1.2.1

MACARA OPTIMIZATION SERVICE NODE

Rev C - 2/20/03

Configuration Guide for Client 1.3

MACARA OPTIMIZATION SERVICE NODE CONFIGURATION GUIDE FOR CLIENT 1.3

NOTICE OF LIABILITY

COPYRIGHT AND TRADEMARKS

OPENSSH

INDEPENDENT JPEG COMPONENTS

The Bytemobile Macara software is based in part on the work of the Independent JPEG Group.

IPFILTER COMPONENTS

The Bytemobile Macara software is based upon IPFilter, Copyright © 1993–2001 by Darren Reed. The author of IPFilter accepts no responsibility for the use of the IPFilter software and provides it on an “as is” basis without express or implied warranty. This IPFilter program is distributed WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

ZLIB COMPONENTS

The Bytemobile Macara software is based upon zlib software, Copyright © 1995–1998 Jean-loup Gailly and Mark Adler. The zlib software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

INFORMATION ON GPL COMPONENTS

The Bytemobile Macara software is dynamically linked to components that are licensed to Bytemobile under the GNU General Public License (including FreeSwan, l2tpd, and squidGuard). Some of these components include code that has been modified by Bytemobile. Each recipient of these components that has obtained a binary license from Bytemobile obtains a license from the original licensor of these components to copy, distribute, or modify them subject to the terms of the GNU GPL (which can be reviewed at http://www.gnu.org/licenses/gpl.html#SEC1). Upon request to Bytemobile, each such recipient may obtain a copy of a complete machine-readable copy of the corresponding source code for these components for a charge no more than Bytemobile's cost of physically performing source distribution.

Contents

About this Guide

This guide supplements the Macara OSN System Administrator's Guide and provides the following information:

- A brief overview of the Macara Optimization Service Node (OSN) and the Macara Client server components.
- Procedures to install, setup, use, and troubleshoot the Macara Client software.

Audience

The intended audience is the installation and administrative teams who install, configure, and maintain the Macara OSN at the Mobile Network Operator's (MNO) site.

Font Conventions

Font	Meaning	Example
`Monospaced AaBbCc123`	Names of commands, files, directories, and on-screen computer output.	Edit the `.login` file.
`Monospaced bold AaBbCc123`	What you type, contrasted with on-screen computer output.	machine_name% **`su`**
`Monospaced italics AaBbCc123`	Command-line placeholder that you replace with a real name or value.	To delete the file, type: # **`rm`** *`filename`*.
italics	Book titles, new words or terms, or words you want to emphasize.	See Chapter 6 in *User's Guide*.

Command Syntax

Command, configuration, and programming syntax follow the conventions illustrated and described in the following examples.

The `bmgetsvcid` command-line utility:

Syntax `# bmgetsvcid -i svcid|-s svcname`

`bmgetsvcid`	**Screen font bold** indicates a command or option that must be typed exactly as shown.
`svcid`	**Screen font italic** represents a variable that must be replaced as appropriate. For example, choosing the `-i svcid` option, the system administrator would type `# bmgetsvcid -i 8`
`-i` *`svcid`*`|`**`-s`** *`svcname`*	**Delimiter bar** separates options, one of which must be chosen. The `bmgetsvid` command has two options; the system administrator must choose one of them.

The pass rule from the Redirector module:

Syntax **`pass`** `[!]`*`destgroup`* `[`**`all`**`|`**`none`**`] {...}`

`[!]`, `[`**`all`**`|`**`none`**`]`	**Brackets** indicate one or more optional parameters.
`{...}`	**Curly braces** indicate a list of options or expressions. The system administrator may choose more than one. **Ellipses** indicate that multiple expressions may be repeated on the same line.

Organization of this Guide

This guide is organized as follows:

Chapter 1	**Overview** Provides a brief overview of the Client-server software.
Chapter 2	**Configuring the Client Manager** Provides background information and procedures to configure the server-resident component of the Macara Client software.
Chapter 3	**Preparing the Macara Client Software** Describes how to modify the `BMInstallation.ini` file installation and package the Macara Client software for distribution.
Chapter 4	**Installing and Running the Macara Client Software** Provides background information and procedures to configure the Macara Client software.
Appendix A	**Client-Server FTP Configuration** Explains FTP settings and parameters and provides procedures to configure FTP settings.
Appendix B	**Files and Parameters** Describes parameters that appear in the `cmgr.conf` and `BMInstallation.ini` files. Parameters in the `cmgr.conf` file are used to start the Client Manager and enable optimization between the Macara OSN on the server and the Macara Client software on the subscriber's device.The `BMInstallation.ini` file parameters are used to prepare the Macara Client software for subscriber distribution.
Appendix C	**Glossary** Lists and defines terms and acronyms used in the Bytemobile, Inc., user documentation.

Related Documents

Bytemobile Documentation	Description and Source
Macara OSN Release Notes for Client 1.3	Describes new features, issues resolved, and issues identified. Macara Client R1.3 CD.
Macara OSN Release Notes 1.2.1	Describes new features, issues resolved, and issues identified. Macara OSN CD.
Macara OSN Client for Windows Release Notes 1.2.1	Describes new features, issues resolved, and issues identified. Macara OSN CD.
Macara OSN System Administrator's Guide 1.2.1	Describes features and procedures to install, configure, and run the Macara OSN.
Macara OSN Network Management Subsystem Administrator's Guide 1.2.1	Details on using the Macara SNMP agent, available on the Macara Documentation CD.
Macara Client SDK Windows Developer's Guide and *Macara Client SDK Pocket PC Developer's Guide 1.2.1*	The software developer's kit (SDK) includes the following components: Overview, Developer's Guide, and Client Deployment Guide.

Bytemobile Online Documentation	Description and Source
Macara OSN Client Software for Windows Help	The Macara OSN Client software for Windows provides Help on how to use the application. Help is accessible from the application.

Third-Party Documentation	Description and Source
Solaris 8 System Administration Guide, Volume 3	Comprehensive instructions on administration procedures for the Solaris platform. http://docs.sun.com

Obtaining Technical Support

Bytemobile provides technical support for the Macara product line at **support@bytemobile.com**.

Overview 1

Information provided in this chapter:

Descriptions and Benefits

The Macara OSN is a network proxy server that provides data optimization, network control, and management services. It compresses data transmission between your PC and the network.

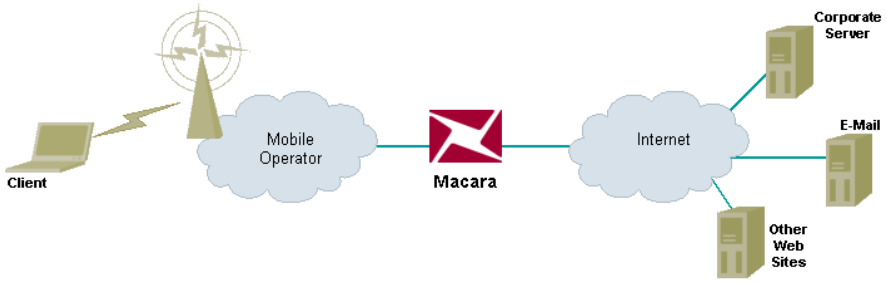


Figure 1. Macara client-server connections and interactions

The entire operation of the Macara OSN is transparent to the user and requires no special equipment or additional configurations. The Macara OSN includes the network resident Server component and the optional Client software that is installed on the subscriber's Windows platform to further enhance optimization of web access and email traffic.

In clientless mode, the Macara OSN performs application- and content-specific optimization over the HTTP, POP, and IMAP protocols. Acceleration, QoS and other services are available to other protocols—but optimization services are not. In a client-server configuration, the Macara Client works in concert with the Macara OSN to

provide lossless optimization for data transmitted over the MAPI, SMTP, NNTP, FTP, and all other TCP-based protocols.

Also, in clientless mode, the benefits of caching and optimization flow only one way—from the Macara OSN to the client. The Macara Client software opens a dual-optimization channel, with optimization occurring in both directions.

Components and Interactions

Setting up a client-server configuration involves the following:

- Installing and setting up the Macara Client software on client devices.
- Configuring the Web proxy, POP proxy, IMAP proxy, Generic proxy, and the Client Manager services on the Macara OSN.

The Macara Client Software

The Macara Client optimizes any TCP/IP packets between itself and the Macara OSN. It performs this optimization in a manner that is transparent to both applications and content servers. The Client-Server option provides optimization features and benefits beyond those provided by the clientless implementation and operates with virtually any dialer software that your subscribers are using. This includes the Windows dialup networking, third party aircard dialers, or other third party dialer software.

The Macara Client software is available in a stand-alone version and as a software developer's toolkit (SDK). The stand-alone version is ready for subscriber distribution and does not require integration with any other software. The SDK comes as a set of application programming interface (API) functions designed to enable integrating the Macara Client Software with your own dialer application.

Note This guide only addresses the stand-alone configuration issues and procedures. For information on using the SDK version, refer to the *Macara Client SDK Windows Developer's Guide and Macara Client SDK Pocket PC Developer's Guide 1.2.1*.

The Client Manager

The Client Manager serves as an intermediary between the Macara Client software on subscriber devices and proxy services on the Macara OSN. The Client Manager performs three main functions:

- **Verifying the Macara Client software resides on the client.** The Client Manager maintains contact with client devices by means of beacon messages. Among other

tasks, it keeps track of the clients that have the software installed and those that do not.

- **Communicating the client state to proxy services.** Because the Client software enables optimization, it is necessary for the proxy services to conclude if the software is installed on a device in order to determine how to respond to a particular request. The primary function of the Client Manager is to provide this client state information to the services.

The Proxy Services

The Macara OSN proxy services provide client-style optimization when the client device is running the Macara Client software, and clientless optimization when it does not. They depend on the Client Manager to determine those client devices (identified by their IP address and interface ID) that have activated their Macara Client.

How it Works

1. **Macara Client is activated**.

   The Macara Client is activated in the following ways.

   - In the SDK form, it is integrated into the dialer software. Thus, after the dialer completes the connection to the wireless network, it activates the Macara Client at that time.
   - In the stand alone form, you can configure the Macara Client to activate automatically or manually, by the user. In automatic mode, it monitors all network connections and activates optimization when the connection is established. In the manual mode, the user activates optimization by clicking the **Start** button.

