**项目编号:XXXXXXX**

需求描述书

**2014年06月**

1. 文档属性

|  |  |
| --- | --- |
| 文档属性 | 内容 |
| 项目/任务名称 | 打印平台库 |
| 项目/任务编号 |  |
| 文档名称 |  |
| 文档版本号 | V1.3 |
| 文档状态 | 正在编写 |
| 文档编写完成日期 |  |
| 作 者 |  |
| 内部发布范围 |  |

1. 文档变更历史清单

|  |  |  |  |
| --- | --- | --- | --- |
| 文档版本号 | 变更日期 | 修改人 | 备注 |
| 1.0 | 2014-6-16 | 徐强 | 初稿 |
| 1.1 | 2014-8-10 | 徐强 | 细化设计 |
| 1.2 | 2014-9-12 | 徐强 | 细化设计 |
| 1.3 | 2014-9-28 | 徐强 | 细化设计 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[第 1 章 总体说明 3](#_Toc401337692)

[1.1 项目背景 3](#_Toc401337693)

[1.1.1 打印指令没有完全解耦 3](#_Toc401337694)

[1.1.2 调用封装的不够细 3](#_Toc401337695)

[1.1.3 难于调试 3](#_Toc401337696)

[1.2 基本定义和术语 3](#_Toc401337697)

[1.3 相关文档人员 3](#_Toc401337698)

[第 2 章 设计思路 3](#_Toc401337699)

[2.1 上述问题的解决方法 3](#_Toc401337700)

[2.1.1 打印指令没有完全解耦 3](#_Toc401337701)

[2.1.2 调用封装的不够细 3](#_Toc401337702)

[2.1.3 难于调试 3](#_Toc401337703)

[第 3 章 软件架构描述 3](#_Toc401337704)

[3.1 系统组成部分 3](#_Toc401337705)

[3.2 总体设计原则 3](#_Toc401337706)

[3.2.1 组件间解耦 3](#_Toc401337707)

[第 4 章 需求描述 3](#_Toc401337708)

[4.1 打印服务器 3](#_Toc401337709)

[4.1.1 规划 3](#_Toc401337710)

[4.1.2 需求场景描述 3](#_Toc401337711)

[4.1.2.1 打印服务调用 3](#_Toc401337712)

[4.1.2.2 解析字符串 3](#_Toc401337713)

[4.1.2.3 计算格式排版 3](#_Toc401337714)

[4.1.2.4 生成打印伪指令 3](#_Toc401337715)

[4.1.2.5 开始JOB 3](#_Toc401337716)

[4.1.2.6 结束JOB 3](#_Toc401337717)

[第 5 章 附录 3](#_Toc401337718)

[5.1 打印例子描述 3](#_Toc401337719)

[5.1.1 例子 3](#_Toc401337720)

[5.1.2 例子解释 3](#_Toc401337721)

[5.1.3 初始值 3](#_Toc401337722)

[5.2 标签描述 3](#_Toc401337723)

[5.2.1 打印设置标签 3](#_Toc401337724)

[5.2.1.1 Page 3](#_Toc401337725)

[5.2.2 格式标签 3](#_Toc401337726)

[5.2.2.1 Underline 3](#_Toc401337727)

[5.2.2.2 Bold 3](#_Toc401337728)

[5.2.2.3 Doubleheight 3](#_Toc401337729)

[5.2.2.4 Doublewidth 3](#_Toc401337730)

[5.2.2.5 Tripleheight 3](#_Toc401337731)

[5.2.2.6 Triplewidth 3](#_Toc401337732)

[5.2.2.7 Sup 3](#_Toc401337733)

[5.2.2.8 Sub 3](#_Toc401337734)

[5.2.3 结构标签 3](#_Toc401337735)

[5.2.3.1 Br 3](#_Toc401337736)

[5.2.3.2 Turnpage 3](#_Toc401337737)

[5.2.3.3 Rect 3](#_Toc401337738)

[5.2.3.4 Table 3](#_Toc401337739)

[5.2.3.5 Tr 3](#_Toc401337740)

[5.2.3.6 Td 3](#_Toc401337741)

