



Universidade do Minho
Escola de Engenharia

Computação Gráfica

Fase 2 - Transformações Geométricas

Relatório Trabalho Prático

Grupo 40

LEI - 3^o Ano - 2^o Semestre

A90969	Gonçalo Gonçalves
A98695	Lucas Oliveira
A89292	Mike Pinto
A96208	Rafael Gomes

Braga,
19 de fevereiro de 2025

Conteúdo

1	Introdução	3
2	Engine	4
2.1	Estruturas de Dados	4
2.1.1	Classe <i>Group</i>	4
2.1.2	Classe <i>Transformation</i>	5
2.1.3	Classe <i>Model</i>	5
2.2	Parsing dos ficheiros <i>XML</i>	5
2.3	Desenho	6
3	Sistema Solar: Demo Scene	7
3.1	Anel de Torus	8
3.2	Primitivas Gráficas utilizadas	9
3.3	Hierarquia utilizada	9
3.4	Transformações utilizadas	9
3.4.1	Sol	9
3.4.2	Planetas	10
3.4.3	Luas	10
3.4.4	Anel de Saturno	11
3.5	Resultados obtidos	11
4	Conclusão	15
A	Anexos	16
A.1	Ficheiro <i>XML</i> do sistema solar	16

Lista de Figuras

3.1	Parâmetros do Torus	8
3.2	Sistema Solar Disperso	12
3.3	Sistema Solar Disperso com Cores <i>Filled</i>	13
3.4	Sistema Solar Alinhado	13
3.5	Sistema Solar Alinhado com Cores <i>Filled</i>	14

Capítulo 1

Introdução

Este relatório é elaborado no âmbito da segunda fase do trabalho prático da Unidade Curricular de Computação Gráfica do 2º semestre do 3º ano do curso de Engenharia Informática da Universidade do Minho.

Nesta fase, foi proposto a alteração da aplicação **Engine** para poder aplicar transformações geométricas a modelos de forma hierárquica. Para tal foi necessário a introdução de novas *tags* a utilizar nos ficheiros do tipo *XML*. Como requisito para esta fase, foi necessário criar uma *demo scene* estática do sistema solar, incluindo o sol, planetas e luas, definidos numa hierarquia.

Ao longo deste documento, serão explorados os métodos utilizados, as funcionalidades implementadas e os resultados obtidos, proporcionando uma visão abrangente do trabalho realizado nesta fase do projeto.

Capítulo 2

Engine

2.1 Estruturas de Dados

As alterações propostas na aplicação Engine passam apenas por definir uma cena (*scene*) como uma “árvore”, onde cada nó contém um conjunto de transformações geométricas, translação, rotação e escala. Opcionalmente poderá conter um conjunto de modelos (*models*) e “subgrupos”.

Foram então definidas três novas classes na aplicação **Engine**, **Group**, **Transformation** e **Model** que irão representar os grupos, transformações de um grupo e os respectivos modelos.

2.1.1 Classe *Group*

Como tal, um objeto da classe **Group** irá conter:

- Um vetor de objetos da classe **Transformation** que irá conter a informação das transformações a aplicar no grupo.
- Um vetor de objetos da classe **Model** que irá conter a informação dos modelos a desenhar.
- Um vetor de objetos da classe **Group** que irá conter os subgrupos do grupo.

Além disso esta classe possui os métodos principais para o *parsing* dos grupos, como:

- **parseTransformation** - Responsável por ler e instanciar um objeto da classe **Transformation** que irá conter a informação da transformação. Além disso, também verifica se a restrição de apenas um tipo de transformação por grupo é respeitada.
- **parseModels** - Responsável por ler os pontos contidos num ficheiro .3d e instanciar um objeto da classe **Model** por cada modelo presente no grupo.
- **draw** - Responsável por desenhar a cena representada em um grupo, respeitando a ordem de desenho e aplicação das transformações.

- Construtor da classe - Esta função recebe como argumento um apontador para um elemento do tipo `XMLElement` da biblioteca *tinyxml*. Esta função é responsável por criar um objeto da classe `Group` com toda a sua informação, utilizando as funções apresentadas acima.

