A Sierra Monitor Company

Utility User Manual

Ruinet

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after May 1, 2001

Version: 1.10
Rev. 10

TABLE OF CONTENTS

Table of Contents

1. General Overview

The Ruinet Utility is used to:

- Transfer files (configuration, firmware, etc) to and from a FieldServer
- Monitor a working FieldServer's internal data and parameters
- Change or update a FieldServer's internal data and parameters
- Delete files on a FieldServer
- Restart a FieldServer

1.1. PC Requirements

1.1.1. Hardware

Ruinet works over an Ethernet network so a TCP/IP enabled PC with a network card is required. The network card must support 10Mbit/s Ethernet.

The PC and FieldServer can either be connected to an established network or connected directly using a cross-over cable.

1.1.2. Software

Ruinet runs under any of the following operating systems:
DOS, Windows 95 (SR2 upwards), Windows 98, 2000, NT, XP

1.2. Installation and Setup

Ruinet is distributed on the FieldServer software installation media sent with the FieldServer. The Ruinet PC's TCP/IP settings may have to be changed to successfully connect to a specific FieldServer if there is more than one FieldServer on the network. Ruinet uses Internet Protocol to connect to a FieldServer and therefore the Ruinet PC and the FieldServer have to be setup with an IP address on the same subnet (e.g. IP addresses 192.168.0.10 and 192.168.0.20 are on the same subnet.).

If a PC is used on an already established network, it is better to change the FieldServer's IP address than the PC's IP address. Please refer to section **Error! Reference source not found.** to change the FieldServer's IP address.

2. Remote User Interface (RuiNet)

The notes in this section explain how to navigate the menu system provided by RuiNet and provide information on the contents of the various screens.

2.1. A – Connecting to a FieldServer

Since RuiNet can only work with one FieldServer at a time, it is necessary to target the FieldServer of interest. (Note that it is possible to run multiple instances of RuiNet at the same time). If RuiNet is run without specifying a target it will provide a list of the FieldServers on the network. Pick the required FieldServer from the list -

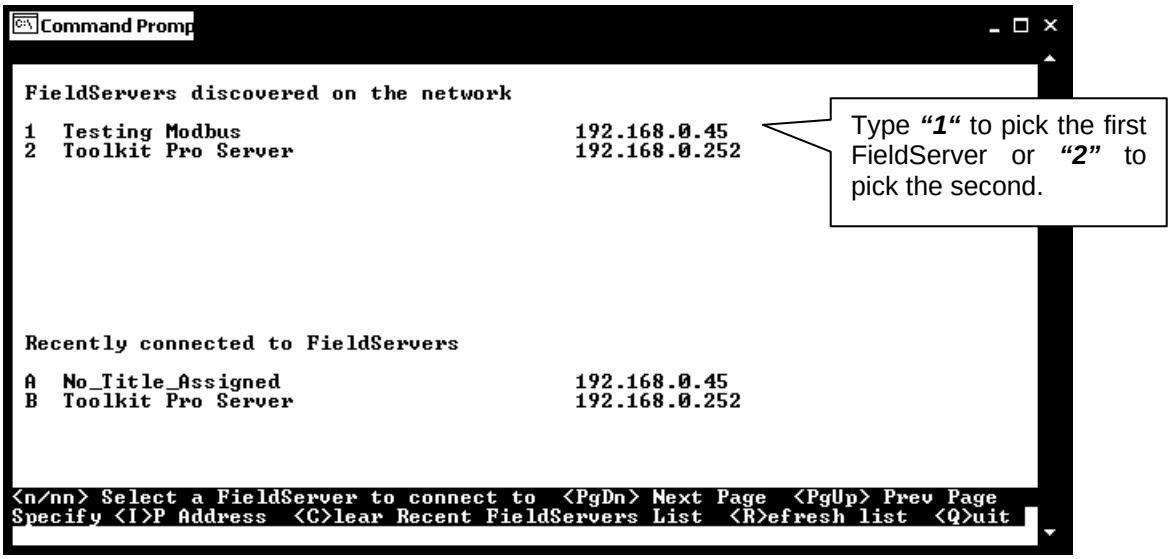


Figure 2.1

If a FieldServer is not selected, RuiNet will automatically connect to the FieldServer last connected to after about 10seconds.

