Extending OBEX Object Push to create a Minimal Basic Printing Profile Client.

Revision 1.0
25th September, 2002

Revision History

Revision	Date	Comments
0.0	1 May, 2002	First draft
0.5	8 May 2002	Changes to Table 4, plus minor editorial changes
0.6	1 June 2002	Added contributor, changed title
0.7	10 July 2002	Clarification of FilePush vs Simple Push Transfer
1.0	25 September 2002	Changed Version # to 1.0

Contributors

Jerry Thrasher	Lexmark
Melinda Grant	HP
Alan Berkema	HP
John Waters	HP

Contents

1 Introduction

This paper describes, in general, the differences between the OBEX Object Push Profile (OPP) push transfer operation and the simple FilePush operation used by the 'Simple Push Transfer' printing model in the Basic Printing Profile (BPP). It also outlines changes that are needed to a Client device that currently supports the Object Push Profile to be able to support the BPP simple FilePush operation.

On the surface the Basic Printing Profile can seem to be a very complex implementation of a printing solution. In reality, the complexity of BPP is shouldered mainly by the BPP Printer. ***There are very few differences between the mandatory features of a BPP Sender (client) and the mandatory features of an OPP Client.***

2 Process Overview

The following is a list of operations that are performed to transfer data from Client to Server for both OPP and BPP.

1 Device Discovery (Inquiry)

2 SDP Query for the desired service (OPP or BPP)

3 SDP Query for Protocol Descriptor List attributes of interest (e.g. data formats that the server supports).

4 Establish L2CAP ACL to the discovered RFCOMM server channel.

5 Establish an OBEX Connection

6 Use OBEX Put operations to "push" the data

7 Close the OBEX Connection

8 Disconnect the L2CAP ACL Connection

3 SDP Device Discovery (Inquiry)

Since a BPP Printer is required to support Public Online mode, the process for discovering the device is the same as if trying to discover a public device in General Discoverable mode in OPP. The only difference is that the Service and Device Class bits/values will be different.

Class Bits	Object Push Server	BPP Server(Printer)
Major Service Class	Object Transfer	Rendering
Major Device Class	Phone or Computer (varies)	Imaging
Minor Device Class	(varies)	Printer

Table 1: Service and Device Class comparison

4 SDP Query for Service Discovery

After device discovery, the desired service must be discovered. The process is the same for both OPP and BPP. The differences are in the UUIDs that are used for the two profiles.

UUID Name	Object Push Server	BPP Server(Printer)
Service Class ID	0x1105	0x1118 (Direct Printing Service)
Profile ID	0x1105	0x1122
Protocol ID's	Same ID's for both OPP and BPP (L2CAP, RFCOMM, OBEX)	

Table 2: UUID Comparison

5 SDP Query for Attribute Discovery

After it is determined that the desired service is located on the discovered device, and the service record handle is returned, the Client may desire to discover certain attributes about the device or service that are contained in the Service Record to assist in device selection.

The BPP Basic Printing Service Record can contain enough information about a BPP Printer for a BPP Sender (or user) to make a very detailed decision about the suitability of the printer for the printing task. For example, if the user has a color print job that needs to be printed on photo paper it may be desirable to make sure the printer can support these requirements during device selection. While these features are desirable, their support is optional for a BPP Sender.

While the BPP Service Record contains many more attributes than the OPP Service Record, the critical attribute definition (Data Format) for simple FilePush printing is actually shared by both BPP and OPP (though formatted differently).

BPP Attribute Name	Attribute ID	Attribute Format and Description
Document Formats Supported	0x0350	String that describes the data formats and version information for data that can be understood by the BPP Printer.
Character Repertoires Supported	0x0352	Bit field that indicates the types of characters that can be printed. (US-ASCII is guaranteed)

Table 3: Critical Attributes to discover for a Minimal BPP Sender

5.1 Document Formats Supported

Because there are many types and versions of printer languages and data formats, it is important for the Sender to only send data that can be rendered by the Printer. In the OPP case, a mistake results in a file or object that can't be viewed or used on the OPP Server and is subsequently discarded. For a BPP Printer, a mistake could result in many useless and unwanted pages being printed.

For BPP, the Document Formats Supported (0x0350) attribute is included in the Service Record. This attribute is a string of bytes formatted such that the data format (MIME media-type) and any associated version number can be discovered. A String was chosen instead of a byte sequence to make this attribute future-proof such that new printer languages and versions could be dynamically added and discovered without changing the profile or affecting interoperability. (I.e., from a client's perspective, parse between the commas for the MIME media-types you support, ignore the things you don't. Client may have to add support for queries with continuation-state information since this string could be long.)

For OPP, the attribute for discovering data (object) formats is called Supported Formats List (0x0303). This attribute is a byte sequence with currently six types of objects defined.

5.1.1 Required Data Formats for BPP

All BPP Printers support the XHTML-Print v0.95 specification, which includes support for baseline JPEG images. Other formats such as vCard and vCal are optional.

