梦幻诛仙Show（SVG版）实现方案概述

V0.1 刘东学 2011-6-24

# 概述

梦诛Show是梦幻诛仙仿照流行的QQSHOW实现的一套角色形象展示系统。本文档从技术方案实现角度描述QQSHOW的客户端渲染引擎如何完成对梦诛SHOW的渲染。可提供给SNS项目组在WEB端的FLASH中显示梦诛SHOW的实现方案进行参考。

本文档暂不对梦诛秀的商业方案进行论述：比如svgshow的定价方案，svgshow的商场售卖方案，包括上架显示定义等相关细节。

本文档重点解释如下方面：

1. 美术资源的组织和管理。
2. 客户端使用的配置数据。
3. 服务器端保存的有关角色自身的可变化的配置数据信息。
4. 与渲染策略有关的特别需要说明的问题，比如“缺省形象的处理方式”。

# 梦诛Show 的美术资源的组织和管理

梦诛秀的美术资源主要分两部分组成。一种是用于显示梦诛秀的部件缩略图，（TGA位图文件），本文档暂不阐述。

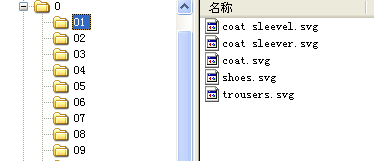
另一种是所有梦诛SHOW的原始矢量图（svg矢量图文件，xml格式），其中这一部分资源目录结构如下：

|  |  |
| --- | --- |
| 男性svgshow根目录 | 1 – 女性svgshow根目录。 |
|  |  |

|  |  |
| --- | --- |
| 0/Default – 男性svgshow缺省形象根目录。效果图如下： | 1/Default – 女性svgshow缺省形象根目录。效果图如下： |
|  |  |

每个子目录中，是由美术师提供的整套svgshow的资源，由多个svg文件组成。

比如1/01这个目录下的所有svg文件列表：



部分套装的效果图如下：

|  |  |
| --- | --- |
| 0/01套装效果图： | 1/01套装效果图 |
|  |  |

# 梦诛Show相关的配置数据及定义说明

梦诛秀可由一个专门的ShowEditor.exe程序对所有资源进行配置。该工具最后输出qqshow.xml。

下图是女性角色套装1的编辑器属性列表：



这里单独解释与渲染相关的字段。

1. **文件名：**定义了svg美术资源的相对路径。其根路径由客户端程序根据安装环境和打包配置方式确定。比如，梦诛中的show目录位于安装程序的根目录下。参见下图的show.pfs资源包实例图：

|  |  |
| --- | --- |
|  |  |

1. **id ：**

每个svg部件都有一个唯一的id标志。需要特别说明的是，对于裸模资源和可换装资源，其ID是有一个简单范围定义。即0~99范围内的部件是不能被编辑器所移动、调整、替换的。100（含）以上的部件，是普通的服装、首饰等可以被用户替换的部件。

|  |
| --- |
| enum {  CHANGED\_PART\_MIN\_ID = 100, // 可换装模型的最小id，小于这个id为裸模  }; |