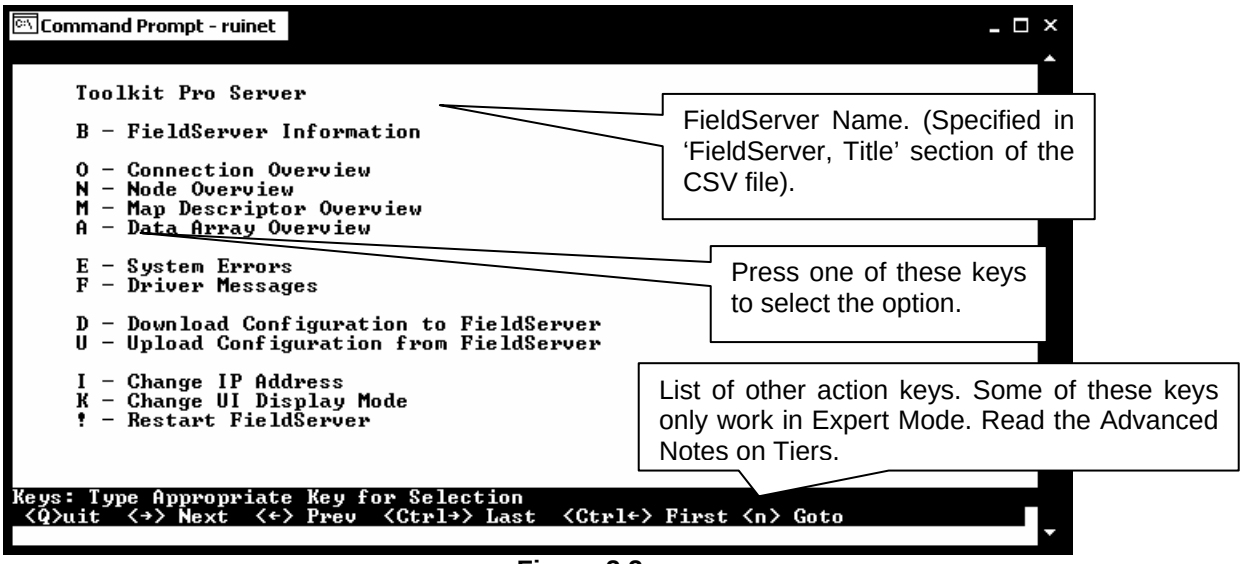


Figure 2.2

2.2. B - FieldServer Information

FieldServers were previously known as bridges, hence the use of the B key to access the screen displaying general information about a FieldServer. The B Screen consists of two aspects (settings and status). Use the spacebar to toggle between the two aspects. The settings screen is depicted below. The various parameters are described more fully in the tables that follow.

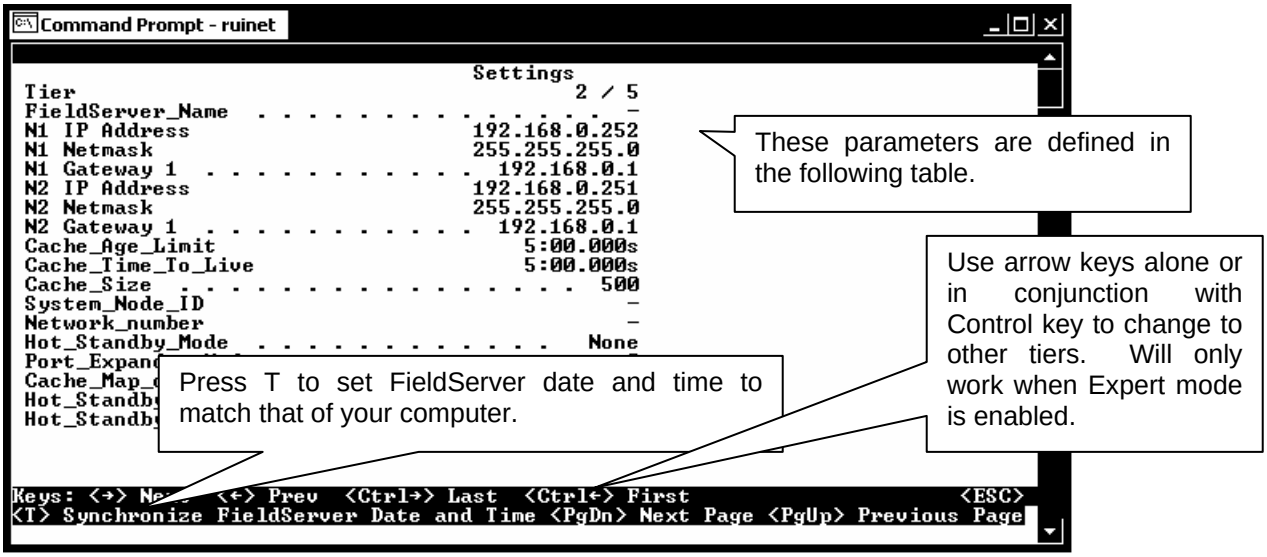


Figure 2.3 – FieldServer Information – Settings Aspect.

