

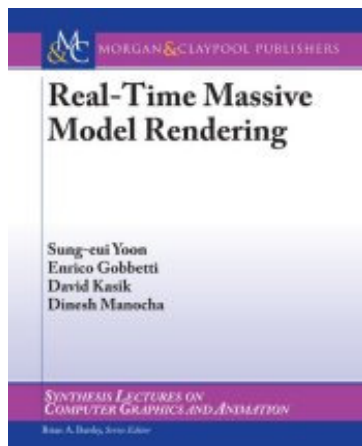
INF 2063 – Tópicos em CG III

Visualização de Modelos Massivos

Prof. Alberto Raposo
Tecgraf / DI / PUC-Rio



Livro



Real-Time Massive Model Rendering

**Sung-eui Yoon, Enrico Gobbetti, David Kasik, and
Dinesh Manocha**

**2008, 122 pages
Morgan & Claypool Publishers**

(Amazon: \$31,58 em 4/3/2010)

Outros textos

- D.J. Kasik et al., “Massive Model Visualization Techniques: Course Notes,” ACM Siggraph Classes, ACM Press, 2008, pp. 1–188.
- I. Wald et al., “State of the Art in Ray Tracing Animated Scenes,” Computer Graphics Forum, vol. 28, no. 6, 2009, pp. 1691–1722.
- Vários artigos e tutoriais referenciados durante o curso.

Aula 01

Introdução à Renderização de Modelos Massivos

Explosão de Dados

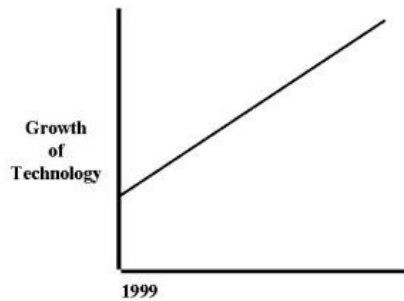
- Todas as mídias de armazenamento produziram 5 exabytes de novas informações em 2002
 - 92% disso foi armazenado em meio magnético (principalmente hard disks)
- Essa quantidade de informação nova é o dobro do armazenado em 1999
- Fluxo de informação através dos canais eletrônicos (telefone, rádio, TV, e Internet) foi de ~18 exabytes em 2002
 - Isso é 3 1/2 vezes maior que o que foi armazenado
 - 98% é voz e dados enviados por telefone (fixo e móvel)

<http://www2.sims.berkeley.edu/research/projects/how-much-info/>

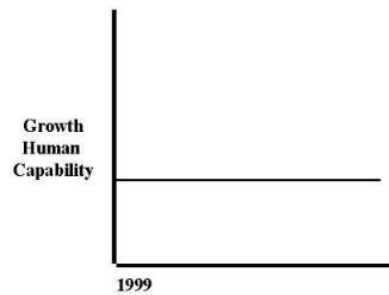
Exabytes

Name	Value	Equivalent
Kilobyte (KB)	1,000 bytes, 10^3	Typewritten page
Megabyte (MB)	1,000,000 bytes, 10^6	Small novel
Gigabyte (GB)	1,000,000,000 bytes, 10^9	Pickup truck filled with books
Terabyte (TB)	1,000,000,000,000 bytes, 10^{12}	2 TB: academic research library
Petabyte (PB)	1,000,000,000,000,000 bytes, 10^{15}	200 PB: All printed material
Exabyte (EB)	1,000,000,000,000,000,000 bytes, 10^{18}	5 EB: All words ever spoken

Dilema

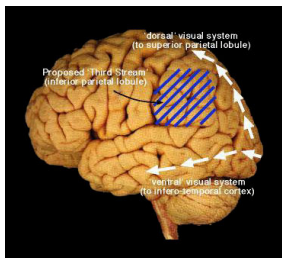


Moore's Law



God's Law

Processador de comunicação visual humano

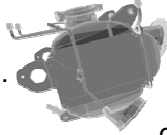


- O sistema visual humano provê maneira extremamente eficiente de comunicar o contexto e o detalhe
- As pessoas podem olhar para quantidades massivas de dados e adicionar significados com interação e construtos visuais próprios