1. **层**

定义了svg部件所渲染的层信息。层定义如下（以C/C++枚举表示）：

|  |
| --- |
| enum SHOW\_LAYER // QQ 秀 层定义  {  MH\_SHOW\_BACK\_GROUND = 0,// 0. 背景  MH\_SHOW\_VEHICLE, // 1. 交通工具（可移动）  MH\_SHOW\_COMPANION, // 2. 同伴（可移动）  MH\_SHOW\_WING, // 3. 翅膀  MH\_SHOW\_HAT\_B, // 4. 帽子B（可移动）  MH\_SHOW\_HAIR\_B, // 5. 头发B  MH\_SHOW\_MANTLE\_B, // 6. 披风B  MH\_SHOW\_ARM\_R, // 7. 手臂(右)  MH\_SHOW\_LEG, // 8. 腿  MH\_SHOW\_BODY, // 9. 胸口  MH\_SHOW\_ARM\_L, // 10. 手（左手）  MH\_SHOW\_SHIRT, // 11. 衣服T恤  MH\_SHOW\_SHIRT\_SLEEVE\_R, // 12. 衣服T恤 袖子(右)  MH\_SHOW\_COAT\_SLEEVE\_R, // 13. 外衣 袖子(右)  MH\_SHOW\_SOCKS, // 14. 袜子  MH\_SHOW\_TROUSERS\_INNER, // 15. 裤子(鞋里面)  MH\_SHOW\_SHOES, // 16. 鞋  MH\_SHOW\_TROUSERS\_OUTER, // 17. 裤子(鞋外面)  MH\_SHOW\_WAISTBAND, // 18. 皮带  MH\_SHOW\_COAT, // 19. 外衣  MH\_SHOW\_SHIRT\_SLEEVE\_L, // 20. 衣服T恤 袖子(左)  MH\_SHOW\_COAT\_SLEEVE\_L, // 21. 外衣 袖子(左)  MH\_SHOW\_MANTLE\_A, // 22. 披风A  MH\_SHOW\_HEAD\_EAR, // 23. 脸型和耳朵  MH\_SHOW\_BEARD, // 24. 胡子  MH\_SHOW\_EARRING, // 25. 五官和耳环 (耳环)  MH\_SHOW\_BROWS, // 26. 五官和耳环 (眉毛)  MH\_SHOW\_MOUTH, // 27. 五官和耳环 (嘴)  MH\_SHOW\_NOSE, // 28. 五官和耳环 (鼻子)  MH\_SHOW\_EYE, // 29. 五官和耳环 (眼睛)  MH\_SHOW\_HAIR\_A, // 30. 头发A  MH\_SHOW\_TATTOO, // 31. 文身（可移动）  MH\_SHOW\_HAT\_A, // 32. 帽子A（可移动）  MH\_SHOW\_ACCESSORIES\_A, // 33. 配饰（可移动）  MH\_SHOW\_SHAWL, // 34. 围巾（可移动）  MH\_SHOW\_ACCESSORIES\_B, // 35. 周边装饰（可移动）  MH\_SHOW\_PET, // 36. 宠物（可移动）  MH\_SHOW\_SUBTITLING, // 37. 字幕（可移动）  MH\_SHOW\_MAGZINE, // 38. 杂志  MH\_SHOW\_PHOTO\_FRAME, // 39. 相框  MH\_SHOW\_LAYER\_MAX, // 40 计数  }; |

层决定了svgshow的渲染引擎安排的svg部件的叠放顺序。层越小，svg部件越靠下。反之，svg部件越靠上。换言之，只有层相对比较高的部件才会遮挡层相对比较低的部件。比如，0层（背景层）最低，会被所有上面的部件遮挡。

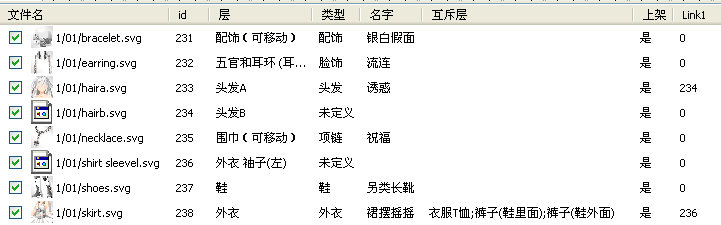
1. **类型**

类型，是部件类别的定义。在ShowEditor.exe中，可以按照类型进行分类，另外，在梦诛客户端的SHOW商城中，也可以根据类型划分到不同的显示列表中，方便用户查找。另外在换装过程中，多数相同类型的部件是互斥的。比如帽子，一个show戴上帽子A，然后再戴帽子B时，帽子A会自动被show引擎进行替换。但具体互斥定义是由层来确定的，而不是由类型来定义的。有关互斥的解释，参见5）互斥小节。

