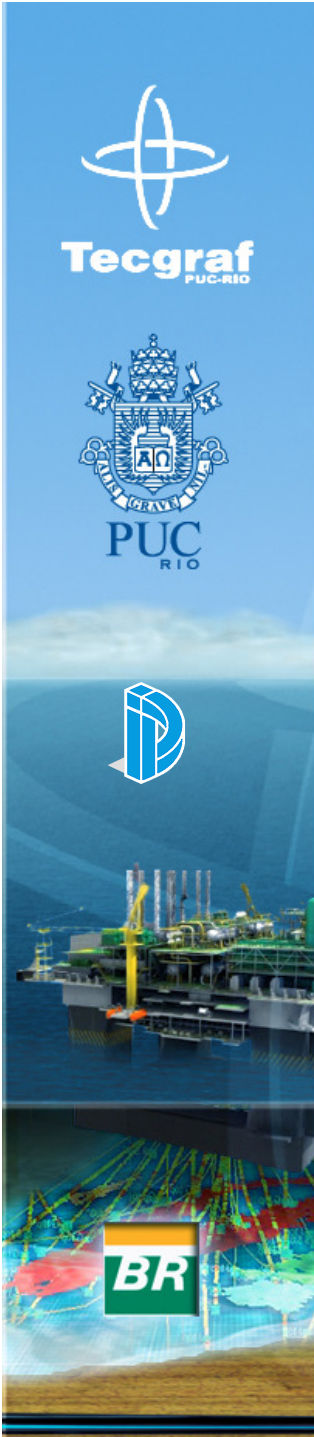


Modelos Massivos e Multi-Escala em Realidade Virtual

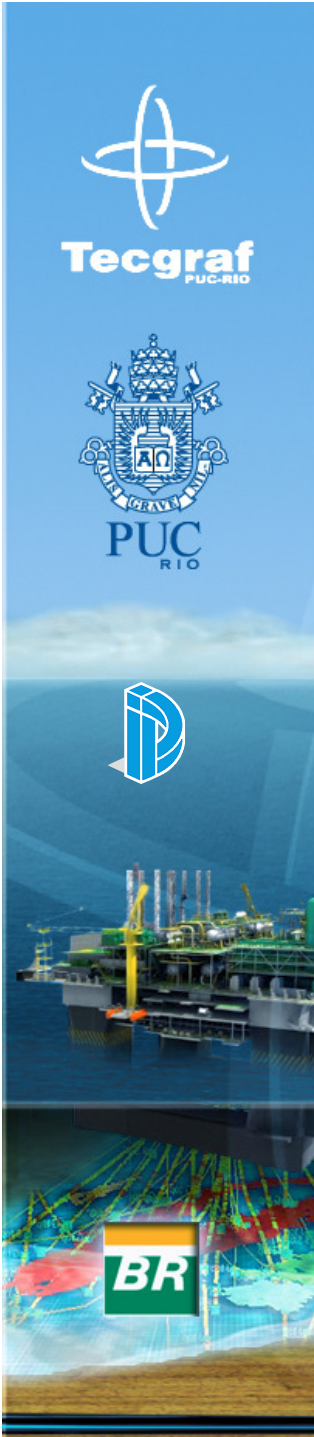


Alberto Raposo

Tecgraf – Grupo de Tecnologia em Computação
Gráfica

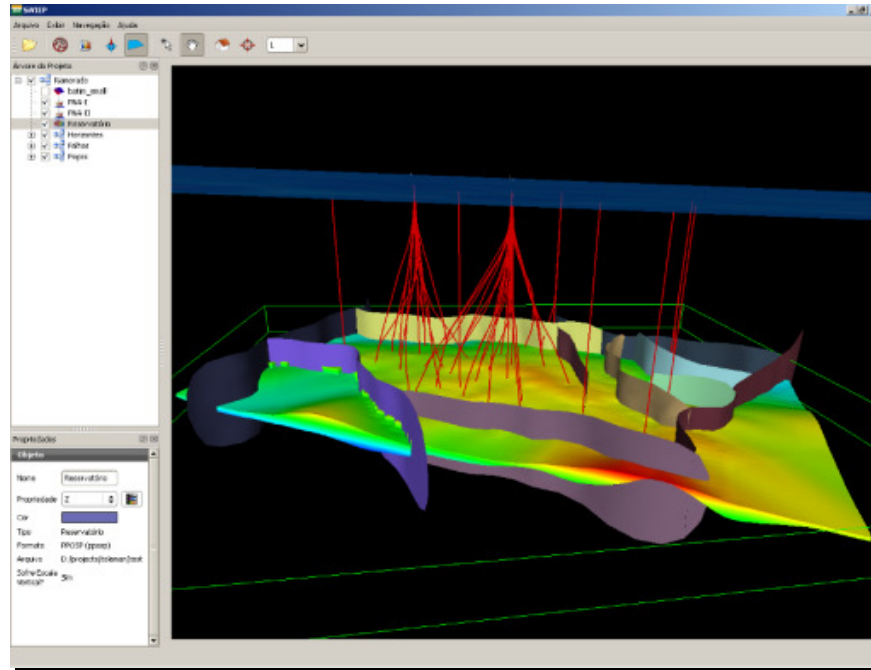
Departamento de Informática,
Pontifícia Universidade Católica do Rio de Janeiro

<http://www.inf.puc-rio.br/~abraposo>

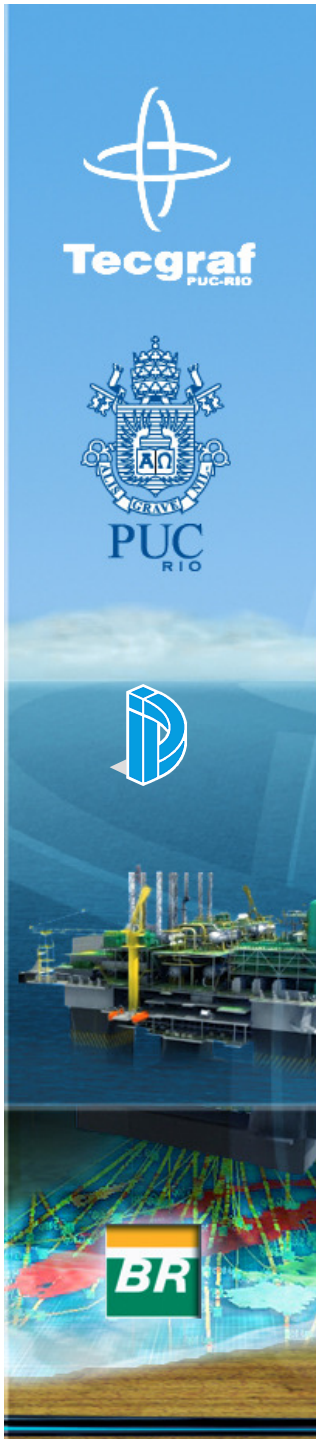


Realidade Virtual

Novas tecnologias possibilitam a criação de ambientes virtuais 3D cada vez maiores

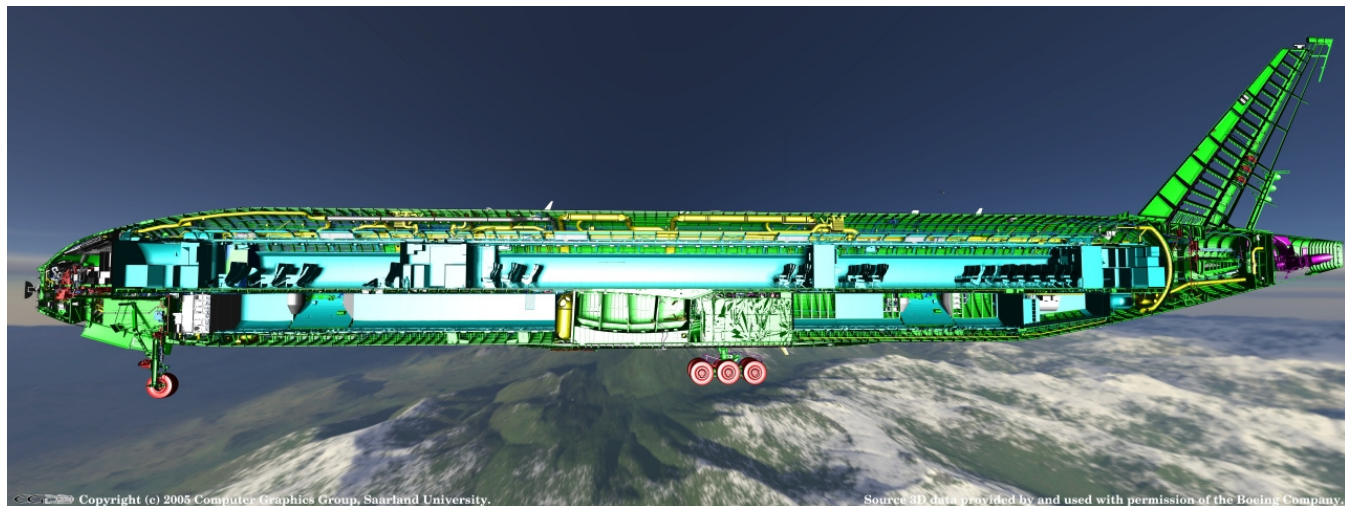


Modelos Massivos
Modelos Multi-Escala



Modelos Massivos (1)

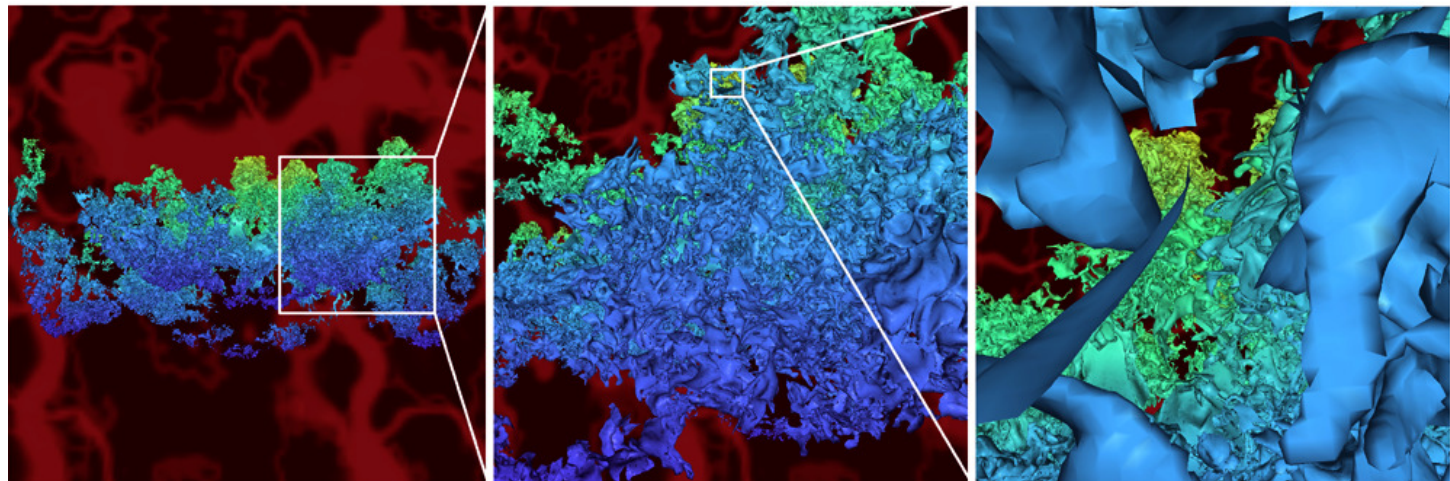
- Podem ser massivos nos 3 sentidos:
 1. Alto nível de detalhes, que não pode ser visto pelo olho humano sem ampliação
 2. Dados consomem centenas de GB ou TB para armazenamento, possuem bilhões de primitivas geométricas e têm unidades de medidas variando de angstroms a anos-luz
 3. Dados excedem a capacidade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