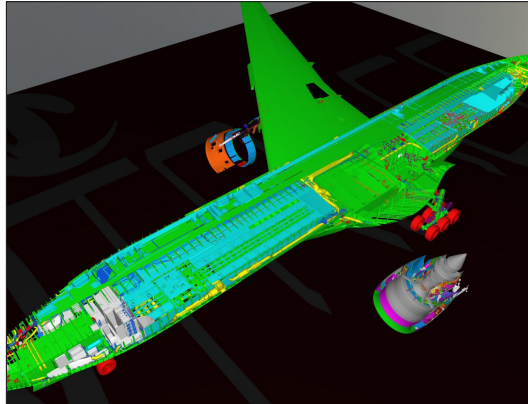
- Complexos datasets visuais, como o Google Earth e o Visible Human, se tornaram a norma nos seus respectivos domínios
- “A visão dá aos humanos o canal de entrada mais largo e flexível para o cérebro”

Exemplo: Análise Visual

- Encontre isso...



dentro disso...



Definição de “Massive”

1. Consisting of or making up a large mass; bulky, heavy, and solid: *a massive piece of furniture.*
2. Large or imposing, as in quantity, scope, degree, intensity, or scale: *"Local defense must be reinforced by the further deterrent of massive retaliatory power"* (John Foster Dulles).
3. Large in comparison with the usual amount: *a massive dose of a drug.*

The American Heritage Dictionary of the English Language, 4th edition, published by Houghton Mifflin Company

Modelos Massivos

- Podem ser massivos nos 3 sentidos anteriores:
 1. Alto nível de detalhes, que não pode ser visto pelo olho humano sem ampliação
 2. Dados consomem centenas de GB ou TB para armazenamento, possuem bilhões de primitivas geométricas e têm unidades de medidas variando de angstroms a anos-luz
 3. Dados excedem a capacidade convencional de processamento e armazenamento

Modelos Massivos

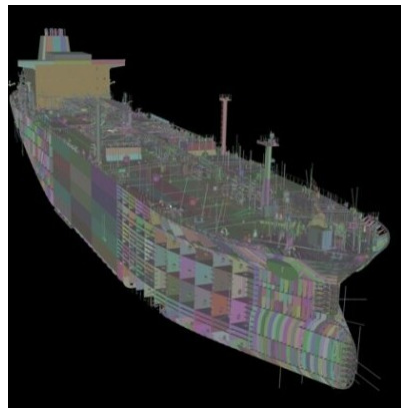
- Tamanho: de 1 a 1 bilhão de triângulos
 - Isso dá até 26 GB de raw data!
- Datasets com bilhões de polígonos estão se tornando disponíveis
- “Naïve rendering” não é rápido o suficiente
- Continuam querendo renderização em tempo real

Exemplo



Paisagem com mais de 1 bilhão de polígonos – Saarland University

Exemplo



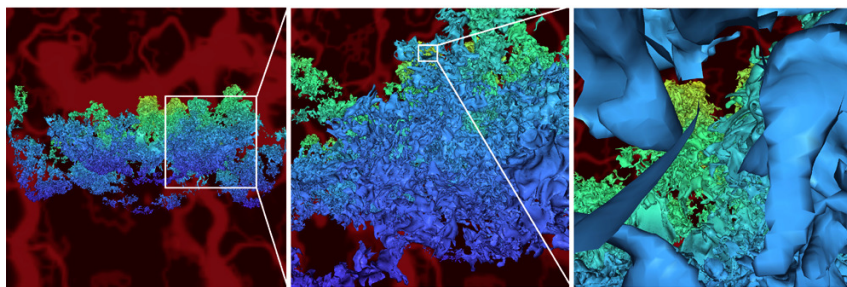
Petroleiro com 82 milhões de triângulos – UNC / Newport News Shipbuilding)

Exemplo



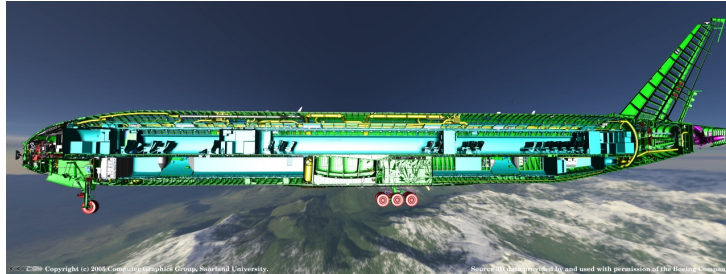
Scan da St. Matthew (Michelangelo), 372 milhões de triângulos (10GB)

