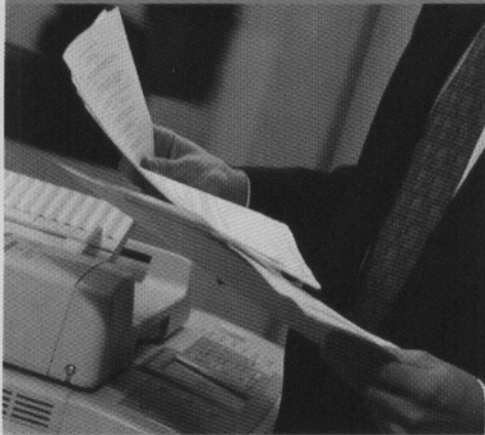


封玮 邱秀芹 (烟台大学计算机学院  
264005)



# Java 平台的打印 功能研究

## The Research on Printing in Java

**摘要:** 本文重点论述了 Java(tm) 2 Platform Standard Edition (J2SE(tm)) 在打印服务方面的最新进展和 JPS (Java Print Service) API 的特色、包结构和功能等。通过实例展示了 JPS (Java Print Service) API 的先进技术和使用方法, 并探讨了 Java 1.4 平台打印服务应用的相关问题。

**关键词:** JPS (Java Print Service) API Java API 打印服务 IPP JSR

### 1 引言

无论是本地应用还是网络应用, 打印服务都是各类计算机用户最为需要, 也是各种平台必须提供的一项基本的服务。毫无例外地, Java 环境也应该且必须满足这种需求。但是, 由于历史原因, Java 平台提供的打印服务功能一直都比较弱。实际上, 在最初的 JDK 1.0 版本中根本没有提供任何支持打印服务的功能, 直到 1997 年 JDK 1.1 问世, Java 才有了最基本的打印支持。由于其提供的能力有限, 在有关 Java 技术的应用中, 一些较复杂的打印功能大都通过其它途径来实现, 如: 调用 ActiveX/OCX 控件或者 VB/VC 程序……, 十分繁琐、不便。

可喜的是, 近年来, 随着 SUN 公司对于 Java 打印功能的不断完善, 现在, 我们已经可以实现许多相当复杂的打印服务应用需求了。本文在回顾 Java 平台打印服务的进化演变过程的基础上, 介绍了最新的 JPS (Java Print Service) API 的技术特色、主要贡献、包结构和常用类与接口的功能等, 探讨了相关的服务及应用问题。

### 2 Java 打印服务发展过程

(1) JDK1.1。1997年, JDK1.1 为用户提供初步的、基本的打印支持 (java.awt.PrintJob 类, 面向 AWT 组件, 只能打印图形的 API 1.1);

(2) JDK1.2。1998年, JDK1.2 提供了 java.awt.print 包, 增强了 API 1.1 的能力, 使得 Java 应用程序可以打印各种 Java 2D 图形、文

本和图像 (java.awt.print.PrinterJob 类, 面向 Java 2D(tm) 技术的 API, 本质上仍是页面映象的 API);

(3) JDK1.3。2000年, JDK1.3 在 java.awt.print 包中增加了 JobAttributes 和 PageAttributes 类, 此时, Java 客户端应用程序可以规定打印任务的特性和打印页面的属性;

(4) IPP/1.1。2000年9月, Xerox 公司提交了 IPP/1.1 (Internet Printing Protocol), 为分布式打印提出了一个新的工业标准协议;

(5) JSR006。2001年11月, 由 Xerox 公司领导的四方专家组 (IBM, JustSystem Corporation, Sun Microsystems, Inc. 和 Xerox) 提交了 JSR006 (Java Specification Request), 从而

- ① 统一了 Jini 和 Java 技术的打印服务 API;
- ② 制定了打印服务 API 的规范;
- ③ 促使了打印服务 API 与 Java 2D API 的紧密结合。

