



Samba and Printing

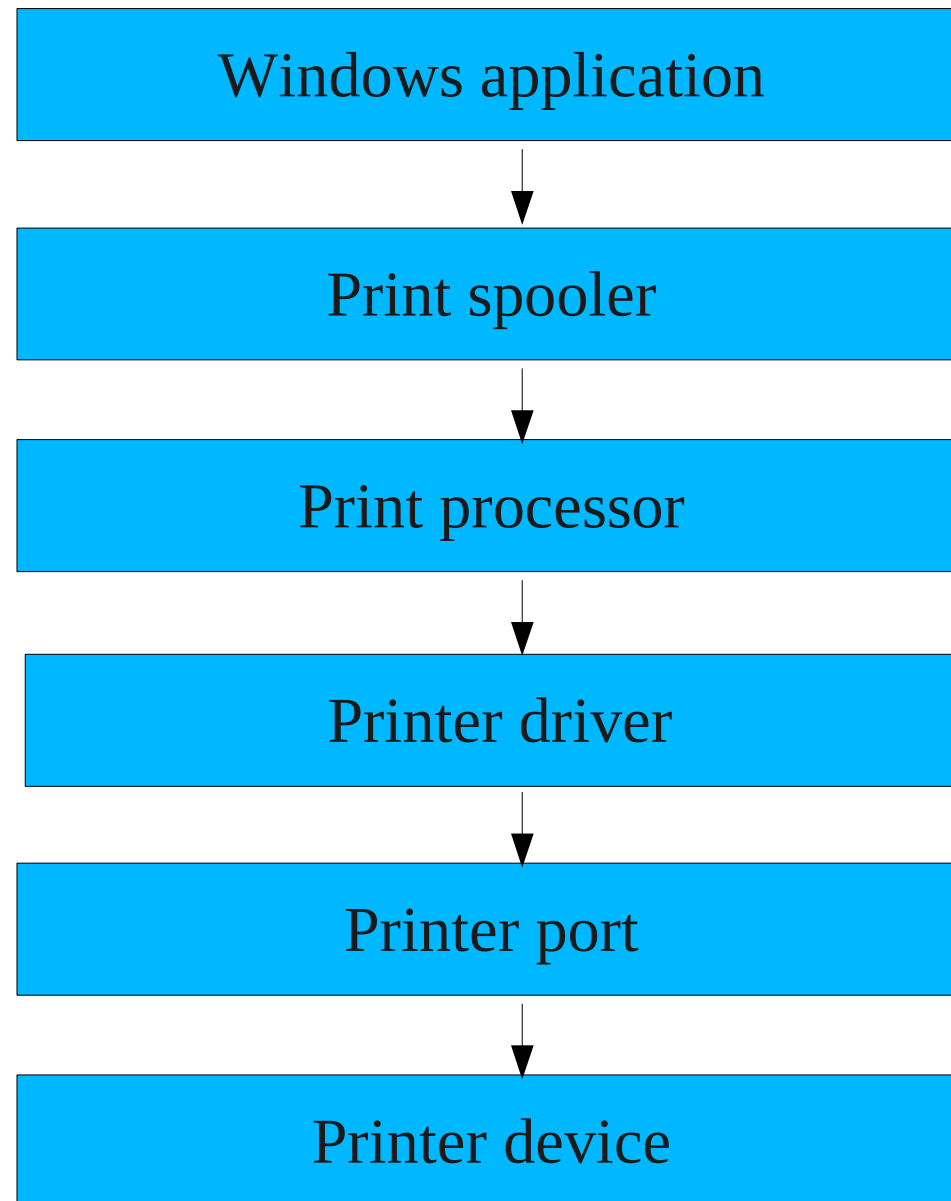
Günther Deschner
<gd@samba.org>

(Red Hat / Samba Team)

What is printing ?

=> Printing output from a Windows application to a printer device

What is involved in printing ?



