

Connecting the Symbian OS Emulator to the Internet using Ethernet

Revision 1.4, July 2004

Authors: Nao Kagabu, Nishaat Rajabali, Hamish Willee, Panos Asproulis

1 Introduction

This paper shows how to set up the Symbian OS Emulator to connect to the Internet using the host PC's Ethernet card. If you work with applications/middleware software requiring Internet connectivity, you should consider this method (a comparison of the alternative solutions is given in section 2).

Support for Ethernet-based connectivity has been in the platform since Symbian OS v7.0, but was never widely publicised; mainly because the driver and Communications database (CommDb) configuration was more difficult than other connection mechanisms. The zip package supplied with this paper delivers tools to simplify the Ethernet setup, including improved drivers, a Perl script to automate the driver configuration, and reference CommDb configuration files.

2 Comparison of Symbian OS Emulator Connection Methods

The available Emulator connectivity solutions and their relative advantages/disadvantages are listed below. In summary, the configuration of the Ethernet drivers themselves is trivial; however configuration of the CommDb *may* be more problematic, depending on your expertise with Symbian's networking sub-system.

The Ethernet solution is to be considered if you are a laptop user, work in an organisation (probably large) with DHCP servers already set up to hand out IP/DNS addresses, or know a bit about Internet configuration. If you don't fit into any of these categories, then the WinSock plug-in solution is recommended for Symbian OS v7.0 (UIQ), and mRouter for other platforms.

Connection Method	Description	Advantages/Disadvantages
NTRAS	Windows NTRAS routes PPP packets from Symbian OS to the LAN/Internet. Symbian OS is configured to output PPP packets to a serial port; this is connected (using a NULL modem cable) to the serial port on the host PC that is set up for NTRAS. In the Emulator case both serial ports are on the same machine (loopback).	<p>Advantages:</p> <ul style="list-style-type: none"> Symbian OS configuration is static and well understood Symbian OS Configuration is same on both Emulator and Target <p>Disadvantages</p> <ul style="list-style-type: none"> NTRAS configuration is complicated and prone to error. NTRAS must be configured and run for every PC that the Emulator is used on. NTRAS configuration cannot be automated NTRAS configuration changes with every revision of MS Windows. Emulator setup requires a null-modem loopback cable, so not suitable for using with laptops (only 1 serial port).
M-Router	M-Router routes PPP packets from Symbian OS to the LAN/Internet. This solution is similar to NTRAS, except	<p>Advantages</p> <ul style="list-style-type: none"> Symbian OS configuration is the same as for NTRAS; it is static, well understood, and the same on both

	that mRouter does packet forwarding.	<p>Emulator and Target</p> <ul style="list-style-type: none"> • Unlike for NTRAS, laptop use is possible on DevKits as tools are supplied to enable internal loopback i.e. requires no NULL-Modem cable • No additional configuration required; The standard installation of mRouter (as used for connectivity on real devices) does not need further configuration in order to connect to the internet • mRouter functions as a NAT server; hence, requires no DHCP interaction <p>Disadvantages</p> <ul style="list-style-type: none"> • Requires mRouter to be installed on Host PC, and for every PC it is used on. • Requires mRouterDeveloper.exe to be installed and run in order to support loopback • As a NAT server, mRouter makes the Emulator invisible to the local network; hence, packet pushing to the Emulator is NOT possible.
WinTunnel	<p>Remote server on network routes packets from all Emulators on network. This requires modification to the Emulator's serial driver so that it sends and receives from a Win32 TCP socket rather than a Win32 serial port. Note that this solution still uses the Symbian OS TCP/IP stack, PPP nif, C32, and the logical serial driver.</p> <p>This solution is not currently "supported" by Symbian; hence it is not made generally available.</p>	<p>Advantages:</p> <ul style="list-style-type: none"> • Emulator configuration same as for NTRAS; it is static, well understood • Single setup of server for routing on network (rather than need for setup of mRouter/NTRAS on every computer requiring internet access) • Works on laptops • Solution under Symbian control; no dependency on other products (mRouter/NTRAS) <p>Disadvantages:</p> <ul style="list-style-type: none"> • Setup of server on network is beyond abilities of most users.
WinSock Plug-in	<p>Winsock Plug-in (Symbian OS v6.1 and UIQ SDK – Symbian OS v7.0) allows the Emulator to make use of the host PC's networking connection by forcing the Emulator to use the Windows networking layer instead of the Symbian OS networking layer.</p>	<p>Advantages:</p> <ul style="list-style-type: none"> • Extremely simple setup <p>Disadvantages:</p> <ul style="list-style-type: none"> • Does not use Symbian OS TCP/IP networking stack. Any issues with your usage of the stack will not be revealed when working with this solution. • This solution should not be used if there are any dependencies on Symbian's Networking stack e.g. when measuring throughput, latencies etc.
Ethernet driver (described in this document)	<p>Symbian OS shares host PC Ethernet driver, and appears to the local LAN as another machine in the network.</p>	<p>Advantages:</p> <ul style="list-style-type: none"> • Solution under Symbian control; no dependency on other commercial products (mRouter/NTRAS), only on the open-source product WinPcap. • If local LAN supports DHCP supplied IP and DNS addresses, this solution works

