



(12) 发明专利

(10) 授权公告号 CN 102099824 B

(45) 授权公告日 2015. 09. 30

(21) 申请号 200980127886. 3

H04N 21/258(2011. 01)

(22) 申请日 2009. 07. 16

H04N 21/4402(2011. 01)

(30) 优先权数据

H04N 21/8543(2011. 01)

10-2008-0069343 2008. 07. 16 KR

H04L 29/08(2006. 01)

(85) PCT国际申请进入国家阶段日

(56) 对比文件

2011. 01. 17

CN 1689046 A, 2005. 10. 26,

(86) PCT国际申请的申请数据

CN 101196818 A, 2008. 06. 11, 第 4 页第 6

PCT/KR2009/003927 2009. 07. 16

行 - 第 5 页 25 行 .

(87) PCT国际申请的公布数据

审查员 郭姝梅

W02010/008220 EN 2010. 01. 21

(73) 专利权人 三星电子株式会社

地址 韩国京畿道

专利权人 庆熙大学校产学协力团

(72) 发明人 宋在涓 黄棲煥 金奎宪

(74) 专利代理机构 北京市柳沈律师事务所

11105

代理人 邵亚丽

(51) Int. Cl.

H04N 21/2343(2011. 01)

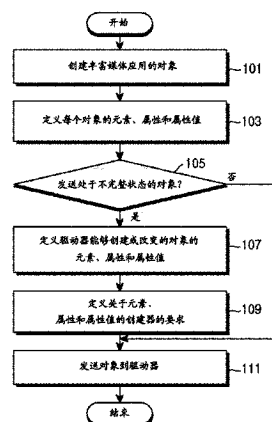
权利要求书3页 说明书8页 附图3页

(54) 发明名称

用于发送和接收丰富媒体内容的方法和装置

(57) 摘要

一种用于发送和接收丰富媒体内容的方法和装置。创建器以其中驱动器能够改变配置信息的可改变格式创建按分级结构形成的每个节点的配置信息, 以及向驱动器发送包括每个节点的配置信息的内容。驱动器接收该内容, 确定在接收的内容中按分级结构形成的每个节点的配置信息是否是以可改变格式创建的, 当配置信息是以可改变格式创建的时重新配置每个节点, 以及执行在重新配置的节点上的应用。



1. 一种用于在创建器中发送丰富媒体应用内容的方法,包括:
通过创建器确定是否发送不完整状态的配置信息;
如果创建器确定发送不完整状态的配置信息,则以其中驱动器能够改变配置信息的可改变格式来创建按分级结构形成的每个节点的配置信息;以及
向驱动器发送包括每个节点的配置信息的内容,
其中,创建每个节点的配置信息包括将创建器的要求包括在每个节点的配置信息中,
其中,所述可改变格式包括如下格式之一:
其中元素、属性和属性值中的至少一个不完整的格式;以及
其中当所述驱动器没有通过基于所述驱动器的唯一权利的选择来选择用于完成场景的元素时,所述创建器指定通过其所述驱动器将完成场景的默认元素的格式,
其中,通过所述驱动器将不完整状态的配置信息改变为完整状态的配置信息。
2. 根据权利要求1所述的方法,其中,所述创建器的要求指示创建器指定所述驱动器不能改变的元素和属性值中的至少一个。
3. 根据权利要求1所述的方法,其中,所述创建器的要求指示创建器指定所述驱动器不能创建的元素和属性值中的至少一个。
4. 根据权利要求1所述的方法,其中,所述创建器的要求指示创建器指定所述驱动器能够选择的值。
5. 一种用于在驱动器中接收丰富媒体应用内容的方法,包括:
接收来自创建器的内容;
确定在接收的内容中按分级结构形成的每个节点的配置信息是否是以其中驱动器能够改变配置信息的可改变格式创建的;
当配置信息是以可改变格式创建的时,根据驱动器的判决条件重新配置每个节点;以及
执行在重新配置的节点上的应用,
其中,创建每个节点的配置信息包括将创建器的要求包括在每个节点的配置信息中,
其中,所述可改变格式包括如下格式之一:
其中元素、属性和属性值中的至少一个不完整的格式;以及
其中当所述驱动器没有通过基于所述驱动器的唯一权利的选择来选择用于完成场景的元素时,所述创建器指定通过其所述驱动器将完成场景的默认元素的格式,
其中,所接收的内容包括不完整状态的配置信息;以及
其中,通过所述驱动器将不完整状态的配置信息改变为完整状态的配置信息。
6. 根据权利要求5所述的方法,还包括:
确定在对元素、属性和属性值中的至少一个不完整的节点中是否包括创建器的要求;
当在节点中包括创建器的要求时,根据创建器的要求创建每个节点;以及
执行在创建的节点上的应用。
7. 根据权利要求5所述的方法,其中,所述创建器的要求指示创建器指定所述驱动器不能改变的元素和属性值中的至少一个。
8. 根据权利要求5所述的方法,其中,所述创建器的要求指示创建器指定所述驱动器不能创建的元素和属性值中的至少一个。