[第 6 章 概要设计 3](#_Toc401337742)

[6.1.1.1 格式化XML文件 3](#_Toc401337743)

[6.1.1.2 计算排版格式 3](#_Toc401337744)

[6.1.1.3 生成伪指令 3](#_Toc401337745)

# 总体说明

## 项目背景

因为公司的产品需要适应各种硬件产品,各种操作系统。因此,平台要具有良好的可扩展性和设备无关性。现有的打印库不能满足这两个要求,造成了开发中问题较多。主要表现在以下几个方面：

### 打印指令没有完全解耦

如：设置行间距。现在是1b 02 XX。这种调用方式,对于客户端是个负担。因为要查手册,而应用程序开发者不熟悉指令集。而且,指令是和特定指令集相关的。如：换成热敏打印机后,行间距是否还是这个指令?

### 调用封装的不够细

把格式控制,字符换行之类交给客户端来处理。使得客户端很难使用。

### 难于调试

指令是否执行,执行结构如何,字符断行结果如何?现在很难知道,使得程序内容行为不透明。造成开发困难。

## 基本定义和术语

|  |  |
| --- | --- |
| 硬件设备服务化 | 将硬件调用看成是一种服务。由服务提供者提供服务。 |
| 打印指令 | 程序与打印机间的沟通格式。不同类型的打印机(如热敏,针式),具有不同的格式。 |

## 相关文档人员

|  |  |
| --- | --- |
| 库开发者 | 开发库的人员 |
| 库使用人员 | 使用库的人员 |

# 设计思路

## 上述问题的解决方法

### 打印指令没有完全解耦

使用一套类似于HTML排版格式系统的伪语言来描述。客户端只要记住标签即可。

### 调用封装的不够细

封装页面格式。即：把排版和打印作为一个接口提供给客户端。即：提供的API需要能够描述格式

### 难于调试

提供调试接口。详尽的记录日志。使得调用者能够知道问题在哪。

# 软件架构描述

## 系统组成部分

下图是系统调用的总的一个序列图。其中涉及了五部分组件。

## 总体设计原则

### 组件间解耦

# 需求描述

## 打印服务器

### 规划

1 对调用者隐藏打印机细节

2 提供各种基本格式服务(如表格)

3 提供调试服务

### 需求场景描述

#### 打印服务调用

主执行者：客户端

范围：

层次：用户

主成功场景：

1 传入需要打印的内容

1.a 如果打印机正在打印,返回忙的信息。

2 解析字符串(case 4.1.2.2)

2.a 格式错误,返回给客户端

3 计算格式排版(case 4.1.2.3)

3.a 计算排版格式错误,返回给客户端

4 生成伪指令

5 调用打印引擎进行调用

5.a 打印错误,返回

6 打印完毕,返回客户端信息

#### 解析字符串

主执行者：打印引擎

范围：

层次：系统

主成功场景：

1 获取需要打印的字符串

1.a 如果不符合xml格式,则返回格式错误信息

1.b 如果有不符合xml格式的tag,返回tag不支持的tag信息(具体的tag,请参见第五章)