		out-of-the-box. Disadvantages: <ul style="list-style-type: none"> • Ethernet and CommDb setup is a “black art” to many users. May need support of network administration to set up correct parameters for CommDb if DHCP not enabled.
--	--	---

3 Architectural Overview

The Symbian OS Emulator shares the host PC’s Ethernet Network Interface Card (NIC) with the PC’s operating system. This is achieved by using the publicly available WinPcap API, which interfaces directly with the NIC via Windows OS NDIS (Network Driver Interface Specification) Driver; refer to <http://netgroup-serv.polito.it/winpcap> for more information.

Symbian OS calls WinPcap APIs through EtherWins.pdd; the Symbian OS Ethernet device driver used on Windows with the emulator. The relationships between the different components are shown in Figure 1.

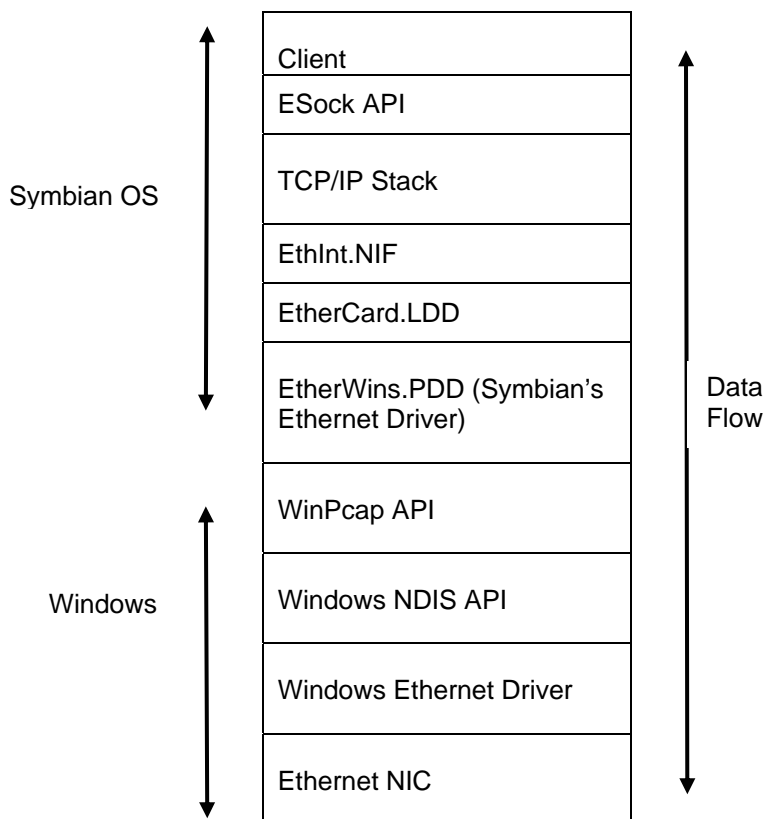


Figure 1. Data flow over the EtherWins PDD

There are two main issues with this architecture:

1. The first is that the Symbian OS Emulator is not really a unique machine on the Ethernet LAN. As a result, it does not have the locally unique MAC address that it needs in order to receive Ethernet frames. The solution is to generate a locally unique MAC address, which the EtherWins driver gets from a file on the emulated "C" drive:

```
c:\system\data\ethermac.dat
```

2. The second issue is that the driver needs to know which of the Host PC's Network adaptors it should use. This information is supplied to the driver in the file (on the emulator C drive):

```
c:\system\data\etherdriver.dat
```

The configuration file (configchange.pl) accompanying this paper automatically generates both the MAC address and Network adapter .dat files for you. If there is more than one possible Network adapter on your PC, you will be prompted to select which one to use.

The configuration file can optionally back up the current CommDb and load a CommDb that has been pre-configured for Ethernet connections.