9. 根据权利要求 5 所述的方法, 其中, 所述创建器的要求指示创建器指定所述驱动器能够选择的值。

10. 根据权利要求 5 所述的方法, 其中, 所述驱动器的判决条件包括所述驱动器的资源能力和丰富媒体应用用户的偏好中的至少一个。

11. 一种用于在创建器中发送丰富媒体应用内容的装置, 包括:

内容创建器, 通过创建器确定是否发送不完整状态的配置信息, 如果创建器确定发送不完整状态的配置信息, 则以其中驱动器能够改变配置信息的可改变格式来创建按分级结构形成的每个节点的配置信息; 和

发送器, 用于向驱动器发送包括每个节点的配置信息的内容,

其中, 创建每个节点的配置信息包括将创建器的要求包括在每个节点的配置信息中,

其中, 所述可改变格式包括如下格式之一:

其中元素、属性和属性值中的至少一个不完整的格式; 以及

其中当所述驱动器没有通过基于所述驱动器的唯一权利的选择来选择用于完成场景的元素时, 所述创建器指定通过其所述驱动器将完成场景的默认元素的格式,

其中, 通过所述驱动器将不完整状态的配置信息改变为完整状态的配置信息。

12. 根据权利要求 11 所述的装置, 其中, 所述创建器的要求指示创建器指定所述驱动器不能改变的元素和属性值中的至少一个。

13. 根据权利要求 11 所述的装置, 其中, 所述创建器的要求指示创建器指定所述驱动器不能创建的元素和属性值中的至少一个。

14. 根据权利要求 11 所述的装置, 其中, 所述创建器的要求指示创建器指定所述驱动器能够选择的值。

15. 一种用于在驱动器中接收丰富媒体应用内容的装置, 包括:

接收器, 用于接收来自创建器的内容;

分级结构创建器, 用于确定在接收的内容中按分级结构形成的每个节点的配置信息是否是以其中驱动器能够改变配置信息的可改变格式创建的, 并且当配置信息是以可改变格式创建的时, 根据驱动器的判决条件重新配置每个节点; 和

内容执行器, 用于执行在重新配置的节点上的应用,

其中, 创建每个节点的配置信息包括将创建器的要求包括在每个节点的配置信息中,

其中, 所述可改变格式包括如下格式之一:

其中元素、属性和属性值中的至少一个不完整的格式; 以及

其中当所述驱动器没有通过基于所述驱动器的唯一权利的选择来选择用于完成场景的元素时, 所述创建器指定通过其所述驱动器将完成场景的默认元素的格式,

其中, 所接收的内容包括不完整状态的配置信息; 以及

其中, 通过所述驱动器将不完整状态的配置信息改变为完整状态的配置信息。

16. 根据权利要求 15 所述的装置, 其中, 分级结构创建器确定在对其未定义元素、属性和属性值中的至少一个的节点中是否包括创建器的要求, 并且当在节点中包括创建器的要求时, 根据创建器的要求创建每个节点。

17. 根据权利要求 15 所述的装置, 其中, 所述创建器的要求指示创建器指定所述驱动器不能改变的元素和属性值中的至少一个。

18. 根据权利要求 15 所述的装置, 其中, 所述创建器的要求指示创建器指定所述驱动器不能创建的元素和属性值中的至少一个。

19. 根据权利要求 15 所述的装置, 其中, 所述创建器的要求指示创建器指定所述驱动器能够选择的值。

20. 根据权利要求 15 所述的装置, 其中, 所述驱动器的判决条件包括如下各项中的至少一个:

所述驱动器的资源能力 ; 和
丰富媒体应用用户的偏好。

用于发送和接收丰富媒体内容的方法和装置

技术领域

[0001] 本发明一般涉及一种用于发送和接收丰富媒体内容的方法和装置,且更具体地涉及一种用于发送和接收丰富媒体应用的对象的方法和装置。

背景技术

[0002] 丰富媒体应用 (rich media application) 是这样的应用,其包括:诸如音频、视频和文本的多媒体内容;在服务器和客户端之间的交互信息交换和控制;对用户的交互信息交换和控制;在这些元素之间的时间和空间排列。最显著地,该丰富媒体通常就其交互性而言区别于多媒体。也即,丰富媒体是以如下结构来创建的:除了媒体数据、图形元素数据和各个元素值的属性数据之外,利用诸如指示时间/空间关系的数据、用于表达交互性的事件和操作数据的信息来描述所述结构,并且以由特定丰富媒体引擎提供的语言来表达这些。能够以各种方式应用的语言包括例如 MPEG4 轻便应用场景表现 (LAsE R)、MPEG-4 场景描述的二进制格式 (BIFS)、万维网联盟 (W3C) 的可缩放矢量图形 (SVG) 等。