Implementation difficulties

- **Fundamental assumptions make it difficult to map Windows printing to Unix printing (Graphics Device Interface - GDI)**
- **Server side processing of printjobs**
- **Windows clients can forward raw GDI to a print server and let the print server generate output for a printer**
- **Printer Drivers are native Windows binary DLLs (Unix cannot execute these)**
- **Printing architecture more complex than Unix printing architecture**

Printing support in current Samba versions

Samba 3 as a Print Server

- **Printing support is one of the oldest features of samba**
- **Samba only supports RAW printing mode**
 - **printjobs are rendered on the client**
 - **Server only queues jobs and sends them to printing device**
- **Samba implements Windows 2000 model of printing (mostly done by Gerald (Jerry) Carter)**
- **Few, critical fixes for Vista and other modern Windows versions**
- **Printing support was working (mostly) but was not in the best shape to suite Vista and Windows 7**
- **Some windows drivers only partially work and some not at all**

Printing implementation in Samba < 3.4

- **Full implementation**
 - **Printers**
 - **Forms**
 - **Drivers**
- **Configuration displayed via registry interface**
- **Hand-marshalled parsing**
- **Tested in production**

Printing implementation in Samba 4

- **Skeleton implementation for some calls**
- **Cannot print anything**
- **No registry storage of configuration**
- **Autogenerated parsing**
- **“pull” tested with smbtorure**
- **Basically Samba 4 lacks printing support completely**

Printing implementation in Samba > 3.4

- Core spooler code remained untouched (mostly)
- Beginning with Samba 3.2.0 all hand marshalled RPC protocols have been replaced with autogenerated code
- Autogenerating parsers for a hand marshalled protocol
- Samba 3.4.0 finally had spoolss autogenerated
- First version got still parsing problems

Why is it so hard to make printing work ?

spoolss – a really bizarre protocol

- **Very irregular and very odd protocol**
- **Network signature is quite verbose:**
 - **GetPrinter request**
client asks for size
 - **GetPrinter reply**
server replies that client needs to allocate 1024 byte buffer
 - **GetPrinter request**
client sends 1024 bytes of 0 to server (!)
 - **GetPrinter reply**
server fills in 1024 byte buffer

spoolss – a pidl challenge

- spoolss in Windows is partly IDL based/autogenerated and partly not
- Sophisticated code to support spoolss code generation with pidl (Stefan Metzmacher <metze@samba.org>)
- spoolss.idl was completed before documentation got available

spoolss – pidl 64bit client support

- Windows 64bit clients did not see any printer but Wireshark and Samba logfiles did
- Famous endnote #28 in MS-RPRN:

“<28> Section 2.2.2: Windows XP 64-Bit Edition, Windows Server 2003 64-bit Edition, Windows Vista 64-bit Edition, Windows Server 2008 64-bit Edition, Windows 7 64-bit Edition, and Windows Server 2008 R2 64-bit Edition do not correctly handle custom-marshaled INFO structures where the unused space is not between the end of the last Fixed_Portion block and the beginning of the first Variable_Data field.”

(<http://msdn.microsoft.com/en-us/library/cc244870%28PROT.13%29.aspx#id28>)

spoolss – pidl 64bit client support

- Buffer 1024 bytes with relative pointers in it:
 - [offset1] [offset2] [offset3] [ptr1] [ptr2] [ptr3] fillup with 0
 - [offset1] [offset2] [offset3] zeros... [ptr3] [ptr2] [ptr1]
- Added pidl extension for relative pointers in reverse order
 - LIBNDR_FLAG_RELATIVE_REVERSE
- Fixed in Samba 3.4.8 and 3.5.0

**Why is my driver xyz not
working ?**

Printer and driver settings

- **Windows executes driver to generate devicemode**
- **PrinterData and DeviceMode is stored on the server**
- **Unix cannot execute driver to implement same behaviour**
- **Samba needs to rely on the client to set it up all correctly**
- **Samba currently cannot store all of this information properly (incomplete registry for printer/driver settings)**

How does Samba address stabilization of printing support ?