4 Requirements

This paper assumes that the user is set up with the following environment:

4.1 Host PC environment

- Windows NT/2000
- LAN access
- A fully functioning Ethernet NIC (with a globally scoped MAC address).
- WinPcap version 3.0 (note that other versions may not work)

4.2 Supported Symbian OS platforms

The installation packages have been tested for these development environments at the time this document was published:

- UIQ 2.1 SDK
- Symbian OS v7.0.17 DevKit
- Nokia Series 60 v2.1 Beta SDK
- Symbian OS v7.0s v200422 DevKit

Future releases of the Symbian OS v8.0 DevKit will include the latest versions of the tools supplied with this whitepaper. Therefore even though the tools supplied with this paper may work, you should use versions accompanying the DevKit.

4.3 Contents of the installation package

The installation package contains the latest drivers, configuration tool, and CommDb configuration files for each of the target platforms.

File list and description is given below:

```
\epoc32\tools\configchange.pl : The CommDb/MAC address/ configuration tool
\epoc32\tools\netcards.exe    : Tool used by configchange.pl to get the network adaptor list
\epoc32\release\wins\udeb\etherwins.pdd : Wins version of driver
\epoc32\release\wins\urel\etherwins.pdd : Wins version of driver
\epoc32\release\wincsw\udeb\etherwins.pdd : Wincsw version of driver
\epoc32\release\wincsw\urel\etherwins.pdd : Wincsw version of driver
```

```

\epoc32\wins\c\7.0sDevKit_ethernetced.cfg : Template CommDb for Symbian OS v7.0s DevKit
\epoc32\wincsw\c\7.0sDevKit_ethernetced.cfg : Template CommDb for Symbian OS v7.0s DevKit
\epoc32\wins\c\series60_2.1_ethernetced.cfg : Template CommDb for Series 60 SDK
\epoc32\wincsw\c\series60_2.1_ethernetced.cfg : Template CommDb for Series 60 SDK
\epoc32\wins\c\v7.0_ethernetced.cfg : Template CommDb for Symbian OS v7.0 (both DevKit and UIQ SDK)
\epoc32\wincsw\c\v7.0_ethernetced.cfg : Template CommDb for Symbian OS v7.0 (both DevKit and UIQ
SDK)

```

It also contains a CommDb Wizard which can be used with the Symbian CommDb Editor Version 3.1 or higher (it can be downloaded from <http://www.symbian.com/developer/downloads/tools.html#cdb>) in order to automate the process of configuring the CommDB settings.

5 Setting up Ethernet Connection

The following installation process explains how to set up the Ethernet connection.

In summary, the process is:

- Download and install the WinPcap library
- Unzip the package over the target SDK/DevKit
- Ensure the SDK is set as the default device, and that EPOCROOT is correct
- Configure the CommDb for Ethernet connectivity.
- Run the configchange.pl tool
- Test the connection

The following sections provide detail on each of the steps. Section 8 has “Troubleshooting” information, should you have any problems.

5.1 Install WinPcap library

Symbian’s Ethernet driver requires the WinPcap library to send and receive Ethernet frames. WinPcap is the Windows ported version of the ‘libPcap’ library used in Unix/Linux. To install WinPcap please run the installation file available at <http://winpcap.polito.it/install/>. Please note, only WinPcap v3.0 has been tested.

Disclaimer: Symbian gives NO guarantees to the state and usability of this version of WinPcap, and WILL NOT be liable for any damage and/or loss caused during/after the driver is installed.

Note: It is important that only one copy of WinPcap is installed – use the Control Panel Add / Remove Programs to uninstall any existing copies of the software before installing the new one.

5.2 Enable the correct Device/EPOCROOT

The driver and configuration tool require that the correct Symbian OS “Device” and EPOCROOT are set for your target SDK/Devkit. Information about how to set up the Symbian OS Device is provided in the Symbian Developer Library at: [» Developer Library » Tools and Utilities » Kit management](#)

5.2.1 How to set the correct device

Run the following on the command line to get the list of devices:

```

C: \>devi ces
Series60_2_1b: com. Noki a. Seri es60_2_1b
UIQ_21: com. symbi an. UIQ - defaul t
02335: com. symbi an. TechVie w

```

In my case the UIQ SDK is set as the default device, you can change this to Series 60 (for example) by using:

```
C:\>devices -setdefault @Series60_2_1b: com. Nokia. Series60_2_1b
```

5.2.2 How to set EPOCROOT

The devices command *should* set the EPOCROOT to match your current device. However if it does not, this is easy to do manually:

First get the root using the devices “-info” flag

```
C:\>devices -info @Series60_2_1b: com. Nokia. Series60_2_1b
Device: Series60_2_1b: com. Nokia. Series60_2_1b
Root is D:\Symbian\7.0s\Series60_v21b\
Tools path is D:\Symbian\7.0s\Series60_v21b\
```

Then set the EPOCROOT. Note that the drive is not specified:

```
set EPOCROOT=\Symbian\7.0s\Series60_v21b\
```

5.3 Unzip the installation package

Unzip the package over your SDK/DevKit installation.