2. **Macara Client registers with Macara OSN.**

   Once activated, the Macara Client attempts to contact the Macara OSN. Specifically, it sends a registration message (referred to as a beacon) to the Client Manager running on the Macara OSN. The Client Manager is configured to capture a specified IP address and TCP port, and the Macara Client is configured to send its beacon messages to this same IP address and port.

3 **Macara Client and Macara OSN exchange heartbeat messages.**

Once the beacon connection is established, the Macara Client sends periodic beacon messages (heartbeats) to the Macara OSN at an interval configured on the Macara Client known as the “beacon interval.” These heartbeats allow the Macara Client to maintain contact with the Macara OSN and detect failures on the Macara OSN. As long as the Macara OSN responds to the heartbeat messages, the Macara Client continues to optimize traffic. If the Macara OSN fails to respond to the beacon heartbeat (or the beacon connection is reset), the Macara Client closes the beacon connection, and tries to establish a fresh connection. If this attempt fails, the Macara Client makes two more attempts to connect before turning off optimization. In addition, the Macara OSN knows that as long as it receives heartbeat messages, the given device is running an active Macara Client. If the Macara OSN does not receive a heartbeat message during the specified interval (configured on the Client Manager as the beacon interval) it knows that the device is no longer running an active Macara Client.

4 **The proxy services query the Client Manager.**

Before processing requests, the proxy services on the Macara OSN query the Client Manager to determine if the Macara Client software is active on the client device making the request. Two results are possible:

- If the Macara Client software is present, client-server optimization is used by all proxies.
- If the Macara Client software is not present, clientless optimization is performed.

The Macara OSN uses this information to perform the appropriate optimization for devices that are running the Macara Client and devices that are not running the Macara Client.

Configuring the Client Manager 2

Information provided in this chapter:

Assumptions

Material presented in this chapter assume that you have already:

- **Installed the Macara OSN** software and are running it in a clientless configuration.
- **Received the Client** software that you must configure and package for subscriber distribution.

Configuration Tasks

Use the Macara OSN Command Line Interface (CLI) to configure the Client Manager. See *Macara OSN System Administrator's Guide 1.2.1.* The server-resident Client Manager coordinates with the Macara Client software to enable optimization of all TCP-based Internet traffic.

Task 1: Configuring the Heartbeat Interval

For procedures see, "Task 1: Configuring the Heartbeat Interval" on page 7.

Task 2: Configuring the Beacon and Proxy Services

For procedures see, "Task 2: Configuring the Beacon and Proxy Services" on page 7.

Task 3: Verifying Client Connection in CMGR Log File

You must first install the Macara Client software on a PC and then verify the connection in the CMGR log file.

- To install the Macara Client, see "Installing Software and Starting Optimization" on page 21.
- To verify that the connection is established, see "Task 3: Verifying Client Connection in CMGR Log File" on page 9.

Task 1: Configuring the Heartbeat Interval

The Heartbeat interval is a Client-Server option. It specifies the frequency of the beacon messaging. The default settings will work for most installations. It is recommended that you contact Bytemobile (see “Obtaining Technical Support” on page vi) before making any changes or to make sure these settings are appropriate for your environment.

Task 2: Configuring the Beacon and Proxy Services

As described earlier, the Macara Client must coordinate with the Macara OSN for client-server optimization to work properly. Specifically, it sends a registration message (referred to as a beacon) to the Client Manager running on the Macara OSN. In this step, you configure the Client Manger to capture these messages on the specified beacon IP address and beacon port.

Important Do not select a non-existent client beacon IP address in the LAN that is attached to Macara's server-side interface. You must choose either an existing machine's IP on the Macara server-side interface LAN, or an IP address that is more than one hop away from Macara's server-side interface (would normally be routed through the router associated with that interface).

The following procedure configures the Client Manager to capture the beacon IP address of 1.2.3.4 and a beacon port of 4201. You must choose the actual IP address so that the beacon message passes through the Macara OSN. In some environments, the default IP address achieves this function. Other configurations may require the use of a different address.

Note The following procedure may only be used if the default beacon port of 4201 is used. If you must use a different port, you must configure the capture rules manually with the `bmrule` command. For information on the `bmrule` command, refer to *Macara OSN System Administrator's Guide 1.2.1*.

To configure the Macara OSN with these beacon values:

1 Use the `bmconfig` command as follows:

```
# /opt/bmi/bin/bmconfig -o client,beaconip=1.2.3.4
```

Important Blanks are not allowed between a comma (“,”).

You are prompted to start or restart the proxy services. This is required for the changes to take affect.

```
Restart web_1 (y/n)? y
Restart imap (y/n)? y
Restart pop (y/n)? y
Restart generic (y/n)? y
client manager started.
```

2. Type **y** (as shown above) to restart the required services. At the conclusion, the message that the Client Manager is started also appears.

3. Verify that the a capture rule for the beacon IP and TCP port exists by typing the following command. Note that rule 9 is capturing the beacon IP and port and forwarding them to the CMGR service.

   Note In the output listing below, there is a second CMGR rule (rule number ten). The second rule is used for health checks and can be safely ignored.

   # **/opt/bmi/bin/bmrule -l**

```
RECID    ID       SRC IP/(M)        DEST IP/(M)       DP/(M)    SP/(M)       DEV/(M)  SERVICE   REG
-------- -------- ----------------  ----------------- --------  ----------   -------  -------- ---
1        1         0.0.0.0           0.0.0.0                0         0         0   generic   yes
                  (0.0.0.0        ) (0.0.0.0        ) (    0.0) (    0.0) (    0.0)
2        2         0.0.0.0           0.0.0.0              445         0         0             yes
                  (0.0.0.0        ) (0.0.0.0        ) (255.255) (    0.0) (    0.0)
3        3         0.0.0.0           0.0.0.0                0       445         0             yes
                  (0.0.0.0        ) (0.0.0.0        ) (    0.0) (255.255) (    0.0)
4        4         0.0.0.0           0.0.0.0              139         0         0             yes
                  (0.0.0.0        ) (0.0.0.0        ) (255.255) (    0.0) (    0.0)
5        5         0.0.0.0           0.0.0.0                0       139         0             yes
                  (0.0.0.0        ) (0.0.0.0        ) (    0.0) (255.255) (    0.0)
6        6         0.0.0.0           0.0.0.0              143         0         0   imap      yes
                  (0.0.0.0        ) (0.0.0.0        ) (255.255) (    0.0) (    0.0)
7        7         0.0.0.0           0.0.0.0              110         0         0   pop       yes
                  (0.0.0.0        ) (0.0.0.0        ) (255.255) (    0.0) (    0.0)
8        8         0.0.0.0           0.0.0.0               80         0         0   web_1     yes
                  (0.0.0.0        ) (0.0.0.0        ) (255.255) (    0.0) (    0.0)
9        9         0.0.0.0           1.2.3.4             4201         0         0   cmgr      yes
                  (0.0.0.0        ) (255.255.255.255) (255.255) (    0.0) (    0.0)
10       10        0.0.0.0           127.1.0.2           4201         0         0   cmgr      yes
                  (0.0.0.0        ) (255.255.255.255) (255.255) (    0.0) (    0.0)
```

Important In the Client-Server configuration, if you want to stop a proxy service, you must remove its classifier rule. This will direct traffic that would normally have been captured by that service to the generic proxy. This enables the Macara OSN to operate properly. When the generic proxy service is stopped, you must also stop CMGR to effectively prevent optimization in new connections.

Task 3: Verifying Client Connection in CMGR Log File

After configuring the Client Manager and installing the Macara Client (see "Installing and Running the Macara Client Software") you must check to see if the client can connect to the server. You can make this verification by checking the information in the Client Manager's Log file (`cmgr_info.log`) as shown below.

To verify the client connection from the CMGR log file parameters or modify default settings:

1. Type the following command to verify connection from log entries in the `cmgr_info.log` file.

   `# more /opt/bmi/var/log/cmgr/cmgr_info.log`

2. Verify the following entries are listed in the log file:

   "`Service request state for client <IP Address> <Interface ID>`"

   "`New client beacon connection from client <IP Address> <Interface ID>`"

Client-Server Options

Client-server options govern communication between the Macara Client software installed on subscriber devices and the Client Manager on the Macara OSN. The Client-Server Configuration window contains these configuration parameters.

To configure client-server options:

In the `/opt/bmi/etc/cmgr/cmgr.conf` file, set the following parameters:

forward_beacon_port

This parameter specifies the port on which the Client Manager listens for newly captured connections.

Note: Although its default value is the same, this parameter is not the same as the beacon IP configured above in "Task 2: Configuring the Beacon and Proxy Services" on page 7.

Default `4201`

Important Make sure the listen port is consistent with the Remote Beacon Port field set in the Admin Tool of the Macara Client software on the client.

Syntax `forward_beacon_port: portnumber` (*portnumber* is a valid port number)

beacon_interval

This parameter specifies the time in seconds between heartbeat broadcasts between the client and the Macara OSN.

Default `1800`

Important The heartbeat interval must be greater than the beacon interval set in "Task 2: Configuring the Beacon and Proxy Services" on page 7.

Syntax `beacon_interval: n` (*n* is a positive integer)

vlan

This parameter enables or disables tunneling (VLAN) support.

Default `0`

Syntax `vlan: 0|1` (set to 1 to enable)

TO_long

This parameter specifies the frequency in seconds, that the Client Manager checks for expired clients. These are clients that have not sent an updated beacon message because their connection was lost or they were abruptly shutdown without closing the beacon connection.

Default `100`

Syntax `TO_long: n` (*n* is a positive integer)

max_rtt

This parameter specifies the duration in seconds that a service waits for a response to a request for the client state validation before responding with a “no client software present” message for a particular client.

Default `2`

Syntax `max_rtt: n` (*n* is a positive integer)

TO_short

This parameter specifies how often in seconds the Client Manager’s service interface checks for a state update when one or more of the client states was in an intermediate value (for example, pending: after the beacon connection was accepted, but before the beacon was read and verified).

Default `3`

Syntax `TO_short: n` (*n* is a positive integer)

The Log Files

There are two log files associated with the Client Manager, `cmgr_info.log` and `cmgr_error.log`. The path to both files is `/opt/bmi/var/log/cmgr`. You can edit these parameters to change these file paths.

info_log

This parameter specifies the file path of the Client Manager’s informational log.

