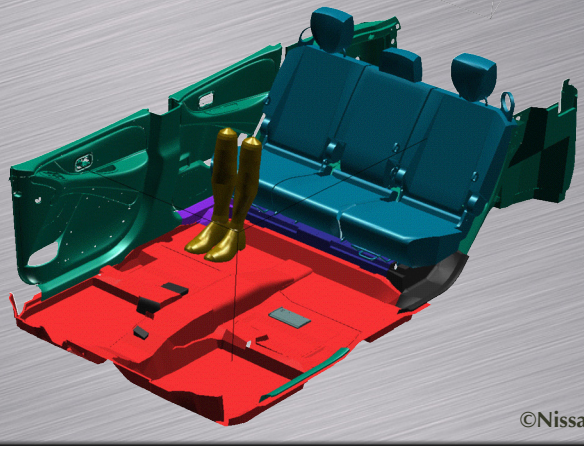
Example

- 車室内の空調風シミュレーション
  - $8x10^6$セル x 10^3step

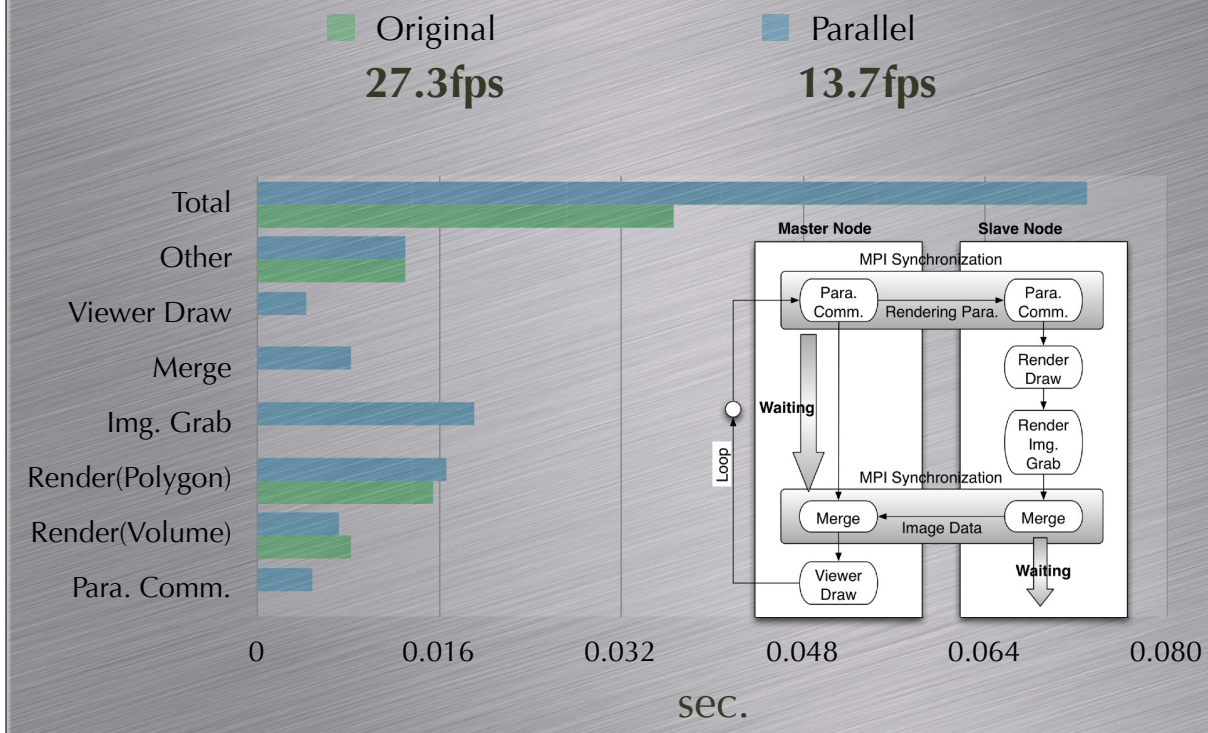
    > 8GB >Quantize> 1GB



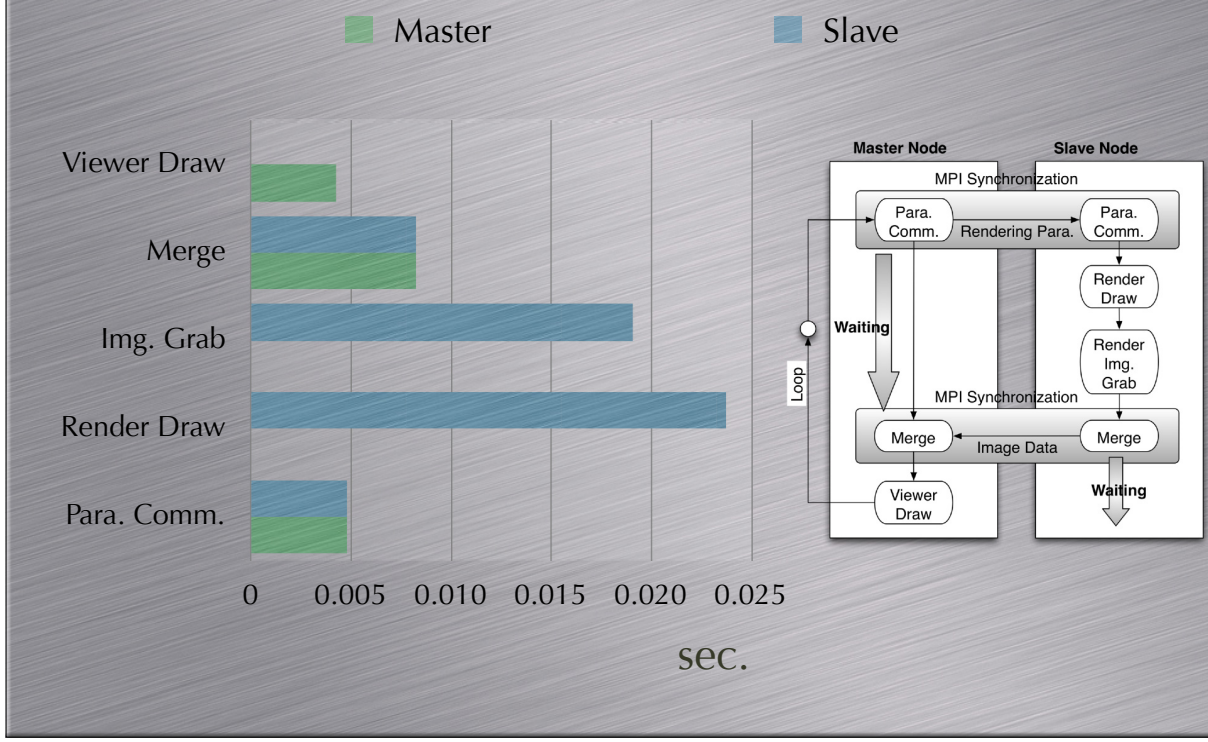