The package should copy over your SDK at the at the EPOCROOT (this is the folder location which contains the binary tree \epoc32\). Carrying on the example from the previous section, unzip to D:\Symbian\7.0s\Series60_v21b\.

5.4 Set up the CommDb

5.4.1 Networks with DHCP

\epoc32\release\[wins | wincw]\c\ contains template CommDb configuration files for each of the tested SDKs detailed in 4.2. Rename the configuration file for your specific SDK/DevKit to ethernetcd.cfg. Note that the configuration file for Symbian OS v7.0 is suitable for both the DevKit and the UIQ SDK. Alternatively, you can use the Symbian CommDB Editor and the Ethernet Connection Wizard in order to configure the CommDb settings.

The configuration files connect the emulator to a network supporting DHCP-supplied IP and DNS addresses, where no proxy server is specified. No editing of these files is necessary, and you can proceed to section 5.5.

5.4.2 Networks with static IP addresses (no DHCP)

You must place your network’s IP settings into the CommDb. You can edit your current CommDb configuration if you wish, but we recommend editing one of the template CommDb configuration files provided with this whitepaper. They are stored in \epoc32\release\[wins | wincw]\c\. Rename the file appropriate for your specific SDK/DevKit to ethernetcd.cfg.

Now edit the configuration file as described in section 7, or use the Symbian CommDB Editor and the Ethernet Connection Wizard in order to generate an appropriate CommDb configuration file. After this, proceed to section 5.5.

5.5 Run the Ethernet driver configuration tool

In order to configure the driver *and* the default CommDb, on a wins/udeb platform, enter the following on the command line:

```
configchange.pl --config ethernetWithCommDB --kernel EKA1 --target wins --variant UDEB
```

If there is more than one Network adapter on your machine the tool will prompt you to select which one to use; If there is only one, then the tool will simply write this to the emulated C drive:

```
c:\system\data\etherdriver.dat
```

The tool will automatically create a “locally unique” MAC address based on the MAC address of your PC and write it to:

```
c:\system\data\ethermac.dat
```

Notes:

- To use the winscw Emulator for Codewarrior, replace with `--target winscw`, and for UREL version, replace the `--variant` with UREL. The `--config` options of `ethernetWithCommDB` and `ethernetNoCommDB` enable and disable the use of the template CommDb configuration file.
- To restore the CommDb, replaced by using `ethernetWithCommDB` in the original invocation, run `configchange.pl` with the `--config restoreCommDB` parameter, e.g.:

```
configchange.pl --config restoreCommDB --kernel EKA1 --target wins --variant UDEB
```
- Some DHCP servers are set up to only allow specific computers on the network. If this is the case on your network, then you should ask your network support team to add the MAC address in `ethermac.dat` to the list of allowed computers.

6 Testing the Ethernet connection

This section shows how to verify the Emulator’s Internet connection. Two applications may be used:

1. TechView IAP status application
2. HTTP Example Client application

6.1 Test using the IAP Status Application

The IAP status application tests IP connections; effectively whether packets can be sent to a specified IP address. This is therefore a good “sanity check” of your CommDb configuration. However it does not test usage of DNS, and so is not as complete a test as the HTTP example client.

In addition, the IAP Status application is available only on the Techview Emulator; therefore this test cannot be used for testing the UIQ and Nokia Series 60 SDKs.

To run this test:

1. Start the TechView Emulator
2. Start the ‘IAP Status’ application from the Extras menu.
3. Select the appropriate IAP on the IAP list and press ‘Start’
If the IAP status changes from ‘Inactive’ to ‘Active’ then there is a valid Internet connection.
4. Select the IAP and press ‘Stop’
Confirm that the IAP status changes back to ‘Inactive’.