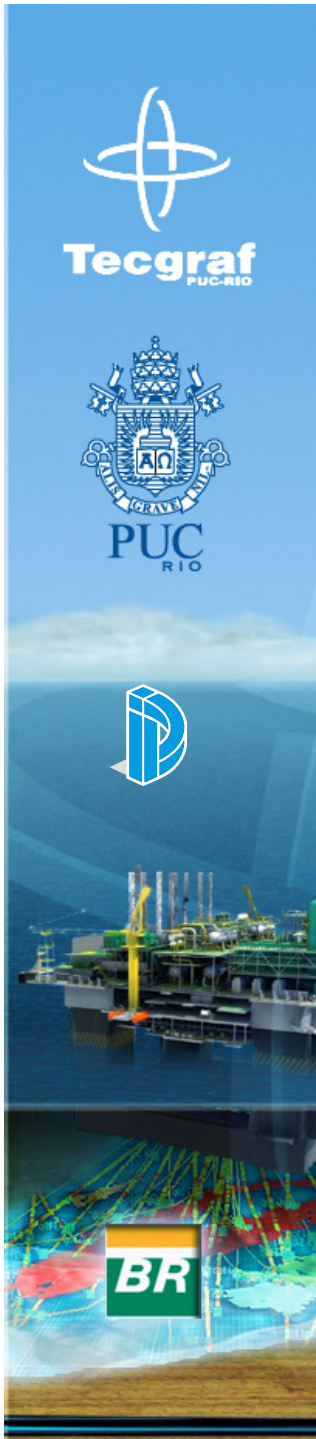
Boeing 777 - 470 milhões de polígonos (14 GB)

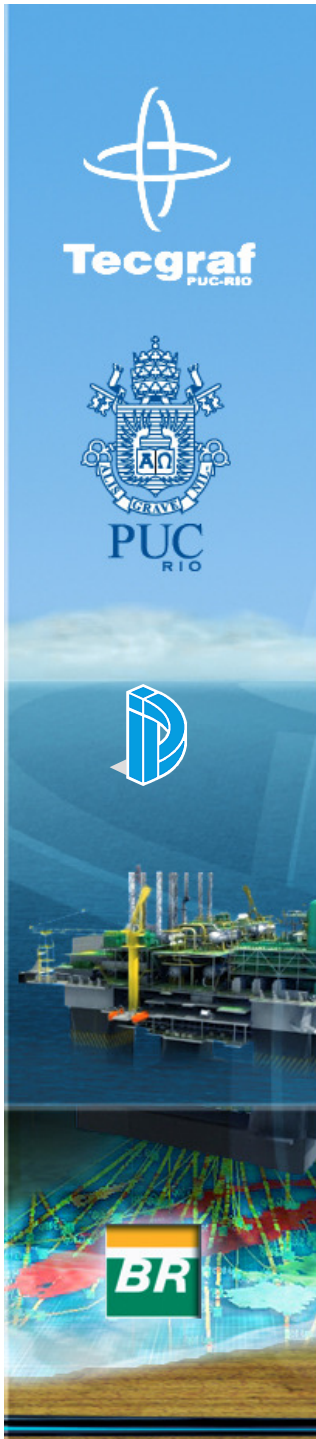
Modelos Massivos (2)

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



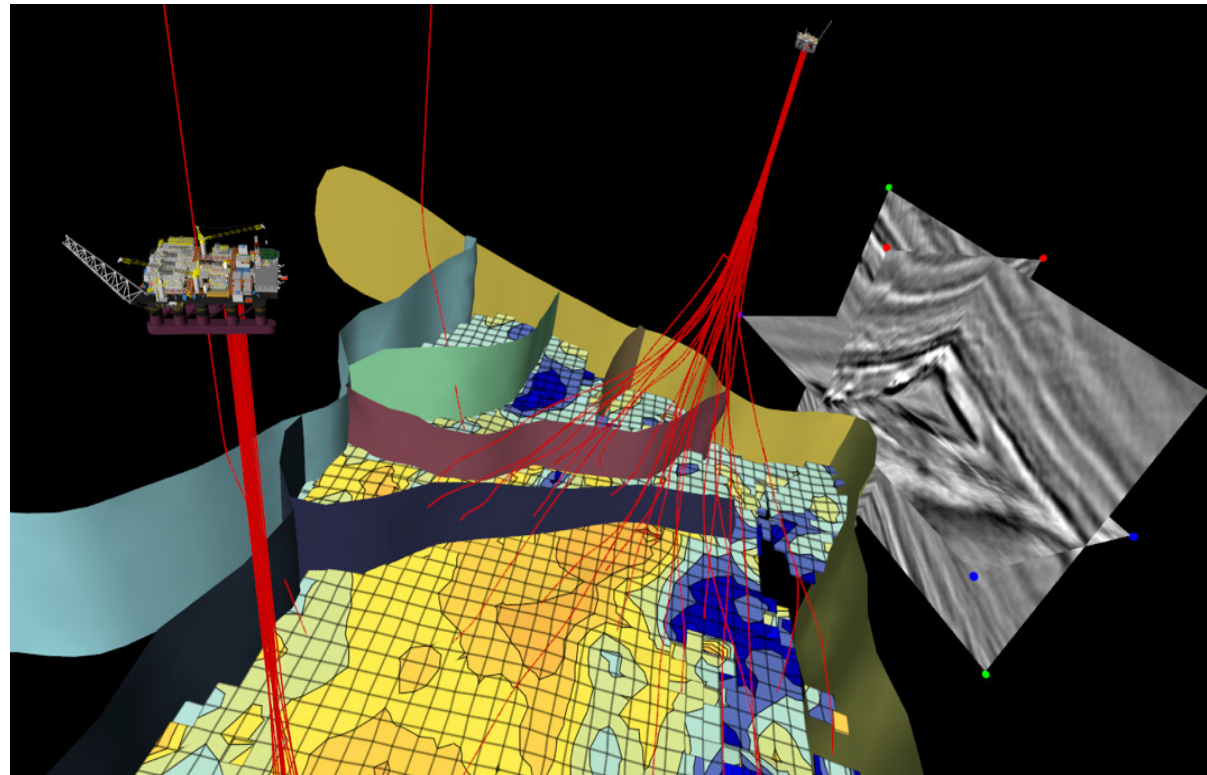
Iso-superfície com 100 milhões de triângulos

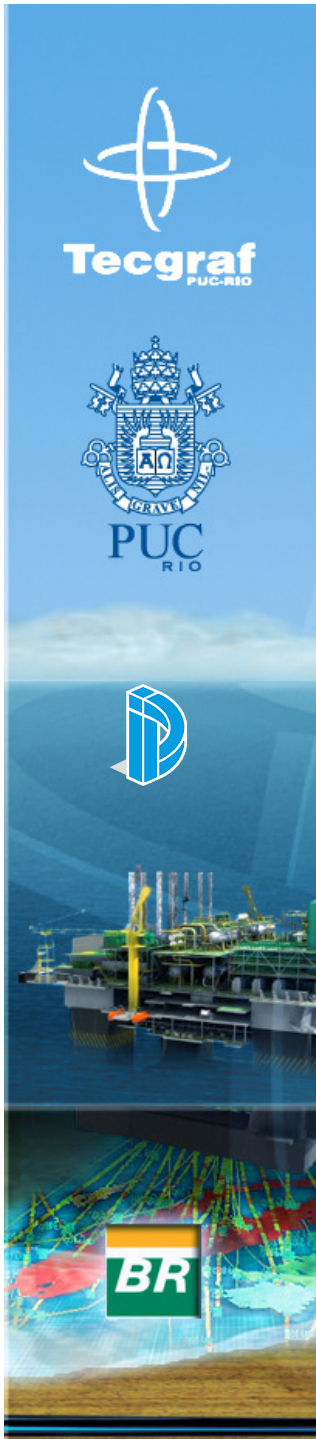




Modelos Multi-Escala

A informação pode existir em vários níveis de detalhes – por exemplo, a cena pode conter objetos que vão desde um pequeno parafuso até campos de exploração de petróleo.