[0003] 为了驱动该应用,需要将以上信息片段转换为分级结构的操作。上述操作用于按照驱动元素的时间/空间顺序从根元素(最高节点)到根的子元素(最低节点)来对丰富媒体元素进行分层或层化。以树结构形成该分级(或分层)结构。一个根可以具有多个子节点,并且每个节点由构成节点的元素、表示每个元素的属性的类型和各个属性值组成。驱动该应用的丰富媒体引擎使用它的关联属性类型和属性值来以分级结构实现每个元素。这里,属性类型将被称为“属性”。

发明内容

[0004] 技术问题

[0005] 传统丰富媒体应用使用在内容对象的创建期间创建的内容对象的分级结构信息来提供丰富媒体应用。也即,丰富媒体应用实现实际上创建的丰富媒体信息片段的完整分级结构信息片段,因为如果未定义用于实现丰富媒体服务的分级结构的节点信息片段中的至少一个(即,元素、属性和属性值),则它不能实现丰富媒体服务。基于该原因,传统驱动器应该仅驱动由创建器创建的应用。

[0006] 技术方案

[0007] 设计本发明来解决至少以上所述的问题和/或不足并且提供至少以下所述的优点。

[0008] 因此,本发明的一方面提供一种装置和方法,其中使用丰富媒体应用的创建器向驱动器发送分级结构信息片段,其中元素、属性和属性值中的至少一个未定义或仅具有指定的默认元素。驱动器能够通过识别分级信息片段将其中元素、属性和属性值中的至少一个未定义的分级结构改变为完整分级结构,然后执行该应用。

[0009] 本发明的另一方面提供一种装置和方法,其中驱动器接收其中元素、属性和属性值中的至少一个未定义的分级结构信息,检查组成形成层的每个节点的信息片段,然后针

对不完整的或能够被改变的节点信息片段,基于由创建器展示的条件使用新元素或属性值来创建完整分级结构。

[0010] 依据本发明的一方面,提供一种用于在创建器中发送内容的方法。该方法包括:以其中驱动器能够改变配置信息的可改变格式来创建按分级结构形成的每个节点的配置信息;以及向驱动器发送包括每个节点的配置信息的内容。

[0011] 依据本发明的另一方面,提供一种用于在驱动器中接收内容的方法。该方法包括:接收内容;确定在接收的内容中按分级结构形成的每个节点的配置信息是否是以其中驱动器能够改变配置信息的可改变格式创建的;当配置信息是以可改变格式创建的时,根据驱动器的判决条件重新配置每个节点;以及执行在重新配置的节点上的应用。

[0012] 依据本发明的另一方面,提供一种用于在创建器中发送内容的装置。该装置包括:内容创建器,以其中驱动器能够改变配置信息的可改变格式来创建按分级结构形成的每个节点的配置信息;和发送器,用于向驱动器发送包括每个节点的配置信息的内容。

[0013] 依据本发明的另一方面,提供一种用于在驱动器中接收内容的装置。该装置包括:接收器,用于接收内容;分级结构创建器,用于确定在接收的内容中按分级结构形成的每个节点的配置信息是否是以其中驱动器能够改变配置信息的可改变格式创建的,并且当配置信息是以可改变格式创建的时根据驱动器的判决条件重新配置每个节点;和内容执行器,用于执行在重新配置的节点上的应用。

[0014] 有益效果

[0015] 根据本发明的实施例,应用创建器能够创建和发送具有不完整的分级结构的内容,并且驱动器能够完成具有不完整的分级结构的内容,然后驱动该完整内容,由此有助于减少创建器的创建负荷。

[0016] 另外,本发明能够使用由创建器做出的一个公共应用来在多个驱动器之间提供适应个别差异的(differentiated)服务。

附图说明

[0017] 通过结合附图的以下描述,本发明的一些实施例的以上和其他方面、特征和优点将更明了,其中:

[0018] 图 1 是示出根据本发明的实施例的创建器的操作的流程图;

[0019] 图 2 是示出根据本发明的实施例的驱动器的操作的流程图;

[0020] 图 3 是根据本发明的实施例的创建器装置的框图;以及

[0021] 图 4 是根据本发明的实施例的驱动器装置的框图。

[0022] 贯穿附图,相同的附图参考数字应当被理解为表示相同组件、特征和结构。

具体实施方式

[0023] 提供在说明书中定义的主题(诸如详细的结构和组件)以助于本发明的各种实施例的完整理解。因此,本领域的普通技术人员将认识到在不背离本发明的范围和精神的情况下,这里所述的实施例的各种改变和修改可以被实现。此外,出于清楚和简洁之故,公知的功能和结构的说明被省去。