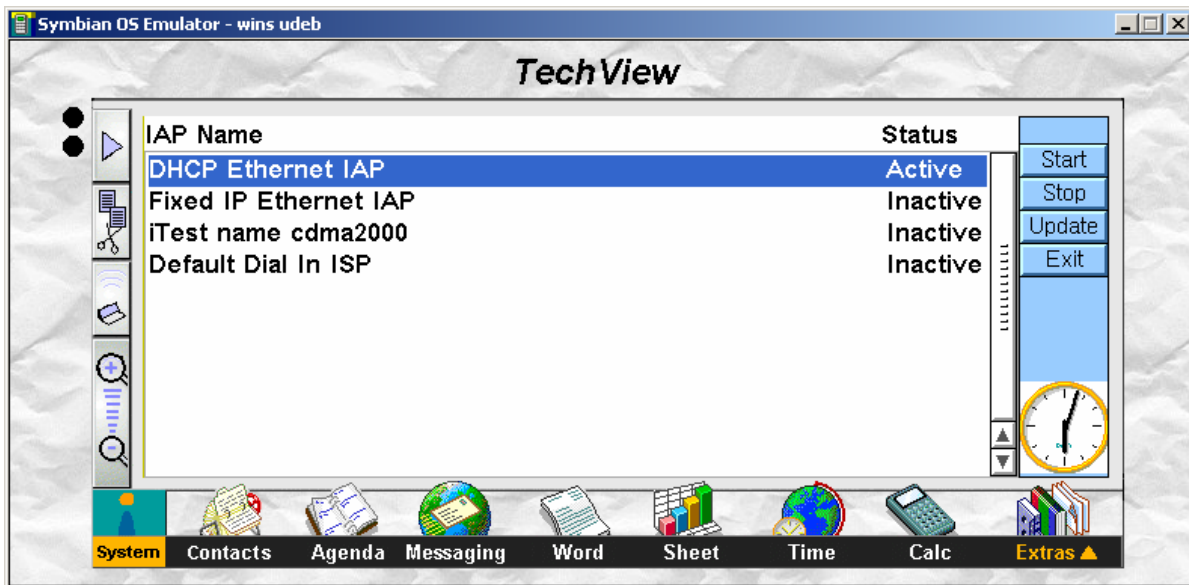


Figure 2. IAP Status Application

6.2 Test using the HTTP Example Client

The HTTP Example Client is the best available test for Internet connectivity on Symbian OS. It allows you to download a web-page and save it to a file.

The httpexampleclient.exe can be found on all DevKits (and on the UIQ SDK) at:

```
...\\epoc32\\release\\[wins|wincsw]\\[udeb|urel]\\httpexampleclient.exe
```

On the Nokia Series 60 SDK, the example client is not pre-built. However the source is available for you to build at:

```
\\Symbian\\7.0s\\Series60_v21b\\Examples\\appropts\\exampleclient
```

To run the test:

1. Run httpexampleclient.exe from Windows explorer or the command line
2. Enter "1" to make an HTTP GET request
3. Enter a target URL i.e. <http://www.symbian.com> and press return



Figure 3. HTTP Example Client GET request

The program may now prompt you to select an IAP (unless the communications database is set to select the IAP automatically). After an IAP is selected the program will attempt to retrieve the web page.

If the request succeeds your Internet connection is good, and you will be prompted to retrieve the page:

4. Select 'Y' to acknowledge the request to "Save the selected pay to disk?"
5. Type a file name to save the retrieved content. E.g. c:/Symbian.html
This will be saved to the Symbian OS emulated "C:" drive, i.e. /epoc32/[wi ns|wi nsw]/c
6. When the transaction completes successfully, type any key to return.
7. Check if the HTML file is fully retrieved by opening the downloaded file with a web browser.

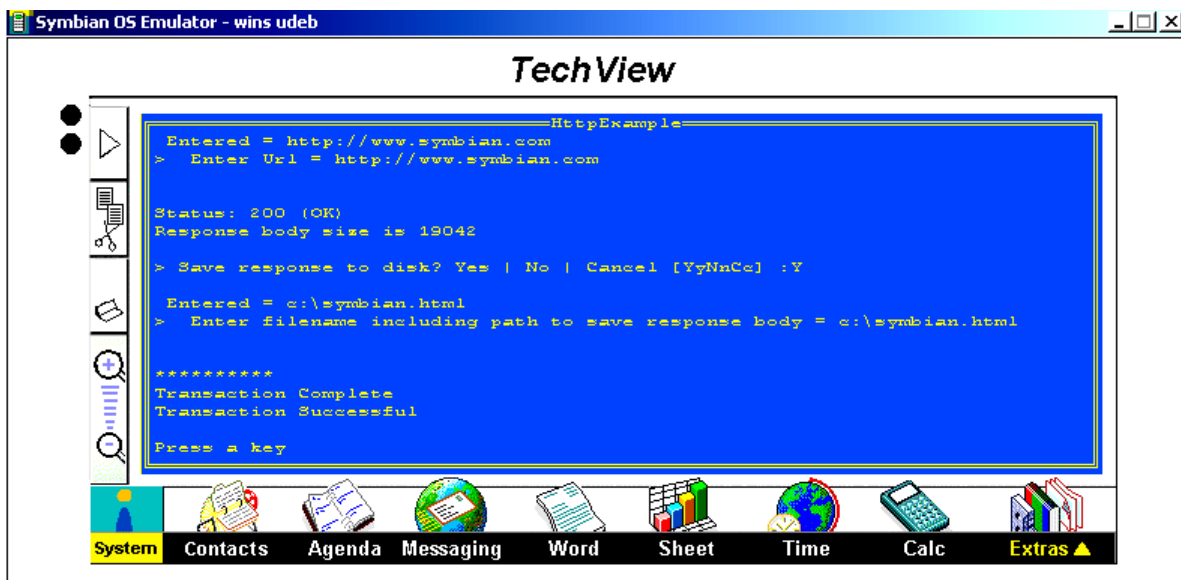


Figure 4. Completed HTTP Example GET request

Troubleshooting: The above response indicates the case where the connection works. For information on how to troubleshoot an unsuccessful connection, see section 8.

