

# Realidade Virtual e Interfaces Modernas

Prof. Carlos Henrique Q. Forster

IEC-ITA

Julho/2005

# Introdução ao Java3D

# Tópicos

- Java 3D
- Mundo virtual básico
- Nós do Java 3D

# Java 3D

- <http://java.sun.com/products/java-media/3D>
- É uma hierarquia de classes Java
- É parte da API Java Media
- Centrado em rede e grafo de cenas
- Permite criar aplicativos e applets interativos com objetos tridimensionais
- Existe desde Dezembro 1998

# Java 3D

- Gráficos em baixo-nível (modo imediato) juntando as melhores idéias do Direct3D, OpenGL, QuickDraw3D e XGL
- Gráficos em alto-nível (modo retido) juntando idéias do VRML, Direct3D, WTK etc
- Multiplataforma (Java 2)

# Java 3D

- Objetivos
  - Rico em elementos para mundos 3D
  - Orientado a objetos
  - Suporta carregadores de diversos formatos. Wavefront (OBJ), VRML (WRL), Lightwave (LWS, LWO)
  - Portável
  - API simples para programadores Java

# Java 3D

- Compressão de geometria 10:1
- Modelo de ponto de vista próprio
- Independência de display e suporte a rastreo
- Execution culling
- Audio espacial
- Rendering automático
- Escrito em Java
- Modos imediato e retido
- Níveis de detalhes

# Pacotes para usar Java 3D

javax.media.j3d	Docs do j3d
com.sun.j3d.utils	Docs do j3d utils
javax.vecmath	Docs do j3d
java.awt	Docs do Java
java.applet	Docs do Java



# Grafos de cena do Java 3D

- Cada caminho até uma folha descreve completamente o estado do objeto representado pela folha
- Nenhum objeto do Java 3D pode ter 2 ou mais pais
- Objetos compartilhados: subgrafo separado (SharedGroup) e objeto folha (Link)
- Exceção de múltiplos parentes

# Grafos de cenas do Java 3D

- Cada grafo de cenas tem um único VirtualUniverse
- Objetos de um VirtualUniverse não podem ser utilizados simultaneamente por outro universo
- É comum ter apenas um universo por aplicação
- Cada VirtualUniverse tem uma lista de objetos Locale que representam um ponto de referência no universo (exemplo: planeta)
- É comum haver um único Locale por aplicação

# Grafos de cenas do Java 3D

- Locales servem de raízes para múltiplos subgrafos de cena
- Apenas objetos ligados direta ou indiretamente a locales são desenhados
- BranchGroup é uma raiz para um subgrafo de cena (branch graph)

## Nodes and NodeComponents (objects)



VirtualUniverse



Locale



Group



Leaf



NodeComponent



other objects

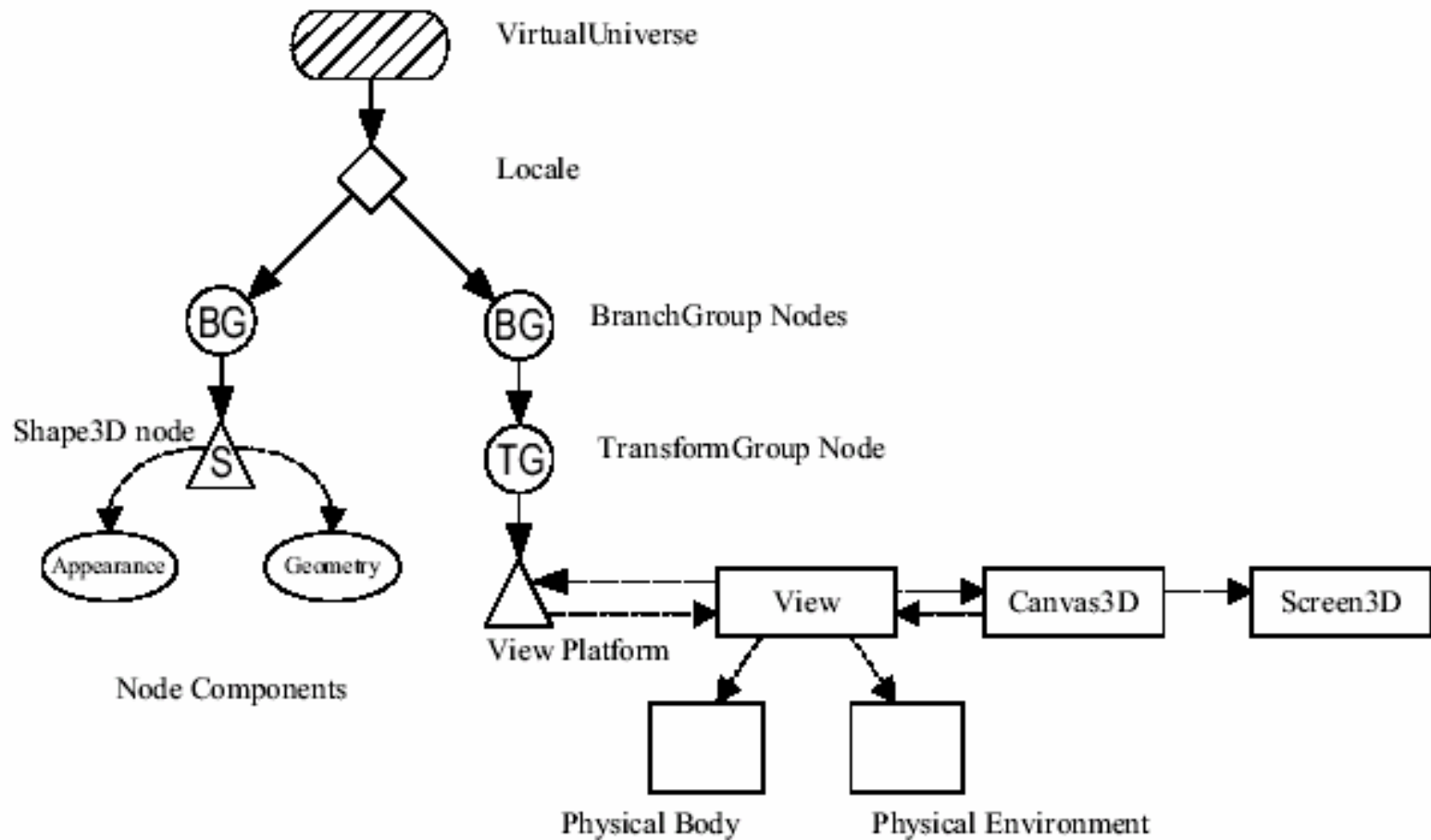
## Arcs (object relationships)