2.2.1. FieldServer Information – Settings Aspect

Parameter	Description
Tier	FieldServers have the ability to run as "multiple" FieldServers on one platform. To differentiate between the different running applications, each of the applications is referred to as a Tier with a specific name.
FieldServer_Name	A name by which a FieldServer is identified - need not be unique.
N1 IP Address	The IP address of the N1 Ethernet Adapter.
N1 Netmask	Netmask of N1 Ethernet Adapter.
N1 Gateway 1	The IP address of the gateway that N1 Ethernet messages use if the destination IP is not found on the local network.
N2 IP Address	The IP address of the N2 Ethernet Adapter.
N2 Netmask	Netmask of N2 Ethernet Adapter.
N2 Gateway 1	The IP address of the gateway that N2 Ethernet messages use if the destination IP is not found on the local network.
Cache_Age_Limit	Maximum age of data in a cache Map Descriptor for immediate response to poll. Default 5 minutes. See **Error! Reference source not found.**
Cache_Time_To_Live	The time that the FieldServer maintains the port expanded polling in Port Expander Mode. Default 5 minutes. See Configuration Manual for more information.
Cache_Size	The maximum number of cache Map Descriptors allowed.
System_Node_ID	Use is driver dependent. Generally used to identify the FieldServer as a node when it is configured as a server.
Network_number	Displayed where a protocol requires the FieldServer to be assigned a network number (e.g. BACnet).
Hot_Standby_Mode	Where specified, this parameter defines the behavior of the standby FieldServer in Hot Standby mode. In Mode 1 the FieldServer is completely passive; in Mode 2 the standby FieldServer polls the connected devices through alternate communication paths. Refer to the Configuration Manual for more information.
Port_Expander_Mode	Indicates whether the port expander mode function is enabled or not.
Cache_Map_Descriptor_ Scan_Interval	Default is two seconds. If the value 65535 is displayed, then this is an error and it indicates that there is no setting.
Hot_Standby_Designation	Primary or Secondary. On boot the primary tries to become the active and the secondary tries to become the standby FieldServer. This behavior may be different if the so called secondary FieldServer gets re-booted first.
Hot_Standby_Pair_Name	A name by which a pair of FieldServers configured as a Hot Standby pair is known. When one of a pair boots, it broadcasts a message with its pair name in order to try and locate the other FieldServer that forms the hot standby pair.

2.2.2. FieldServer Information – Status Aspect

Parameter	Description
Tier	FieldServers have the ability to run as "multiple" FieldServers on one platform. To differentiate between the different running applications, each of the applications is referred to as a Tier with a specific name.
Driver Configuration	The part of the FieldServer firmware that contains the drivers ordered. Each combination of drivers is known as a DCC.
DCC Version	A DCC version number is allocated to each DCC. This version increases with changes/updates to drivers. Tech support are able to track a DCC version to determine what features of each driver are available and what bugs may have been present in a particular version.
Kernel Version	The version number of the kernel. The kernel is that part of the firmware that provides support and resources to the individual drivers. Tech support may require this number.
BIOS Version	The version number of the FieldServer's BIOS. This seldom changes.
Data Points Used	Each FieldServer has a combination of drivers and a maximum number of data points that may be managed. A data point is an element of a Data Array with a responsible Map Descriptor. Responsible Map Descriptors are client side, active and almost always read Map Descriptors. Example: A CSV file configured with a RDBC Map Descriptor with a length of 100 may use 100 data points. If the number of points used exceeds the maximum then the FieldServer will continue to operate for 24 hours and then shutdown. See Enote024 for details on point count. The file slots.ini controls the maximum number of data points. Only use slots.ini supplied with the FieldServer or by tech support. The user cannot edit or generate this file. The default for an X40 is 1000 points. The default for an X20 is 500 points. The FieldServer bridge ID must be provided to tech support for them to generate a new slots.ini file.
Data Points Max	
Cycles Now	Number of times the FieldServer executes all its software per second. This number will change continuously.
Cycles Max	The maximum value since the FieldServer started.
Cycles Min	The minimum value since the FieldServer started.
Avg Cycle Time	The average time in milliseconds the software took to complete since last restart. Cycle timers are only started after the CSV files have been loaded.
Min Cycle Time	The minimum/maximum time in milliseconds the software took to complete. Use the R key to reset this value.
Max Cycle Time	
Cache Age Ave[1]	The average data age of the cache blocks currently in existence.
Cache Age Max[1]	The maximum data age of the cache blocks currently in existence
Cache Age Max Ever[1]	The maximum data age of cache blocks that existed on the FieldServer since startup, i.e. the oldest that cache data ever got.
Cache usage (RDB)[1]	The number of active cache blocks reading data at the current time.
Cache usage (WRB)[1]	The number of active cache blocks writing data at the current time.
Memory Blocks	The number of memory blocks reserved by drivers and other system functions.
Last Time Rebooted	The time that the FieldServer was last restarted. New FieldServers are