7 Configuring CommDb for Ethernet

Symbian OS stores communications-related settings in an internal database called *CommDb*. The database is not edited directly, but through intermediary text files that are exported and imported using the CED and CEDDUMP tools, respectively [R1].

The tables that may need to be either changed or added to the CommDb, and their relationships, are shown in Figure 5.

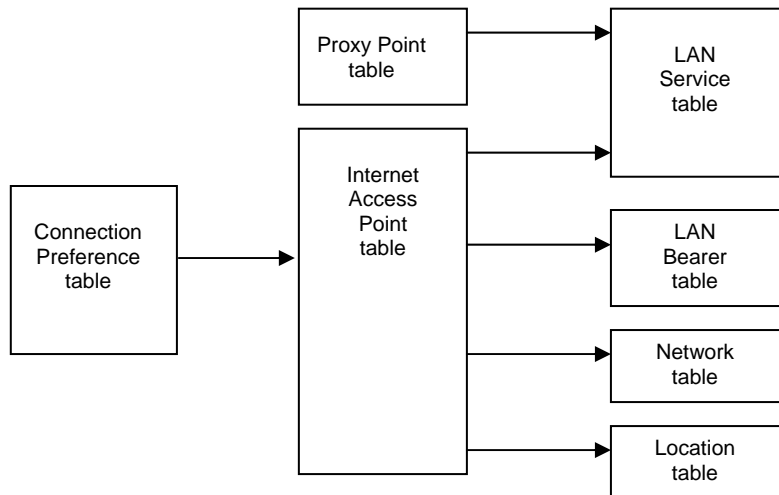


Figure 5. CommDb tables affecting Ethernet setup

The manual configuration of each table is discussed below. If the Symbian CommDB Editor is used for configuring the CommDB tables then the following sections can be skipped.

7.1.1 The LANBearer table

The LANBearer table contains information about the physical connection to the network. Field definitions are shown below.

Field	Description
Name	Arbitrary name for this bearer
Agent	Name of the Agent. For Ethernet, set to nullagt.agt, the default PSD agent used for packet/Ethernet connections.
IfName	The name of the NIF to use. For Ethernet, is always ethint (.nif)
LDDName	Set to "not used" in Symbian OS v7.0, v7.0s
PDDName	Set to "not used" in Symbian OS v7.0, v7.0s
LastSocketActivityTimeout	Socket inactivity timeout. Set to 9999 or -1 to disable.
LastSessionClosedTimeout	Session close timeout. Set to 9999 or -1 to disable.
LastSocketClosedTimeout	Socket close timeout. Set to 9999 or -1 to disable.

To set up the LANBearer, simply add the following section into the CED configuration file.

```
[LANBearer]
```

```

ADD_SECTION
  Name= LAN_bearer_1
  Agent=nullagt.agt
  IfName=ethint
  LDDName=not used
  PDDName=not used
  LastSocketActivityTimeout=-1
  LastSessionClosedTimeout=-1
  LastSocketClosedTimeout=-1
  FIELD_COUNT=8
END_ADD

```

7.1.2 The LANService table

This table contains information about the network to which you are going to connect. The fields are explained in the table below:

Field	Description
Name	Arbitrary name for this LANService
IfNetworks	Supported IP types. Possible values 'ip', 'ip6' or 'ip, ip6', indicating whether IPv4 packets, IPv6 packets, or both types are supported.
IpNetMask	IP subnet mask
IpGateway	Gateway IP address
IpAddrFromServer	TRUE if IP address is obtained from DHCP server, FALSE otherwise.
IpAddr	IP address to use for the Emulator.
IpDNSAddrFromServer	TRUE if address of DNS server is obtained from DHCP server, FALSE otherwise.
IpNameServer1, IpNameServer2	Primary and secondary DNS server addresses

The LANService can be configured to support IP addresses dynamically using DHCP, or to use static IP addresses that you supply. Your network administrator will tell you whether your network supports DHCP, or if not, what range of IP addresses you should use.