2.1.2 Classe *Transformation*

Esta classe representa uma transformação a aplicar em um grupo. Como variáveis de instância, esta classe possui:

- `String type` - Representa o tipo da transformação, e.g. *scale*, *translation* e *rotate*.
- `float x, y, z` - Representa os valores cartesianos a utilizar nas transformações.
- `float angle` - Ângulo de transformação caso a transformação geométrica seja uma rotação.

Como métodos, esta classe possui, para além de um construtor, um método `Transformation` que liberta a memória utilizada por um objeto desta classe e um método `transform` que aplica a transformação (utilizada durante o desenho de um grupo).

2.1.3 Classe *Model*

Esta classe é responsável por possuir e desenhar os vértices necessários para o desenho de um modelo. Como tal como variável de instância, possui um vetor de objetos do tipo `Point` que representa as coordenadas cartesianas de um ponto. Como métodos desta classe, apenas possui um método `getPoints` que devolve um vetor com os pontos desse modelo, e um método para desenhar os vértices desse modelo.

2.2 Parsing dos ficheiros *XML*

Para interpretar os novos campos/*tags* adicionados aos ficheiros *XML* relativamente à primeira fase, foi necessário alterar a interpretação, pela **Engine**, da *tag group*.

Inicialmente é inicializado um objeto `Group` que será imediatamente responsável pelo *parsing* do XML ao receber como argumento um apontador para um objeto `XMLElement` que constitui uma hierarquia, percorrendo os elementos que o constituem um por um.

No caso do elemento ser do tipo `transform`, que representa transformações geométricas a aplicar no grupo, é criado um vetor de objetos da classe `Transformation`. Por cada elemento filho é instanciado um objeto da classe `Transformation` que irá acolher o tipo de transformação e os valores a utilizar sendo inserido este objeto no vetor de transformações. Desta forma é garantido que as transformações são guardadas pela ordem correta e se cada grupo possui no

máximo uma de transformações por cada tipo.

No caso do elemento ser do tipo `models` o processo será análogo ao das transformações explicitado anteriormente, onde é criado um vetor de objetos da classe `Model`, e por cada modelo encontrado nos elementos filho, são lidos os pontos que constituem o ficheiro `.3d` explicitado como o atributo `file` de uma `tag model`.

Caso o elemento seja um `group` é criado um objeto da classe `Group` fornecendo o elemento `XML` atual, e repetido o processo explicitado acima, funcionando, portanto, recursivamente. Uma vez terminada a construção desse objeto `Group` este será guardado no `Group` "pai", funcionando de forma análoga a uma árvore.

2.3 Desenho

Após a fase de *parsing* do ficheiro `XML`, é utilizado o método `draw` no objeto `Group`.

Esta irá armazenar a matriz de desenho através da função `glPushMatrix`, uma vez que será necessário guardar o estado da matriz para que possa ser recuperado.

Assim, por cada `Group` será feito um `glPushMatrix`, sendo numa fase inicial executadas todas as transformações armazenadas, recorrendo às funções `glTranslatef`, `glScalef` e `glRotatef`, consoante o seu `type` da classe `Transformation`. Seguidamente, são desenhadas as figuras guardadas no vetor de objetos da classe `Model`, recorrendo à função `draw` da classe `Model` que utiliza o código que utilizado na primeira fase para desenhar os ficheiros 3D. Por último, aplica-se recursivamente o método `draw` da classe `Group` ao `subGroup` correspondente. Uma vez terminada essa aplicação recursiva será reposto o estado prévio da matriz através do método `glPopMatrix`.

Capítulo 3

Sistema Solar: Demo Scene

Para a segunda fase deste trabalho prático, foi proposta a elaboração de uma *Demo Scene* de um modelo estático do Sistema Solar, contendo o Sol, os planetas e as suas respetivas luas.

Esta *Demo Scene* foi definida num ficheiro *XML* compatível com a aplicação **Engine** utilizando os novos conceitos de hierarquia e transformações adicionadas.

O ficheiro contempla o Sol, os principais planetas, (Mercúrio, Venus, Terra, Marte, Júpiter, Saturno, Urano e Neptuno), assim como os satélites naturais do planeta Terra, de Marte e os anéis de Saturno.

3.1 Anel de Torus

Com o intuito de representar os anéis de Saturno, foi adicionado à aplicação **Generator** a primitiva gráfica de *Torus*.