Default `/opt/bmi/var/log/cmgr/cmgr_info.log`

Syntax `info_log: filepath` (*filepath* is the path to the `Manager_info.log` file)

error_log

This parameter specifies the file path of the Client Manager’s error log.

Default `/opt/bmi/var/log/cmgr/cmgr_error.log`

Syntax `error_log: filepath` (*filepath* is the path to the `Manager_error.log` file)

Advanced Configuration Parameters

This section lists configuration parameters that are not available from the Command Line Utility. Some parameters are included here because they are seldom used. Others are removed from general view due to the potential for harm if set in error.

Note It is recommended system administrators familiar with the TCP protocol and proxy operations modify the default values in these parameters.

register_service

This parameter enables or disables the automatic registration of the Client Manager into the Service Control Framework.

Syntax **`register_service:`** `0|1`

where `1` enables and `0` disables automatic registration.

Default `1`

service_name

This parameter specifies the unique name of the Client Manager.

Syntax **`service_name:`** *`name`*

where *`name`* is expressed in alphanumeric characters, plus the underscore.

Default `cmgr`

service_addr_path

This parameter specifies the path on which the Client Manager listens to service requests.

Syntax **`service_addr_path:`** *`filepath`*

where *`filepath`* is the full path to the address.

Default `/tmp/service.req_str`

Important This value must be consistent with the value specified in `libcmgr.conf` (the configuration file for the API library used by the services to interface with CMGR).

pid_file

This parameter specifies the path to the file in which process IDs are stored.

Syntax **`pid_file:`** *`filepath`*

where *`filepath`* is the full path to the pid file.

Default `/opt/bmi/var/run/cmgr_pid`

TO_broadcast

This parameter specifies the time in seconds between heartbeat broadcasts.

Syntax **`TO_broadcast:`** *`n`*

where `n` is the send-interval expressed as a whole number in seconds.

Default `10`

Note The `macara_dist` parameter must be set to `1` in order for this parameter to take effect.

trans_capt

This parameter enables the Client Manager to operate in either transparent or explicit mode.

Syntax **`trans_capt: 0|1`**

where `0` indicates explicit mode and `1` indicates transparent mode.

Default `1`

wireless_network

This parameter, along with `wireless_mask`, defines the range of IP address that wireless clients may occupy.

Syntax **`wireless_network:`** *`ipaddress`*

where *`ipaddress`* is expressed in the 0.0.0.0 format.

Default `none`

wireless_mask

This parameter, along with `wireless_network`, defines the range of IP address that wireless clients may occupy.

Syntax **`wireless_mask:`** *`ipnetmask`*

where *`ipnetmask`* is expressed in the 0.0.0.0 format.

Default `none`

The libcmgr.conf File

The `libcmgr.conf` file specifies the settings for the CMGR API that the proxy services use to link to the CMGR. The path to this file is `/opt/bmi/etc/cmgr/libcmgr.conf`

Syntax *`parameter: option`*

File Structure

The `libcmgr.conf` file consists of parameter/option pairs. Configuration parameters are separated from options by a space. Each pair must be on a separate line. The # symbol indicates a comment. The CMGR ignores comments as well as blank lines.

`service_address_path`

Specifies the address on which the CMGR listens for service requests.

```
service_address_path: /tmp/service.req_str
```

Important This parameter also exists in cmgr.conf file. Both parameters must specify the same path.

`TO`

Specifies how long in seconds to wait for a response to a request from the CMGR before timing out.

```
TO: 5
```

`max_retries`

Specifies the maximum number of times to request information about a particular client from a CMGR.

```
max_retries: 2
```

Troubleshooting the Client Manager

The server-resident Client Manager symptom that requires resolution is when optimization does not occur. To troubleshoot Client Manager problems, try the following methods.

	Action	Method
1	Verify that the Client Manager is running.	Use the **ps** command as follows: `# ps -ef | grep cmgr` If no output appears, the Client Manager is not running.
2	Confirm the licensing information.	View the log file `/var/adm/messages` and search for the string `FLEXlm`. A message containing this string (and the related messages immediately before and after it) indicates a problem with the license management software or the license key.
3	Verify that classification rules to capture packets and redirect them to the Client Manager are registered.	Use this command: `# bmrulelist` If none are registered, use **bmrule** to define the appropriate rules.
4	Verify that the Macara Client configuration is consistent with the Client Manager configuration.	Verify that the beacon interval and beacon port settings are the same on both the client and the Client Manager.
5	Verify CMGR is receiving beacons.	View `cmgr_info.log` file.
6	Verify client coordination and VLAN are activated	Use this command to view configuration: `# bmconfig -l`
7	Stop and then restart the Client Manager.	Restart the module by typing the following command: `# /opt/bmi/bin/bmproc --start | restart | stop cmgr`

Preparing the Macara Client Software 3

Information provided in this chapter:

Assumptions

Material in this chapter assume that you have already configured the server-resident Client Manager and restarted all modified proxy services. This is necessary to properly test the Macara Client after it has been configured.

The Macara Client Software

The Macara Client works with the Macara OSN to optimize Internet traffic. This results in speeding up web access, sending and receiving email, and other types of Internet traffic.

Preparing the Macara Client Distribution Package

Task 1: Unpack the Macara Client Files

For procedures see “Task 1: Unpack the Macara Client Files” on page 17.

Task 2: Configure the Default Parameters for the Macara Client

For procedures see “Task 2: Configure the Default Parameters for the Macara Client” on page 18.

Task 3: Package the Macara Client Software For Distribution

For procedures see “Task 3: Replace Macara Client Logo in logo.bmp” on page 18.

Task 4: Install and Test the Macara Client Software

For procedures see “Task 4: Install and Test the Macara Software” on page 20.

Task 1: Unpack the Macara Client Files

The Macara Client software for Windows, consists of the following files, typically distributed as a Zip file.

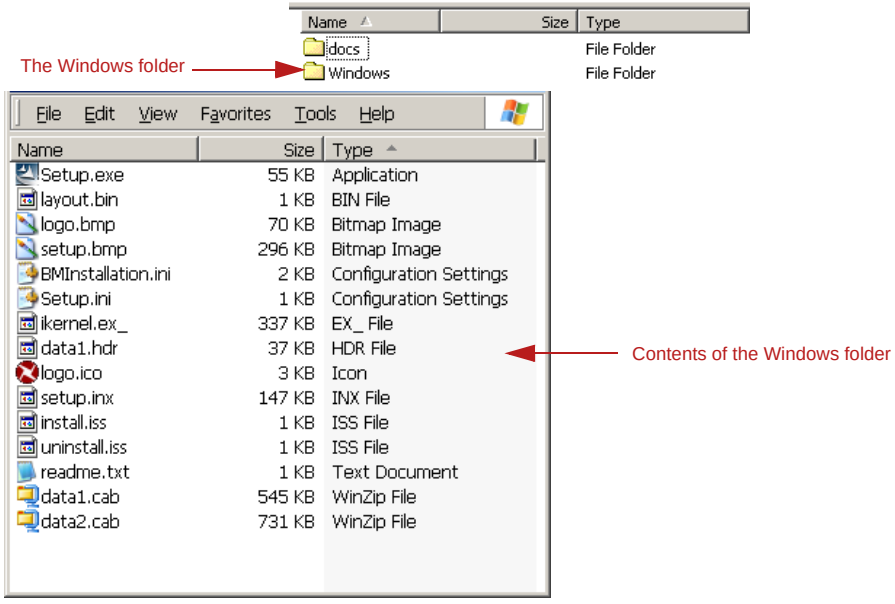


Figure 2. Client software files and folders

Unpack these files in the Windows folder to a directory where you can modify the files shown above and replace them with the ones tailored for your distribution.

Task 2: Configure the Default Parameters for the Macara Client

Bytemobile provides the `BMInstallation.ini` file as one of the components of the Macara Client software shown in Figure 2 on page 17. When your subscribers install the software, values set in this file will be used as the default options. You must use this file to customize the Macara Client software for installation in your network.

For a description of these parameters, see Coordination with Macara in Appendix C.

Parameter and default value:	Verify these values as follows:
`Remote Beacon Port= 4201`	This value must match the `forward_beacon_port` value set in "Client-Server Options" on page 9.
`Remote Beacon IP= 1.2.3.4`	This value must match the `beaconip` value set in "Task 2: Configuring the Beacon and Proxy Services" on page 7.
`Beacon Interval= 900`	This value must be less than the `beacon_interval` set in "Client-Server Options" on page 9. **Note** This value is in seconds. In earlier releases, it was in milliseconds.

Task 3: Replace Macara Client Logo in logo.bmp

Bytemobile provides a copy of the `logo.bmp` file with an image of the Bytemobile logo for branding purposes. This logo appears in all tabbed pages of the Macara Client's main window. You can replace this file with your own logo to rebrand the software in the distribution package.

Specifications for the logo.bmp File

The `logo.bmp` file must adhere to the following specifications:

- Maximum dimension: 330 pixels by 75 pixels
- Maximum color depth: 8 bits (256 colors)
- Transparent color: Red=0, Green=128, Blue=128

Note Use the transparent color for background color of your logo bitmap in areas where you want client window's background color to appear.

Task 4: Package the Macara Software

After you have a proper `BMInstallation.ini` file, you must package this with the other Client files for distribution. This process differs depending on whether you intend to distribute the software on the World Wide Web, a custom CD, or as a part of another piece of software. See below for further details.

Web distribution

For distribution on the web, we will create a self-extracting executable file which can be downloaded by subscribers.

1. Using Windows Explorer, create a folder to hold these files. It is recommended that you use **Macara_Client_1.2.1** for folder name.
2. Copy the entire set of files in the **Windows** folder, to the folder. See Figure 2 on page 17 for a list of these files.
3. Copy the modified `BMInstallation.ini` file you created in "Task 1: Configuring the Heartbeat Interval" and "Task 2: Configuring the Beacon and Proxy Services" into this folder. Be sure to click **OK** to copy over the old files.