7.1.2.1 Dynamically allocated IP Addresses

For networks that support DHCP server allocated IP addresses dynamically, you can simply add a service entry as shown below:

```

[LANService]
ADD_SECTION
  Name=Lan_service_1
  IfNetworks=ip
  IpNetMask=0.0.0.0
  IpGateway=0.0.0.0
  IpAddrFromServer=TRUE
  IpAddr=0.0.0.0
  IpDNSAddrFromServer=TRUE
  IpNameServer1=0.0.0.0
  IpNameServer2=0.0.0.0
  FIELD_COUNT=9
END_ADD

```

The values of "IpAddrFromServer" and "IpDNSAddrFromServer" fields are set to TRUE. The values of the other fields are ignored.

7.1.2.2 Statically allocated IP Addresses

If the values of “IpAddrFromServer” and “IpDNSAddrFromServer” fields are set to FALSE, then Symbian OS will use the (static) values in the other fields. Your network administrator (or Internet supplier) will be able to tell you your gateway address, IP address, netmask, and DNS server addresses.

The section below shows a typical configuration:

```
[LANService]
ADD_SECTION
  Name= Lan_service_1
  IfNetworks=ip
  IpNetMask=255.255.255.0
  IpGateway=10.1.2.254
  IpAddrFromServer=FALSE
  IpAddr=10.1.2.1
  IpDNSAddrFromServer=FALSE
  IpNameServer1=10.1.2.64
  IpNameServer2=10.1.2.96
  FIELD_COUNT=9
END_ADD
```

7.1.3 The Internet Access Point (IAP)

The IAP is the name given to a connection in Symbian OS; it links the LANService and LANBearer tables.

The IAP entry will look almost exactly as below. The values of the IAPService and IAPBearer fields are the respective positions of your Ethernet LANService and LANBearer entries in their respective tables. In this example, the new entries are the first entry in their tables. Note that the xml version of the CED configuration file [R1] supports named rather than number ordered entries.

Note the name of the IAP. When you connect with the emulator, you may be asked to choose an IAP by name.

```
[IAP]
ADD_SECTION
  Name=Ethernet Intranet
  IAPService=1
  IAPServiceType=LANService
  IAPBearer=1
  IAPBearerType=LANBearer
  IAPNetwork=1
  IAPNetworkWeighting=0
  Location=2
  FIELD_COUNT=8
END_ADD
```

The IAPNetwork and IAPNetworkWeighting have little impact on Ethernet setup, and are discussed in section 7.1.7. In this example, the IAP belongs to the first network entry in the Network table.

The Location table is discussed in 1.1.1. In this case the Location entry used is the second in the Location table.

7.1.4 The Connection Preferences table

The connection preferences table defines which IAP is used by default, and whether it is used automatically or a prompt is displayed.

A maximum of two connections are supported, distinguished by the “Ranking” field. Symbian OS initiates the connection using the IAP with the highest ranking – the next ranked IAP is used if the connection attempt fails then.

The section below assumes that the IAP to be used is the first IAP entry in the IAP table.

```
[Connecti onPreferences]
ADD_SECTI ON
  Ranki ng=1
  Di recti on=OUTGOI NG
  Di al ogPref=PROMPT
  BearerSet=LAN
  I AP=1
  FI ELD_COUNT=5
END_ADD
```

7.1.5 The Proxy table

If your network uses a proxy server, you may need to add a proxy record to *CommDb*. Your network administrator will know whether you *need* to connect to a proxy server or not.

The example below shows the proxy settings used within Symbian.

```
[Proxi es]
ADD_SECTI ON
  I SP=1
  ProxyServi ceType=LANServi ce
  UseProxyServer=TRUE
  ProxyServerName=proxy. i ntra
  Protocol Name=http
  PortNumber=80
  Excepti ons=*. i ntra
  FI ELD_COUNT=7
END_ADD
```

7.1.6 The Location table

The location table is used to set the dialling behaviour etc. As these settings are irrelevant for Ethernet (and GRPS) you can use any valid location table entry (normally just select any existing entry in the Location table)

The location table settings are only meaningful for CSD based connections.

7.1.7 The Network table

The network table has minimal impact on Ethernet setup; you can use an existing network or define your own, e.g:

```
[Network]
ADD_SECTI ON
# COMMDB_I D = 1
  Name=I ntranet
  FI ELD_COUNT=1
END_ADD
```

The Network table forms part of the infrastructure for defining the connection behaviour when there are multiple networks e.g. GPRS and corporate LAN networks. The IAP table (7.1.3) defines which network an IAP belongs to and its weighting where there are multiple IAPs in a network.