Modelos Massivos vs Multi-Escala

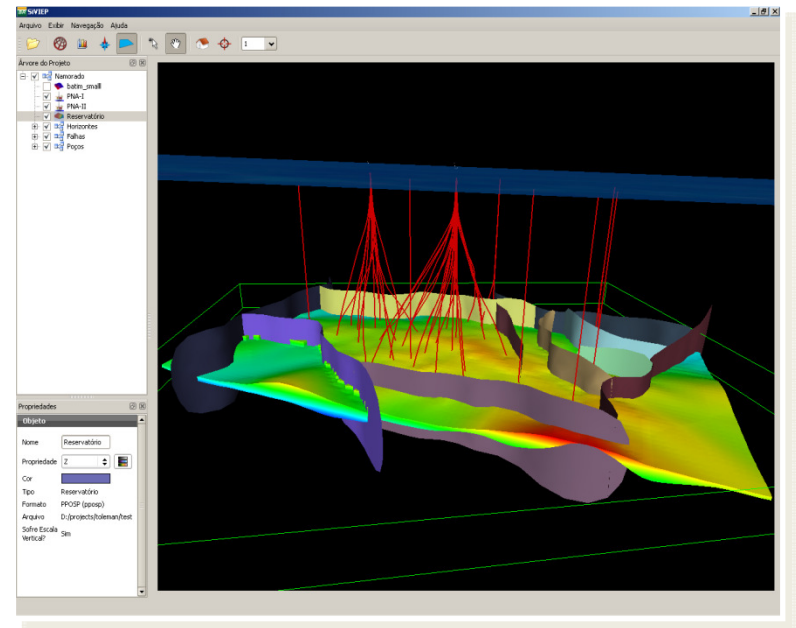
Embora possam estar relacionados, são conceitos independentes

Massivo e não multi-escala

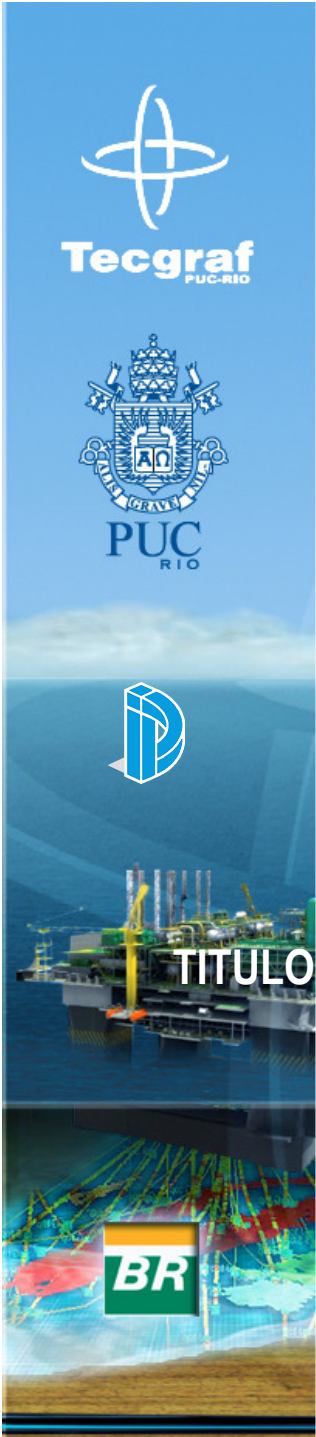


**Scan da St. Matthew (Michelangelo),
372 milhões de triângulos (10GB)**

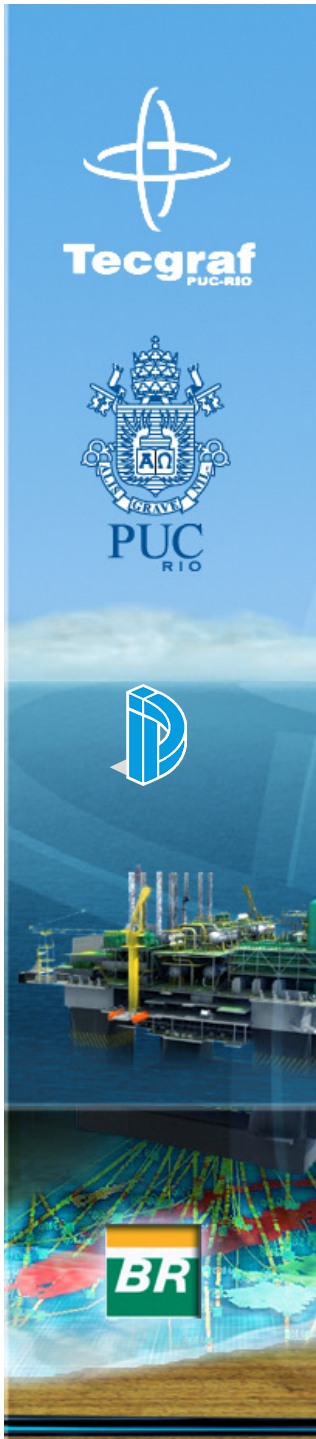
Multi-escala e não massivo



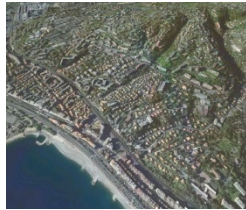
**Campo de namorado, Petrobras
Menos de 1 milhão de triângulos**



Modelos Massivos



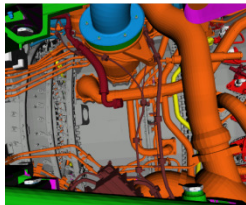
Domínios de Aplicação / Fontes de Dados



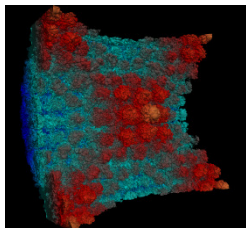
Modelos de terrenos



Modelos escaneados a laser



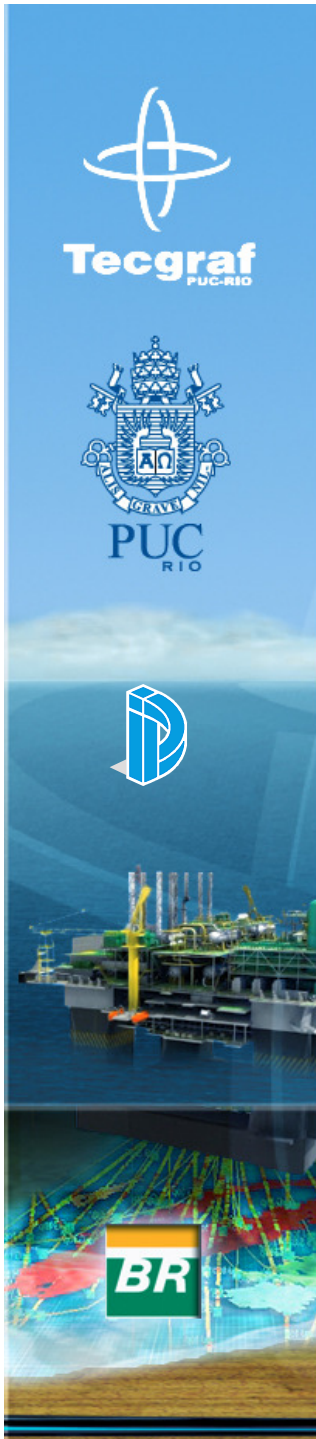
Modelos CAD



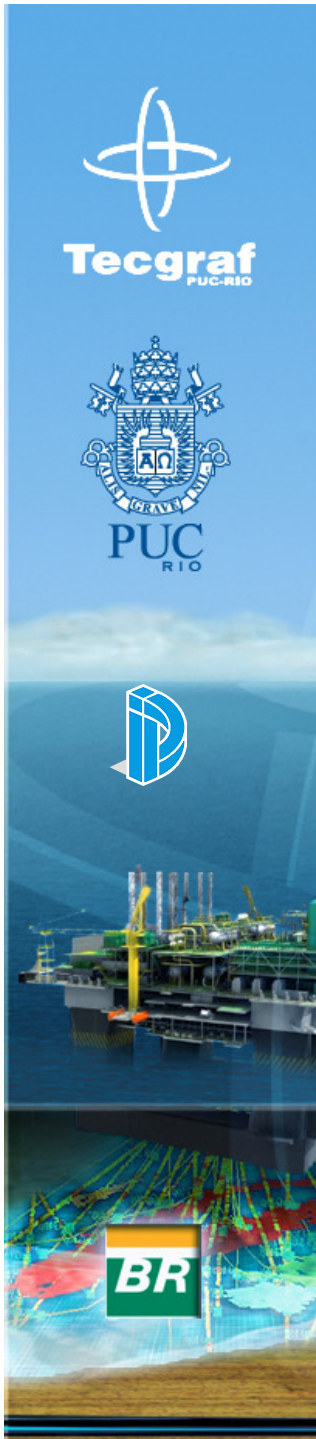
Objetos naturais / Simulações

- Vários domínios de aplicação importantes
- Modelos atuais excedem facilmente
 - $O(10^8-10^{10})$ amostras
 - $O(10^9-10^{11})$ bytes
- Variações
 - Na dimensionalidade (2.5D, 3D)
 - Na topologia
 - Na distribuição das amostras

Visualização de Modelos Massivos

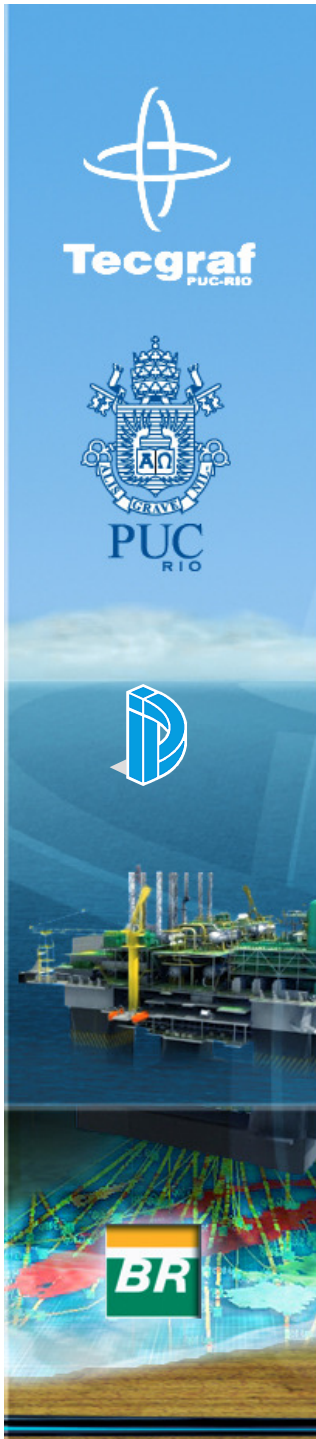


- Tenta prover aos usuários a capacidade de interagir com modelos 3D de tamanho e complexidade praticamente ilimitados
 - Principalmente em relação à geometria
 - Foto-realismo (iluminação, etc): demanda crescente atualmente