Exemplo



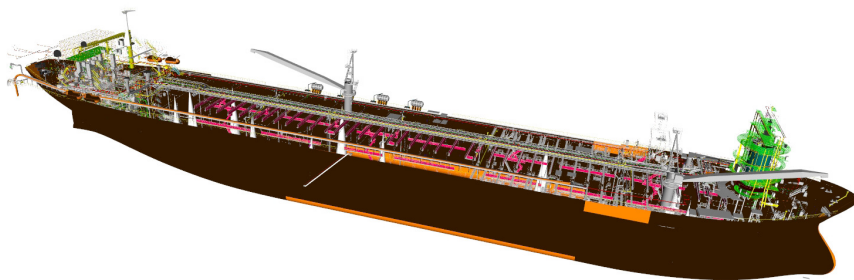
Iso-superfície com 100 milhões de triângulos

Exemplo



Boeing 777 - 470 milhões de polígonos (14 GB)

Exemplo



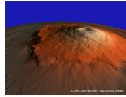
P38 (Petrobras) – 38 milhões de triângulos (1.94 GB)

Domínios de Aplicação / Fontes de Dados



Local Terrain Models

2.5D – Flat – Dense regular sampling



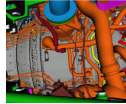
Planetary terrain models

2.5D – Spherical – Dense regular sampling



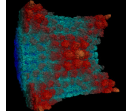
Laser scanned models

3D – Moderately simple topology – low depth complexity - dense



CAD models

3D – complex topology – high depth complexity – structured - 'ugly' mesh



Natural objects / Simulation results

3D – complex topology + high depth complexity + unstructured/high frequency details

- Many important application domains
- Today's models easily exceed
 - $O(10^8-10^{10})$ samples
 - $O(10^9-10^{11})$ bytes
- Varying
 - Dimensionality
 - Topology
 - Sampling distribution

Visualização de Modelos Massivos

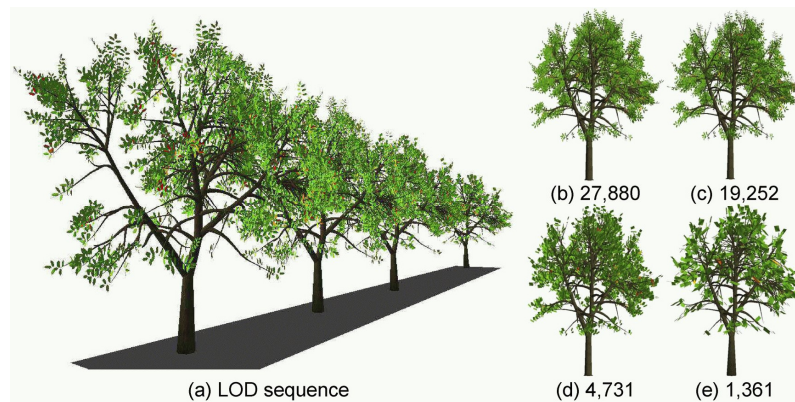
- Tenta prover aos usuários a capacidade de interagir com modelos 3D de tamanho e complexidade praticamente ilimitados
 - Principalmente em relação à geometria
 - Foto-realismo (iluminação, etc): demanda crescente atualmente

Desempenho Interativo

- Prover uma taxa de quadros por segundo (fps) rápida o suficiente para convencer o sistema visual do usuário de que o movimento é contínuo
- Desempenho interativo inclui também
 - Tempo de carregamento do modelo
 - Seleção de objetos com feedback (hand-eye coordination)
- Números variados
 - Navegação: 16 a 24 fps
 - Estudo da Boeing: 16 fps (útil), 10 fps (aceitável).
 - Carregamento de modelos: 1 min (até 5 min, em alguns casos)
 - Feedback de seleção: < 0,25s

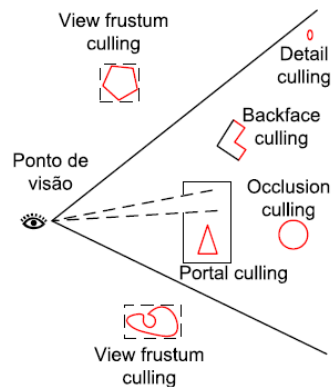
Inovações históricas importantes em CG

- LOD (níveis de detalhes)
 - [Clark 1976]