[1] Cache blocks are temporary Map Descriptors created by the FieldServer to achieve certain objectives such as a write through or port expansion. Because they are temporary this number rise and fall is not visible. The read cache blocks persist until they expire after the cache_age_to_live time has expired.

Parameter	Description
	shipped without the time or date set and hence the value shown here is meaningless until the FieldServer time is synchronized with the computer's.

2.3. O - Connection Overview

This is one of the most important information screens provided by RuiNet. It supplies information on communication between the FieldServer and remote devices. A number of aspect screens are available, and some of the aspect screens have more than one page. Use the space bar to toggle between aspects and the PgUp, PgDn keys to toggle between pages of the same aspect. The Connection Overview and Settings Aspect screens are depicted below. The various parameters are described more fully in the tables that follow.

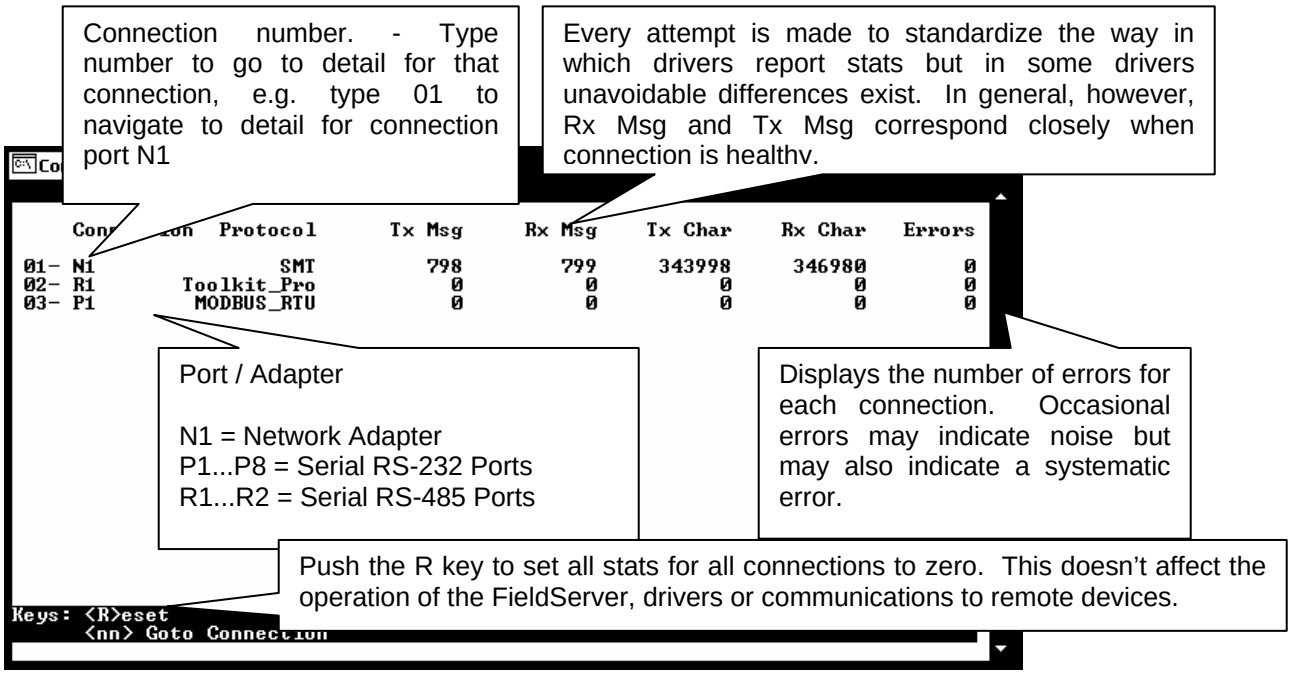


Figure 2.4 – Connection Overview Screen

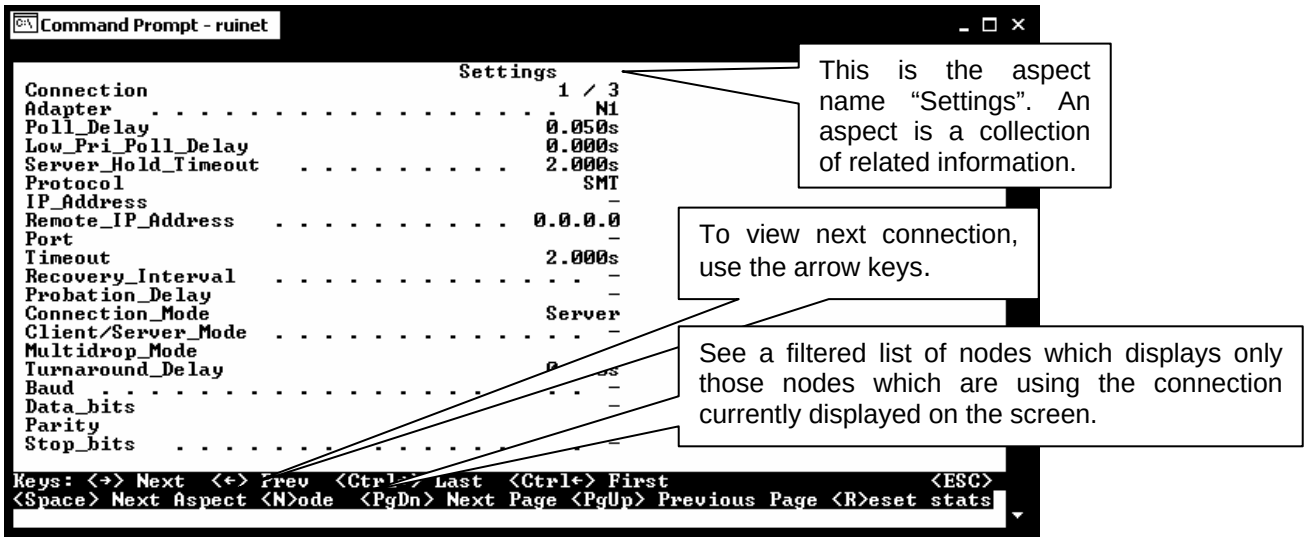


Figure 2.5 – Connection Overview – Settings aspect – Page 1

2.3.1. Connection Overview – Settings Aspect.

Parameter	Description
Connection	The current connection out of the total number of connections.
Adapter	An adapter ID is displayed if the driver is an adapter driver.
Poll_Delay	The minimum amount of time that must pass between one Client Map Descriptor completing its task and the next Client Map Descriptor being serviced. Refer to Driver Manuals.
Low_Pri_Poll_Delay	The poll delay used for lower priority Map Descriptors.
Server_Hold_Timeout	When an upstream device polls the FieldServer, and the data is unavailable or too old, the driver generates a poll to the downstream device for fresh data, (port expansion). The Server_Hold_Timeout defines the time available for this transaction to complete before an error is returned. The default is 2 seconds.
Protocol	The name of the protocol used by this connection.
IP_Address	An IP address for the connection if applicable.