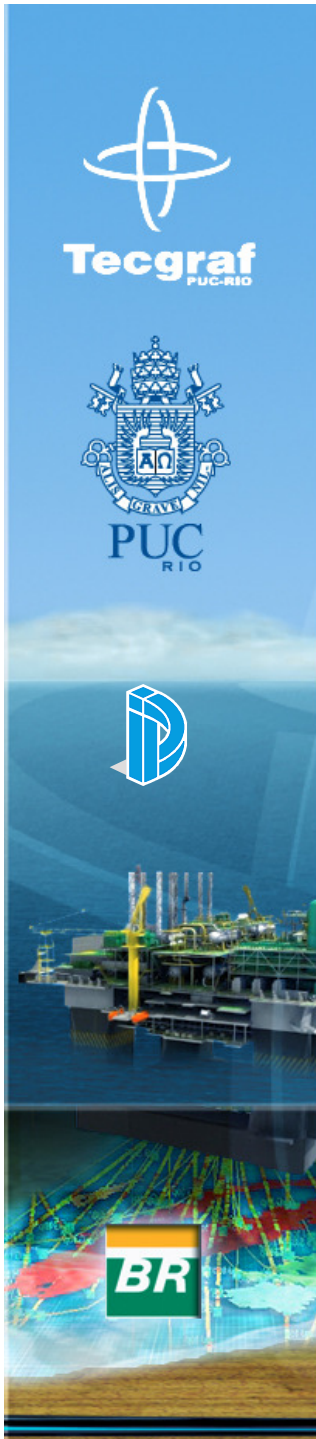


Desempenho Interativo

- 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

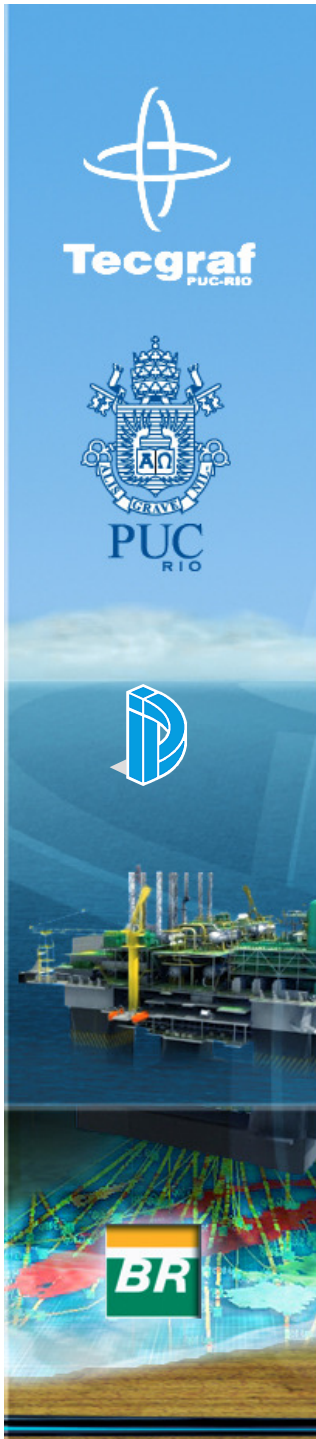


- GPU-based rasterization
 - Fazer uso paralelo das centenas de fragment processors das GPUs modernas
- Ray-tracing interativo
 - Ray-tracing “ressurge” devido ao crescimento exponencial da capacidade de processamento e tendência de arquiteturas multi-core
 - É algoritmo facilmente paralelizável e adaptável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



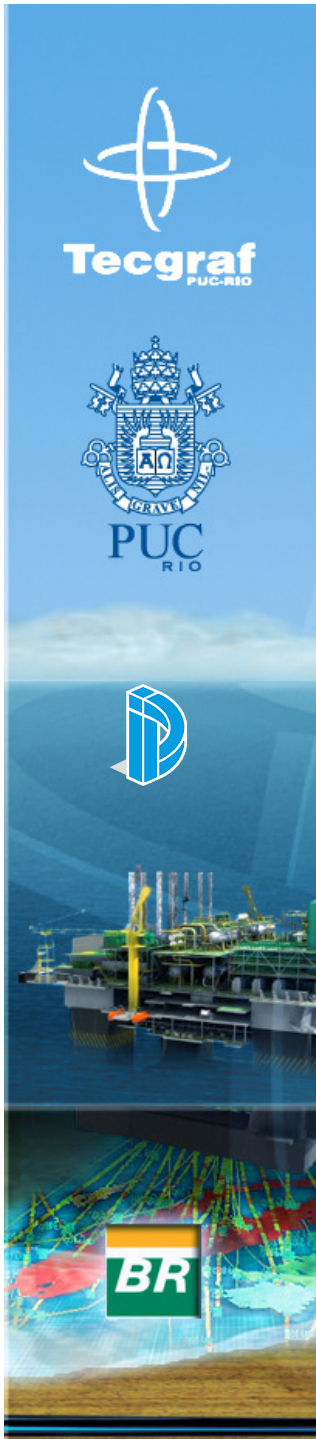
Gargalos no Desempenho

- Converter muita quantidade de dados 3D em pixels para determinar o que está visível ou não
 - Visibility Culling
- Reduzir a complexidade dos dados a serem processados (frame a frame)
 - LOD / Adaptive mesh simplification
 - Representações alternativas



Visibility Culling

- Ideia básica: rejeitar grandes partes da cena que não estejam visíveis, para reduzir a complexidade da renderização ao conjunto de primitivas (potencialmente) visíveis
- O processo de criação do subconjunto (potencialmente) visível da cena é chamado Visibility Culling
- Geralmente modelos massivos pedem algum tipo de pré-processamento para poderem executar o Visibility Culling
 - Pré-computação do próprio PVS (caso de portal culling, por ex.)
 - Organização espacial das primitivas geométricas em estruturas que facilitem os testes de visibilidade
- Modelos massivos também requerem técnicas especiais para occlusion culling



Simplificação e LOD (1)

- Simplificação geométrica: processo de tentar reduzir os polígonos de um modelo detalhado, mantendo sua aparência

- Modelos massivos precisam de técnicas especiais de simplificação, visto que suas malhas geralmente não cabem em memória



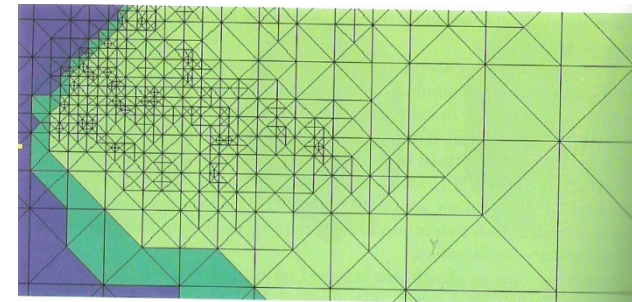
(a) Original; 108,588 faces

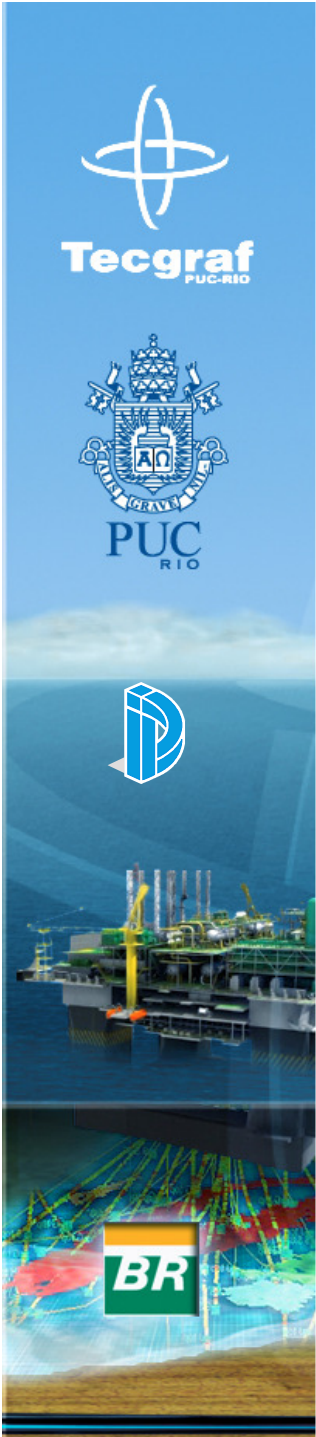


(b) 25,000 face mesh

- O que se busca em LOD

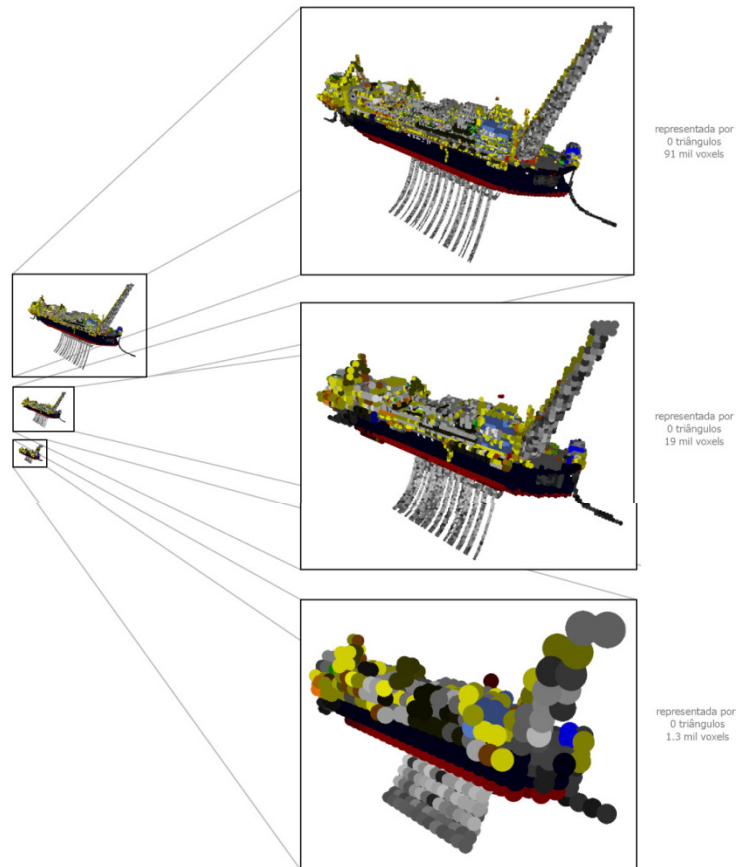
- Real time adaptive view-dependent LOD

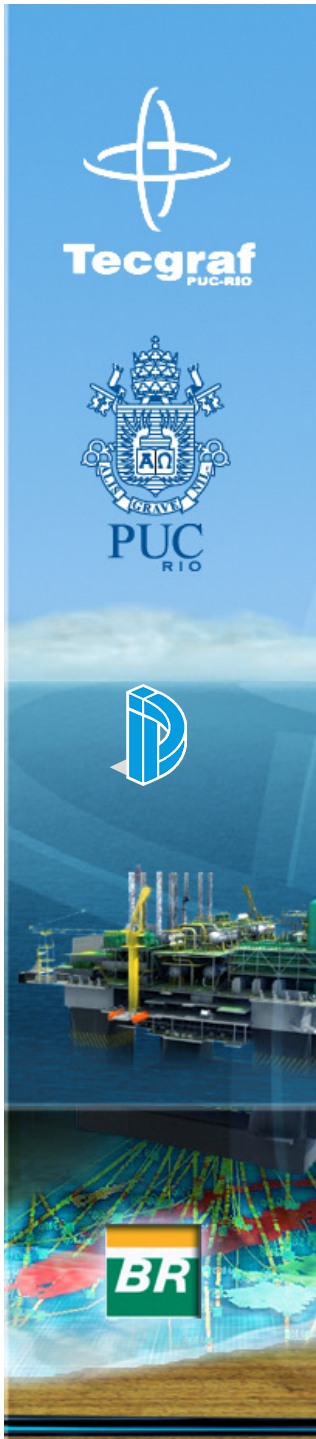




Simplificação e LOD (2)