Para gerar os vértices desta nova primitiva gráfica é necessário fornecer à aplicação **Generator** os seguintes argumentos:

- R_1, R_2 que representam o raio das *stacks* e *slices* respetivamente.
- O número de *sides*
- O número de *rings*

Cada *side* está dependentes de um ângulo u e cada *ring* de um ângulo v como podemos verificar pela figura 3.1.

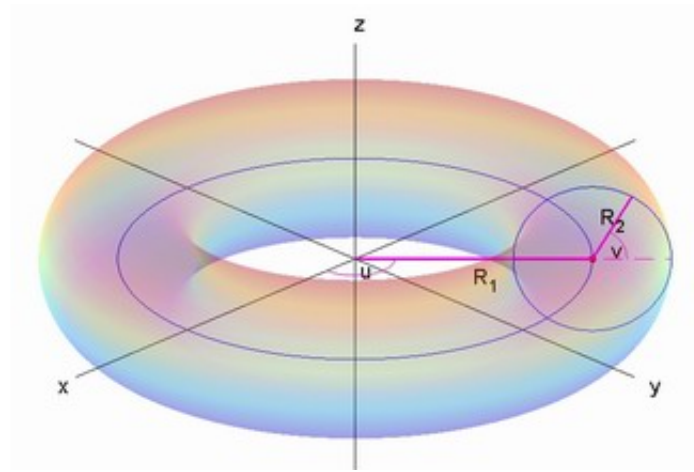


Figura 3.1: Parâmetros do Torus

Para o desenho desta primitiva procedeu-se em desenhar progressivamente em *slices* verticais, onde para cada vértice foi utilizado as seguintes equações para o cálculo das suas coordenadas:

$$x = (R_1 + R_2 \times \cos(v)) \times \cos(u) \quad (3.1)$$

$$y = (R_1 + R_2 \times \cos(v)) \times \sin(u) \quad (3.2)$$

$$z = R_2 \times \sin(v) \quad (3.3)$$

Para o desenho das fatias manipulou-se os valores dos ângulos u e v conforme o número de *sides* e *rings* fornecidos.

3.2 Primitivas Gráficas utilizadas

As primitivas gráficas utilizadas para a representação do Sistema Solar, foram gerados a partir da aplicação **Generator**.

Como tal, foi necessário apenas duas primitivas gráficas, uma esfera de raio = 1, 20 *stacks* e 20 *slices*, a ser utilizada na representação dos corpos celestes, e um anel de torus com $R_1 = 7$, $R_2 = 1$, 20 *sides* e 20 *rings* a utilizar na representação dos anéis de Saturno. Ambas as primitivas gráficas foram guardadas em ficheiros .3d designados de `sphere_demo.3d` e `torus_demo.3d`

3.3 Hierarquia utilizada

O Ficheiro *XML* desenvolvido consiste em vários grupos, com um grupo principal que irá conter diversos subgrupos. A divisão em grupos foi realizada da seguinte forma: cada planeta/estrela possui um grupo próprio inserido no grupo principal. No caso de terem algum satélite natural (luas) ou, no caso de Saturno, anéis, esses modelos estarão num grupo inserido no grupo do respetivo planeta.

Foi adotada esta hierarquia para representar os planetas relativamente ao Sol e os satélites naturais e anéis conforme o seu planeta correspondente.

3.4 Transformações utilizadas

Após definir a hierarquia utilizada, deu-se início ao cálculo dos valores das transformações a utilizar. Durante todo o desenvolvimento da *Demo Scene* tentou-se obter um resultado o mais próximo da realizada, objetivando a utilização de escalas adequadas e realistas ¹.

3.4.1 Sol

O Sol foi desenhado no centro do referencial, apenas sofrendo uma escala de 20 com o intuito de o tornar maior, como podemos verificar pelo seguinte excerto:

```
1  <group> <!-- Sol -->
2    <transform>
3      <scale x="20" y="20" z="20" />
4    </transform>
5    <models>
6      <model file="sphere_demo.3d" /> <!-- generator sphere 1 20 20
7        sphere_demo.3d -->
8    </models>
9  </group>
```

Excerto XML 3.1: Sol.