|  |
| --- |
| enum SHOW\_TYPE // QQ 秀 类型  {  SHOW\_TYPE\_UNDEFINE = -1, // 未定义  //------------------------------------------------------------------------------  // 服装  SHOW\_TYPE\_COAT = 0, // 外衣  SHOW\_TYPE\_TROUSERS, // 裤子  SHOW\_TYPE\_SUIT, // 套装  SHOW\_TYPE\_SHAWL, // 围巾  SHOW\_TYPE\_HAT, // 帽子  SHOW\_TYPE\_SHOES, // 鞋  SHOW\_TYPE\_MANTLE, // 披风  //------------------------------------------------------------------------------  // 美容  SHOW\_TYPE\_HAIR, // 头发  SHOW\_TYPE\_EXPRESSION, // 表情  SHOW\_TYPE\_HEAD\_EAR, // 头像  //------------------------------------------------------------------------------  // 场景  SHOW\_TYPE\_BACK\_GROUND, // 背景  SHOW\_TYPE\_MAGZINE, // 杂志  SHOW\_TYPE\_PHOTO\_FRAME, // 相框  SHOW\_TYPE\_SUBTITLING, // 字幕  SHOW\_TYPE\_AMBITUS, // 周边装饰  SHOW\_TYPE\_ACCESSORIES\_A, // 配饰  SHOW\_TYPE\_ASSOCIATE, // 同伴  SHOW\_TYPE\_VEHICLE, // 交通工具  //------------------------------------------------------------------------------  // 配饰  SHOW\_TYPE\_HEADWEAR, // 头饰  SHOW\_TYPE\_FACEWEAR, // 脸饰  SHOW\_TYPE\_NECKLACE, // 项链  SHOW\_TYPE\_EARRING, // 耳环  SHOW\_TYPE\_GLASSES, // 眼镜  SHOW\_TYPE\_BAG, // 包包  SHOW\_TYPE\_WING, // 翅膀  SHOW\_TYPE\_BROOCH, // 胸针  SHOW\_TYPE\_WAISTWEAR, // 腰饰  SHOW\_TYPE\_OTHERWEAR, // 其他  //------------------------------------------------------------------------------  // 宠物  SHOW\_TYPE\_NORMAL\_PET, // 普通宠物  SHOW\_TYPE\_SPECAL\_PET, // 特殊宠物  //------------------------------------------------------------------------------  // 套装  SHOW\_TYPE\_COMPETITIVE, // 精品  SHOW\_TYPE\_POPULARITY, // 人气  //------------------------------------------------------------------------------  // VIP  SHOW\_TYPE\_VIP, // VIP  //----------------------------------  SHOW\_TYPE\_MAX, // 计数  }; |

1. **互斥**

在下图中，裙摆摇摇这件外衣，有一个互斥层的定义。说明该部件与“衣服T恤”“裤子（鞋里面）”“裤子（鞋外面）”3个层的部件是互斥的。



渲染引擎在渲染角色穿着的svg部件时，根据互斥的定义，实现在添加一个部件时，摘除身上已有的部件。

另外，如果摘除的是缺省形象的部件，注意在svg部件被移除时，需要还原回来。这一点，ShowEditor.exe程序并未完全按照这个要求执行，但梦诛客户端SHOW商城中已经做到。其他仅仅用于显示的渲染引擎，不必考虑这个还原策略。

1. **link**

link定义了原本是一个应用逻辑部件（比如头发）由于层的遮挡关系，被拆分成了多个部件的一个关联关系。比如：

在上图中，id为233的“诱惑”头发A层有一个Link1属性，其值为234。

这说明，头发A和头发B是一对关联部件，它们共同定义了头发这个逻辑部件。渲染引擎在渲染233（头发A）这个部件时，要同时加载234（头发B）这个部件，但，二者肯定是未在同一个层上渲染。