[0024] 传统上,利用由创建器创建的分级结构的节点信息来实现应用驱动器,但是创建

器创建具有完整分级结构的节点。然而,依据本发明的实施例,提供一种方法和装置,其中应用创建器创建和发送具有不完整的分级结构的内容,并且驱动器完成具有不完整的分级结构的内容然后驱动该完整内容。

[0025] 本发明支持这样的方案,其中当驱动器即使驱动不完整的分级结构中的应用时也应该没有错误发生,并且即使创建器没有提供完整分级结构,驱动器也能够表达创建器的意图。也即,虽然创建器不能指定分级结构值,但是它向驱动器提供条件或限制,以使得它的意图能够被反映在完整应用中。然后,驱动器能够在定义的条件下来完成该应用的对象创建。以此方式,驱动器能够创建创建器期望的应用,即使创建器不向驱动器提供该应用的全部分级结构信息片段。

[0026] 另外,传统上,驱动器仅驱动由创建器创建的应用。然而,本发明提供另外的功能在于,当驱动器成功地解码来自创建器的不完整分级结构且存在关于相关对象值的条件时,驱动器能够分析条件并且基于驱动器的判决完成合适的对象创建。

[0027] 下面将描述的本发明的以下实施例使用 MPEG4 轻便应用场景表现 (LAsE R) 语言。但是,对本领域技术人员来说显然的是,本发明不限于该特定语言并且可以使用任何其他合适的语言。

[0028] 当按分级结构表达丰富媒体应用的对象配置时,如上所述,每个对象被表达为一个节点,并且每个节点包括三个信息片段,即,元素、属性和属性值。因此,依据本发明的实施例,创建器可以创建未指定三个信息片段中的至少一个的分级结构,并且本发明单独地提供关于元素、属性和属性值的每个的建议 (proposal)。

[0029] 下表 1 示出其中没有指定属性值的情况的示例。

[0030] 表 1

[0031]

```
***
<g id="adaptation01" ... > ... </g>
<rect id="adaptation02" fill="red"> ... </g>
<rect id="adaptation03" width="80" height="100"> ... </g>
<Change xlink:href="#adaptation02" attributeName="fill"/>
***
```

[0032] 在表 1 中,“<g>”和“<rect>”是表示丰富媒体对象的命令,它们在 LAsE R 中定义。例如,作为预定属性的“<rect>”是用于绘制矩形的命令。另外,“id”、“fill”、“width”和“height”是每个丰富媒体对象的元素值的属性,并且分别指示标识符 (id)、特定色彩或图案的表达、宽度和高度。依据本发明的实施例提出的“<change>”元素值指示具有“adaptation02”id 属性值的丰富媒体对象的“fill”属性的值没有被创建器指定,或者是能够由驱动器重新创建的属性值。

[0033] 下表 2A 示出其中没有指定属性值的情况的另一示例。

[0034] 表 2A

[0035]

```

    <g id="adaptation01" ... > ... </g>
    <rect id="adaptation02" fill="red"> ... </g>
    <rect id="adaptation03" width="80" height="100"> ... </g>
    <Change xlink:href="#adaptation02" attributeName="fill" recommendValue="red
yellow"/>
    <Change xlink:href="#adaptation03" attributeName="fill" recommendValue="green|blue"/>

```

[0036] 依据表 2A, 驱动器能够将具有“adaptation02”id 属性值的丰富媒体对象的“fill”属性的值选择为“红”和“黄”其中之一。

[0037] 下表 2B 示出当没有指定属性值时定义用于指示驱动器能够选择的值的 recommendValue(推荐值)的示例。

[0038] 表 2B

[0039]

```

<Attribute name="recommendValue" type="recommendValueType" use="optional"/>

<complexType name="recommendValueType">
  <complexContent>
    <union>
      <simpleType>
        <xs:restriction base="NMTOKENS"/>
      </simpleType>
      <simpleType name="recommendValueList">
        <list itemType="string"/>
      </simpleType>
      <simpleType name="RestrictedList">
        <xs:restriction base="recommendValueList">
          <xs:pattern value="( )"/>
        </xs:restriction>
      </xs:restriction>
    </union>
  </complexContent>
</complexType>

```

[0040] 下表 3 示出其中没有指定元素值的情况的示例。

[0041] 表 3

[0042]

```

    <g id="adaptation01" ... > ... </g>
    <rect id="adaptation02" fill="red"> ... </g>
    <rect id="adaptation03" width="80" height="100"> ...
    <empty id="element01" defaultElementName="rect"/>
    <empty id="element02" defaultElementName="circle" width="40" height="30"/>
    ...
    <Change xlink:href="#element01"/>
    <Change xlink:href="#element02" elementName="rect"/>