¹Informação dos corpos celestes do sistema solar e respetivos tamanhos e distancias retirados de: https://pt.wikipedia.org/wiki/Sistema_Solar

3.4.2 Planetas

Para os restantes planetas do sistema solar, foram aplicadas **translações** para representar a sua distância ao Sol, **escalas** para representar o seu tamanho e ainda foram aplicadas **rotações** para representar a inclinação axial do planeta.

```
1      <group> <!-- Mercurio -->
2          <transform>
3              <translate x="31" y="0" z="0" />
4              <scale x="0.2" y="0.2" z="0.2"/>
5              <rotate x="1" y="0" z="1" angle="0.01"/>
6          </transform>
7          <models>
8              <model file="sphere_demo.3d" />
9          </models>
10     </group>
11     <group> <!-- Venus -->
12         <transform>
13             <translate x="36.25" y="0" z="0" />
14             <scale x="0.70" y="0.70" z="0.70"/>
15             <rotate x="1" y="0" z="1" angle="177.36"/>
16         </transform>
17         <models>
18             <model file="sphere_demo.3d" />
19         </models>
20     </group>
```

Excerto XML 3.2: Mércurio e Vénus.

3.4.3 Luas

Já para representar as Luas do planeta Terra e de Marte foram aplicadas, a cada um foi aplicada uma translação para representar a sua distância ao Planeta e uma escala.

```
1 <group> <!-- Terra -->
2     <transform>
3         <translate x="40.5" y="0" z="0" />
4         <scale x="0.72" y="0.72" z="0.72"/>
5         <rotate x="1" y="0" z="1" angle="23.45"/>
6     </transform>
7     <models>
8         <model file="sphere_demo.3d" />
9     </models>
10 </group>
11 <group> <!-- Lua -->
12     <transform>
13         <translate x="3" y="0" z="0" />
14         <scale x="0.2" y="0.2" z="0.2" />
15     </transform>
16     <models>
17         <model file="sphere_demo.3d" />
18     </models>
19 </group>
20 <group> <!-- Marte -->
21     <transform>
22         <translate x="48.75" y="0" z="0" />
23         <scale x="0.4" y="0.4" z="0.4"/>
24         <rotate x="1" y="0" z="1" angle="25.19"/>
25     </transform>
26     <models>
```

```

27     <model file="sphere_demo.3d" />
28 </models>
29 <group> <!-- Fobos -->
30     <transform>
31         <translate x="1.5" y="0" z="0" />
32         <scale x="0.1" y="0.1" z="0.1" />
33     </transform>
34     <models>
35         <model file="sphere_demo.3d" />
36     </models>
37 </group>
38 <group> <!-- Deimos -->
39     <transform>
40         <translate x="2" y="0" z="0" />
41         <scale x="0.05" y="0.05" z="0.05" />
42     </transform>
43     <models>
44         <model file="sphere_demo.3d" />
45     </models>
46 </group>
47 </group>

```

Excerto XML 3.3: Planeta Terra, Marte e respectivas Luas.

3.4.4 Anel de Saturno

Por fim, para o anel de Saturno aplicou-se uma rotação para representar a inclinação do anel relativamente ao planeta e uma escala. Note-se que a rotação do anel é afetada pela rotação do planeta Saturno, tendo os cálculos utilizados previsto essa rotação.

```

1  <group> <!-- Saturno -->
2      <transform>
3          <translate x="173" y="0" z="0" />
4          <scale x="6.94" y="6.94" z="6.94"/>
5          <rotate x="1" y="0" z="1" angle="26.73"/>
6      </transform>
7      <models>
8          <model file="sphere_demo.3d" />
9      </models>
10 <group> <!-- Anel de Saturno -->
11     <transform>
12         <rotate x="1" y="0" z="0" angle="90"/>
13         <scale x="0.25" y="0.25" z="0.05" />
14     </transform>
15     <models>
16         <model file="torus_demo.3d"/> <!-- generator torus2 torus_demo
17         .3d -->
18     </models>
19 </group>

```

Excerto XML 3.4: Saturno e respetivo anel.