另外，所有部件的关联关系，最多有4个。

|  |
| --- |
| enum {  LINK\_MAX\_COUNT = 4, // linkID 最大数  }; |

ShowEditor.exe 工具导出的xml定义如下：

为了简化，下图仅仅包含了1/01,0/01两套服装的定义。

|  |
| --- |
| <?xml version="1.0" encoding="UTF-16"?>  <data>  <Show id="231" fileName="1/01/bracelet.svg" iconFileName="1/01/bracelet.tga" showSex="1" layer="33" showType="15" price="150" name="银白假面" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;146.5;62.5" box="0;0;0;0"/>  <Show id="232" fileName="1/01/earring.svg" iconFileName="1/01/earring.tga" showSex="1" layer="25" showType="19" price="80" name="流连" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;68;38" box="0;0;0;0"/>  <Show id="233" fileName="1/01/haira.svg" iconFileName="1/01/haira.tga" showSex="1" layer="30" showType="7" price="150" name="诱惑" showDescription="" flag="0" issaling="1" linkID="234" conflict="0" transform="1;0;0;1;43.5;10.5" box="0;0;0;0"/>  <Show id="234" fileName="1/01/hairb.svg" iconFileName="1/01/hairb.tga" showSex="1" layer="5" showType="-1" price="0" name="" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;35.5;14.5" box="0;0;0;0"/>  <Show id="235" fileName="1/01/necklace.svg" iconFileName="1/01/necklace.tga" showSex="1" layer="34" showType="20" price="80" name="祝福" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;86;60.5" box="0;0;0;0"/>  <Show id="236" fileName="1/01/shirt sleevel.svg" iconFileName="1/01/shirt sleevel.tga" showSex="1" layer="21" showType="-1" price="0" name="" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;111;88.5" box="0;0;0;0"/>  <Show id="237" fileName="1/01/shoes.svg" iconFileName="1/01/shoes.tga" showSex="1" layer="16" showType="5" price="150" name="另类长靴" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;57;206.5" box="0;0;0;0"/>  <Show id="238" fileName="1/01/skirt.svg" iconFileName="1/01/skirt.tga" showSex="1" layer="19" showType="0" price="150" name="裙摆摇摇" showDescription="" flag="0" issaling="1" linkID="236" conflict="165888" transform="1;0;0;1;30.5;59" box="0;0;0;0"/>  <Show id="267" fileName="0/01/coat sleevel.svg" iconFileName="0/01/coat sleevel.tga" showSex="0" layer="13" showType="-1" price="0" name="" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;0;0" box="0;0;0;0"/>  <Show id="268" fileName="0/01/coat sleever.svg" iconFileName="0/01/coat sleever.tga" showSex="0" layer="21" showType="-1" price="0" name="" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;0;0" box="0;0;0;0"/>  <Show id="269" fileName="0/01/coat.svg" iconFileName="0/01/coat.tga" showSex="0" layer="19" showType="0" price="150" name="时尚型男" showDescription="" flag="0" issaling="1" linkID="267;268" conflict="0" transform="1;0;0;1;0;0" box="0;0;0;0"/>  <Show id="270" fileName="0/01/shoes.svg" iconFileName="0/01/shoes.tga" showSex="0" layer="16" showType="5" price="150" name="运动男孩" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;0;0" box="0;0;0;0"/>  <Show id="271" fileName="0/01/trousers.svg" iconFileName="0/01/trousers.tga" showSex="0" layer="17" showType="1" price="150" name="黑色经典" showDescription="" flag="0" issaling="1" linkID="" conflict="0" transform="1;0;0;1;0;0" box="0;0;0;0"/>  </data> |

该定义中，transform属性定义了一个部件的坐标微调参数。比如transform="1;0;0;1;146.5;62.5"定义了该部件按照如下仿射变换矩阵进行变换后再显示到界面上。

# 服务器有关梦诛show的数据保存方案

每个角色可以购买多套svg show。其中一个时刻穿在身上的只有一套。相关数据库定义如下：

|  |
| --- |
| 角色购买的svgshow定义： |
| <xbean name="QQShowItem">  <variable name="key" type="int"/>物品在包裹中的key  <variable name="typeid" type="int"/>物品类型ID  <variable name="itemid" type="int"/>对应的物品ID  <variable name="expiretime" type="long"/>到期时间  </xbean>  <xbean name="QQShowBag">  <variable name="nextid" type="int" default="1"/>下一个可用的ID  <variable name="items" type="map" key="int" value="QQShowItem"/>购买的QQ秀  </xbean>  <table name="qqshow" key="long" value="QQShowBag" lock="rolelock" daemon="true" foreign="key:basic"  cacheCapacity="15000" cachehigh="512" cachelow="256"/> |
| 角色穿在身上的svgshow定义： |
| <xbean name="QQShowInfo">  <variable name="ratio" type="int"/>放大比率  <variable name="posX" type="int"/>位置坐标（X）  <variable name="posY" type="int"/>位置坐标（Y）  <variable name="showMap" type="map" key="int" value="QQShowDetail"/>具体的效果列表  </xbean>  <xbean name="QQShowDetail">  <variable name="posX" type="int"/>位置坐标（X）  <variable name="posY" type="int"/>位置坐标（Y）  <variable name="direction" type="int"/>旋转角度  ~~<variable name="conflict" type="int"/>冲突记录~~  </xbean> |
| 有关的表： |
| <xbean name="FriendSetting">  <!—此处省略其它属性若干-->  <variable name="qqShow" type="QQShowInfo"/>QQ秀  </xbean>  <table name="friend\_setting" key="long" value="FriendSetting" lock="rolelock" daemon="true" foreign="key:basic" cacheCapacity="10240" cachehigh="512" cachelow="256"/> |