(6) JDK1.4。2002年, 新问世的 JDK1.4 增加了 javax.print 包, 它不再是另一个图形打印 API, 而是在跨平台的基础上, 为用户带来了一套完整的、全新的“Java 打印服务 API”——JPS (Java Print Service) API, 可以满足各种文档格式的打印需求。

现在, Java 环境真正提供并实现了一整套丰富的打印服务功能的支持, 具体体现为:

- ① 有为用户提供打印服务的标准 GUI (Graphics User Interface) 组件;

② 有创建和分派打印任务的API(Application Programming Interfaces);

③ 有建立并驱动第三方提供的打印服务的接口SPI(Service Provider Interfaces)。

所以, JPS (Java Print Service) API的时代到来了。

### 3 JPS (Java Print Service) API

应用程序与打印系统交互的目的: 以打印机为目标, 最终产生出一个打印的页面。如图1所示, 一个完整的打印系统由“应用程序”层、“打印机发现”层、“打印机映象”层、“打印机驱动”层和“打印机”层五部分组成。介于应用程序与打印机之间的“打印机发现”层、“打印机映象”层和“打印机驱动”层将由打印系统的API来提供。因此, Java平台打印系统服务功能的强弱、好坏都将取决于打印系统API的性能。而JPS (Java Print Service) API以比原有的打印模型大大增强的特性, 改善了由于历史原因造成的Java平台打印服务功能较弱的状况。

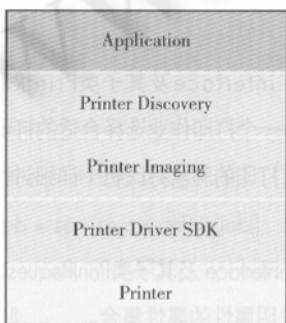


图1 打印系统

#### 3.1 JPS (Java Print Service) API 的特色

应用程序代码可以完全控制整个打印过程;

不需要通过用户对话框来选择非缺省的打印机及选项, 这一点有利于服务器端的打印服务;

为客户端应用程序增强了用户对话框的功能(应用程序可以检查、重载用户对话框的选择);

提供了对加入第三方打印服务的支持, 它对已有的打印功能是积极的补充。

#### 3.2 JPS (Java Print Service) API 的贡献

JPS (Java Print Service) API统一并拓展了Java平台上已有的打印服务功能, 满足了客户端与服务器端双方对于打印服务方面的各种需求。

另外, 在支持已有的Java 2D打印特性的基础上, JPS (Java Print

Service) API进行了许多改进, 包括:

(1) 客户端和服务器端双方均可以根据自己的能力来发现和选择打印机并可规定打印任务的特性。

(2) 作为first-class objects, JPS (Java Print Service) API中包含了标准IPP (Internet Printing Protocol) 属性的实现。

(3) 应用程序可以扩展包含在IPP (Internet Printing Protocol) 中的属性。

(4) 通过SPI(Service Provider Interfaces), 第三方可以加入他们自己的打印服务。

#### 3.3 JPS (Java Print Service) 的体系结构

JPS (Java Print Service) API由四个包组成: javax.print及其子包 javax.print.attribute, javax.print.attribute.standard和javax.print.event。

javax.print包

javax.print包是JPS API的一个核心包, 主要有发现打印服务、规定打印数据格式、根据打印服务建立打印作业以及将打印作业送至打印机或流等功能。

javax.print.attribute和javax.print.attribute.standard包

javax.print.attribute和javax.print.attribute.standard包用以定义相关的打印属性, 描述打印服务的能力、文档的特征、处理文档或整个打印作业的指令或描述打印作业/打印机的状态等。

javax.print.attribute包描述了属性的类型以及如何将需要的属性加入到打印请求的属性集合中。其中, Attribute接口是所有属性的父接口。javax.print.attribute包定义了五种不同类型的属性, 每一种用以描述打印过程中的一部分能力。例如, PrintRequestAttribute接口定义了客户用来描述打印作业特征的属性(如: 打印份数)。

javax.print.attribute.standard包列举了所有由API支持的标准属性, 它们中的大多数是指定在IPP (Internet Printing Protocol) 规范中实现的属性。

javax.print.event包

javax.print.event包给用户提供了为打印专业及打印服务注册事件监听类和接口。

#### 3.4 JPS (Java Print Service) API 中常用的类与接口

(1) PrintServiceLookup: class, 用以查找、发现打印服务;