1.c 如果有不符合的tag属性,则仅在后台打印出警告信息,并不抛错。

2 分解成数据结构

#### 计算格式排版

主执行者：打印引擎

范围：

层次：系统

主成功场景：

1 获得了数据结构

2 根据数据结构计算所占的范围

3 调整范围

#### 生成打印伪指令

设备服务程序所能提供的服务,参见附录：硬件服务。

主执行者：程序

范围：

层次：用户

主成功场景：

1 获取要调用的设备

1.a 检查此设备上有无job.如果有,则返回错误。

1.b 检查此设备是否正在执行。如果在执行,则返回错误。

2 调用此设备进行服务,同时阻塞住客户端。

2.a 调用设备错误,返回给客户端

3 获得硬件输入,或者超时后,返回信息。

#### 开始JOB

主执行者：程序

范围：

层次：用户

主成功场景：

1接收到客户端提出的申请Job指令

2 检查所需要锁定的硬件

2.a 如果硬件空闲,则新建jobid,返回

2.b 如果硬件忙,则返回JOB错误信息

#### 结束JOB

主执行者：程序

范围：

层次：用户

主成功场景：

1接收到客户端提出的退出Job指令,获得jobid。

1.a 没有jobid,返回报文错误。

2 解除对资源的锁定

# 附录

## 打印例子描述

### 例子

<?xml version='1.0' encoding='utf-8'?>

<Root>

<Setting>

<Page>

<Width>100mm</ Pagewidth>

<Height>60mm</ Pageheight>

<Leftmargin>5mm</ Leftmargin >

<Rightmargin>60mm</ Rightmargin >

<Topmargin>60mm</ Topmargin >

<Bottommargin>60mm</ Bottommargin >

<Cpi>10</Cpi>

<Lpi>6</Lpi>

</Page>

</Setting>

<Content>

//绝对定位的例子。离纸张左边距3毫米,上边距2毫米。打印12345678

//最大宽度为10毫米。若12345678超出了10毫米,则自动进行换行处理

// Align:居中对齐

<Rect Pos=”Absolute” Top=”2mm” Left=”3mm” Width=”10mm” Align=”Middle”>

12345678

</Rect>

//相对定位的例子。离纸张左边距3毫米,上边距2毫米。离上一个元素下浮10毫米距离。打印12345678

//最大宽度为10毫米。12345678超出了10毫米,则自动进行换行处理

// Align:居中对齐

<Rect Pos=”Relative” Top=”+10mm” Left=”3mm” Width=”10mm” Align=”Middle”>

12345678

</Rect>

//Table:表示打印表格

//Border:打印表格线.1表示表格格式打印

<Table Border=”1”>

<Tr Width=”60”>

< Td Width=”20”><Bold>1111</Bold></Td >

< Td Width=”40”>2222</Td>

</Tr>

<Tr Width=”60”>

<Td Width=”20”>3333</Td>

< Td Width=”40”>4444</Td>

</Tr>

</ Table>

//表格2：第一和第二列合并

<Table Border=”1”>

<Tr Width=”60”>

< Td Width=”20” Colspan=”2”><Br>1111</Br></Td >

< Td Width=”40”>2222</Td>

</Tr>

<Tr Width=”60”>

<Td Width=”20”>3333</Td>

< Td Width=”40”>4444</Td>

</Tr>

</ Table>

//表格3：第一行和第二行合并

<Table Border=”1”>

<Tr Width=”60”>

< Td Width=”20” Rowspan=”2”><Br>1111</Br></Td >

< Td Width=”40”>2222</Td>

</Tr>

<Tr Width=”60”>

< Td Width=”40”>4444</Td>

</Tr>

</ Table>

<Br>

</Content>

</Root>

### 例子解释

长度：因为毫米到英尺的换算过程中会出现小数。应此规定：只保留小数点后四位。

1inch = 25.4 mm

1mm = 0.03938inch

长度,坐标的单位：均以标准ASCII码字符长度表示。1个ASCII标准字符长度=1/10inch=0.24cm。

单位：

1 毫米：mm

定位方式：分为两种定位方式。一种是绝对定位,另一种是相对定位。

绝对定位：绝对定位下,以打印纸的Leftmargin,Topmargin为起始坐标(0,0)。

相对定位：相对于当前打印头所在位置的定位。注：为了在各种型号的打印机间保持兼容。不支持打印头回退定位(即：在此种定位方式下,left和top属性不能出现负数)。

Page:表示纸张的设置。

Width:纸张的宽度。

Height: 纸张的高度。

Leftmargin:纸张的左边界。

Rightmargin: 纸张的右边界。

Topmargin: 纸张的上边界。

Bottommargin: 纸张的下边界。

Lineseperate:行距。

Content:表示需要打印的内容。

打印格式控制：

Br:表示略过一行

Turnpage:表示手动设置翻页。

Text:表示输出文字

Table:表示定义一个表格。

Border=”1” //表示标准格式打印表格线

Tr:定义行

属性：定义width。

Td:定义列

属性：RowSpan:表示跨行

属性：ColSpan:表示跨列

属性定义:定义各种需要打印内容的修饰信息。

Pos:表示定位的方式。Absolute:绝对定位。Relative:相对定位

Left:左边坐标。

Top:上边坐标。

Width:定义宽度。

Height:定义高度。

Align:表示对齐方式。

### 初始值

Cpi:character per inch。每英尺字符数。默认是10cpi.