   Important Do not change these file names.
4. Copy the modified `logo.bmp` file that you created in "Task 3: Replace Macara Client Logo in logo.bmp." Be sure to click **OK** to copy over the old files.
5. Create a self-extracting executable of the folder using WinZip as follows:
   a. In Windows Explorer, right-click on the folder and then click **Add to Zip**.
   b. Click **Add** on the dialog that appears.
   c. Once Winzip opens, click **Action**>**Make .exe File**.

CD distribution

For CD distribution, there is no need to create a zip file. Simply create a CD with the appropriate files on it.

1. In Windows Explorer, create a folder to hold the files.
2. Copy all files shown in Windows folder in “Task 1: Unpack the Macara Client Files” to the folder.
3. Copy the new `BMInstallation.ini` file you created in tasks 2 through 3 above, to this folder. You must click **OK** to copy over the old files.
4. Use your CD burning software to create a CD containing the files from this directory.

Integration With Other Software

To integrate the Macara Client installation with a piece of third party software, you must modify the installation of the third party software to invoke the Macara Client installer.

Note: This is different from the full integration of the SDK client.

1. Create a self-extracting executable as described above in the Web distribution method.
2. Modify the third party software installer to invoke the executable and unzip the files.
3. Modify the third party software installer to invoke the Setup.exe program in the unzipped files, to install the Macara Client software.

Task 4: Install and Test the Macara Software

See “Installing and Running the Macara Client Software” on page 21 to install the Macara Client software and then see “Task 3: Verifying Client Connection in CMGR Log File” on page 9 to test client connection.

Installing and Running the Macara Client Software

4

Information provided in this chapter:

Installing Software and Starting Optimization

System Requirements

The Macara Client software runs in the following environments.

Windows PC Environments

- Windows 95 (OSR2 only) with Internet Explorer 5.0 or higher
- Windows 98, ME
- Windows NT 4.0 (SP4, 5, 6A only)
- Windows 2000
- Windows XP

Default Settings

When the Macara Client software is installed, it will automatically load default values specified in the `BMInstallation.ini` file and will use them to establish connection and start optimization. See “Configuration Tasks” on page 6.

Installing the Macara Client

Important If you are running Windows NT, 2000, and XP, you must have **administrator privileges** to install the Macara Client. This is not necessary for Windows 95, 98, and ME systems. Also, be sure to **close all applications** before running the setup program, as it will automatically shutdown and restart your Windows workstation once the Macara Client is installed.

To install the Macara Client

1. Close all open files and open applications. The installer will automatically restart the system upon completion of the installation process.
2. Locate and double click **`Setup.exe`** to start the installer program.

   The installer displays a series of dialogs, beginning with the Macara Client Installation dialog.

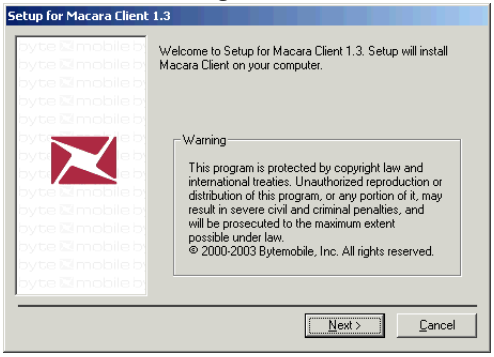


Figure 3. Macara Client Installation dialog

3. Click **Next>** to continue.

4 When the Macara Client License Agreement dialog appears, click **I Accept** to continue.

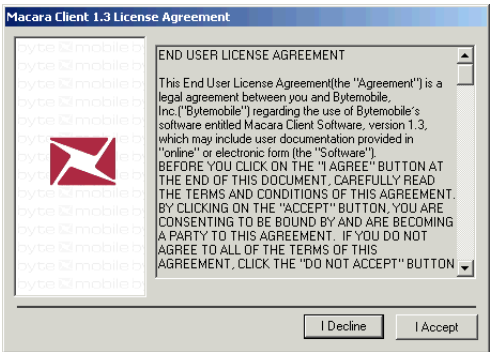


Figure 4. License agreement

The installation takes a few seconds and the following dialog appears.

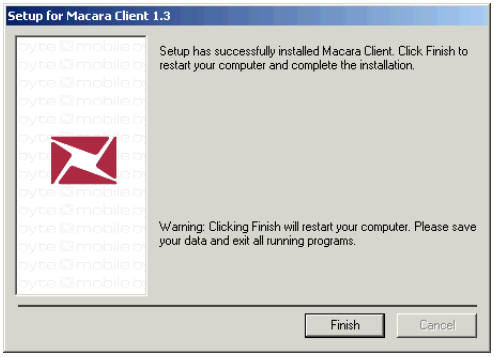


Figure 5. Macara Client installation message

Note When you click **Finish**, the Macara Client installer will automatically start optimization after restarting the system. See “Default Settings” on page 22.

5 Click **Finish**.

Once the installation is complete, the Macara Client installer will:

a Automatically restart the system

b Place the Macara icon on the **Desktop**, in the **Program section of the Start menu**.

c Start optimization automatically.

Starting Optimization

As indicated above, the Macara Client requires no user interactions to start optimization. It will attempt to automatically start optimization and display the status and outcome of its efforts after restarting the system, provided:

- You have not modified the default **Start optimization automatically** option in the Macara Optimization Client dialog box.

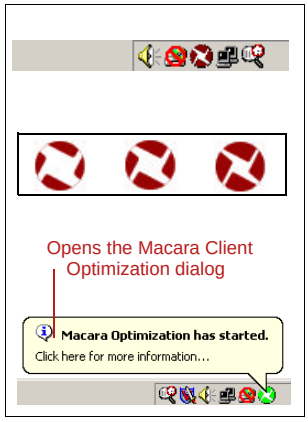
and

- A Macara OSN is available on the network connection. This can be a dial-up or a local area networking (LAN) connection.

In this process, the Macara Client performs the following tasks and displays the outcome using the Macara icon in the system tray.

1. Displays a red Macara icon indicating the application is running, but **optimization is not enabled** and searches for an available networking connection
2. Connects to the Macara OSN specified in `BMInstallation.ini` and displays a changing (pulsing) red Macara icon while trying to establish connection.
3. Establishes connection with the Macara OSN, starts optimization, and displays a green Macara icon.

   Note The tray balloon is only available on Windows 2000, Windows ME, and Windows XP. The **Optimization has started** message will disappear in a few seconds.



Operating the Macara Client

Use the Macara Client icons in the system tray menu, on desktop, and in Windows Start menu to perform operations listed below. Necessary procedures appear in this chapter.

Using the system tray icon

- Check optimization status
- Stop or start optimization
- Open the Macara Optimization Client dialog to:
  - Start or stop optimization and view optimization and connection statistics in the General page of the dialog.
  - Display the About page to view software version number and copyrights information.
  - Change personal preferences in the Options page of the dialog.
  - Access online help.
- Exit the Macara Client application

Using the desktop and Windows Start Menu icons

Restart the Macara Client application after quitting the application using system tray's Exit command.

Using the Admin Tool dialog

Verify Macara Client filter and beacon settings.

Operations Using the System Tray Icon

Checking Optimization Status

The Macara icon in the system tray displays the latest status of the application.

Icon Color	Indication
Red 	The application is **running but is not active**.
Pulsing 	The Macara Client is trying to contact the Macara OSN using the specified beacon parameters.
Green 	The Macara OSN is contacted and **optimization is enabled**.

Using the System Tray Icon Commands

In the system tray, point to the Macara icon and then right-click the mouse button. Depending on optimization status, one of the following shortcut menu appears.

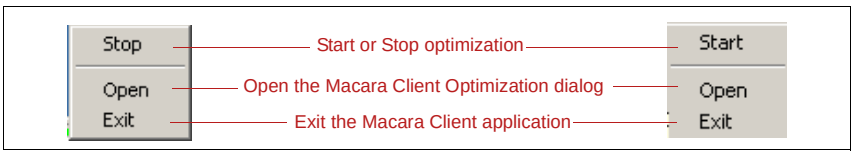


Figure 6. The System tray icon shortcut menu

To stop or start optimization:

1. Select the Macara icon in the system tray and then right-click the mouse button to open the shortcut menu. See Figure 6.
2. You can stop or start optimization as follows:
   - In the shortcut menu, click **Start** to start or **Stop** to stop optimization.

   Or,

   - In the shortcut menu, click **Open** to display the General page of the Macara Optimization Client dialog, and then click **Stop** to stop or **Start** to start optimization.

   The Macara icon in the system tray changes color as optimization status changes.

To open and view data in the Macara Optimization Client dialog

In the System tray icon's shortcut menu, click **Open**.

The Macara Optimization Client dialog opens in the General page and displays the following information in the General, Options, and About pages:

- **General page**, displays optimization and connection statistics in the **Optimization** and **Connection** boxes. See Table 1.
- **Options page**, startup and notification options.
- **About page**, software version number and copyrights information.

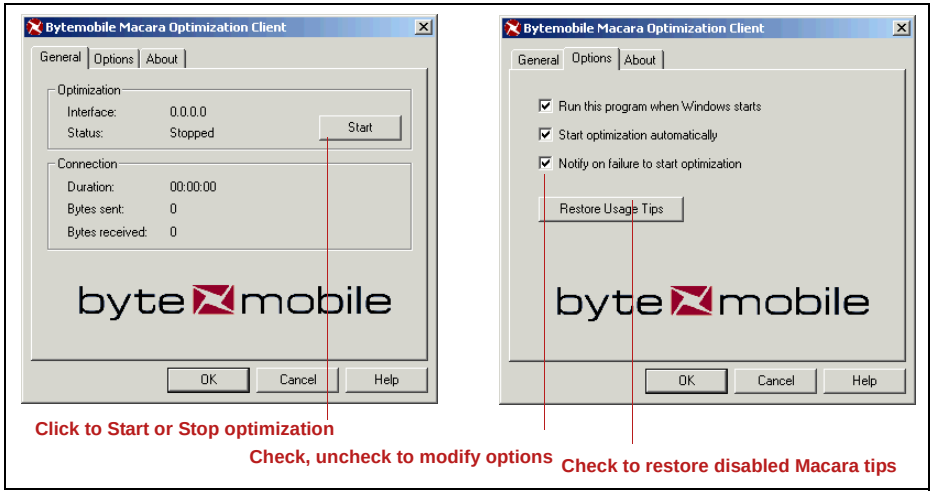


Figure 7. Macara Optimization Client dialog - General and Options pages

Table 1. Data in the General page

Name	Description
Interface	IP address of the Macara OSN server at the start of optimization services. This IP address changes to **0.0.0.0** when optimization stops.
Status	Optimization status, displays one of the following: -- Started -- Stopped -- Coordination in progress ... -- Stopping ... -- Service not available
Duration	Counter displaying the length of time, to the nearest second, since optimization began.
Bytes sent	Counter displaying the cumulative number of bytes sent since optimization began.
Bytes received	Counter displaying the cumulative number of bytes received since optimization began.