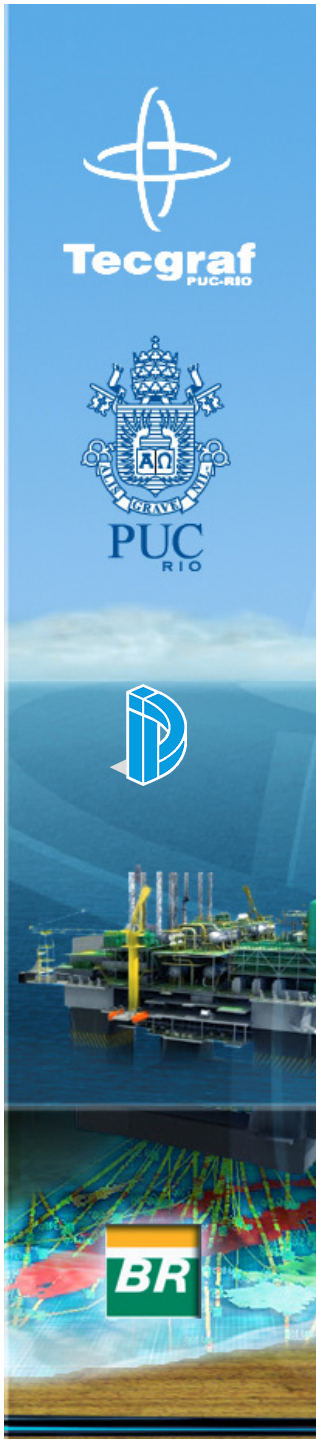
- Representações alternativas (ex., Far Voxels)





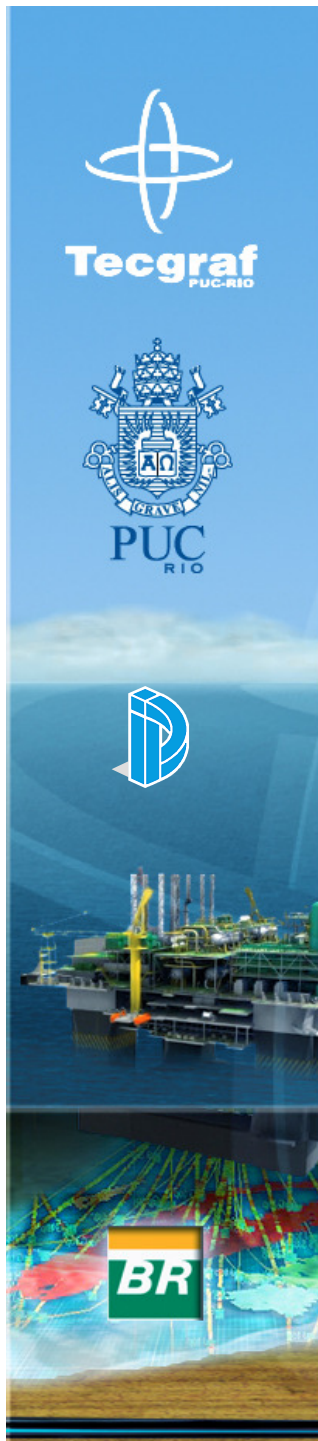
Visibility Culling vs Simplificação / LOD

- Ambas são essenciais para a renderização de modelos massivos
- Dependendo do tipo de modelo, uma ou outra técnica pode ser mais importante
 - Modelos escaneados e terrenos, são geralmente de low depth complexity
 - Favorecem o LOD puro
 - Modelos CAD geralmente tem large depth complexity
 - Exigem técnicas de visibilidade antes de LOD



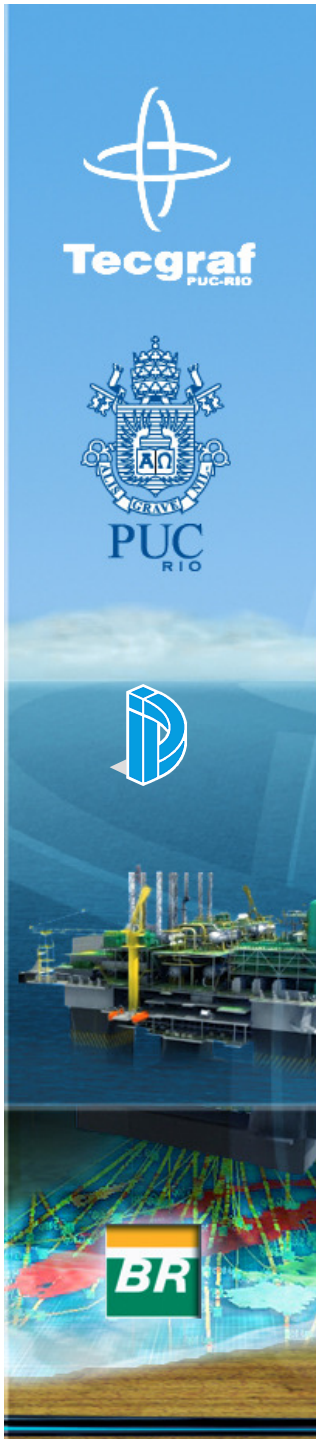
Outros Desafios

- Criar métodos eficientes de storage-to-memory
- 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

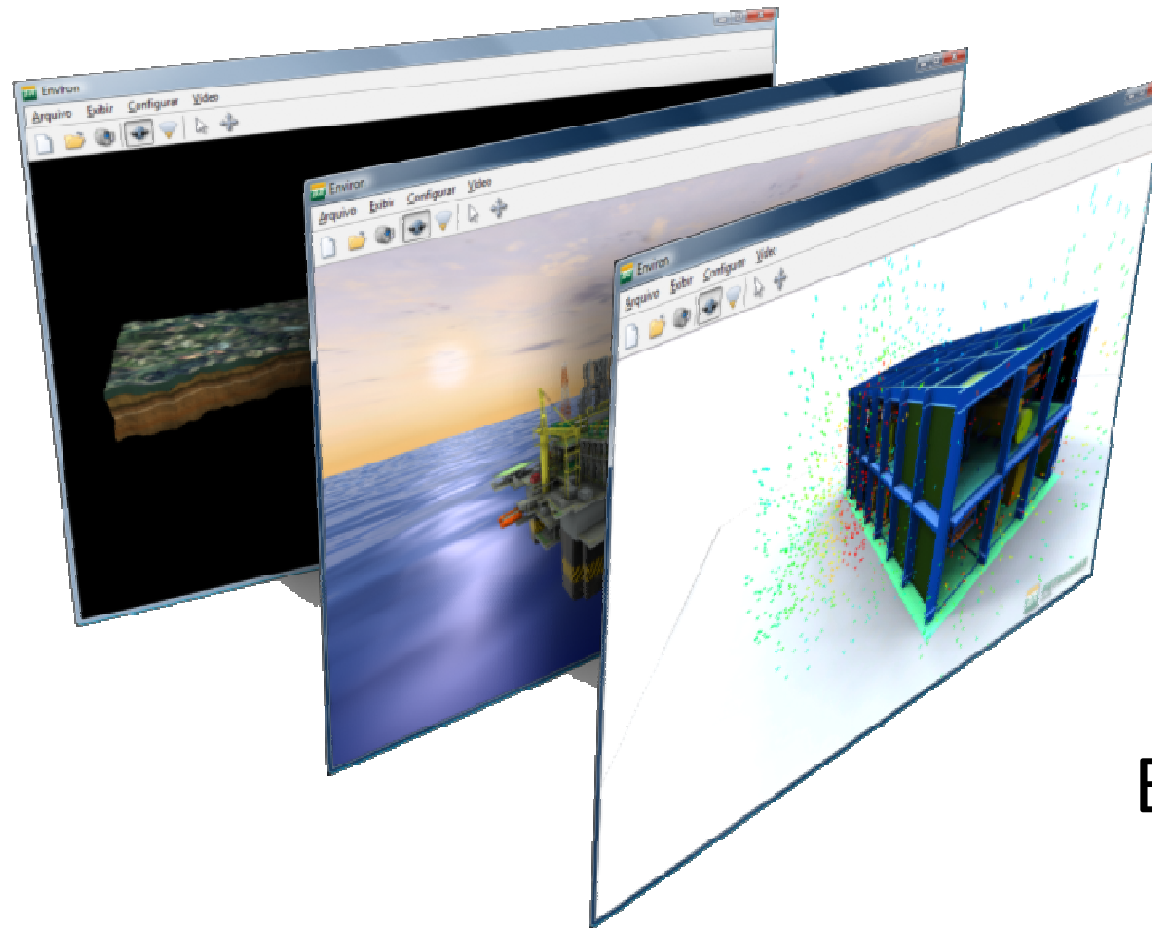


Solução é Sistêmica

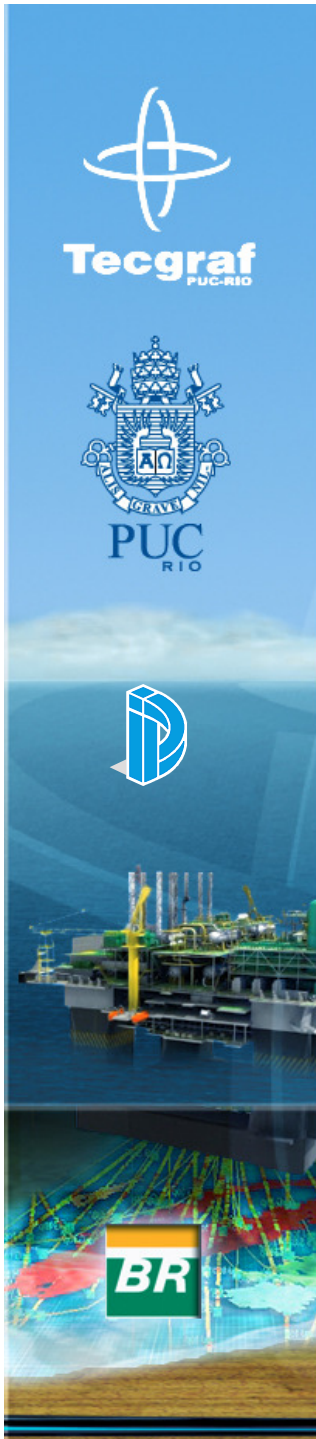
- Renderização interativa de modelos massivos que atinja nível de desempenho consistente e sustentável requer solução em nível de sistema
 - Atacar apenas um ou dois aspectos do sistema pode fazer com que os outros falhem
- Soluções para uma classe de modelos massivos podem não funcionar bem para outras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