Inovações históricas importantes em CG

- Culling



Inovações históricas importantes em CG

- Gerenciamento de memória

- A partir de 2000
- Várias abordagens ortogonais
 - Cache-coherent layouts
 - Random-accessible compressed meshes
 - Cache-oblivious ray reordering
 - Hybrid parallel continuous collision detection

2 linhas atuais

- GPU-based rasterization
 - Fazer uso paralelo das centenas de fragment processors das GPUs modernas
- Ray-tracing interativo
 - Ray-tracing “ressurge” devido ao crescimento exponencial da capacidade de processamento e tendência de arquiteturas multi-core
 - É algoritmo facilmente paralelizável e adaptável às arquiteturas multi-core/multi-processor

Solução é Sistêmica

- Renderização interativa de modelos massivos que atinja nível de desempenho consistente e sustentável requer solução em nível de sistema
 - Atacar apenas um ou dois aspectos do sistema pode fazer com que os outros falhem
- Soluções para uma classe de modelos massivos podem não funcionar bem para outras classes

Performance bottlenecks

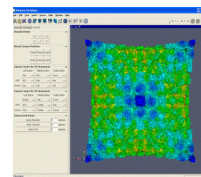
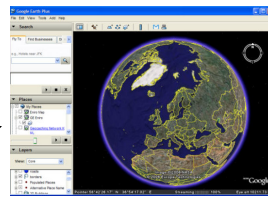
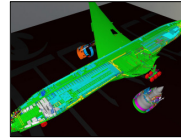
- Converter muita quantidade de dados 3D em pixels para determinar o que está visível ou não
- Reduzir a complexidade dos dados a serem processados (frame a frame)
 - Adaptive mesh simplification
 - Representações alternativas
- Criar métodos eficientes de storage-to-memory

Outros desafios

- Métodos de aquisição e modelagem de dados
- Time-dependent models
- Preparação dos dados (pré-processing)
- Técnicas de programação para multi-processadores, multi-core, multi-threading
- Estratégias de distribuição para quantidades massivas de dados
- Qualidade e interoperabilidade dos dados
- Hardware dedicado à visualização interativa

Tipos de Organização de Dados

- Dados organizados arbitrariamente
- Dados espacialmente coerentes
- Dados geograficamente coerentes
- Visualização Científica



Dados organizados arbitrariamente

- Tipicamente: modelos CAD
 - Complexidade dos modelos CAD cresceu porque eles têm se tornado a “build authority”, e não mais os desenhos 2D
- As peças individuais são coerentes, mas o modelo como um todo não é organizado espacialmente (hierarquia funcional, e não espacial)
 - São construídos como nos projetos de engenharia: por sistemas (hidráulicos, cabos, dutos, etc)
 - Esses sistemas são geralmente diferenciados por cores (sem necessidade de foto-realismo)
- Problema: dificulta algoritmos de otimização de rendering, geralmente baseados em organização espacial
 - Atenuante: dá para criar uma organização espacial em pré-processamento

Dados Coerentes Espacialmente

- Polígonos (ou voxels) são distribuídos regularmente no espaço
- Exemplos típicos: scanners 3D, iso-superfícies, terrenos
- Problema: por serem muitas vezes um objeto contínuo, algoritmos como LOD estático ou culling de objetos não funcionam

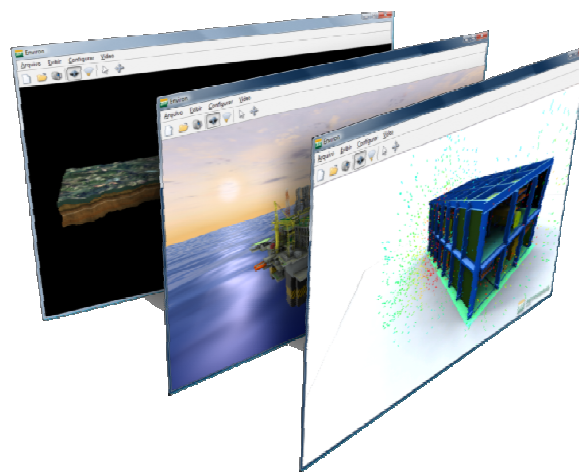
Dados Coerentes Geograficamente

- Relativamente poucos dados 3D (topografia do terreno), mas com dados massivos de textura (fotografias aéreas, fotos de ruas)
 - Malha espacialmente coerente e geograficamente coordenada com sua posição na Terra
 - Texturas correlacionadas com a malha
- Exemplos: Google Earth, Microsoft Live Earth