Table 2. Data in the Options page

Name	Description
Run this program when Windows starts	Automatically runs the application when Windows starts.
Start optimization automatically	Enables connection detection to automatically start or stop optimization at system.
Notify failure to start optimization	Displays failure message in the system tray icon if unable to start optimization. Enables connection detection to automatically start or stop optimization.
Restore usage tips 	Restores all disabled Macara tips. See Figure 8.

Figure 8. Macara Client Tip dialog and Tip dialog restored notification

Note Once data optimization is stopped, these counters display the duration (hr.min.sec) and byte counts for the last *successful* optimization connection. If **Start optimization automatically** is checked, Macara Client will try to restart optimization when a network connection becomes available. This is not the case if you quit the application using the **Exit** command.

To exit the Macara Client application:

1 Right-click the Macara icon in the system tray menu to display the shortcut menu.

2 In the shortcut menu, click **Exit**.

The following dialog appears.

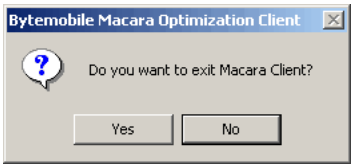


Figure 9. Client exit warning dialog

3 Click **Yes**.

The Macara Client application stops running and the Macara icon in the system tray menu disappears.

Note This procedure does not delete the Macara Client application. It simply stops running the application. For this information, see "Operations Using Desktop or Windows Start Menus."

Operations Using Desktop or Windows Start Menus

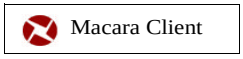
To restart the Macara Client application:

To restart the Macara Client, do as follows:

- Click the Macara icon on the desktop

Or,

- In Windows Start menu, select **Programs>Macara Client>**



The Macara icon reappears in the System tray icon and starts optimization if Start optimization automatically is checked.

Operations Using the Admin Tool Dialog

Viewing and Modifying Macara Client Beacon and Filters Settings

Beacon default parameters are first set in the `BMInstallation.ini` file. Filters can be set in the `BMInstallation.ini` file, but it is not mandatory to do so. The application provides the Macara Client Admin (Administration) Tool dialog to verify and modify the beacon and filter settings using a graphical user interface.

Important If you are using Windows NT, 2000, and XP operating system, you must have administrator level privileges on the device to access this dialog.

To view Beacon and Filter Settings:

Use the following procedure to open the Macara Client Admin tool dialog and check these settings in the different pages of the dialog.

- Double-click the Admin tool shortcut in the Macara Client folder.

  Note The default location is: C:\Program Files\Bytemobile\Macara Client.

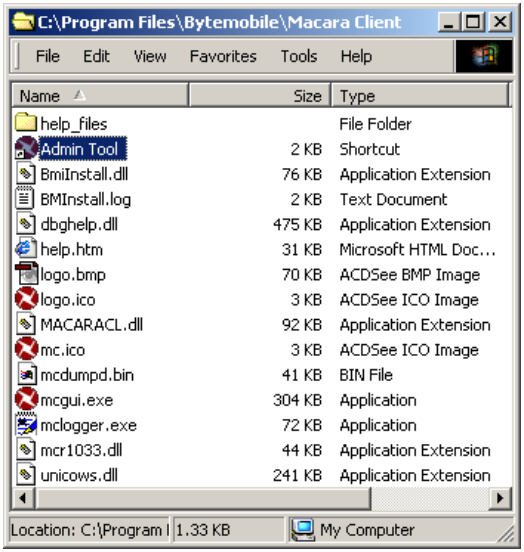


Figure 10. Admin Tool shortcut in the Macara Client folder

Or,

- At the Command prompt, change to this directory, and then type:

```
mcgui -a
```

The Macara Optimization Client, displaying the Filters and Beacon tabs (Admin Tool facilities) opens in the General page. Data in the first three pages are described in Table 1 and Table 2 above. Data in the Filters and Beacon pages are described in Table 3 below.

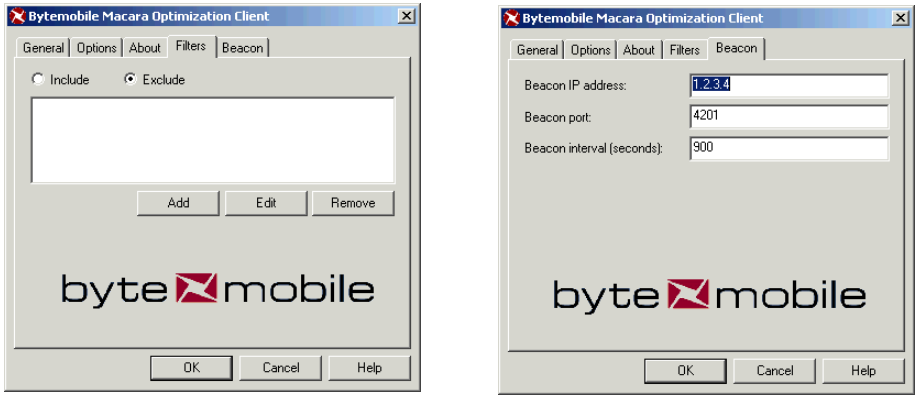


Figure 11. Admin Tool dialog Filters and Beacon pages

Table 3. Data in Filters and Beacon pages of the Admin Tool dialog

Name	Description
Include/Exclude	IP addresses, ports, and masks that are included in, or excluded from optimization.
Beacon IP address	The IP address of the Macara OSN server that the Macara Client beacons.
Beacon port	The port number assigned by the Macara OSN server for the Macara Client to connect to and beacon the server.
Beacon interval (seconds)	The length of time measured in seconds, between any two succeeding heartbeat beacons.

To modify Beacon settings

1. Using the system tray icon, check the current optimization status and stop optimization using procedures described on *Page 27*. Otherwise, the three fields in the Beacon page of the dialog are not available for modification.
2. In the Beacon page of the dialog, modify beacon IP address, port number, and interval settings.

To modify Filters settings:

1. Repeat step one above. Otherwise, the fields in the Filters page are not available for modification.
2. In the Filters page click **Add**.

   The Edit Filter dialog of the Admin Tool opens, displaying current settings.

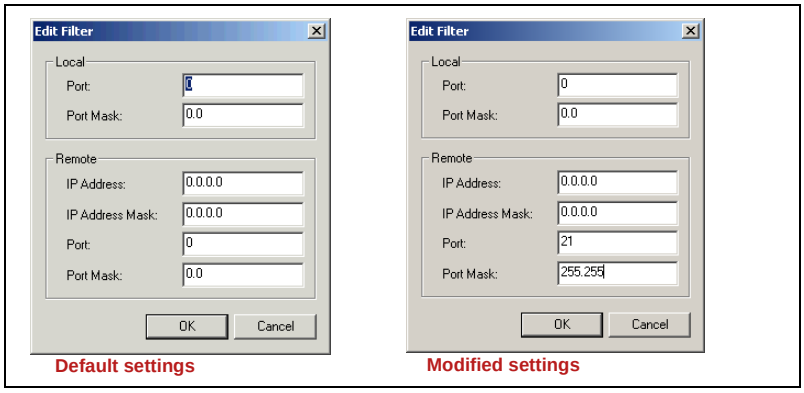


Figure 12. Default and modified settings in Admin Tool's Edit Filter dialog

Table 4. Data in the Edit Filters dialog

Name	Description
Port: (local)	The port on the device to include or exclude for optimization.
Port Mask: (local)	The port mask for the device to include or exclude for optimization. Value is either 0.0 or 255.255.
IP Address: (remote)	The IP address on the Macara OSN server to include or exclude for optimization.
IP Address Mask: (remote)	The IP address mask on the Macara OSN server to include or exclude for optimization.
Port: (remote)	The port on the Macara OSN server to include or exclude for optimization.
Port Mask: (remote)	The port mask on the Macara OSN server to include or exclude for optimization. Value is either 0.0 or 255.255.

3. Type the required local and remote port, IP address, and port mask values and then click **OK**.

   These values appear in the Filter page.
4. In the Filters page, click **Add** to add additional local and remote ports, IP addresses, and port masks.
5. Click **OK**.
6. In the Filters page, check either **Include** or check **Exclude** as required, and then click **OK**.

Uninstalling the Client Software

To uninstall the Macara Client

1. Quit the Macara Client application as described above. See “To exit the Macara Client application:” on page 29.
2. From Start menu, select **Settings>Control Panel>Add/Remove Programs**. The following dialog is displayed.

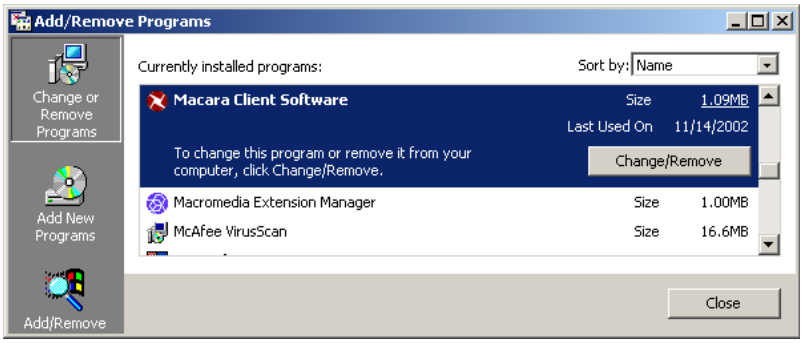


Figure 13. Add/Remove Programs dialog

3. In Add/Remove Programs dialog, select **Macara Client Software>Change/Remove**. The InstallShield Wizard, followed by the following dialog appear.