Nossa Experiência com Modelos Massivos

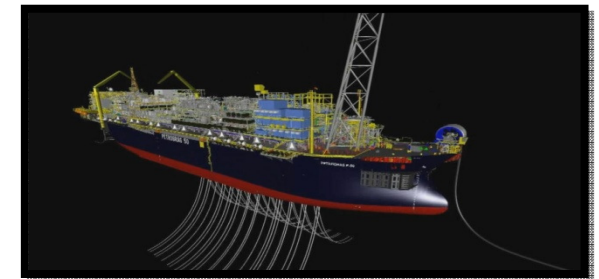


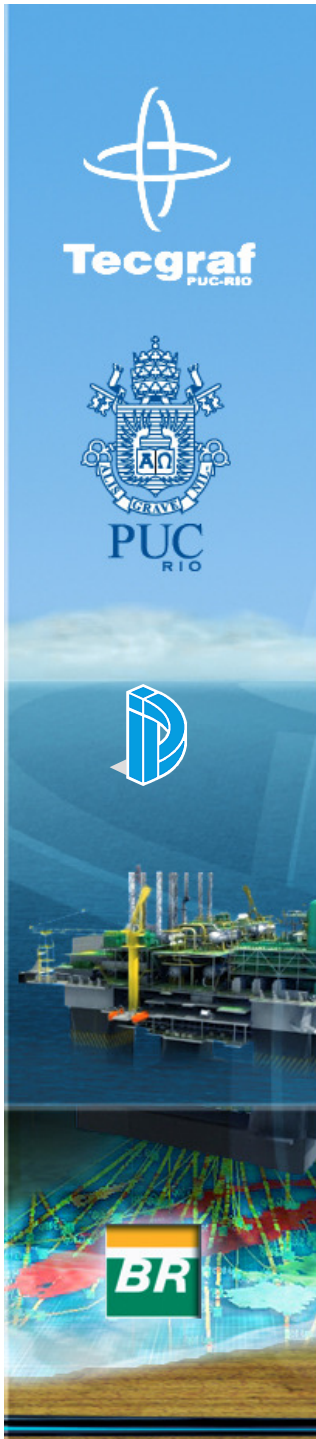
Environ



Environ: Principais Funcionalidades (1)

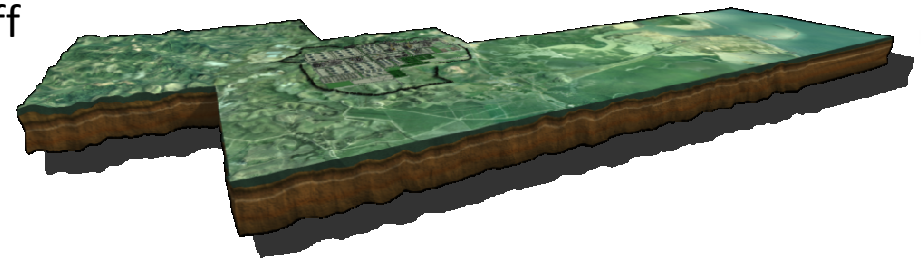
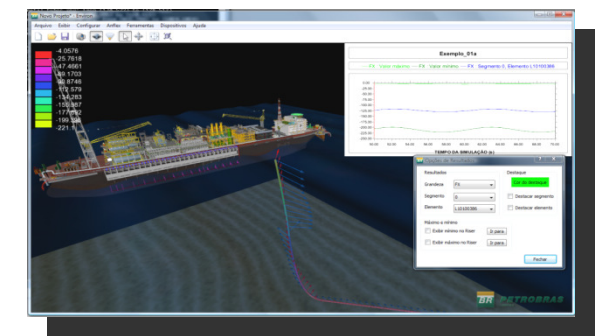
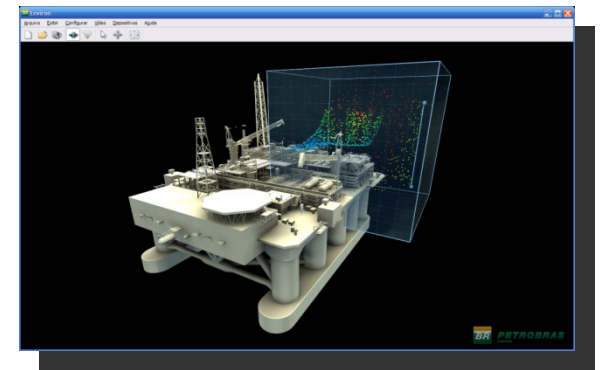
- Visualização em Tempo Real de Modelos Massivos de Engenharia
 - Otimizações
 - Capacidade de visualização de modelos grandes
 - Interoperabilidade
 - Facilidade para conversão CAD ↔ RV
 - CAD/CAE (PDS/Microstation e PDMS/Aveva)
 - Visualização de informações eng. do PDS e do PDMS
- Foto-realismo
 - 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

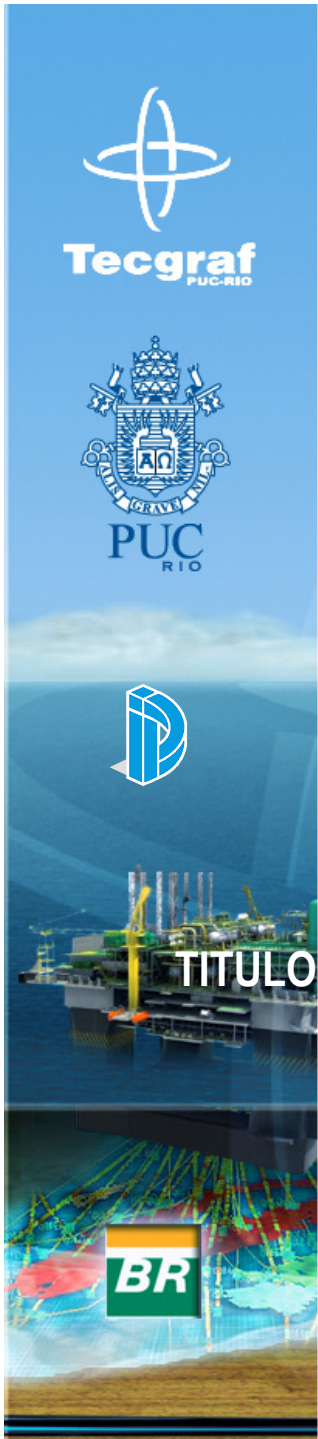




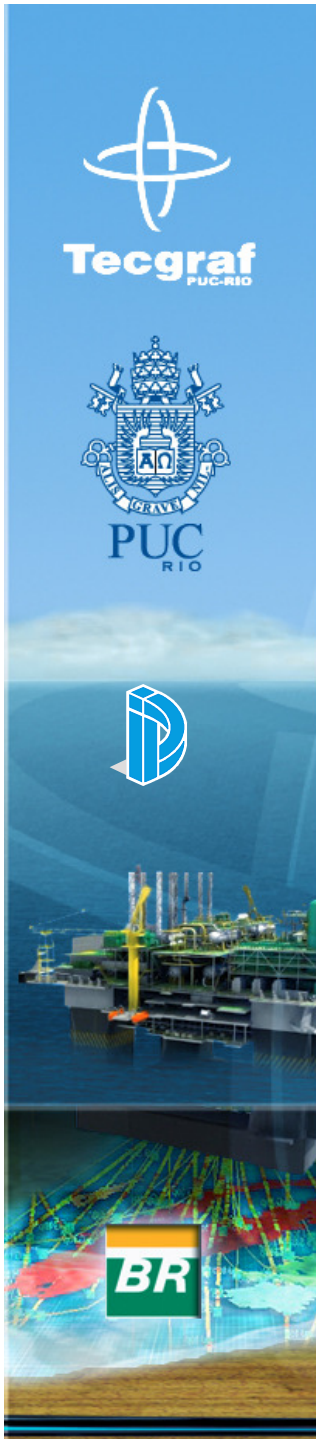
Environ: Principais Funcionalidades (2)