3.5 Resultados obtidos

Ao correr a engine com o ficheiro *solarSystemDisperso.xml* e *solarSystemDispersoCor.xml* como argumento, obtiveram-se os seguintes resultados, respetivamente, em que simplesmente aplicou-se a fórmula 3.4 aos pontos apresentados

na secção anterior e calcularam-se pontos para obter um sistema solar mais realista, mantendo a distância e escala.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (3.4)$$

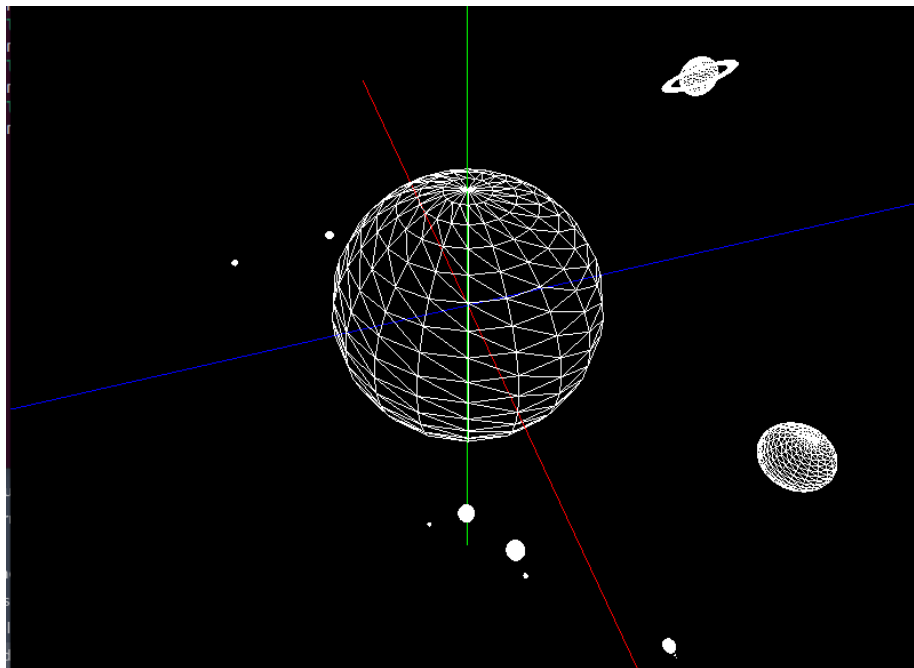


Figura 3.2: Sistema Solar Disperso

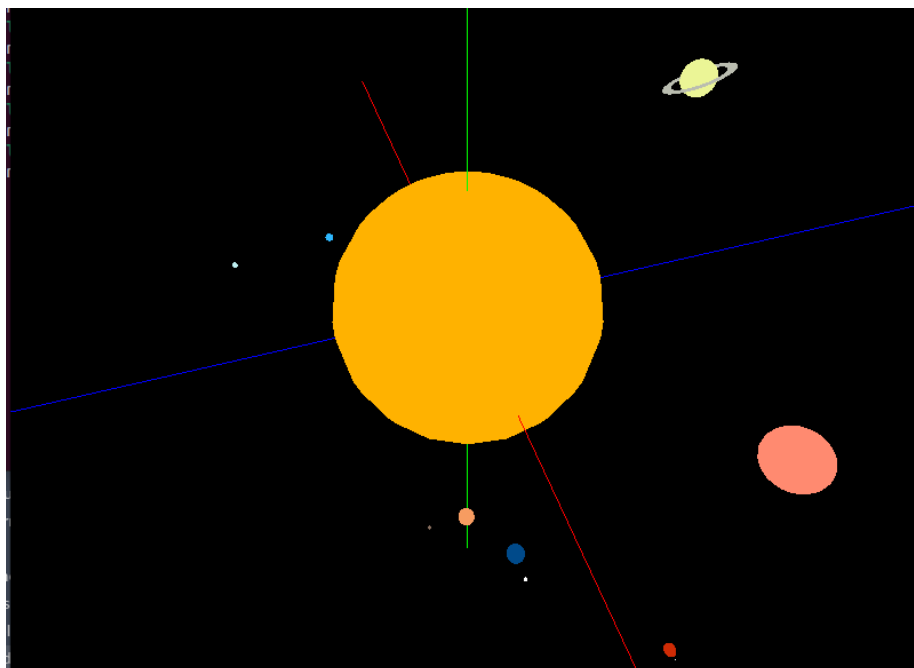


Figura 3.3: Sistema Solar Disperso com Cores *Filled*

Foi também elaborado um ficheiro que mostra os planetas do Sistema Solar em linha reta, para permitir uma visualização mais comparativa entre os planetas e as suas distâncias relativas.