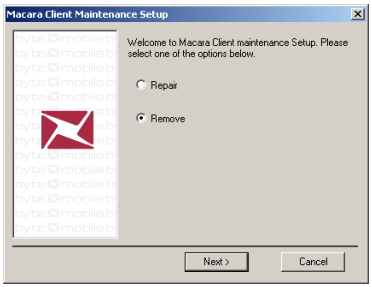


Figure 14. Macara Client Maintenance Setup dialog

4. Select **Remove** and then click **Next >**.

   The Macara Client Unistallation dialog appears and prompts you close all network applications such as e-mail, Netscape, and others.

5. Select **Yes**.

   The Macara Client Unistallation dialog prompts you to restart your system.

6. Select **Finish**.

   Your system will automatically restart and the Macara Client software is removed.

Troubleshooting

Symptom: Unable to activate Optimization

Recommended actions:

- Verify the device is connected to the network where the specified Macara OSN resides.
- Verify the beacon parameters on the beacon page of the Admin tool.

Symptom: Unable to remove the Macara Client Using Windows Control Panel

You tried to uninstall the client software using **Start>Settings>Control Panel>Add>Remove Programs** and the Client software is not removed. Under normal operation, this should never happen. In fact it can only happen if something very unusual has happen to the device. For example, if the Windows system registry becomes corrupt, the Macara Client (and many other programs) may fail to uninstall properly. If you find yourself in such a situation, you can remove Macara Client manually using the following process.

Warning Perform these instructions exactly as stated. Otherwise, you can damage your system's network operations.

Note: Depending on the extent of progress the uninstaller made before it encountered the error, some of the steps listed below may not apply to your device.

Recommended actions:

1. Exit the Macara Client application. See “To exit the Macara Client application:” on page 29.
2. Locate and delete the entire contents of the Macara Client program folder from the system.

   Note Unless you installed the application in a specific directory, the default location is: `Program Files\Bytemobile\Macara Client` or `Program Files\Bytemobile\Macara Client Software`
3. Use the following path to locate and delete the installer folder labeled *{74AC9178-8328-11D5-A9D4-00B0D02DBBB4}*.

   In Windows Explorer, select **[*Drive*] >Program Files>InstallShield Installation Information>{74AC9178-8328-11D5-A9D4-00B0D02DBBB4}**
4. Delete the Macara Client shortcut from the desktop as follows:

   a. Select to the Macara Client icon.

   b. Right click the mouse.

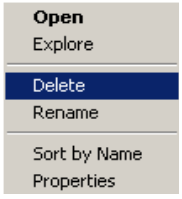
   c. Select **Delete** from the drop-down menu.
5. Delete the Start menu folder as shown below.

   a. Click **Start>Programs>Macara Client Software**, or **Macara Client**, depending on the software version that you are using.

   b. Right click the mouse.

   c. Select **Delete** from the drop-down menu.

   d. When the following warning dialog appears, click **Yes**.

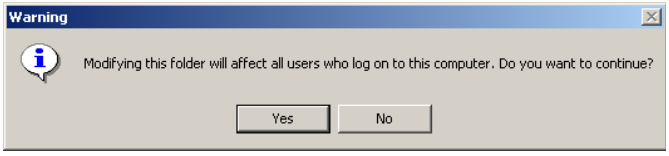
   

Figure 15. Deleting the Macara client software

6 Delete Bytemobile keys in the Registry entries as follows:

a Click **Start>Run** to open the Run dialog.

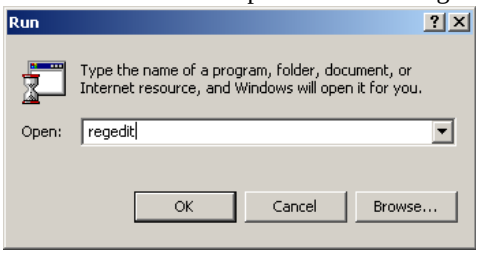


Figure 16. Opening the Registry Editor from the Run dialog

b In the Run dialog, type **regedit** and then click **OK** to open the Registry Editor.

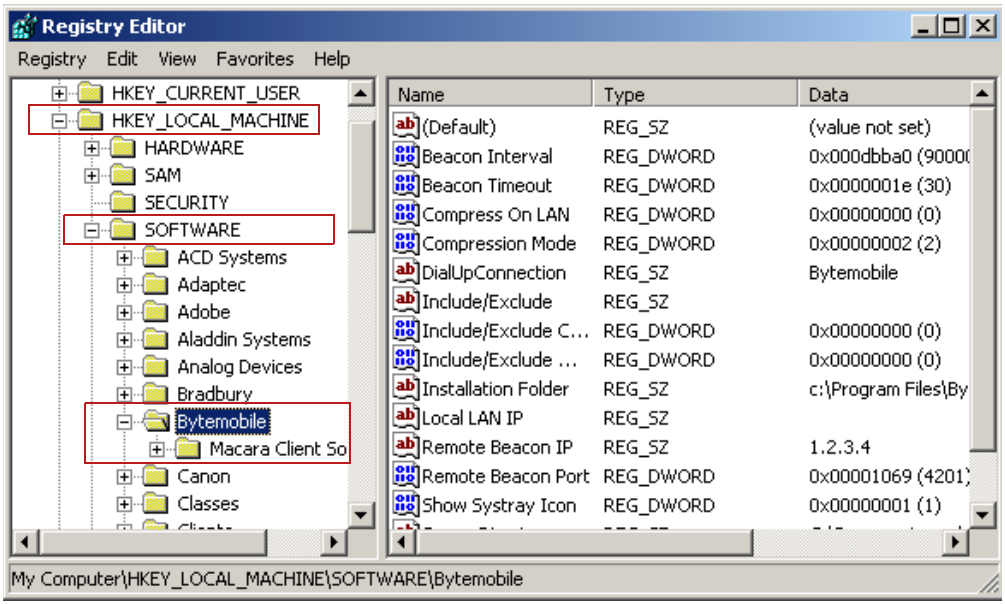


Figure 17. The Registry Editor

c In the Registry Editor, use the following paths to locate and edit **Bytemobile** and **Macara Client** keys.

– **HKEY_LOCAL_MACHINE>Software>Bytemobile**. Delete all values and the key **Macara Client Software**. **Do NOT** delete any other keys under this key.

– **HKEY_LOCAL_MACHINE>Microsoft>Windows>CurrentVersion>Run>Macara Client**. Delete **Macara Client** only.

7 In HKEY_LOCAL_MACHINE, delete the installer registry entries key labeled **{74AC9178-8328-11D5-A9D4-00B0D02DBBB4}**, as shown below.

HKEY_LOCAL_MACHINE>Microsoft>Windows>CurrentVersion>Uninstall>{74AC9178-8328-11D5-A9D4-00B0D02DBBB4}

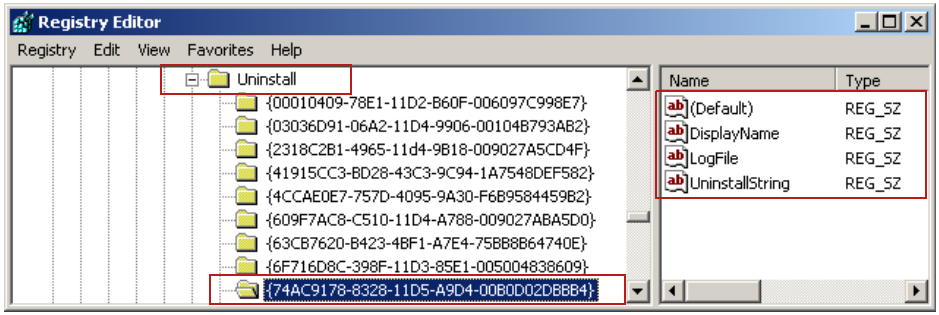


Figure 18. Removing Registry entries

8 In HKEY_LOCAL_MACHINE, locate **Protocol_Catalog9** to remove and update applicable WinSock catalog registry entries using the following path:

HKEY_LOCAL_MACHINE>System>CurrentControlSet>Services>WinSock2>Parameters>Protocol_Catalog9

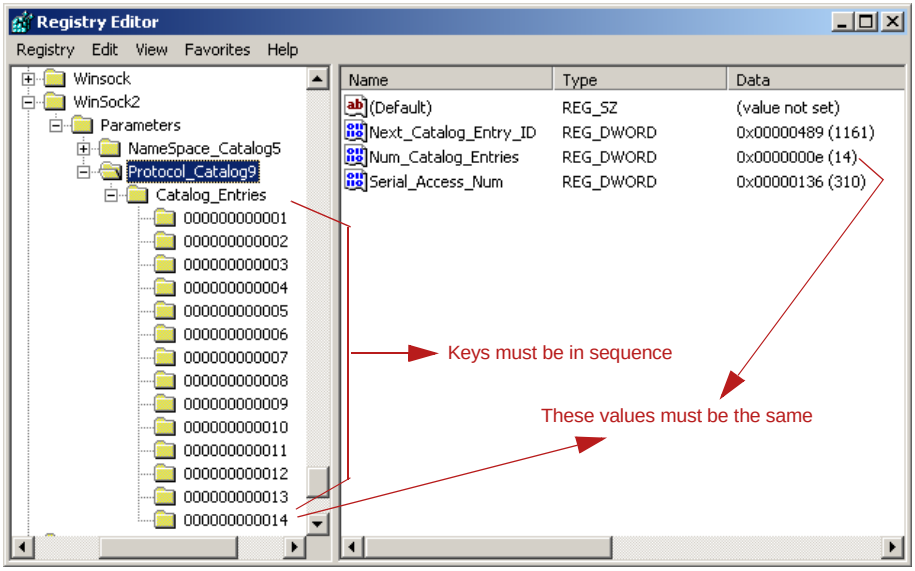


Figure 19. Contents of the Catalog_Entries folder

Using Figure 19 and Figure 20 as guides, locate and delete sub-keys under the Catalog_Entries key that contain a path to **bmi_lsp.dll** and then modify the remaining key names (sequence #s) and data to reflect these changes. Necessary steps are described below.

Warning Perform these instructions exactly as stated. Otherwise, you can damage your system's network operations.

a To locate entries that contain a path to **bmi_lsp.dll**, double-click on each key in **Catalog_Entries**, for example, **000000000001**.