The global settings define the default network

```
[Gl obal Setti ngs]
ADD_SECTI ON
...
  Defaul tNetwork=1
...
END_ADD
```

8 Troubleshooting

HTTP Example Client Reports Error -1

In most cases “-1” indicates that the current CommDb is not compatible with the Emulator. The cause of this error can be very difficult to locate; during testing on this white paper it occurred several times, mainly due to oddities in the way that Series 60 handles connection preferences.

The best way to avoid trouble is to use the template CommDb configuration files for the SDK/Devkit you are using. If using DHCP then you will not need to edit these files. If using static IP addresses ensure that the IAP, LANBearer and LANService are set up exactly as in section 7.

HTTP Example Client Reports Error -5120

This error indicates that the URL cannot be resolved to an IP address. This is usually caused by a problem with DNS; either the location of the DNS server is incorrectly specified, there is no DNS server, or a proxy server (if there is one) is not at the specified IP address.

The first thing to try is to remove the proxy entry from the CommDb configuration file. If a connection is now possible, then the problem was caused by a bad proxy setting.

If using a statically allocated DNS address, check that the value is correct for your network.

HTTP Example Client Reports Other Error

The example client can return other errors from lower levels of the stack. The public knowledgebase (<http://www.symbian.com/developer/techlib/faq.html>) may provide hints at the possible cause.

Windows reports that there is an IP address conflict

The likely cause is that the (static) IP address of your Emulator is the same as that of your computer. This will not occur if your IP addresses are dynamically allocated.

This error occurred when testing this paper on a home network consisting of a router connecting to the Internet via an ADSL modem.

The solution is to ensure that the Emulator is allocated a unique IP address in the LANService table of the CommDb configuration file.

DHCP not handing out IP Addresses

In order to increase network security, DHCP is sometimes configured to only supply IP addresses to known computers on specific VLANs. If this is the case on your network, you will need to ask your network administrators to register the MAC address for your Emulator on the network. This MAC address is created by the configchange tool in the emulator C drive file:

```
c:\system\data\ethermac.dat
```

This file should contain only a hexadecimal number; it must not contain spaces or hyphens. Also ensure that there is no newline or carriage return character in the file.

9 References

No.	Document Reference	Version and Distribution	Description
[R1]	FAQ1071 : What tools are available for editing/configuring the Communications	Public Symbian DevNet	FAQ describing tools for editing the CommDb

No.	Document Reference	Version and Distribution	Description
	Database (CommDb)?		
[R2]	Ethernet basics	Internet page	http://www.mynetwatchman.com/pckidiot/chap04.htm , An overview of Ethernet, including information about the various types of MAC Addresses

10 Document History

Date	Version	Status	Reviser	Description
Feb-2004	1.0	Draft	Nao Kagabu	First draft issued after initial comments from various reviewers
06-04-2004	1.1	On review	Nishaat Rajabali	Did major changes to formatting and content. Included input from the appropriate Technical Architects.
15-06-2004	1.3	On review	Hamish Willee	Major changes to formatting and content in order to reflect actual deliverables.

11 Glossary

Term	Definition
DLL	Dynamic Link Library
GPRS	General packet Radio Service
GSM	Global System for Mobile Communications
IDE	Integrated Development Environment
LSB	Least Significant Bit
MAC	Media Access Control
NIC	Network Interface Card
PDD	Physical Device Driver
WinPcap	Public API for packet-capture and network analysis for the Win32 platforms.
Techview	Symbian's reference UI. See » Developer Library » Symbian OS Guide » IDEs and emulators.
EKA1/EKA2	From Symbian OS v8.0 two versions of the kernel are supported. EKA1 is the original kernel. EKA2 is a new kernel that offers hard real time guarantees to kernel and user mode threads.
Symbian OS Development Kit (DevKit)	The DevKit offers access to all the Symbian OS APIs, and supplies nearly all of the Symbian OS source code, and supports software tools and development boards from several vendors. See » Developer Library » About Symbian OS Kits » Introduction to the Development Kit.

12 Acknowledgements

The authors wish to thank the many Symbian staff who supplied valuable information and review comments.

Symbian licenses, develops and supports Symbian OS, the platform for next-generation data-enabled mobile phones. Symbian is headquartered in London, with offices worldwide. For more information see the Symbian website, <http://www.symbian.com/>.

Trademarks and copyright

'Symbian', 'Symbian OS' and other associated Symbian marks are all trademarks of Symbian Ltd. Symbian acknowledges the trademark rights of all third parties referred to in this material. © Copyright Symbian Ltd 2003. All rights reserved. No part of this material may be reproduced without the express written permission of Symbian Ltd.

Disclaimer

Symbian Ltd makes no warranty or guarantee about the suitability or accuracy of the information contained in this document. The information contained in this document is for general information purposes only and should not be used or relied upon for any other purpose whatsoever.