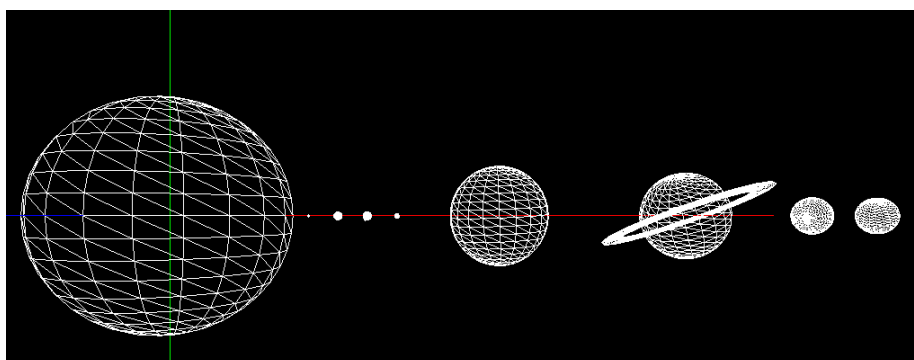


Figura 3.4: Sistema Solar Alinhado

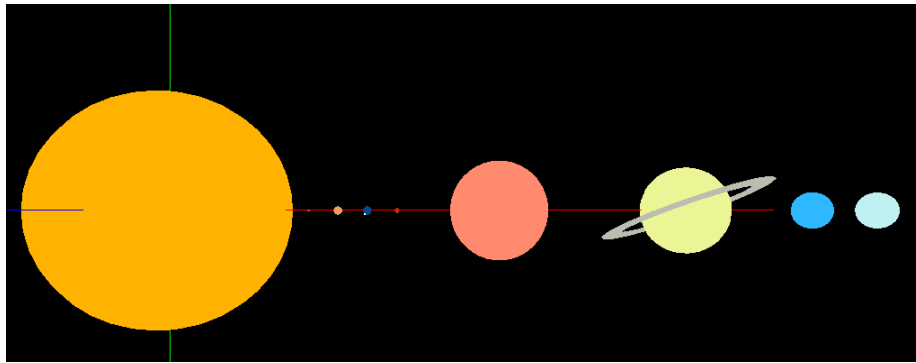


Figura 3.5: Sistema Solar Alinhado com Cores *Filled*

Capítulo 4

Conclusão

Na segunda fase do trabalho prático foi implementado a capacidade de transformações gráficas na criação de cenas hierárquicas contidas num ficheiro *XML* na aplicação **Engine**. Esta fase introduziu a capacidade de definir uma cena, a ser desenhada, na forma de “árvore”, no qual cada nó contém um conjunto de transformações geométricas, sendo elas **translações**, **rotações** e **escalas** a serem aplicadas a um conjunto de **modelos** e **subgrupos**.

No final desta fase foi ainda criada uma cena estática do sistema solar, incluindo o Sol, planetas que o compõem e algumas Luas. Além disso, como passo adicional foi introduzido na aplicação **Generator** uma nova primitiva gráfica, o **Anel de Torus**, utilizado para a representação do Anel de Saturno.

Geralmente, considera-se que a realização desta fase foi bem sucedida, uma vez que acreditamos que cumprimos com todos os objetivos propostos para a mesma e ainda visamos introduzir alguns extras, como o caso do Anel de Torus.

Além disso, esta segunda fase permitiu a consolidação das transformações geométricas assim como a gestão de matrizes de transformação.