Visualização Científica

- Desenvolvimento de técnicas gráficas para fazer o usuário entender mais claramente algo sobre algum tipo de dado
- Diferentemente dos demais, são altamente dinâmicos
 - Coerência espacial via pré-processamento não serve

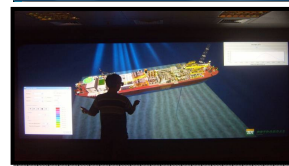
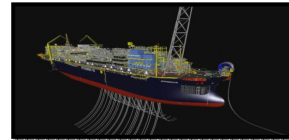
Nossa experiência com modelos massivos CAD



Environ

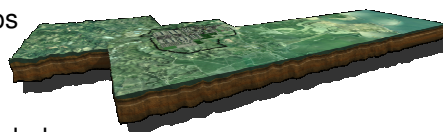
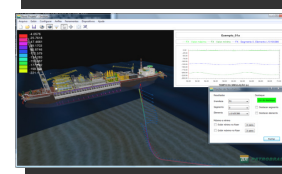
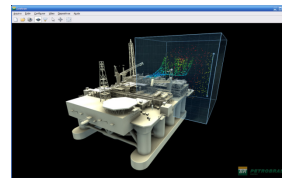
Principais Funcionalidades

- Visualização em Tempo Real de Modelos Massivos de Engenharia
 - Otimizações
 - Capacidade de visualização de modelos grandes
 - Interoperabilidade
 - Facilidade para conversão CAD ↔ RV
 - CAD/CAE (PDS/Microstation e PDMS/Aveva)
 - Visualização de informações eng. do PDS e do PDMS
- Foto-realismo
 - Efeitos de Ambiente 3D, sombras, shaders, etc
 - Foto-realismo associado as informações de projeto para agregar valor as atividades de engenharia
- Visualização Imersiva
 - Suporte a visualização estéreo
 - Dispositivos de tracking: BraTrack, Flock of birds
 - Suporte a interação 3D – vrInput/Viral e VRPN
 - Suporte a visualização multifrustum - vrOutput



Principais Funcionalidades

- Visualização de Simulações de Engenharia
 - CFD para dispersão de gases em plantas de processo;
 - Análise de Risers Rígidos e Flexíveis;
 - Projeto de Controle Anti-corrosivo (cálculo área pintura)
- Manipulação de Modelos
 - Movimentação de objetos
 - Otimizador de modelos - TecOptimizer
 - Remoção de elementos redundantes
- Suporte a Visualização de Terrenos
 - Modelo shapefile e geotiff
- Geração de vídeos das cenas simuladas

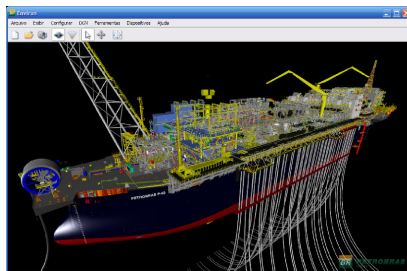


Environ – Modelos CAD

- Visualização em tempo real de resultados de engenharia
- Importação de diversos formatos: PDS, PDMS, Autocad



Foto Realismo



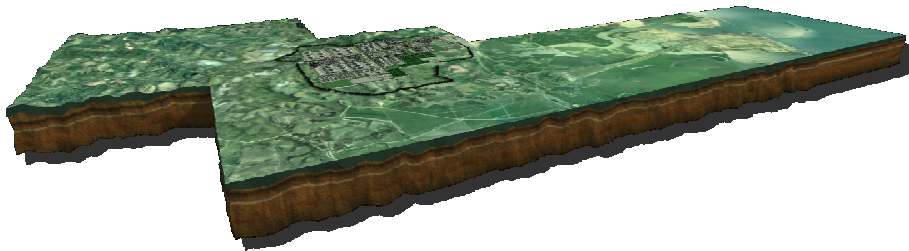
Iluminação



Gera ambientes configuráveis e com realismo (céu e mar)