(2) PrintService: interface, 用以返回打印服务对象;

(3) Attribute: interface, 用以描述打印服务、说明打印请求;

(4) DocFlavor: class, 用以描述打印数据的格式, 如JPEG或PostScript……;

(5) DocPrintJob: interface, 用以打印提交的打印作业;

(6) PrintJobEvent: class, 用以监听打印作业和打印服务事件;

## 4 提供打印服务的标准GUI(Graphics User Interface) 组件

GUI(Graphics User Interface)是应用程序与用户交互的“窗口”，而GUI ( Graphics User Interface ) 组件是图形用户界面标准化的结果。Java平台的为用户提供打印服务的标准GUI (Graphics User Interface) 组件，以直观的形式呈现了打印服务可以提供哪些功能以及应用程序能够进行哪些控制等方面的选择，由此提供了一个良好的交互界面。可以方便、简化用户请求打印服务时的相关操作。

与以往的Printable的对话框不同的是，在jdk1.4提供ServiceUI的打印机对话框的缺省行为已经用新的API更改了：缺省情况下对话框不显示。我们必须使用ServiceUI类调用printDialog方法创建如下图2所示的打印对话框。

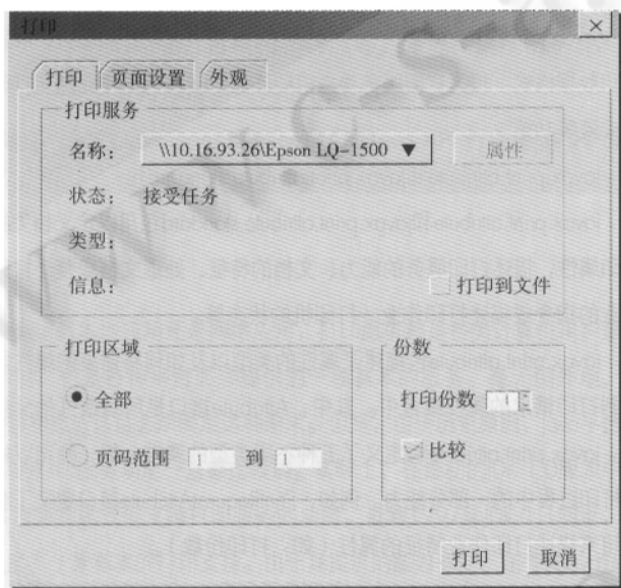


图2 ServiceUI 打印机对话框

## 5 应用JPS (Java Print Service) API实现打印作业

### 5.1 应用要求与解题步骤

应用要求：

实现打印一个HTML格式的文档。

解题步骤：

- 步骤1：描述打印的数据；
- 步骤2：规定如何打印数据；
- 步骤3：查找合适的打印服务；
- 步骤4：建立打印作业并指定打印数据；
- 步骤5：提交打印作业。

### 5.2 JPS (Java Print Service) API 应用举例

#### 5.2.1 描述打印的数据

使用DocFlavor class。

格式：DocFlavor.<获取数据的Java技术表达形式>.<打印数据的类型>

JSP API 有预定义的DocFlavor 实例，如：

DocFlavor.BYTE\_ARRAY.POSTSCRIPT, Byte Array形式的POSTSCRIPT文档；

DocFlavor.INPUT\_STREAM.TEXT\_PLAIN\_UTF8, Input Stream形式的文本文档；

DocFlavor.INPUT\_STREAM.TEXT\_HTML\_US\_ASCII, Input Stream形式的HTML文档；

.....