parent-child link



reference



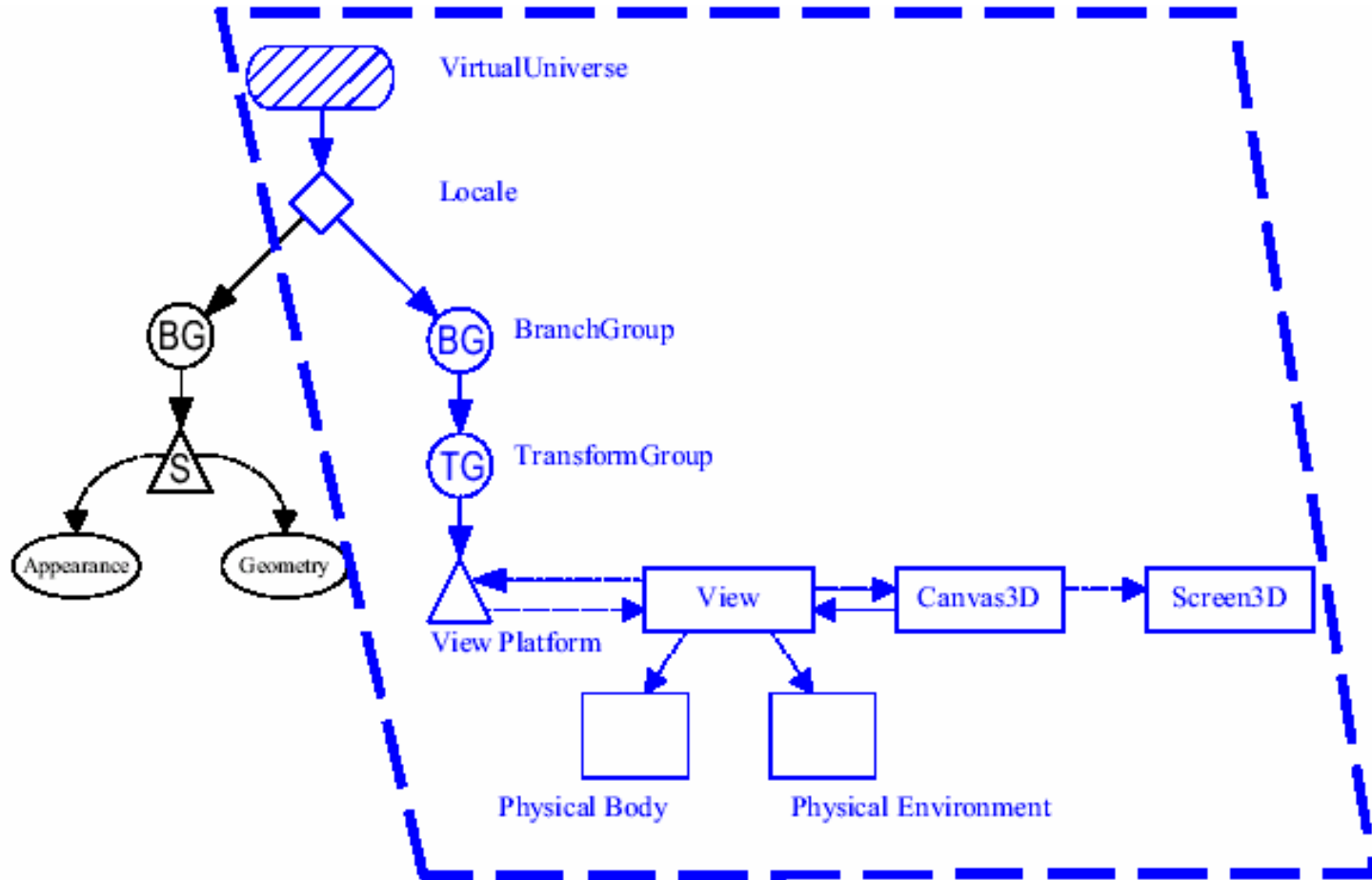
# Mundo Virtual básico do Java 3D

1. Criar objeto Canvas3D
2. Criar VirtualUniverse
3. Criar Locale e conectar a universo
4. Criar os objetos das classes
  1. View
  2. ViewPlatform
  3. PhysicalBody
  4. PhysicalEnvironment

Conectar ViewPlatform, PhysicalBody, PhysicalEnvironment, Canvas3D a View e ViewPlatform a um BranchGroup
5. Construir e compilar branch graphs
6. Conectar branch graphs aos locales

# Receita simplificada

- Criar Canvas3D
- Criar SimpleUniverse (do j3dutils)
- Construir e compilar branch graphs
- Conectar subgrafos ao Locale do SimpleUniverse





# Nós do Java 3D

- Nós (Node)
  - Nós de Grupo (Group)
    - Transformações (TransformGroup)
    - Comutadores (SwitchGroup)
  - Nós Folha (Leaf)
    - Forma (Shape3D)
    - Fonte de Luz (Light)
    - Comportamento (Behavior)
- Componentes de nós
  - Geometria (Geometry)
  - Aparência (Appearance)

# BranchGroup

- Quando se insere um BranchGroup num Locale, seus objetos se tornam “vivos”, isto é
  - São sujeitos a serem desenhados
  - Seus parâmetros não podem mais ser modificados a não ser se especificados como capabilities
- Quando se compila um BranchGroup, se o converte numa forma mais eficiente para o sintetizador de imagens.
- Recomenda-se fazê-lo imediatamente antes de ligar ao locale

# Objetos de ponto de vista e ambiente

- Não segue a metáfora de câmera.
- View – não pertence ao grafo de cenas, múltiplos Canvas3D. Diversos parâmetros.
- ViewPlatform – folha no grafo de cenas. Corresponde ao ponto de vista.
- PhysicalBody – classe que modela a cabeça do usuário, posição dos olhos e ouvidos