其中，有关渲染的参数是穿在用户身上的show的信息。解释如下：

QQShowInfo包含的放大比率、位置坐标（posX,posY）参数是整套Svgshow的参数。用户可以通过客户端界面调整svgshow的放大比率以及显示的位置坐标，以达到自己感觉理想的形象展示效果。

|  |  |
| --- | --- |
| **梦诛的SHOW商城中的女性show展示效果示意图** | |
| 商城中的缺省形象 | 商城中经过放大平移之后的形象 |
|  |  |
| 在好友设置中看到的形象 | 在好友设置中看到的形象 |
|  |  |

QQShowInfo包含的showMap字段，是这套svgshow的所有部件的定义字典。主键是部件的id，映射的内容是QQShowDetail所包含的信息。由QQShowDetail的定义可见，每个部件自身也有一个坐标定义（posX,posY），旋转角度(direction)和冲突定义(conflict)。其中坐标、旋转角度是用于可自由移动部件（比如手包、配饰等）的用户编辑的参数信息。冲突定义(conflict)，已废弃，不再使用。

# 缺省形象的相关定义及说明

缺省形象是预定义的，它们没有定义在qqshow.xml中。下表是一段C++版本的代码，仅供实现者参考。

**特别说明的是，缺省形象中的部分部件是可以被编辑器和客户端程序替换的。因此，在渲染用户自身穿戴的qqshow时，需要将缺省的qqshow形象中的冲突部件替换下来。不冲突的保留不变。**

**比如嘴巴这个部件是可以被1/17/mouth.svg这个svg进行替换的。**

|  |
| --- |
| // GameClient  {  int nID(0);  // 女 裸模  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_BODY;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/body.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_BROWS;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/brows.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_EYE;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/eyes.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_HEAD\_EAR;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/head.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_ARM\_R;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/hand01.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_ARM\_L;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/hand02.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_MOUTH;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/mouth.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_NOSE;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/nose.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_TROUSERS\_INNER;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/trousers.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_SHIRT;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/underwear.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_HAIR\_B;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/hair01.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_HAIR\_A;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/hair02.svg";  kShowInfo.wstrIconFileName = L"";  kShowInfo.AddLink(nID - 2);  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 1;  kShowInfo.nLayer = MH\_SHOW\_LEG;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"1/Default/leg.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  //------------------------------------------------------------------------------  // 男 裸模  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_BODY;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/body.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_BROWS;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/brows.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_EYE;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/eyes.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_ARM\_L;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/handL.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_ARM\_R;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/handR.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_HEAD\_EAR;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/head.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_LEG;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/leg.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_MOUTH;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/mouth.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_NOSE;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/nose.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_SHIRT;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/underwear.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_TROUSERS\_INNER;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/trousers.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_HAIR\_B;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/hair01.svg";  kShowInfo.wstrIconFileName = L"";  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  {  QQShowInfo kShowInfo;  kShowInfo.nID = nID++;  kShowInfo.nSex = 0;  kShowInfo.nLayer = MH\_SHOW\_HAIR\_A;  kShowInfo.nType = SHOW\_TYPE\_UNDEFINE;  kShowInfo.wstrFileName = L"0/Default/hair02.svg";  kShowInfo.wstrIconFileName = L"";  kShowInfo.AddLink(nID - 2);  kFriendData.AddQQShow(kShowInfo);  kFriendData.SetDefaultShow(kShowInfo);  }  } |