```

[0043] 依据表 3,“empty”指示驱动器能够创建相关的元素,并且驱动器然后具有确定算法的唯一权利,驱动器将通过该算法选择元素值。

[0044] 表 3 也示出一个示例,其中当驱动器没有通过基于驱动器的唯一权利的选择来选择用于完成场景的元素时,创建器使用“defaultElementName(默认元素名称)”来指定驱动器将通过其完成场景的默认元素。这里,“empty(空)”元素可以与对“empty”元素做出引用的“change(改变)”元素合作操作,或可以独立操作。

[0045] 下表 4 示出其中创建器指定了驱动器不能改变或创建的元素和属性值的情况的示例。

[0046] 表 4

[0047]

```
<g id="adaptation01" ... > ... </g>
<rect id="adaptation02" fill="red" unchangeable="true"> ... </g>
<rect id="adaptation03" width="80" height="100"> ... </g>
```

[0048] 依据表 4,针对由创建器指定的值的全部属性值,作为具有“adaptation02”id 属性值的丰富媒体对象的“rect”不能被驱动器改变或创建。

[0049] 下表 5 示出其中创建器指定了驱动器不能改变或创建的元素和属性值的情况的另一示例。

[0050] 表 5

[0051]

```
<g id="adaptation01" ... > ... </g>
<rect id="adaptation02" fill="red"> ... </g>
<rect id="adaptation03" width="80" height="100" fill="blue"> ... </g>
<UnChange xlink:href="#adaptation03" attributeName="width height"/>
```

[0052] 依据表 5,在作为具有“adaptation03”id 属性值的丰富媒体对象的“rect”的属性中,在“UnChange”元素中指定的“width”和“height”不能被驱动器改变和创建。然而,因为在“UnChange”元素中没有指定“rect”的“fill”属性,所以它能够被驱动器改变。

[0053] 下表 6 示出如下情况的示例,其中:给出示出具有由创建器完成的分级结构的对象的元素以及示出驱动器能够改变或创建的对象的分级结构的元素的定义。

[0054] 表 6

[0055]

```

<infra>
  <g id="adaptation01" ... > ... </g>
  <rect id="adaptation02" fill="red"> ... </g>
  <rect id="adaptation03" width="80" height="100"> ... </g>
</infra>
<DeviceDep>
  <Element_List>
    <rect xlink:href="#adaptation02"/>
    <circle/>
  </Element_List>
  <Attribute_List>
    <Attribute_List(Change) Attribute_Name="width"/>
    <Attribute_List Attribute_Name="fill" xlink:href="#adaptation02"/>
    <Attribute_List Attribute_Name="fill" xlink:href="#adaptation03"/>
  </Attribute_List>
</DeviceDep>

```

[0056] 参考表 6, <infra> 元素包括驱动器不能创建或改变的元素和属性, 而 <DeviceDep> 元素枚举了或指定了驱动器能够创建或改变的元素和属性。

[0057] 图 1 和图 2 是示出根据本发明的实施例的创建器和驱动器的操作的流程图。

[0058] 图 1 是示出根据本发明的实施例的创建器的操作的流程图。创建器和驱动器的物理位置经历改变。例如, 服务器和客户端设备可以分别充当创建器和驱动器, 或者客户端设备可以充当创建器和驱动器两者。

[0059] 参考图 1, 在步骤 101 中创建器开始创建丰富媒体应用的对象, 并且在步骤 103 中定义每个对象的元素值、属性和属性值。在步骤 105 中创建器确定是否发送处于不完整状态的对象。如果创建器确定不发送处于不完整状态的对象, 则创建器如在现有技术中进行的一样创建在步骤 103 中定义的对象然后在步骤 111 中发送创建的对象到驱动器。然而, 如果创建器确定发送处于不完整状态的对象, 则在步骤 107 中, 创建器检查或定义被使得不完整的元素、属性和属性值。能够使用表 1 到表 3 的示例来指定被使得不完整的元素、属性和属性值。创建器也能够指定驱动器能够创建或改变的对象元素、属性或属性值。

[0060] 之后, 在步骤 109, 创建器定义和插入其对于对象值的要求。这里, 能够使用表 4 到表 6 的示例来定义和插入该要求。然而, 当没有要求时, 步骤 109 是可选的。之后, 在步骤 111, 创建器完成对象创建并且传递创建的对象到驱动器。