Remote_IP_Address	A remote IP address for the connection if applicable.
Port	Identifies the port for a serial driver. P1... P8 or R1...R2
Timeout*	The timeout defined for the connection. Default 2 seconds. See **Error! Reference source not found.** for further information.
Recovery_Interval	The time after a node goes off-line before the driver tries to poll the device again. Default 30 seconds.
Probation_Delay	The length of time communication needs to be re-established for before an offline Client node is marked on-line again. Default 1 minute.
Connection_Mode	Server (passive) or client (active).
Client/Server_Mode	Optional setting to force a connection mode. Possible values are Server, PLC, Hot_Standby, Hot_Standby_Data, Client_Only, Diagnostic.
Multidrop_Mode	Indicates whether Multidrop mode is enabled or not. Refer to **Error! Reference source not found.** for more information.
Turnaround_Delay	The delay that the driver imposes between receiving a poll and sending a response. The default is 5ms for serial drivers.
Baud	Connection baud rate used by serial drivers. Some drivers override this value because the communication protocol allows the baud rate to be changed by the master.
Data_bits	The number of data bits used for serial communication by the connection. The values are 7 or 8.
Parity	The parity mode used by the connection to detect communication errors. Values are even, odd or none. The default is none.
Stop_Bits	The number of stop bits used for serial communication by the connection. The values are 1 or 2.
Line_drive_on	Time a serial driver using RS-485 will wait before driving the communications line after receiving permission. Default is 1milisecond.
Line_drive_off	Time that a communication line using RS-485 is actively driven before being released, after the last bit has been sent. Default 1milisecond..
IC_timeout	Time a driver will wait between receiving the first and second bytes of a message before generating an IC Timeout. Default 0.5 seconds. See **Error! Reference source not found.** for more information.
IP_port	Determined by specific driver or protocol used. See Driver Manual.
Remote_IP_Port	
Max_Master	
Max_Info_Frames	
Connection_Type	