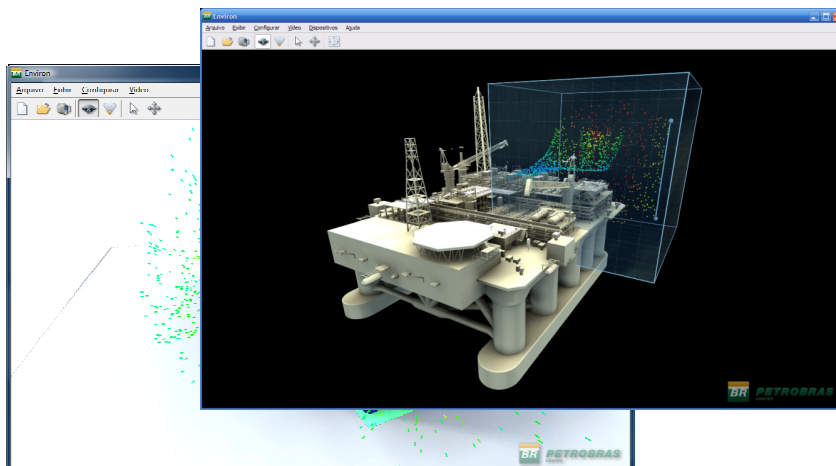
Terrenos

- Suporte a visualizações de dados reais de terreno
 - Altimetria
 - Imagens de satélite



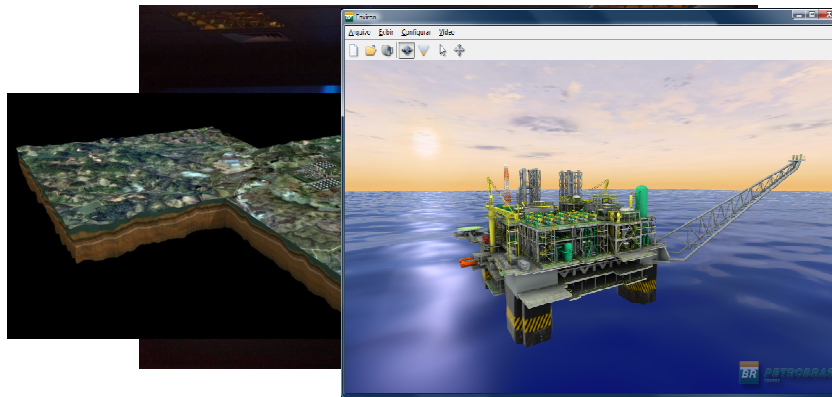
Visualização CFD

- Suporte a visualizações de dinâmica de fluidos geradas no FLACS



Desktop & Salas de RV

- O Environ pode ser usado tanto em ambientes desktop quanto em Salas de Realidade Virtual