PackedCatalogItem appears. See Figure 15 below.

b Next, double-click **PackedCatalogItem**.

The Edit Binary Value dialog appears as shown in Figure 15.

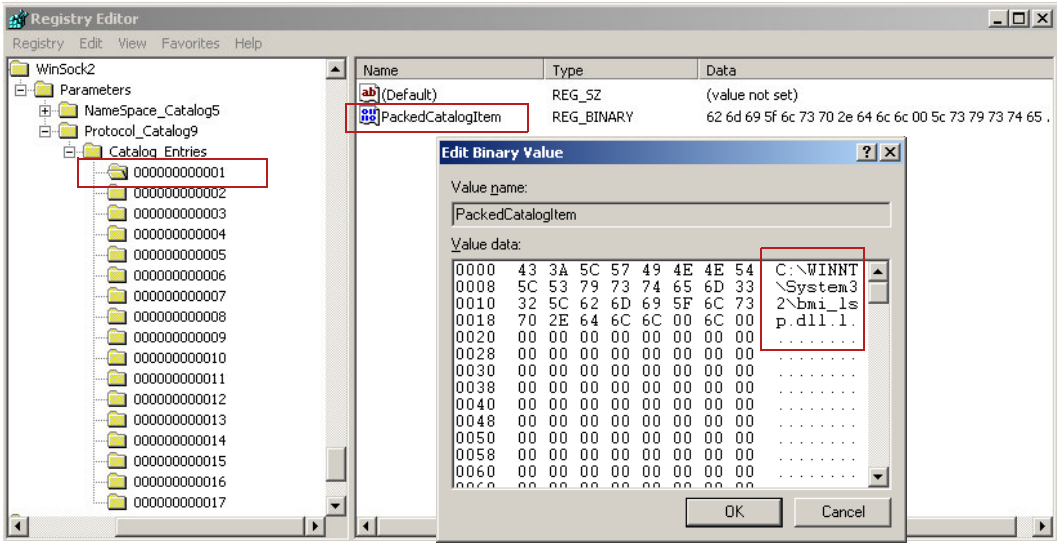


Figure 20. Folders containing bmi_lsp.dll

Important Since the 000000000001 key under Catalog_Entries contains a bmi_lsp.dll entry, it must be removed.

c Right-click the mouse button and select **Delete** from the drop-down menu to delete **000000000001**.

d Continue checking **000000000002** through **0000000000N** and delete all those containing a **bmi_lsp.dll** entry.

e In **Catalog_Entries** folder (see Figure 20), rename the keys in ascending order as described below.

f Point to the key that you want to rename, right-click the mouse button, select **Rename** from the list of options and type the new name (sequence #).

For example, if you deleted keys 000000000001, 000000000003, 000000000010, and 000000000020, then key 000000000002 is renamed to 00000000000, key 000000000004 becomes 000000000003, until the last key, 000000000019 becomes **000000000016** since you deleted four (4) of the twenty (20) keys.

g Under the **Protocol_Catalog9** key, select the value **Num_Catalog_Entries** and then right-click the mouse button. In the drop-down menu, select **Modify**.

The Edit DWORD Value dialog appears.

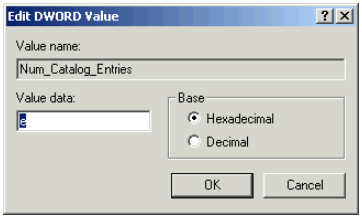


Figure 21. Edit DWORD Value dialog

h In Edit DWORD Value dialog, select the **Decimal** option.

i Type the new value in the **Value data:** field and then click **OK**.

9 Restart your system.

10 Remove the **System** files as described below:

Under the system folder, locate and remove files labeled **bmi_lsp.dll** and **bmzlib.dll**. Depending on your operating system, you can find the **System** folder in the following locations:

- **Windows 95/98/Me:** Windows\System
- **Windows NT/2000:** WINNT\System32
- **Windows XP:** Windows\System3

CLIENT-SERVER FTP CONFIGURATION

A

Note This configuration is only necessary if firewall exists between the Macara Client and Macara OSN.

Most firewalls implement FTP rules by peering into the data layer of the FTP packet coming from the client and looking for the PORT command, which specifies the port to which the server connects on the reverse connection. The firewall then "punches a temporary hole" from the server to the client on only that port. Because the Macara Client compresses the FTP data layer, the PORT command is in a compressed format that the firewall cannot read. Thus, Port 21 data must be excluded from optimization on both the Macara Client and Macara OSN if FTP call setup is to be successful. (The subsequent data transfer on Port 20 is not affected by the firewall.) Configuring the Macara Client and Macara OSN for FTP requires adding a rule on both the Macara Client and Macara OSN to exclude all Port 21 packets from optimization.

To exclude port 21 from optimization on the Macara OSN:

1 Since the Client Manager (cmgr) rule is recommended to be last in the rule list, use the -b option to place the new port 21 rule before the last rule.

```
# opt/bmi/bin/bmrule -a --dport 21 -b 10
```

```
Subscriber classification ID is 10, entry ID is 10.
```

Note Place a single dash before the `-a` option, a double dash before the `--dport` option.

2 Confirm the rule addition, using the `bmrulelist` command

```
# opt/bmi/bin/bmrulelist
```

```
ID   SRC IP/(M)     DEST IP/(M)          DP/M        SP/M      DEV/M     SERVICE  REG
---  ------------   ---------------      -------     -------   -------   ------   ---
1 -7
8    0.0.0.0        0.0.0.0              80          0         0         web_1    yes
     (0.0.0.0    )  (0.0.0.0          )  (255.255)   (   0.0)  (   0.0)
10   0.0.0.0        0.0.0.0              21          0         0                  yes
     (0.0.0.0    )  (0.0.0.0          )  (255.255)   (   0.0)  (   0.0)
9    0.0.0.0        1.2.3.4              4201        0         0         cmgr     yes
     (0.0.0.0    )  (255.255.255.255  )  (255.255)   (   0.0)  (   0.0)
```

To exclude Port 21 from the Client:

1. Using the system tray icon's shortcut menu, stop optimization.
2. Open the Admin Tool dialog. See "To view Beacon and Filter Settings:" on page 30.
3. In the Filters page of this dialog, click **Add** to add the following filters:
   The following dialog opens.

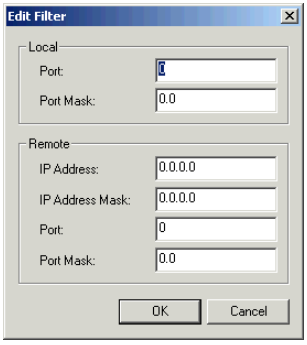


Figure 22. Adding Filters rule

4. Add the following filters and then click **OK**.
   - Local: [:0/:255.255]
   - Remote: [0.0.0.0:21/0.0.0.0:255.255]

   The Filters page will reappear, displaying the added rules.

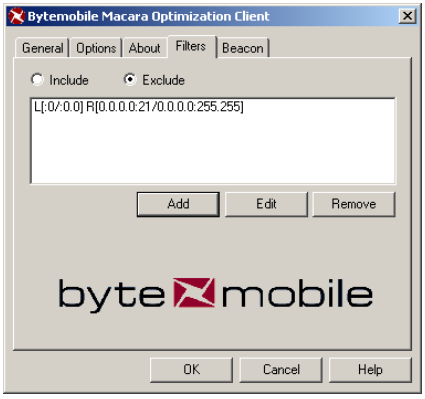


Figure 23. Excluding port 21 for FTP configuration

5. In the Filters page, select **Exclude** and then click **OK** to exit the Admin Tool.
6. Restart the Macara Client to implement modifications.

FILES AND PARAMETERS

B

Client-Manager Configuration File Parameters

he following table lists all Client Manager configuration file parameters alphabetically. This file is located at `/opt/bmi/etc/cmgr/cmgr.conf`.

Parameter	Function
`beacon_interval`	Specifies how long, once a beacon is received, the Client Manager retains its client state information in the database.
`error_log`	Specifies the file path of the Client Manager's error log.
`info_log`	Specifies the file path of the Client Manager's informational log.
`pid_file`	Specifies the path to the file in which process IDs are stored.
`service_name`	Specifies the unique name of the Client Manager for the service control framework.
`TO_broadcast`	Specifies the time in seconds between heartbeat broadcasts to other Macara OSNs in a cluster (when `macara_dist` is enabled).
`wireless_mask`	This field, along with `wireless_network`, defines the range of IP addresses that wireless clients may occupy.
`wireless_network`	This field, along with `wireless_mask`, defines the range of IP addresses that wireless clients may occupy.

BMInstallation.ini File Parameters and Default Values

You can customize the software by modifying the parameters in this file.

Parameter	Description
`Target Directory`	Default directory created on the client for the Macara Client software components. **Default** `Program Files\Bytemobile\Macara Client`
`Remote Beacon Port`	Required for coordination between the Macara Client and Macara OSN. See Chapter 3, “Preparing the Macara Client Software.” **Default** `4201`
`Remote Beacon IP`	Required for coordination between the Macara Client and Macara OSN: See Chapter 3, “Preparing the Macara Client Software.” **Default** `1.2.3.4`
`Beacon Interval`	Required for coordination between the Macara Client and Macara OSN: See Chapter 3, “Preparing the Macara Client Software.” **Default** `900`
`Beacon Timeout`	For internal use only. **Default** `7`

`Include/Exclude`	List of local / remote IP address pairs to Include/Exclude for optimization. Pairs should be represented as shown below.
	Local Ports:
	`<local IP>:<local port>/<local ip mask>:<local port mask 1>.<local port mask 2>`
	Remote Ports:
	`<remote IP>:<remote port>/<remote ip mask>:<remote port mask 1>.<remote port mask 2>`
	Separate pairs using a semicolon.
	Default `none`

`Include/Exclude Count`	Number of paris specified in the “Include/Exclude” parameter specified above. **Default** `0`
`Include/Exclude Option=`	Setting this parameter to a “0” will include only TCP connections matching one of the above rules for optimization. Setting this parameter to a “1” will include all TCP connections for optimization except those matching one of the above specified rules. **Default** `0`

GLOSSARY

C

This glossary defines terms and acronyms used in the Bytemobile, Inc., user documentation. If a definition includes a term in *italics*, the italicized term is also defined in the glossary.