- PhysicalEnvironment – classe que controla os dispositivos de entrada

# Shape3D

- Folha
- Composto de Geometria e Aparência
- Uma Aparência e várias Geometrias

# Geometry

- Pode ser criada a partir de um objeto das classes util
- Pode ser criada através da especificação de coordenadas
- Pode ser criada através de carregadores de objetos a partir de arquivos
- Text3D

# Objetos utilitários

- ColorCube
- Box
- Cone
- Cylinder
- Sphere

# Appearance

- Cor
- Material
- Textura
- Transparência

# Behavior

- Classe que provê suporte a
  - Animação e movimento
  - Colisão
  - Interpolação
  - Morphing
  - Regiões de alcance (execution culling)



# Applets e Aplicativos

- Definir sua classe principal como Applet permite
  - Execução dentro de um Browser
  - Execução a partir da função main como aplicativo
- Objeto MainFrame de j3dutils
- Objeto Frame de awt

# Capabilities

- Permitem modificar parâmetros de subgrafos compilados
  - TransformGroup
    - ALLOW\_TRANSFORM\_READ
    - ALLOW\_TRANSFORM\_WRITE
  - Group
    - ALLOW\_CHILDREN\_EXTEND
    - ALLOW\_CHILDREN\_READ
    - ALLOW\_CHILDREN\_WRITE

# Modos retido e imediato

- Modo retido é padrão
  - Não há necessidade de invocar o sintetizador gráfico. Isto é feito sempre que um branch graph que contém um ponto de vista é ligado a um Locale em um universo virtual
  - Loop infinito
- Modo imediato
  - Ver documentação da classe Canvas3D para o modo imediato e off-screen

# TransformGroups

- Contém um subgrafo e uma matriz de transformação geométrica aplicada ao subgrafo todo
- Transform3D
  - Vários métodos para construir composições de rotações, translações e escalas
  - Rotações na forma de matrizes 3x3, ângulos de Euler e quaternions