Lpi：line per inch。初始值为6

## 标签描述

### 打印设置标签

#### Page

定义打印纸。

1 Bottommargin:底部结束位置

2 Topmargin:顶部开始区域。注：不能小于6.35mm

3 Leftmargin:左边开始区域。

4 Rightmargin:右边结束区域。

5 Width:介质介质宽度。

6 Height:定义介质高度。

7 LineInterval:定义行距。

8 ColInterval: 定义列距。

### 格式标签

#### Underline

表示加下划线。

#### Bold

黑体

#### Doubleheight

倍宽

#### Doublewidth

倍高

#### Tripleheight

三倍高

#### Triplewidth

三倍宽

#### Sup

上标

#### Sub

下标

### 结构标签

#### Br

表示打印空行

#### Turnpage

跳过当前纸张,从下一页开始打印

#### Rect

定义了矩形结构元素

|  |
| --- |
|  |

属性：

Width:表示矩形宽(如果不设置,默认为纸张宽度)

Height:表示矩形高(如果不设置,默认为纸张高度)

Left:表示矩形的起始左边点位(如果不设置,默认为纸张左边距)

Top:表示了矩形的起始上边点位(如果不设置,默认为上一个元素的下一个元素)。

Align:表示对齐方式。Middle:居中。Left:左对齐。Right:右对齐

Pos:表示定位方式。Absolute:绝对定位。Relative:相对定位

#### Table

定义了表格元素

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

属性:

Border:表示打印Table的表格线。

Width:定义表格的整个宽度。

Height:定义表格的整个高度。

Top:定义上边起始部分。

Left:定义左边起始部分。

Pos:表示定位方式。Absolute:绝对定位。Relative:相对定位

#### Tr

定义Table中的一行。

#### Td

定义table中的一列

属性：

Width:表示宽度。

Align:表示对齐方式。Middle:居中。Left:左对齐。Right:右对齐

Colspan:跨列

Rowspan:跨行

1 文件自带MD5校验。

2 子目录。

ADSM:

# 概要设计

#### 格式化XML文件

目的：读取配置文件,然后形成程序内部的数据结构。

大致思路：鉴于1 xml是通用格式 2 不需要兼容老格式.因此本部分不做接口处理,仅作XML验证。

#### 计算排版格式

目的：根据给定的各种标签,计算所占的空间面积,然后排列成符合打印机的数据结构。

大致思路：标签的计算具有通用性,可以建立一个元接口进行抽象计算。生成的数据结构也具有通用性,(是否可以适用于不同的打印机)也可以考虑下。

WORD中的主要属性：

1 每一行有段前和段后属性。而且是以标准行的倍数(包括分数)来的。倍数可以自己自定义。

2 每一行具有左缩进和右缩进属性。且是以标准字符的倍数(包括分数)来的。倍数可以自定义。

3 表格的高度：每一行具有固定的高度。这个纯粹是以厘米来表示的。

Word的计算：

|  |  |
| --- | --- |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 7 | 7 |
| 8 | 8 |
| 9 | 9 |
|  |  |

1

2

3

4

5

6

7

8

9

#### 生成伪指令

目的：根据排版格式,生成适合于打印机的指令。

大致思路：根据上一级形成的数据结构,形成一系列打印的指令。

#### 类的属性组织

文档中一个节点🡨🡪程序中的一个类。

每个节点对应一个类。为了使每个节点都有自己的属性,就要使得每个类有各自的属性。这边主要要注意的是：字节点的属性覆盖父节点的属性。

对于一个table来说,可能有几个属性。（可以参照上面的对比）

1 如果是一个单元格内,其表现与平常的换行差不多。

2 如果不是在一个单元格内,则还需要加上一点距离。还需要加上下划线的距离。