Parameter	Description
Application	

2.3.2. Connection Overview - Status Aspect

Parameter	Description
Connection	The current connection being displayed out of the total number.
Timer	The use of this variable to determine timeouts is driver dependent and has no consistent meaning. In passive drivers this variable may not have been used at all. In simple poll response drivers this variable should decrease from its maximum (equal timeout value) to zero.

2.3.3. Connection Overview - Statistics Aspect

Connection statistics are a roll-up of all the statistics maintained by the Nodes and Map Descriptors using the connection. For example: If three Map Descriptors use a single connection then each time a message is sent for each Map Descriptor, the Map Descriptor, Node and connection statistics are all updated. Thus response time statistics are more meaningful when viewing individual nodes and Map Descriptors.

Parameter	Description
Connection	The current connection being displayed out of the total number
Client Read Msg sent	The number of read messages sent by a driver acting as a Client. May include messages sent to connect to the Server.
Client Read Msg recd	The number of responses received to read messages sent by a driver acting as a Client. For most drivers this statistic quals the number of messages sent.
Client Write Msg sent	The number of write messages sent by a driver acting as a Client. May include messages sent to connect to the Server.
Client Write Msg recd	The number of responses received to write messages sent by a driver acting as a Client. For most drivers this statistic equals the number of messages sent.
Client Passthru messages	This statistic relates to port expansion. Messages that are unrecognized (unsupported) are passed through the FieldServer without the contents being considered.
Client Passthru Msg sent	The number of unrecognized messages passed through the FieldServer.
Client Passthru Msg recd	The number responses to unrecognized messages passed through the FieldServer.
Client Broadcast msg	The number of broadcast messages sent.
Client Bytes Sent	Number of bytes contained in messages sent by the driver acting as a Client. May include bytes of messages used to connect or login to the remote device.
Client Bytes Recd	Number of bytes contained in messages received by the driver when acting as a Client - typically responses to messages sent. May include bytes of messages used to connect or login to the remote device.
Server Msg recd	The number of messages received by a driver acting as a Server. May include non-data messages such as connection or login requests and port supervision messages.
Server Msg sent	The number of messages sent by a driver acting as a server - typically responses to messages received. May include responses to non-data messages as above.
Server Bytes Sent	A count of the bytes sent by the driver when acting as a Server in response polls. May include bytes contained in non-data messages.
Server Bytes Recd	A count of the bytes received by the driver when acting as a server. May include bytes contained in non-data messages.
Cache - Hits	The number of times a cache Map Descriptor containing current data has been polled. See **Error! Reference source not found.**
Cache - Misses	The number of times a cache Map Descriptor containing outdated data has been polled. See **Error! Reference source not found.**