Vídeo Environ

SIGGRAPH 2008
Animation Festival

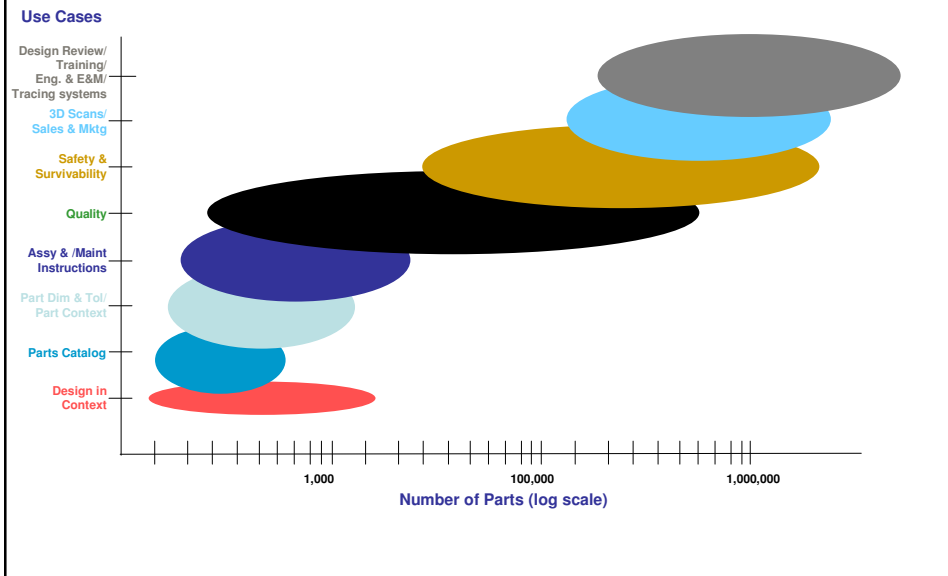
Análise visual de tarefa

- Focar no objeto encontrado para entender suas características de superfície (e.g., smoothness, roughness)
- Encontrado o objeto, olhar o volume no entorno para encontrar
 - Discrepâncias no espaço consumido
 - Interferências entre objetos
 - “buracos”
 - Caminhos para encontrar a outra ponta de objetos conectados
- Observar a dinâmica da cena como um todo (normalmente usando animação)
- Trabalhar com múltiplas versões do mesmo conjunto de objetos para comparar 2 conjuntos

Aplicações Potenciais

- Design in context
- Design reviews/error checking
- Tracing systems
- Engineering analysis
- Electro-magnetic analysis
- Safety
- Survivability
- Complete part context
- Massive 3D scans
- Part dimensions and tolerances
- Quality inspection
- Assembly instructions
- Part catalogs
- Training and familiarization
- Maintenance instructions
- Sales and marketing
- Basically, any process where quick navigation is needed to go anywhere in a digital model

O quanto você precisa ver?



Alternativas de Arquiteturas para Visualização de Modelos Massivos

- Local Data
- Remote Data
- Virtual Terminal

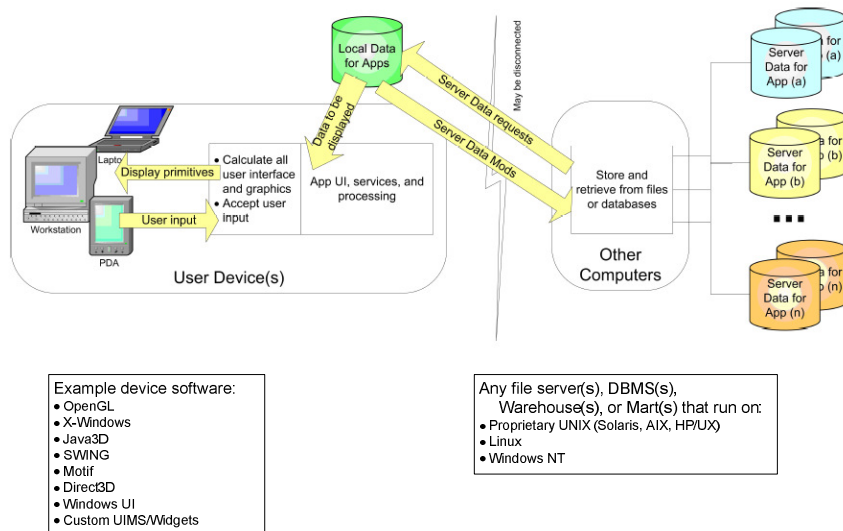
O problema, de uma maneira geral, é a cópia dos dados da unidade (secundária) de armazenamento massivo para a RAM (CPU) ou VRAM (GPU).

- Armazenamento remoto provê, em tese, capacidade ilimitada de armazenamento, mas depende de transferência em rede
- Armazenamento local ganha na transferência para a memória, mas pode perder na capacidade de armazenamento e processamento

Local Data

- Solução mais comum
- Rendering usa apenas dados armazenados em unidades locais (por exemplo, um HD rápido)
 - Visualização e armazenamento na mesma máquina
 - Ocasionalmente, software consulta servidor sobre atualizações
- Vantagens
 - Pode-se trabalhar desconectado
 - Desempenho geralmente bom
- Desvantagens
 - Primeiro carregamento do modelo pode demorar horas
 - Geralmente depende de ação de usuário para atualizar dados do servidor
 - Necessária capacidade de modificar dados alterados desde o último refresh. Caso contrário, seria necessário a transferência de multi-GB pela rede a cada instante