Here is an example of a minimal XHTML-Print job that could be used to print plain text. The tags in *italicized blue* could be used as a template for inserting any

text to be printed. The title does not actually get printed, however, it is considered good practice to include one.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//PWG//DTD XHTML-Print
1.0//EN"
"http://www.xhtml-print.org/xhtml-print/xhtml-
print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title> My SMS Message from K&amp;amp;R </title>
</head>
<body>
<p>
Hello World!
</p>
</body>
</html>
```

While the example above illustrates the minimum requirement, a sending device could improve the appearance of the output by adding any desired tags or style described in the XHTML-Print specification.

XHTML-Print can be viewed as a universal Page Description Language that eliminates the need for printer-specific device drivers. The interoperability provided by this model is one of the primary reasons to use BPP in the mobile environment. UPnP also requires the use of XHTML-Print and this should further proliferate the availability of XHTML-Print content and user-agents.

Given the nature of XHTML and CSS, it is reasonable to expect a BPP printer to do an acceptable job with other XHTML family documents; for instance, XHTML Basic documents will print correctly, and content is generally preserved across all XHTML variants. (The same is not true for HTML documents, however.)

5.2 Character Repertoires Supported

The other attribute that is critical for device selection and sending data to a BPP Printer is the attribute that indicates what character glyphs can be printed. (E.g., can the Printer print Greek or Japanese characters correctly or just US-ASCII?) This attribute is a 128-bit field with each bit indicating a separate supported character repertoire. Currently only the first 18 bits have been assigned. XHTML-Print content is UTF-8 encoded.

Again, for OPP, the downside of transferring unusable data is having to delete an unreadable file. For a BPP Printer, that data may be rendered (or attempted anyway) to paper.

6 L2CAP ACL and RFCOMM Connections

Since both BPP and OPP are based on the Generic Object Exchange Profile, discovery of channel numbers as well as the process for establishing the L2CAP ACL connection and the RFCOMM server channel connection are the same for both profiles.

7 Establishing an OBEX Connection

In the simplest case, establishing the OBEX connection is basically the same for both OPP and BPP (i.e., if the Target Header is not used to connect to a specific OBEX service UUID in BPP, and if OBEX authentication is not enabled on the BPP Printer.)

The BPP recommends the use of the Target Header (with 128-bit UUIDs) and the return of a Connection ID Header when establishing any OBEX connection; however, connections without a Target Header in the OBEX Connect operation are connected to the Direct Printing (DPS_UUID) service by default in a BPP Printer.

The following table shows the OBEX Header usage or support requirements of the OPP and BPP (taken from the individual profiles). 'M' indicates that the use of the header is mandatory; 'O' that use is optional; 'X' that the header is not used; and 'C' indicates that under certain conditions, the header is mandatory. The asterisk, when used, indicates which item the comment refers to.

Header no.	OBEX Headers	OPP Client	BPP Sender	Comments
1	Count	X	O	Not used in BPP (but not forbidden)
2	Name	M*	M	It's actually forbidden for OPP push operations. (should be conditional)
3	Type	O*	M	It's actually required for OPP push operations.
4	Length	M	M	
5	Time	O	O	
6	Description	X	O*	Its contents are optional for the BPP Sender; however, the Printer may provide status information in this header that can improve usability.

7	Target	X	C*	Not required for simple FilePush in BPP.
8	HTTP	O	O	
9	Body	M	M	
10	End of Body	M	M	
11	Who	X	O*	Not used if no Target Header in OBEX Connect.
12	Connection ID	X	C	Conditional for BPP (i.e., used if Target Header is used in Connect)
13	Authenticate Challenge	X	X	
14	Authenticate Response	X	M*	Note 1.
15	Application Parameters	X	O	Not used for simple FilePush in BPP
16	Object Class	X	X	

Table 4: OBEX Header Usage in OPP and BPP

Note 1. While support for OBEX authentication is required in a BPP Sender, it's recommended that a BPP Printer that initiates OBEX authentication by default provide a way to disable it to support BPP Senders that don't have a user interface. Implementers should be aware that printers may issue an OBEX authentication challenge, and that interoperability may not be possible if the client cannot respond to the challenge. Although this should from a practical point of view be a small concern, it may present a real issue in terms of qualification.

8 Transferring data over OBEX

After an OBEX Connect operation (no Target Header) to a BPP Printer is performed, the "pushing" of data from BPP Sender to the BPP Printer is the same as "pushing" data from OPP Client to OPP Server. Both use a Type Header in the OBEX Put operation to transfer the MIME Media-Type of the object being sent. BPP does allow the Name Header to be used to transfer the name of the file, however this is optional. If a Target Header is used in the OBEX Connection, the Connection ID is also part of the OBEX Put operation.

Client Request:	**Bytes**	**Meaning**

Opcode	0x82 0xnnnn 0x42 0x001C *image/jpeg* 0x01 0x0015 0069 006D 0067 0035 002E 006A 0070 0067 0000 0x48 0x0xxx 0x……	**PUT**, Final bit set Length of packet HI for **Type** header Length of **Type** header MIME Media-Type of object, null terminated US-ASCII text HI for **Name** header (optional) Length of **Name** header. **Name** header content “img5.jpg” (UTF-16 encoded), null terminated. Hi for Object **Body** header Length of **Body** Body object bytes of img5.jpg
Server response:		
Response code	0xA0 0x0003	SUCCESS, Final bit set Length of response packet

Table 5: BPP Simple FilePush Data Transfer Example

9 Closing the OBEX Connection

If the OBEX Connect was previously initiated without the Target Header (i.e., no Connection ID returned), then closing the OBEX connection is the same for both OPP and BPP via a generic OBEX Disconnect operation. A Connection ID is required in the Disconnect otherwise.

10 Disconnecting the L2CAP ACL Connection

The process for closing the ACL Connection is the same for both OPP and BPP.