Parameter	Description
Cache - Created	The number of times a cache Map Descriptor containing outdated data has been polled resulting in the creation of a new cache Map Descriptor. See **Error! Reference source not found.**
Cache - Bumped	The number of expired cache blocks. See **Error! Reference source not found.**
Pex Write Thru	Writes are never cached - the external device is put on hold while the FieldServer resends the message to the PLC and waits for the response. When the FieldServer receives the response from the PLC it in turn responds to the external device. This operation is counted as a PEX Write thru.
Server Response Max	The maximum time that the FieldServer has taken since the last reset to receive a message from an external device, poll the PLC and respond to the external device. The SCADA Hold Timeout parameter should be set higher than this limit
Server Response Avg	This is the average time that the FieldServer has taken since the last reset to receive a message from an external device, poll the PLC and respond to the external device. Reconfigure if response times are slow.
Link Control	BACnet specific – see Driver Manual.
Who-Is-Router-To-Network	
I-Am-Router-To-Network	
Reject-Msg-To-Network	
Messages Reconstructed	Metasys Specific – see Driver Manual.
Unsupported Property	The driver encountered an unsupported property in a message.
Unsolicited Messages Recd	A message was received without the driver polling for it.
Single Write	A write data message containing a single data element.
Single Item Read	A read data message requesting a single data element.
Block Write	A write message containing a block of data elements.
Block Read	A read message requesting a block of data elements.
Sequence Error	Messages containing sequence numbers received out of sequence.
Data Object Startup	Data requested from a node in start-up mode.
Expedite Read	The number of Map Descriptors given the highest priority to complete a read/write first.
Expedite Write	
Fasttrack Read	The number of Map Descriptors given a higher priority to complete a read/write as soon as possible.
Fasttrack Write	
Fasttrack Overrun Read	The number of times the buffer holding fasttrack read/write Map Descriptors overflowed.
Fasttrack Overrun Write	
Max Read Response Time	Maximum/Minimum/Average time in seconds that passed before a response was received to a read message sent by a driver acting as a Client.
Min Read Response Time	
Avg Read Response Time	
Max Write Response Time	Maximum/Minimum/Average time in seconds that passed before a response was received to a write message sent by a driver acting as a client.
Min Write Response Time	
Avg Write Response Time	
Max Passthru Response Time	Maximum/Minimum/Average time in seconds to receive a response to a poll sent using the Passthru mechanism.
Min Passthru Response Time	

Parameter	Description
Avg Passthru Response Time	
TCP Conn Lost	Number of times a TCP connection was lost or interrupted.
TCP Send Failed	Number of times a message sent on TCP connections failed.

2.3.4. Connection Overview - Error Statistics Aspect

Parameter	Description
Connection	The current connection being displayed out of the total number.
PEX No slave	Pex mode tried to create a cache block and failed - normally a failed write-through. See Driver Manual for further information
Server Overruns	A message arrived from the upstream device while the server port was on hold. Increase the timeout setting in the external device.
Server Hold Timeouts	If an upstream device requests data from a cache where the data is too old then the data will be refreshed by polling from the downstream device. The response was not received in time.
Timeouts	A remote device never responded to the FieldServer poll. Either the device is not responding, or one of the timeout parameters is set too low. Ensure that the device is online and addressed correctly and if necessary increase the relevant timeout parameter in the CSV file.
Checksum Errors	External influences e.g. electrical noise corrupted the data. Check that communication cables are shielded, not too long and do not run past power cables.
Protocol Errors	An external device responded with unexpected or unknown messages. Consult the relevant driver manual.
Noise	Corrupted or garbage bytes on a communications line.
Bad Length	A message that looks Ok, but is of the incorrect length.
Bad Node	A no-response/error response from an addressable external device.
Bad Function	E.g. an external device is written to that does not support writes.
No Start	Communications to an external device could not be started.
PLC exception	A variation of a NAK message produced by some protocols.
NAK	A NAK message is received in response to a poll.
Streaming	Data seems to be continuously arriving from an external device. This could be due to a mismatch in baud rates. Streaming errors are typically produced when: 1) The transmitter sends more data than the FieldServer can process. 2) An unexpectedly long message is received or messages have been corrupted so that the end of a previous message cannot be detected. 3) A message longer than the driver expects has been received.
Premature	A response from an external device arrived before it was expected, implying that it is not the response for the poll the FieldServer has just sent. This could occur when there are time delays in communication networks which contain bridges and routers which may delay messages for longer than expected. Increase the timeout parameter for the connection to eliminate these errors.
Preamble	Characters preceding a message were dropped.
IC Timeouts	Too much time between receiving successive bytes in a message.
Address Errors	A driver tried to address a wrong device or a wrong block of data within an external device.
Data Object offline	A poll was received for a node that is offline.
Node Offline	An external device node is offline in response to a driver or