2.5G — Designates “second-and-a-half-generation” mobile phone networks using *GPRS* over a *GSM* network.

2G — Designates “second-generation” mobile phone networks using *GSM* technology.

3G — Designates “third-generation” mobile phone networks using *UMTS* technology.

Application service — A program that accepts and processes packets on one or more predefined sockets. See also *Kernel service*.

ASP — Applications Service Provider. An ASP serves applications to the user device over the Internet, eliminating the need to install the applications on the device.

BS — Base Station for wireless connection. It is composed of the controller (*BSC*) and the transceiver (*BTS*).

BSC — Base Station Controller. The intelligent controller of the radio equipment in the Base Station *(BS)*.

BTCP — Bytemobile Transmission Control Protocol Stack. A set of mechanisms that transparently address the mismatch between TCP and wireless network characteristics, via rate control, error recovery, congestion control, and host-level optimization.

BTS — Base Transceiver System. This is the radio portion of a Base Station (*BS*).

Bytemobile Network Services Control Center — The graphical user interface used to administer the Bytemobile platform.

Bytemobile Service Control Framework — The Bytemobile Service Control Framework supports services, subscribers, profiles, logs, and statistics, and also provides APIs for both kernel and application-level services. The Service Control Framework is accessed by either a *graphical user interface* or a command-line interface.

Bytemobile Service Control Module	The Service Control Module is the implementation of the *Bytemobile Service Control Framework*. The *GUI* interface and the command-line utilities pass commands and data to the Service Control Module.
CDMA	Code Division Multiple Access. One of several standards used in the U.S. by *2G* mobile phone networks. CDMA provides digital voice and low-speed (~14.4Kbps) data services. See also *GSM* and *TDMA*.
CDPD	Cellular Digital Packet Data. A U.S. standard for transmitting and receiving digital data on *2G* wireless networks.
Classification algorithm	A means of sorting incoming packets to ensure that the packets are handled according to the subscriber profile.
Control service	A *kernel service* that controls the way packets are handled, e.g., CPU and link QoS, and security (authentication, authorization, VPN, etc.).
Data service	An *application service* that may modify the content of a packet or serve application-level requests, e.g., web caching, application proxy, optimization.
ETSI	The European Telecommunications Standards Institute.
FTP	File Transfer Protocol. The *IETF* standard protocol for reliable host-to-host file transfers.
GGSN	The Cisco Gateway GPRS Support Node offers European Telecommunications Standards Institute (ETSI) GPRS features and value-added routing functionality on a single Cisco router platform.
GPRS	General Packet Radio Service. A technology standard for providing data connections at up to 128Kbps on a *GSM*-based wireless network.
GSM	Global System for Mobile communications. One of several standards used by *2G* mobile phone networks. GSM provides digital voice and low-speed (~16Kbps) data services. GSM is used by several wireless carriers in the U.S., and by most wireless carriers in Europe and Asia. See also *CDMA* and *TDMA*.
GUI	Graphical User Interface.
HTTP	HyperText Transfer Protocol. The *IETF* standard protocol for transfer of hypertext objects such as web pages.
HTTPS	Secure HyperText Transfer Protocol (S-HTTP). An extension to *HTTP* that uses encryption and digital signatures to provide security in each transaction.

ICMP Internet Control Message Protocol. The *IETF* standard protocol used to report errors encountered in processing packets, and to perform other internet-layer functions such as diagnostics.

ICP Internet Caching Protocol. The *IETF* standard protocol that enables communication among a hierarchical mesh of proxy caches.

IETF Internet Engineering Task Force. The organization responsible for Internet standards. See *www.ietf.org* for more information about the IETF and Internet standards.

IMAP Internet Message Access Protocol (currently Version 4). A client-server protocol in which email is received and held by your Internet server, allowing for the administrator to manipulate and create folders on the server.

Internet object cache A store of frequently-requested Internet objects in a location closer to the requester than the sources of the objects, thus reducing both the time required to fetch the objects and the traffic generated by fetching the objects across the Internet.

IP Internet Protocol. The *IETF* standard IP protocol is the network-layer protocol in the TCP/IP protocol stack.

IPsec A set of *IETF* encryption and authentication standards (part of IPv6) that enable compliant virtual networking products to share public keys and encryption algorithms.

ISP Internet Service Provider. A provider of Internet access and other Internet-based services to end users.

Kernel service A sequence of operations and associated data structures that are invoked on a packet at a predefined *service insertion point* along the packet-processing path. See also *Application service*.

L2F Layer Two Forwarding. Cisco's protocol for forwarding the authentication and authorization process from an Internet service provider to a server elsewhere on the Internet, such as a corporate central office's server.

L2TP Layer Two Tunneling Protocol is an IETF standard that combines aspects of Microsoft's Point-to-Point Tunneling protocol and Cisco's Layer Two Forwarding (L2F) protocol.

LAN Local Area Network. A short-distance network of client and server computers within a building or campus.

Management service A *kernel service* or *application service* that performs system-specific, subscriber-specific, or network-specific functions such as logging, performance and network monitoring, and network management.

MAPI	Messaging Application Programming Interface. An interface developed by Microsoft that provides messaging functions including addressing, sending, receiving, and storing messages.
MIB	Management Information Base. Defines the objects that can be managed by *SNMP.*
MIME	Multi-purpose Internet Mail Extensions. The standard for multimedia mail, graphics, and sound contents in the Internet suite of protocols. Browsers identify, decode, and launch applications based on registered MIME types by categories and file type extensions, separated by a slash (such as image/gif).
MP3	*MPEG*-1 audio layer 3 is used for transmitting digital music and other sound files over the Internet for playback on PCs, handheld devices, or MP3 players.
MPEG	Motion Picture Experts Group. A widespread compressed file format used for downloading or streaming sound and movie files across the Internet.
NAT	Network Address Translation. The *IETF* standard protocol used when a network's internal IP addresses cannot be used outside the network either because they are invalid for use outside, or because the internal addressing must be kept private from the external network. NAT allows hosts in a private network to transparently communicate with hosts on an external network and vice versa.
NEBS	Network Equipment Building System. A standard promulgated by Bellcore for telco-compatible network equipment.
NIC	Network Interface Card. An adapter card placed in a computer to make a physical connection to a network.
PDSN	Packet Data Serving Node. In a CDMA-based wireless network, a PDSN performs a traffic aggregation function similar to a circuit switch. The PDSN uses AAA (Authentication, Authorization, and Accounting) servers for authentication and traffic management, then forwards traffic to a gateway router at the target IP network.
POP3	Post Office Protocol 3. A client-server protocol in which email is received and held for you by your Internet server, allowing the user to retrieve email from a remote server over an Internet connection.
POSIX	Portable Operating Systems Interface. POSIX is also known as IEEE standard 1003.1. It is yet another flavor of UNIX. Linux systems are POSIX compliant. The pthreads library is a set of C calls that provides a mechanism for writing multithreaded code.

QoS Quality of Service. The capability of a network to provide better service to selected network traffic, e.g., dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved loss characteristics.

RADIUS Remote Authentication Dial-In User Service. A standard for authenticating the identity of remote dial-in users (see *TACACS+*).

Service insertion point The point in the packet-processing path at which a kernel service handles a packet. The *Bytemobile Service Control Framework* supports 32 insertion points.

Service profile A specification that determines the range of capabilities of a service.

Service registration The process of adding a service to the *Service Control Framework.*

Service statistics Management tool for displaying statistics for a service.

SMIv2 Structure of Management Information version 2 defines the mechanisms used for describing and naming objects for the purpose of management. A subset of these objects is a *MIB.*

SMTP Simple Mail Transfer Protocol. The standard protocol used for Internet email messages.

SNMP Simple Network Management Protocol. The *IETF* standard protocol used by network management stations to monitor and control network elements such as hosts, gateways, or servers. A network management station communicates with an SNMP agent running on each network element that is part of the managed network.

SSL Secure Sockets Layer is a transport-layer technology, developed by Netscape, that allows secure transactions among compliant browsers and servers—typically Web servers.

Subscriber The end user of the Bytemobile services, typically a person using a laptop or handheld device with a wireless modem or a mobile phone.

Subscriber profile A specification of the subscriber's preferences among the profiles offered by services to which the subscriber is entitled.

Subscriber registration The process of adding a subscriber to the *Service Control Framework.*

TACACS+ Terminal Access Controller Access Control System Plus. A Cisco standard for authenticating transmissions between servers and databases.

TCP	Transmission Control Protocol. The *IETF* standard transport-layer protocol used for reliable, packet-sequenced host-to-host communication.
TDMA	Time Division Multiple Access. One of several standards used in the U.S. by *2G* mobile phone networks. TDMA provides digital voice and low-speed (~8Kbps) data services. See also *CDMA* and *GSM*.
Transport accelerator	The service that accelerates the delivery of *TCP* packets to and from subscribers. The transport accelerator is one of the default *kernel services* running on the Bytemobile platform.
UDP	User Datagram Protocol. The *IETF* standard transport-layer protocol used for simple, transaction-oriented, host-to-host communication. Delivery is not guaranteed.
URL	Universal Resource Locator. An Internet address specifying the protocol, server name, and even the file name to be downloaded.
UMTS	Universal Mobile Telephone System. The generic designation for *3G* mobile phone services.
VLAN	Virtual Local Area Network. A *LAN* logical broadcast domain segment, created through bridging (switching) software.
VPN	Virtual Private Network. A means of providing private communications over a public network such as the Internet by setting up a secure tunnel between two end-points.
WAP	Wireless Application Protocol. A protocol stack that provides access to the Internet from wireless devices such as mobile phones, pagers, and PDAs. Currently, only available on *CDMA* networks.
WCDMA	Wideband *CDMA*. An emerging standard for *3G* mobile phones. Currently being tested in Japan, WCDMA can provide data connections at up to 384Kbps.

Bytemobile, Inc.

2029 Stierlin Court

Mountain View, CA 94043

U.S.A.

Phone: +1 650 641 7700

Fax: +1 650 641 7701

www.bytemobile.com

support@bytemobile.com

Made and Printed in U.S.A.

001.003.DOC.0121.SUX.ENG