- 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



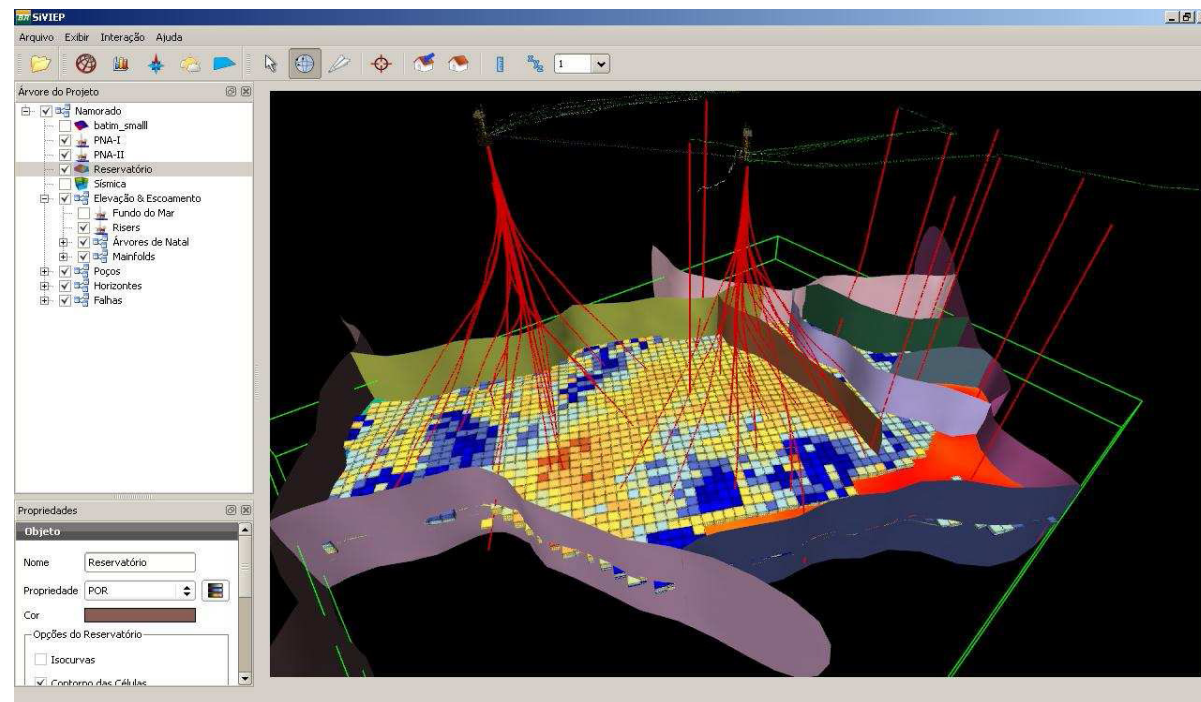


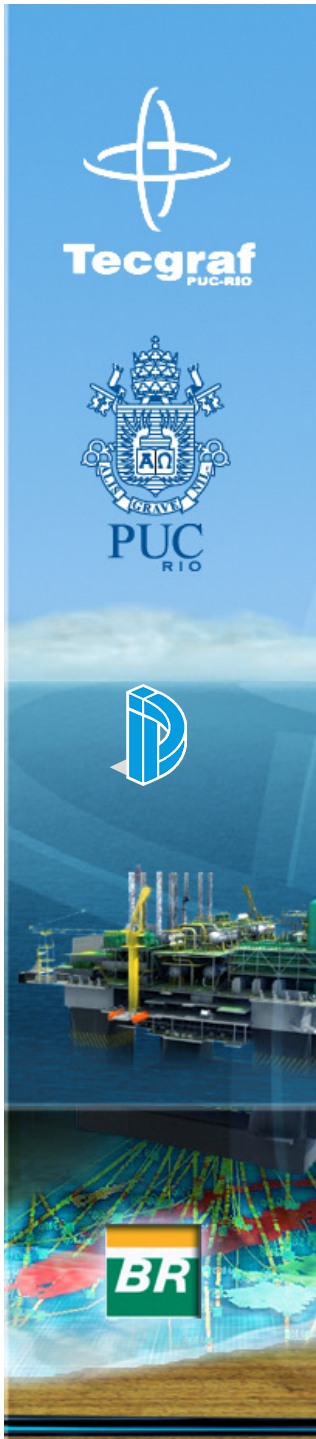
Modelos Multi-Escala



SiVIEP: Sist. Integrado de Visualização de Exploração e Produção

- Permite visualizar de forma integrada modelos de E&P (poços, plataformas, reservatórios,...).
- Ambientes com característica multi-escala.
- Navegação feita basicamente através de duas ferramentas: Examinar e Voar.
- Dificuldade em controlar a velocidade de navegação ao usar a ferramenta Voar.





Siviep

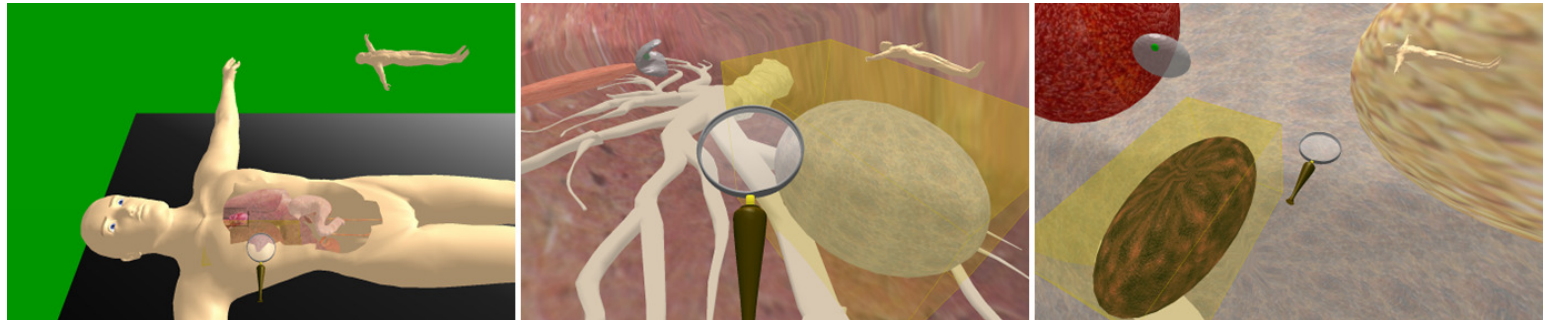
- * Reservatório
- * Falhas e Horizontes
- * Sísmica
- * Poços
- * Plataformas
- * Risers
- * Completação submarina
- * Ancoragem

Siviep_2min30(ago09)

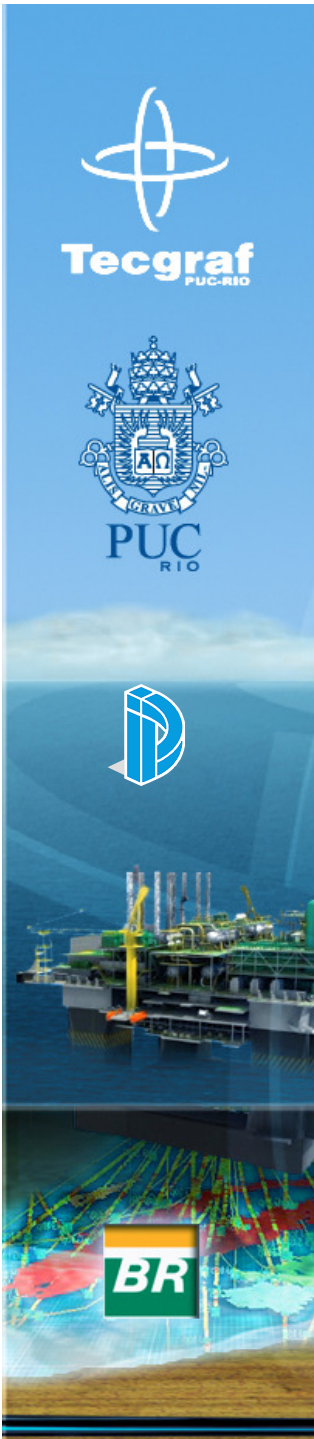


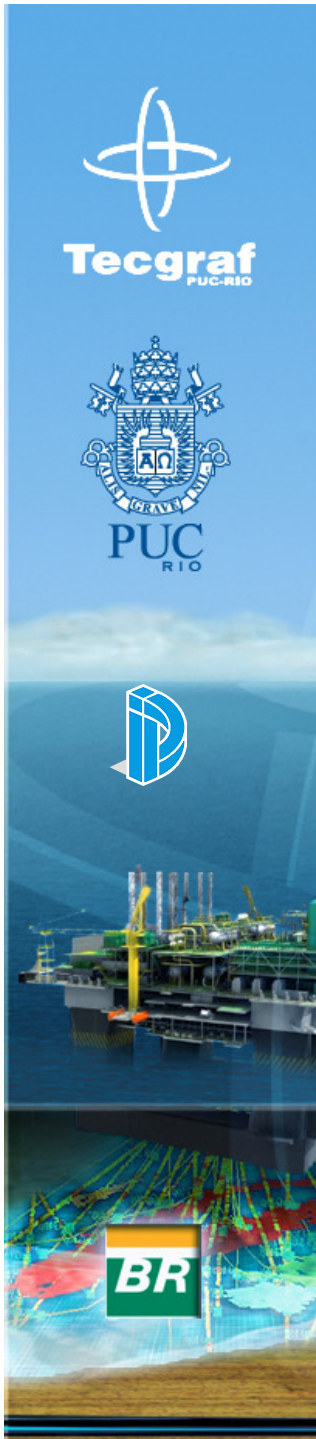
Problema da Navegação em Ambientes Multi-Escala

Soluções normalmente são discretas, para navegar nas várias escalas, i.e., o usuário escolhe manualmente a escala da navegação



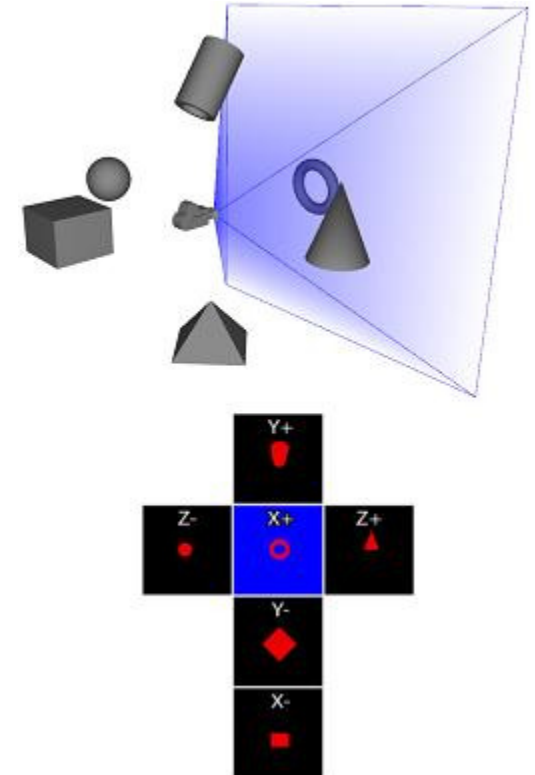
KOPPER, Regis; NI, Tao; BOWMAN, Doug; PINHO, Marcio Serolli. Design and Evaluation of Navigation Techniques for Multiscale Virtual Environments. In: IEEE VIRTUAL REALITY 2006, 2006, Alexandria.
IEEE Virtual Reality. IEEE Computer Society, 2006. p. 24-31

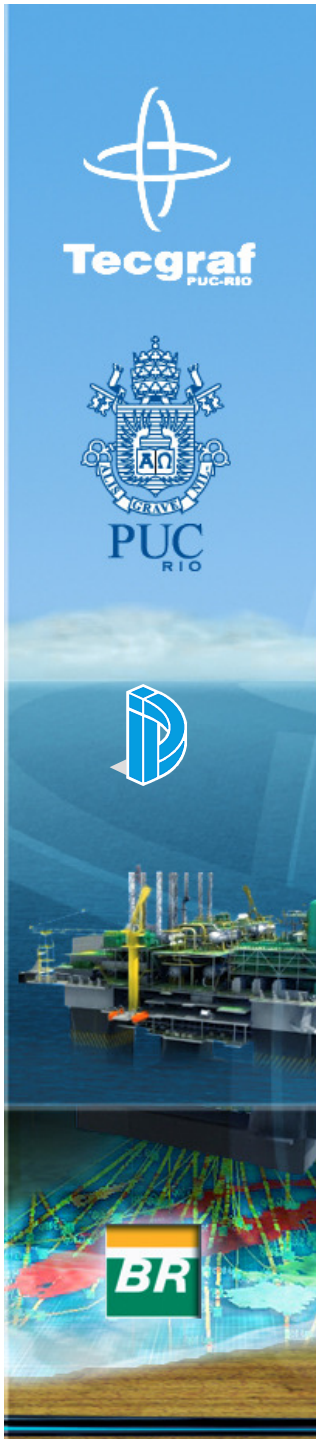




Solução SiVIEP: Ajuste Automático de Velocidade (1)

- A técnica se baseia na construção de uma estrutura chamada de *cubo de distâncias*.
 - Armazena uma amostragem das distâncias da câmera até os objetos.
 - 6 imagens representando todo o espaço, na forma de um cubo.
 - O cubo é orientado em relação à câmera.
 - Canais RGB armazenam um vetor normalizado que aponta do ponto correspondente ao fragmento até a câmera.
 - Canal A armazena a distância da câmera até o ponto.
 - Calculado na placa gráfica.
 - Construído em 6 passadas de renderização.