Testing printing - smbtorure

- Since 3.4.0 a lot of effort has been put into testing
- Samba4 smbtorure:
 - RPC-SPOOLSS
 - RPC-SPOOLSS-WIN
 - RPC-SPOOLSS-NOTIFY
 - RPC-SPOOLSS-PRINTER
 - RPC-SPOOLSS-DRIVER
 - RPC-WINREG
- Attempting to test every possible combination of data for printing settings (PrinterData)

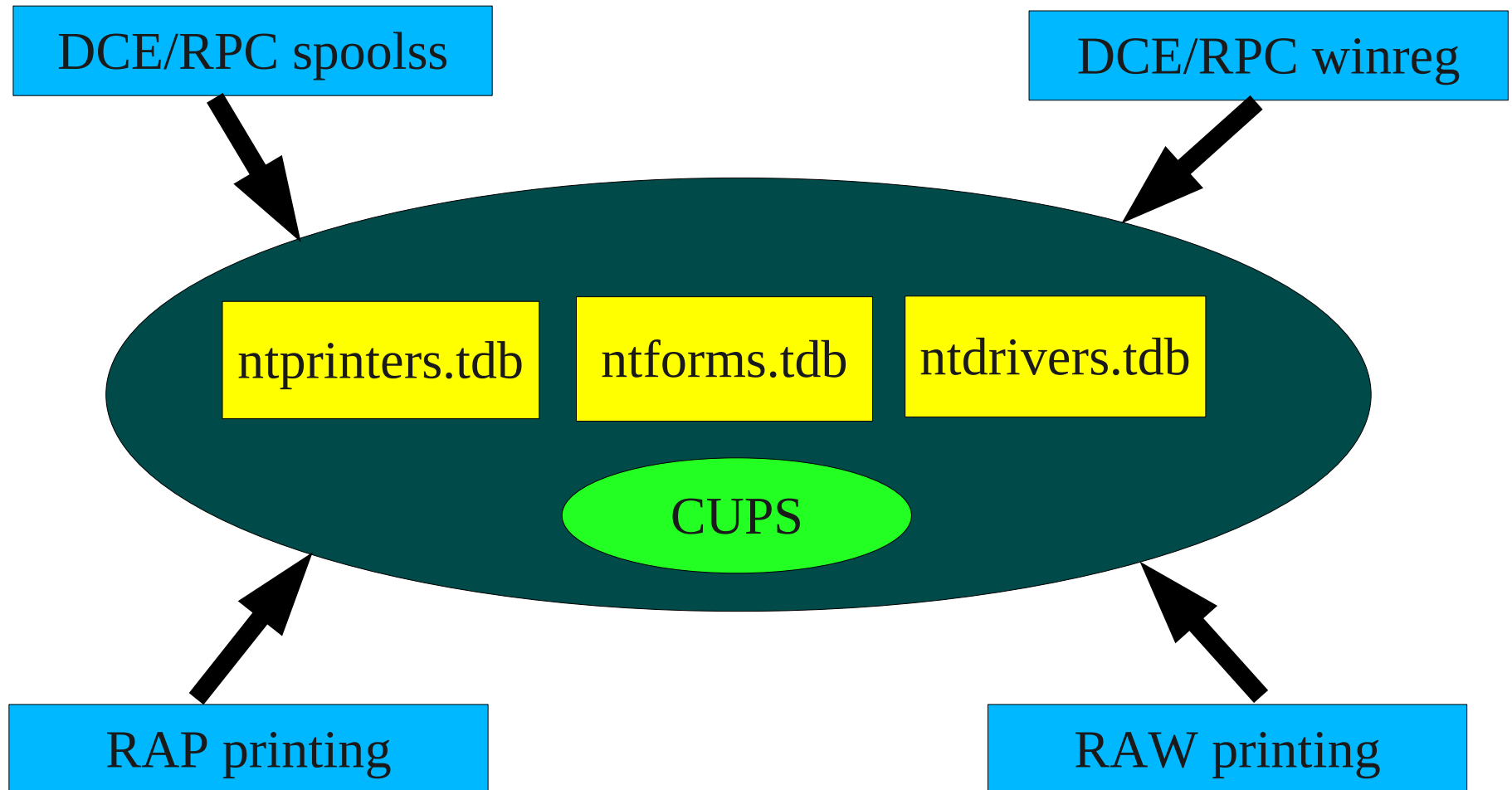
Testing printing - win32

- Only testing with smb torture is **not** sufficient
- Example: spoolss.idl bug in EnumPrinterKey()
- Native printing client code test suite run on Windows
- spoolss.exe in samba tree:
testprogs/win32/spoolss/spoolss.c
- Soon: native win32 printer driver tests on the buildfarm

The samba spooler architecture

- `\\pipe\spoolss` Spoolss DCE/RPC server
- `\\pipe\winreg` Registry DCE/RPC server
- `\\pipe\lanman` RAP server
- raw smbd printing
- `[print$]` share for drivers

Current samba spooler architecture



registry – DCE/RPC winreg pipe

- **Various Microsoft printing tools operate directly on registry:**
 - **printmig.exe**
 - **brm.exe**
 - **etc.**
- **Samba loads content from printing tdb's and displays content dynamically**

Remote Administration Protocol

- What is RAP ?
- Who uses RAP still ?
- In Samba:
 - RAP directly accesses printing tdb's
 - RAP indirectly accesses printing backend (cups)

Implemented RAP calls

- **DosPrintQEnum, DosPrintQGetInfo**
- **WprintQueuePause, WprintQueueResume, WprintQueuePurge**
- **WprintJobEnumerate, WprintJobGetInfo**
- **RDosPrintJobDel, RDosPrintJobPause, RDosPrintJobResume**
- **WPrintDestEnum, WPrintDestGetInfo**
- **PrintJobInfo**
- **WprintDriverEnum, WprintQProcEnum, WPrintPortEnum**