[0061] 图 2 是示出根据本发明的实施例的驱动器的操作的流程图。

[0062] 参考图 2, 在步骤 201 中驱动器开始丰富媒体应用的解码。在步骤 203 中, 驱动器检查在该应用的对象的分级结构中的元素值、属性和属性值。在步骤 205 中, 驱动器确定是否存在任何以不完整状态发送的对象。如果不存在以不完整状态发送的对象, 则在步骤 217 中驱动器执行传统的丰富媒体应用。然而, 如果存在以不完整状态发送的对象, 则在步骤 207 中驱动器检查以不完整状态发送的对象的元素、属性和属性值的详情。也即, 驱动器检查它能够创建或改变的对象元素、属性和属性值。之后, 在步骤 209, 驱动器检查是否存在对于对象的创建器要求。在步骤 211, 驱动器检查在要求内它能够创建和改变的元素、属性

和属性值的列表。该结果可以取决于驱动器的资源和用于在要求内选择的准则。也即,即使在相同的要求下根据驱动器,或即使对于相同的驱动器当场合要求时,该结果也可能经历变化。

[0063] 在步骤 213,驱动器根据驱动器的特定条件从列表中选择元素、属性或属性值。这里,术语“驱动器的特定条件”指在要求内完成不完整的元素、属性和属性值的驱动器的判决条件或处于没有要求的状态的驱动器的判决条件。例如,该判决条件可以包括驱动器的资源能力和丰富媒体应用用户的偏好中的任何一个,且这里将不提供其详细描述。

[0064] 在步骤 215 中驱动器利用选择的值完成构成丰富媒体应用的对象的分级结构,并且在步骤 217 中执行丰富媒体应用。

[0065] 图 3 是根据本发明的实施例的创建器装置的框图。

[0066] 参考图 3,创建器接收媒体对象 301,然后在丰富媒体内容创建器 310 中创建丰富媒体应用的内容。在丰富媒体内容创建器 310 中,使用诸如交互性、时-空排列、事件和在媒体对象之间的命令的丰富媒体功能来执行创建每个组成对象的过程。

[0067] 不完整对象和不完整元素设置器 320 使用不完整对象和不完整元素的设置信息来创建组成对象。不完整对象和不完整元素设置器 320 通过定义在表 4 到表 6 的示例中写出的创建器要求来创建组成对象。

[0068] 丰富媒体内容发送器 340 向驱动器发送包括组成对象的丰富媒体内容。可以在丰富媒体内容创建器 310 中进一步包括编码丰富媒体内容的过程。要注意的是,因为如上所述创建器和驱动器可以物理上存在于一个设备中,故丰富媒体内容发送器 340 不仅包括对另一设备的发送,而且包括从创建器的逻辑实体到驱动器的逻辑实体的传递。

[0069] 图 4 是根据本发明的实施例的驱动器装置的框图。

[0070] 参考图 4,驱动器中的丰富媒体接收器 401 接收丰富媒体应用,并且发送接收的丰富媒体应用到分级结构创建器 420。分级结构创建器 420 创建该应用的分级结构,并且在分级结构创建器 420 中提供的不完整对象和不完整元素控制器 410 检查在每个对象的分级结构中的不完整对象和不完整元素。选择器 430 根据创建器要求和驱动器的选择来选择不完整的元素、属性和属性值。能够使用各种算法和资源来实现该选择过程,并且出于简化这里将不提供其详细描述。

[0071] 丰富媒体内容执行器 440 使用完整分级结构执行丰富媒体内容。这里,执行丰富媒体内容包括在驱动器中播放丰富媒体内容。

[0072] 如通过前述描述所显然的,根据本发明的实施例,应用创建器能够创建和发送具有不完整的分级结构的内容,并且驱动器能够完成具有不完整的分级结构的内容,然后驱动该完整内容,由此有助于减少创建器的创建负荷。

[0073] 另外,本发明能够使用由创建器做出的一个公共应用来在多个驱动器之间提供适应个别差异的服务。

[0074] 本发明的实施例也能够被实现为计算机可读记录介质上的计算机可读代码。计算机可读记录介质是能够存储随后由计算机系统读取的数据的任何数据存储设备。计算机可读记录介质的例子包括,但不限于只读存储器 (ROM)、随机存取存储器 (RAM)、CD-ROM、磁带、软盘、光数据存储设备和载波(诸如经过因特网的经由有线或无线传输路径的数据传输)。计算机可读记录介质也能够分布在网络耦合的计算机系统上从而以分布式方式存储

和执行计算机可读代码。此外,本发明所属领域的熟练的程序员能够容易将完成本发明的功能程序、代码、和代码段解释为在本发明的范围内。

[0075] 尽管已经参考本发明的一些实施例示出和描述了本发明,但是本领域技术人员将理解:在不背离由所附权利要求及其等价物限定的本发明的精神和范围的情况下,可以在这里进行形式和细节上的各种改变。