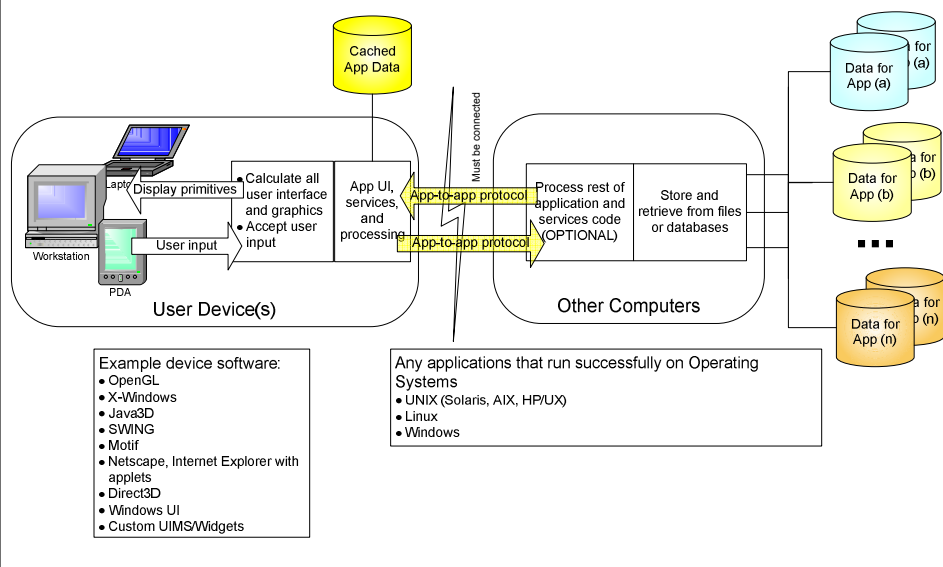
Local Data



Remote Data

- Funciona com cache, a grosso modo, como um Web Browser: antes de abrir a página, verifica se ela está em cache local, se precisa atualizá-la (e também pode ter software para apagar do cache o que não é acessado há algum tempo), etc
- Vantagem: evita necessidade de carregamento demorado inicial
- Desvantagens
 - Se tamanho do cache for menor que o necessário para a aplicação de visualização, é necessário trabalhar com muitos “deletes” e “refreshs” no cache: novamente problema de dowload dos servidores
 - O cache precisa ser checado e a informação necessária para a visualização precisa ser carregada em menos de 0.1s
 - Desempenho pode se tornar imprevisível
 - Precisa necessariamente trabalhar conectado

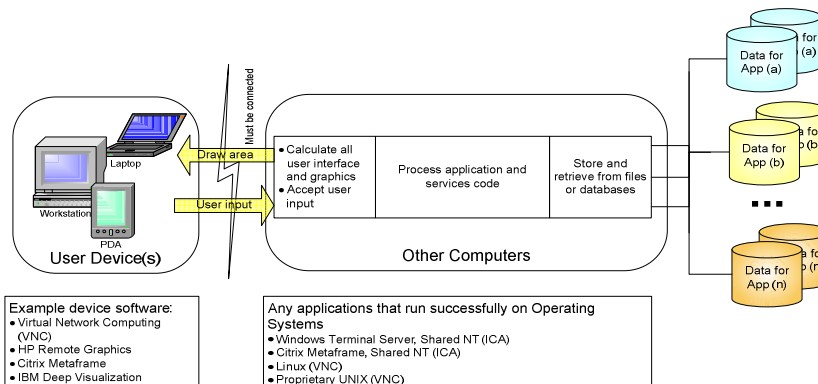
Remote Data



Virtual Terminal / Thin Client

- Todo o processamento (inclusive a renderização) é feita no servidor
- O que trafega pela rede são os inputs do usuário e os bitmaps
- Vantagens:
 - Altamente escalável, pois independente do tamanho do modelo, os bitmaps têm tamanho limitado ao da janela
 - Bitmaps podem ser comprimidos
- Desvantagens:
 - A cada quadro (menos de 0.1s), é necessário: processar o input, renderizar, preparar e empacotar o bitmap, transmiti-lo (servidor) e desempacotá-lo (cliente).
 - Depende de tráfego em rede: desempenho pode se tornar imprevisível

Virtual Terminal



Links

- Gamma Group – Univ. North Carolina
 - <http://gamma.cs.unc.edu/>
 - <http://gamma.cs.unc.edu/research/rendering/>
- CRS4 Visual Computing
 - <http://www.crs4.it/vic/>
- SGLab: Scalable Graphics / Geometric Algorithm Lab – Korea Advanced Inst. of Science and Technology
 - <http://sglab.kaist.ac.kr/>