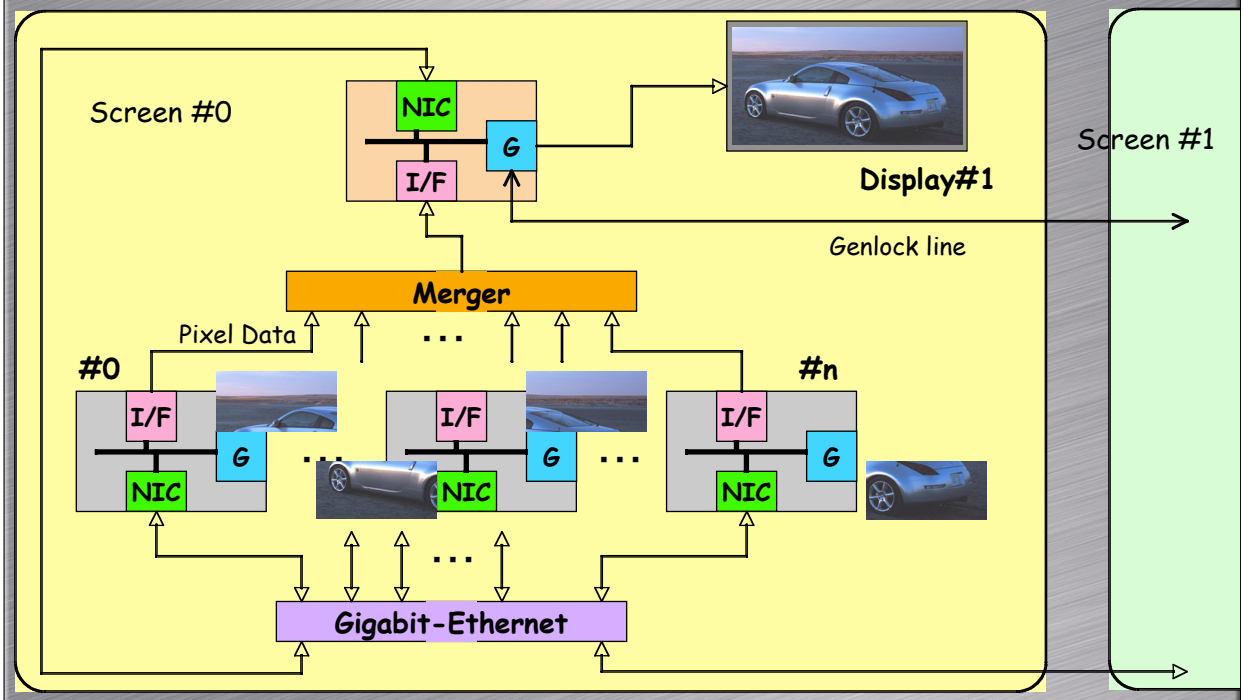
Parallel Performance



Timing of Each Process



Extension to Cabin System



まとめ

- 専用ハードウェアを用いた並列ボリューム/ポリゴンレンダリングシステムの性能評価
- 並列性能が見込めない場合でも，インタラクティブ性を確保できる
- イメージのリードバック処理に時間がかかる
- エンジニアリングには有効なツールとなる