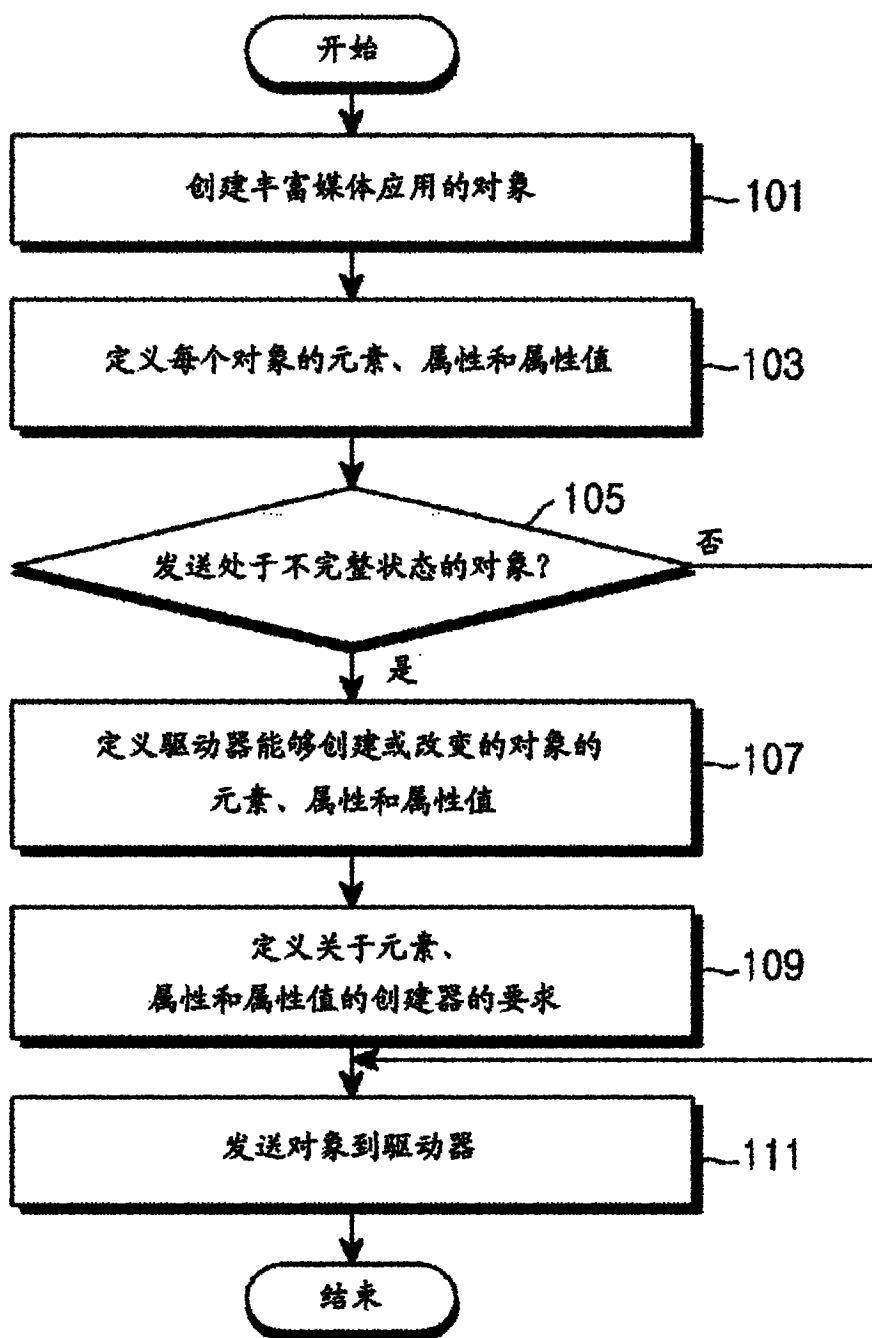


图 1

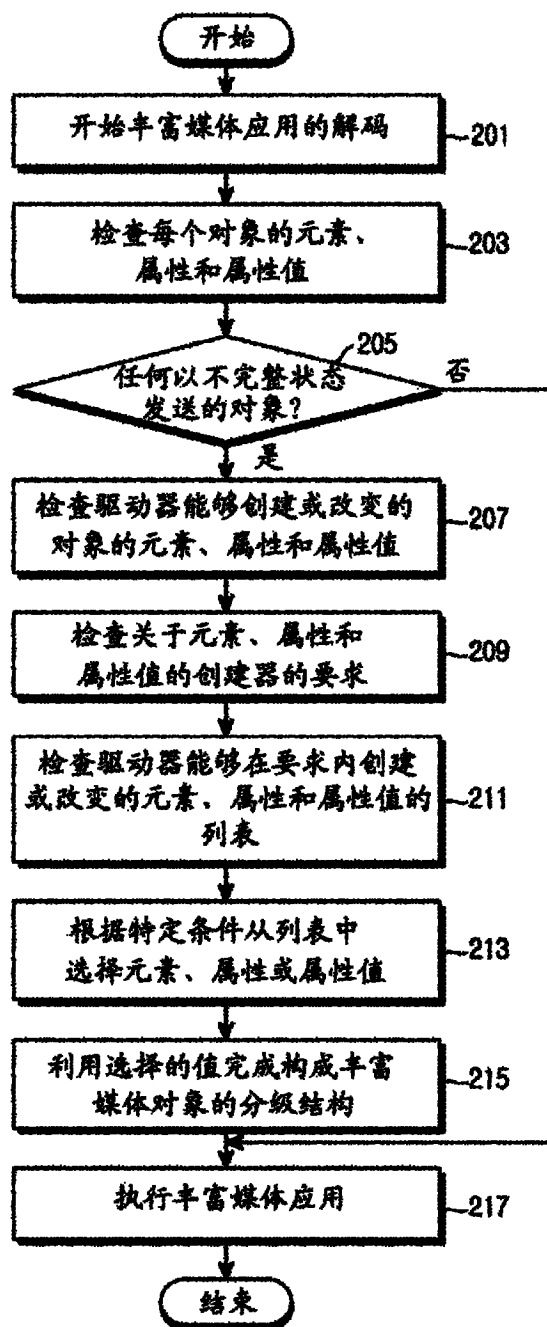


图 2