Solução SiVIEP: Ajuste Automático de Velocidade (2)

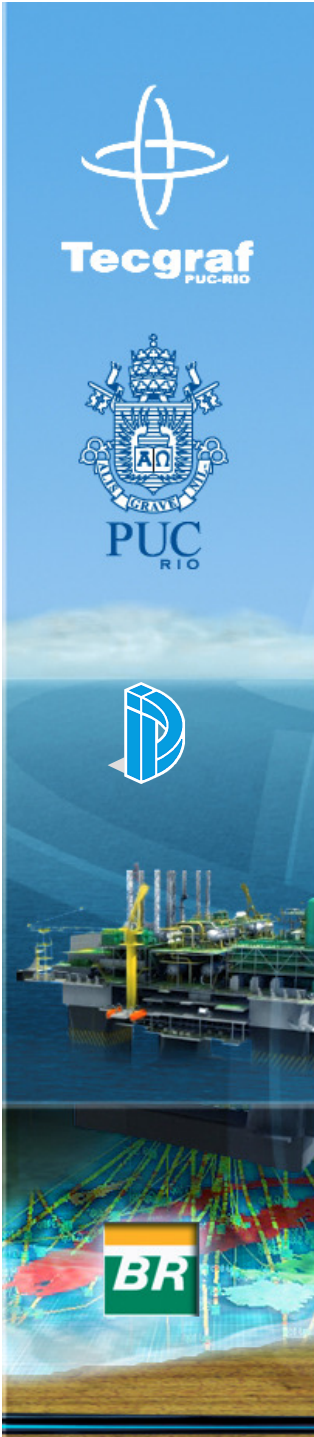
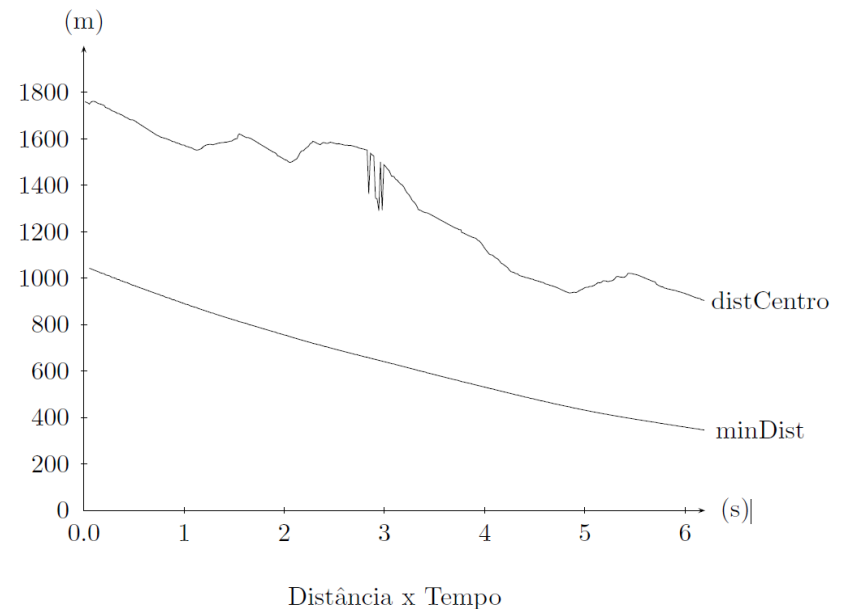
- Velocidade proporcional a menor distância armazenada no cubo de distâncias:

$$V = k \frac{\text{minDist}}{dt}$$

- Problemas:
 - Velocidade muito lenta ao navegar em paralelo a planos ou muito perto de objetos (minDist não reflete o real desejo do usuário em alguns casos).
 - Se k muito alto, V irá variar com mais intensidade (causa desconforto ao usuário).

Solução SiVIEP: Ajuste Automático de Velocidade (3)

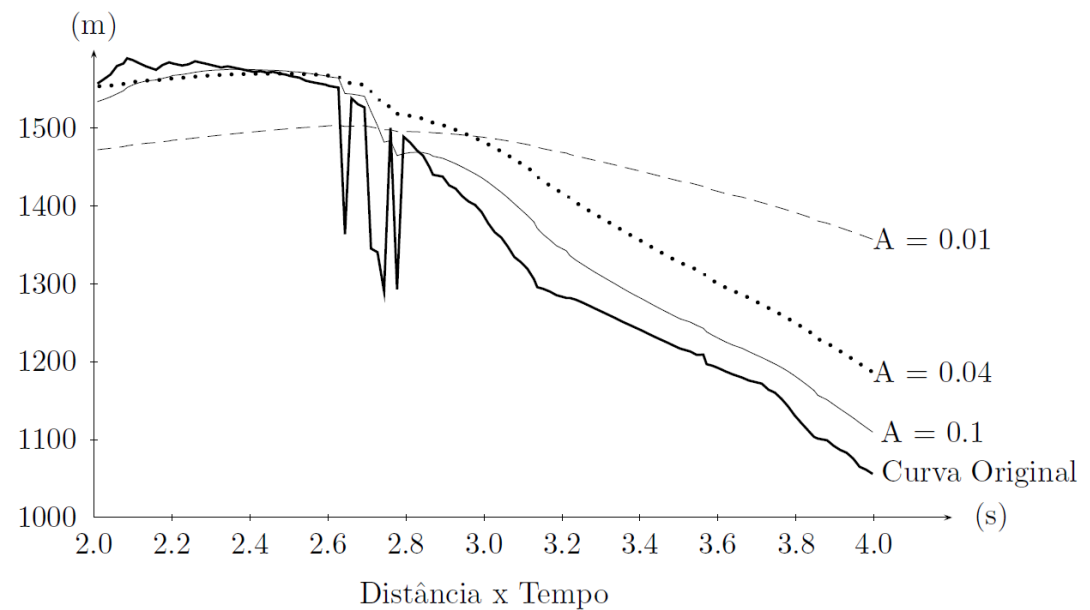
- K ajustável manualmente pelo usuário: ainda houve dificuldades por parte desses.
- Usar a distância da câmera ao centro da tela (*centroDist*) ao invés de *minDist*:
- Problema: *centroDist* não forma uma curva contínua e pode provocar movimentos bruscos na câmera.

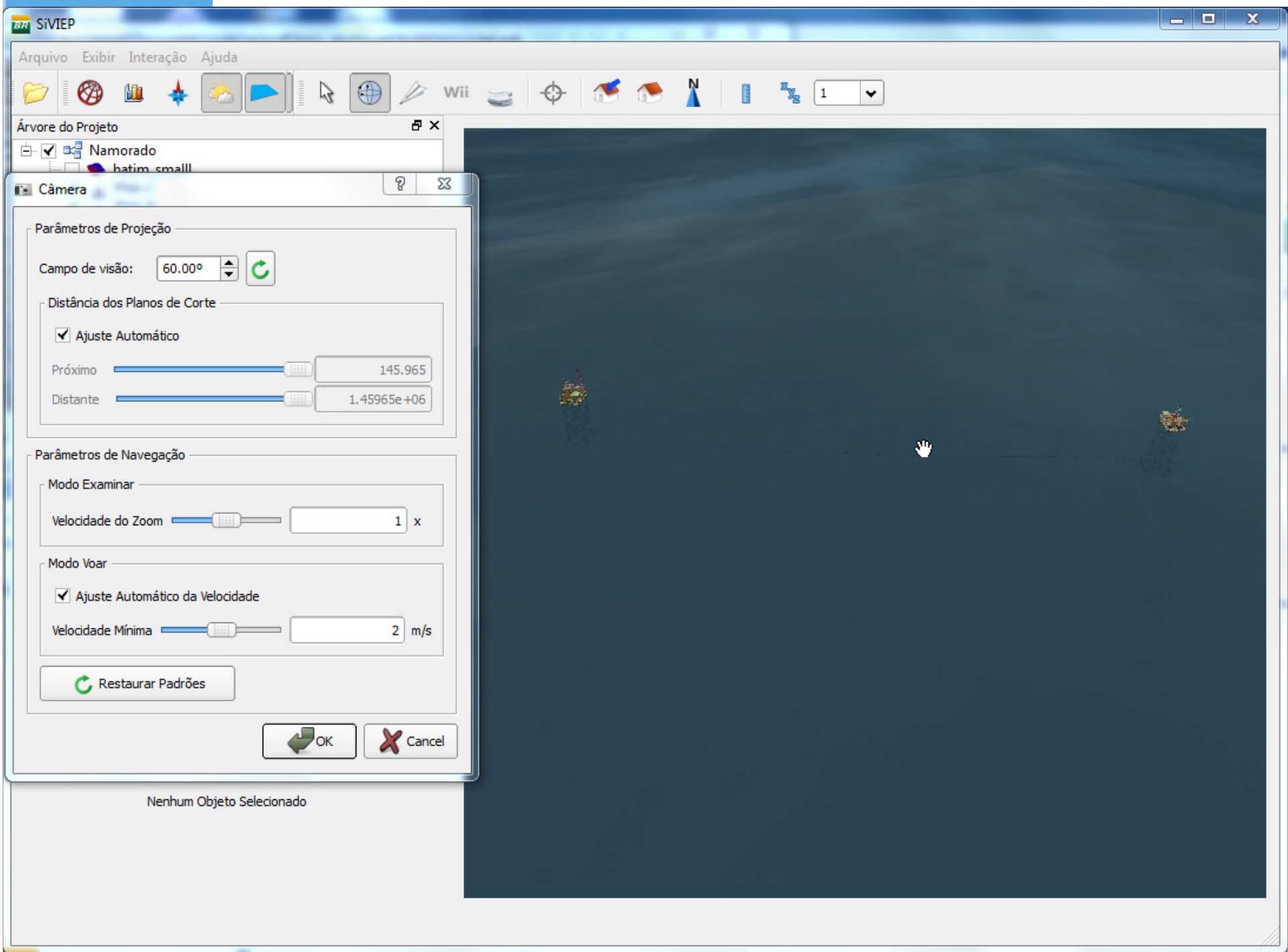




Solução SiVIEP: Ajuste Automático de Velocidade (3)

- Solução: suavizar a curva formada por *centroDist*
- **Média Exponencial Móvel:**
 - $MEM_i = MEM_{i-1} + A * (centroDist_i - MEM_{i-1})$
- *minDist* é usado como limitador de *centroDist*:
 - Se $centroDist > n * minDist$, use $n * minDist$.





Modelos Massivos e Multi-Escala em Realidade Virtual



Alberto Raposo

Tecgraf – Grupo de Tecnologia em Computação
Gráfica

Departamento de Informática,
Pontifícia Universidade Católica do Rio de Janeiro

<http://www.inf.puc-rio.br/~abraposo>

Obrigado!

Perguntas?