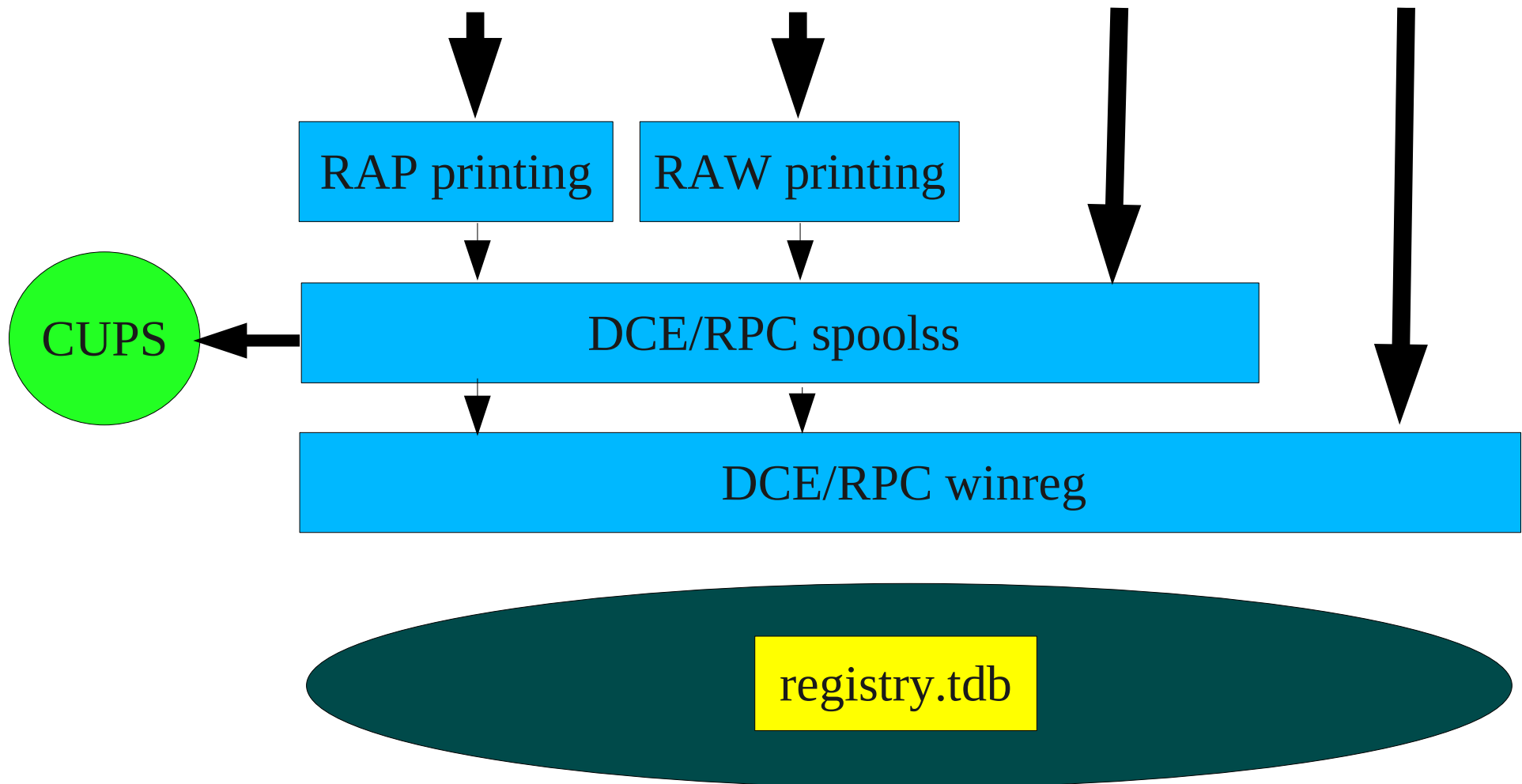
Printing over raw SMB

- **SMBopen**
- **SMBwrite**
- **SMBclose**
- **SMBreply**
- **etc.**

Rearranging samba spooler architecture

- Achieve very clear layering of protocols
- Move all printing related code into the spooler service
- Eventually allow to move spooler into an own binary/daemon
- Next logical step in the ongoing code cleaning initiative
- Mostly done by Andreas Schneider <asn@samba.org> and Simo Sorce <idra@samba.org>
- Test driven development from the early beginning

Future samba spooler architecture



Printing roadmap

- **Re-architecting will be finished for next Samba main release**
- **Patches are constantly pushed (once reviewed) to master**
- **Work on better performance**
- **Even more testing**
- **Create Migration tool**
 - **Convert ntprinters, ntdrivers and ntforms tdb to registry.tdb**
- **Make it possible for Samba4 to use Samba3 print spooler**
- **We need to contact dochelp:
Existing protocol documentation is not sufficient to create a
working spoolss implementation**

Changes in Windows printing

- **IRemoteWinspool**
 - asynchronous printing protocol (MS-PAR)
 - ncacn_ip_tcp
 - Re-uses most of spoolss structures and calls (slight modifications)
- **AD domains often enforce server side spooling via Group Policy**
- **New Windows clients aggressively cache printing traffic**

Further reading

■ Microsoft Protocol Documentation:

- **Print System Remote Protocol Specification ([MS-RPRN].pdf)**
<http://msdn.microsoft.com/en-us/library/cc244528%28PROT.13%29.aspx>
- **Print System Asynchronous Remote Protocol Specification ([MS-PAR].pdf)**
<http://msdn.microsoft.com/en-us/library/cc238080%28PROT.13%29.aspx>
- **Remote Administration Protocol ([MS-RAP].pdf)**
<http://msdn.microsoft.com/en-us/library/cc240190%28PROT.10%29.aspx>

■ Samba Wiki:

- <http://wiki.samba.org/index.php/Spoolss>
-

Thank you for your attention!