根据应用要求，解题步骤1——从一个URL获得HTML的内容，数据为ASCII码，描述如下：

DocFlavor htmlFlavor =DocFlavor.URL.TEXT\_HTML\_US\_ASCII;

#### 5.2.2 说明如何打印数据

指定打印服务的属性，如：纸型（A4、B5...）、单（双）面打印、份数.....。

使用Attribute interface及其子类PrintRequestAttribute和PrintServiceAttribute为一个打印作业选择合适的打印服务。例如：选在三楼的那台大型双面打印的彩色打印机（详细讨论见查找合适的打印服务部分）。

使用AttributeSet interface 及其子类PrintRequestAttributeSet在一个打印服务请求中详细说明属性的属性集合。

#### (1) PrintRequestAttribute的应用

\* MediaSizeName class, 指定纸型名（如：ISO\_A4,NA\_LETTER, JIS\_B5.....）；

\* Copies class, 指定打印文档份数，由客户创建实例对象（如：new Copies(5)）

\* Sides class, 单 / 双面打印（ONE\_SIDED, DUPLEX, TUMBLE）

#### (2) PrintRequestAttributeSet的应用

JPS API中有实现了该接口的子类，主要作用是用于将定义好的属性值加入到打印请求的属性集合中。HashPrintRequestAttributeSet就是其中之一，使用如下：

```
import javax.print.attribute.*;
PrintRequestAttributeSet aset = new HashPrintRequestAttributeSet();
.....
```

根据应用要求，解题步骤2——说明如何打印数据，描述如下：

```
import javax.print.attribute.*;
import javax.print.attribute.standard.*;
```

```
PrintRequestAttributeSet aset = new HashPrintRequestAttributeSet();
aset.add(MediaSizeName.ISO_A4);
aset.add(Sides.TWO_SIDED_LONG_EDGE);
aset.add(new Copies(2));
```

### 5.2.3 查找适合的打印服务

使用PrintService interface描述打印服务，使用PrintServiceLookup class查找打印服务。

根据应用要求，解题步骤3——查找适合的打印服务，描述如下：

```
PrintService []matchingServices = PrintServiceLookup.
lookupPrintServices( aset, htmlFlavor);
if (matchingServices.length == 0) { return; }
// 选取一项服务
PrintService service = matchingServices[0];
```

### 5.2.4 建立打印作业并指定打印数据

使用PrintService interface获得打印作业，而打印数据则由Doc interface提供。

根据应用要求，解题步骤4——建立打印作业并指定打印数据，描述如下：

```
import java.net.*;
import javax.print.*;
URL url = < ... the print data source ...>;
DocPrintJob pjob = service.createPrintJob();
SimpleDoc doc = new SimpleDoc(url, htmlFlavor, null);
```

### 5.2.5 提交打印作业

使用 javax.print 包中DocPrinterJob interface的方法之一：

print ( Doc doc, PrintRequestAttributeSet attributes )，负责提交打印作业，完成打印任务。该方法可能抛出PrintException，所以，应进行

例外处理。

根据应用要求，解题步骤5——提交打印作业，描述如下：

```
try {
pjob.print(doc, aset);
} catch (PrintException e) { }
```

## 6 结语

尽管Java 1.4平台已经提供并实现了一整套丰富的打印服务功能，满足了应用程序对打印服务的种种需求，但随着新技术的不断涌现和发展，对于不断增加新的API、提高并增强API所实现的性能与质量的追求，已成为Java技术研究者和应用者共同的目标。而IPP中所规定的基于工业标准的PS的属性的可扩展性，使得追求成为可行。我们有理由相信：JPS (Java Print Service) API将更加充实、完备，Java平台的打印服务功能将更加强大、易用。

## 参考文献

- 1 Sun Microsystems, Inc. Java™ Print Service API Specification, 2001年。
- 2 Sun Microsystems, Inc. Java(tm) Platform Printing: The Java Print Service API, 2000。
- 3 机械工业出版社,(Java2核心技术 卷II: 高级特性), 2000年11月。
- 4 文枫, Java打印程序设计, IBM:developerWorks中国网站, 2002年10月。