Apêndice A

Anexos

A.1 Ficheiro *XML* do sistema solar

```
1 <world> <!-- a world can have a camera, a window definition, and a single group
   -->
2   <window width="512" height="512" />
3   <camera>
4     <position x="50" y="40" z="50" />
5     <lookAt x="0" y="0" z="0" />
6     <up x="0" y="1" z="0" /> <!-- optional, use these values as default-->
7     <projection fov="60" near="1" far="1000" /> <!-- optional, use these
       values as default-->
8   </camera>
9   <group>
10    <group> <!-- Sol -->
11      <transform>
12        <scale x="20" y="20" z="20" />
13      </transform>
14      <models>
15        <model file="sphere_demo.3d" /> <!-- generator sphere 1 20 20
          sphere_demo.3d -->
16      </models>
17    </group>
18    <group> <!-- Mercurio -->
19      <transform>
20        <translate x="31" y="0" z="0" />
21        <scale x="0.2" y="0.2" z="0.2"/>
22        <rotate x="1" y="0" z="1" angle="0.01"/>
23      </transform>
24      <models>
25        <model file="sphere_demo.3d" />
26      </models>
27    </group>
28    <group> <!-- Venus -->
29      <transform>
30        <translate x="36.25" y="0" z="0" />
31        <scale x="0.70" y="0.70" z="0.70"/>
32        <rotate x="1" y="0" z="1" angle="177.36"/>
33      </transform>
34      <models>
35        <model file="sphere_demo.3d" />
36      </models>
37    </group>
38    <group> <!-- Terra -->
```

```

39     <transform>
40         <translate x="40.5" y="0" z="0" />
41         <scale x="0.72" y="0.72" z="0.72"/>
42         <rotate x="1" y="0" z="1" angle="23.45"/>
43     </transform>
44     <models>
45         <model file="sphere_demo.3d" />
46     </models>
47 </group> <!-- Lua -->
48     <transform>
49         <translate x="3" y="0" z="0" />
50         <scale x="0.2" y="0.2" z="0.2" />
51     </transform>
52     <models>
53         <model file="sphere_demo.3d" />
54     </models>
55 </group>
56 </group>
57 <group> <!-- Marte -->
58     <transform>
59         <translate x="48.75" y="0" z="0" />
60         <scale x="0.4" y="0.4" z="0.4"/>
61         <rotate x="1" y="0" z="1" angle="25.19"/>
62     </transform>
63     <models>
64         <model file="sphere_demo.3d" />
65     </models>
66 </group> <!-- Fobos -->
67     <transform>
68         <translate x="1.5" y="0" z="0" />
69         <scale x="0.1" y="0.1" z="0.1" />
70     </transform>
71     <models>
72         <model file="sphere_demo.3d" />
73     </models>
74 </group>
75 <group> <!-- Deimos -->
76     <transform>
77         <translate x="2" y="0" z="0" />
78         <scale x="0.05" y="0.05" z="0.05" />
79     </transform>
80     <models>
81         <model file="sphere_demo.3d" />
82     </models>
83 </group>
84 </group>
85 <group> <!-- Jupiter -->
86     <transform>
87         <translate x="105.5" y="0" z="0" />
88         <scale x="8.1" y="8.2" z="8.2"/>
89         <rotate x="1" y="0" z="1" angle="3.13"/>
90     </transform>
91     <models>
92         <model file="sphere_demo.3d" />
93     </models>
94 </group>
95 <group> <!-- Saturno -->
96     <transform>
97         <translate x="173" y="0" z="0" />
98         <scale x="6.94" y="6.94" z="6.94"/>
99         <rotate x="1" y="0" z="1" angle="26.73"/>
100 </transform>

```

```

101     <models>
102         <model file="sphere_demo.3d" />
103     </models>
104     <group> <!-- Anel de Saturno -->
105         <transform>
106             <rotate x="1" y="0" z="0" angle="90"/>
107             <scale x="0.25" y="0.25" z="0.05" />
108         </transform>
109         <models>
110             <model file="torus_demo.3d"/> <!-- generator torus2 torus_demo
111                 .3d -->
112         </models>
113     </group>
114     <group> <!-- Urano -->
115         <transform>
116             <translate x="322.5" y="0" z="0" />
117             <scale x="2.94" y="2.94" z="2.94"/>
118             <rotate x="1" y="0" z="1" angle="97.77"/>
119         </transform>
120         <models>
121             <model file="sphere_demo.3d" />
122         </models>
123     </group>
124     <group> <!-- Neptuno -->
125         <transform>
126             <translate x="516.75" y="0" z="0" />
127             <scale x="2.86" y="2.86" z="2.86"/>
128             <rotate x="1" y="0" z="1" angle="28.32"/>
129         </transform>
130         <models>
131             <model file="sphere_demo.3d" />
132         </models>
133     </group>
134 </world>
135 </world>

```

Excerto XML A.1: XML desenvolvido do Sistema Solar.