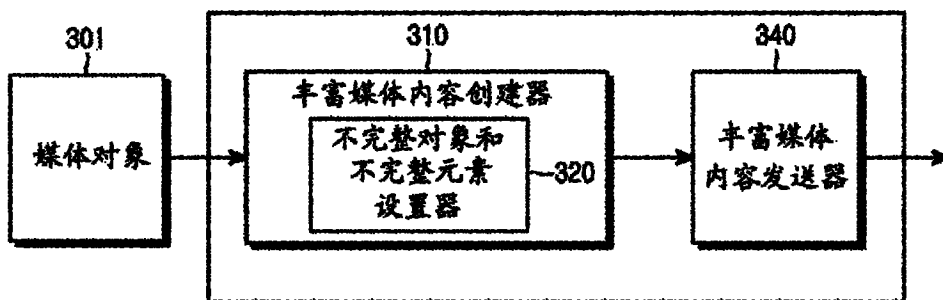


图 3

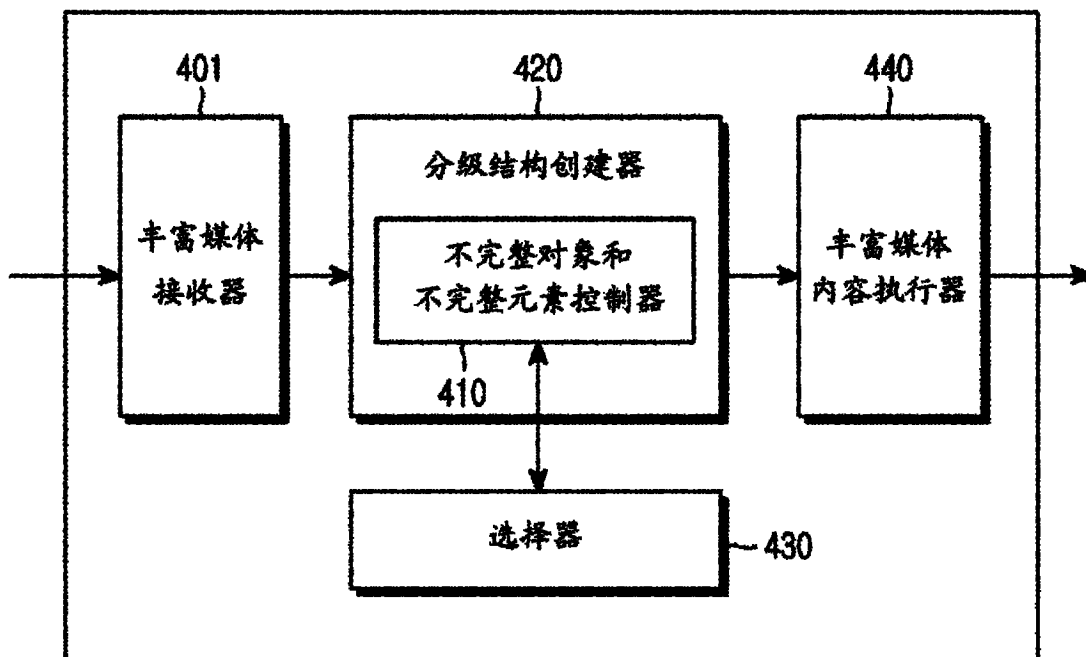


图 4