Parameter	Description
	external device trying to access it.
Msg Ignored	Messages received but unable to be processed - normally because the driver has not implemented functionality for that message.
Sys Cleared	The number of times that Data Arrays were cleared after a system-normal type message was received from a device (e.g. fire alarm panel)
Squelch TX[2]	The number of bytes received during the squelch timing period started when RTS is asserted.
Squelch RX[1]	The number of bytes received during the squelch timing period started when RTS is de-asserted.
Cache Failed	FieldServer unable to create a cache block due to memory shortage or inability to find a downstream node.
Segmentation Not Supported	The received message was segmented but the driver does not support re-assembling segmented messages.
Passthru Retries[3]	Produced when a Passthru poll is busy on the downstream side, and an identical poll (retry) is received on the upstream side.
Passthru Overruns[2]	Produced when a Passthru poll is busy on the downstream side and a different poll (not a retry) is received on the upstream side.
Passthru Early Retries	A Passthru Retry, which occurs when the upstream retry is received before the 1st downstream Passthru poll has been sent.
Passthru Normal Retries	A Passthru Retry, which occurs when the upstream retry is received after the 1st downstream Passthru poll has been sent.
Passthru Early Overruns	A Passthru Overrun, which occurs when the upstream retry is received before the 1st downstream Passthru poll has been sent.
Passthru Normal Overruns	A Passthru Overrun, which occurs when the upstream retry is received after the 1st downstream Passthru poll has been sent.
Passthru Early Overruns Fails	This statistic is currently not used.
Passthru Normal Overruns Fails	
PWT Expired	The Passthru Window Timer expired before the downstream Passthru poll could be sent.

2.3.5. Connection Overview - API Aspect

These statistics are produced by the FieldServer kernel's API (Application Programmer Interfaces). They are intended for advanced users only.

Press the 1 key or the 2 key to change the filter on these stats.
1 = TCP API
2 = Ethernet API (Default)

[2] Information is available in ENOTE19. Enotes are available on the Web at www.fieldserver.com.
[3] Currently only produced by the Metasys driver

Parameter	Description
Connection	The current connection being displayed out of the total number
TX bytes	Number of bytes sent.
TX packets	Number of packets sent.
TX dropped	Number of packets not sent for reasons that include the transmit buffers being full.
TX errors	Errors during transmission such as hardware errors.
RX IP fragments	The number of IP fragmented packets received.
Collisions	Number of collisions - the network is too busy.
TX abort errors	The NIC layer reports this error. Should be zero.
TX carrier errors	
TX heartbeat errors	
TX window errors	
Max TX buffers used	The highest value for the used transmit buffer count since the FieldServer was restarted. An extremely busy network may have a max of 7 or 8 but values this high are generally not expected.
Tx buffers in use	Current transmit buffers in use.
Tx broadcast	The number of broadcast messages sent.
RX bytes	Number of bytes received by the API.
RX pkts total	Number of packets received by the API.
RX pkts for us	The number of received packets destined for transmission.
RX pkts NOT for us	The number of received packets destined for transmission not meant for the API.
RX dropped	The number of received packets which were dropped for reasons such as the Ethernet queue being full.
RX errors	The NIC layer reports this error. Should be zero.
RX length errors	
RX overflow errors	
RX crc errors	
RX frame errors	
RX frame errors	
RX buffer full	
RX ring buffer error	
RX IP type	Number of messages using IP protocol received.
RX ARP type	Number of messages using ARP protocol received.
RX BACNET type	Number of messages using BACnet protocol received.
RX 802_3 type	Number of messages using 802_3 Ethernet protocol received.
RX UNKNOWN type	Number of messages using other Ethernet protocol received.
Frag buf overrun	IP defrag on the receive side. The buffer is (about 80kb) is full.
RX Broadcast	Number of packets received in broadcast.
Exception 1	The Ethernet packet type could not be identified
Exception 2	An Ethernet packet received (i.e. addressed to the FieldServer Ethernet address) was addressed to another IP address - either an Ethernet broadcast was sent with a specific IP address, or another device has an incorrect ARP table.
Exception 3	A UDP packet of length >1500 was received which cannot be handled by the FieldServer.
Exception 4	The Ethernet TX interrupt handler was kick started.
RX global timeout	This statisticis currently not used.
RX local timeout	

Parameter	Description
RX IP not for us	The number of received IP packets not meant for the FieldServer.
Max RX buffers used	The maximum number of receive buffers used since restart.
Rx buffers in use	The number of receive buffers currently in use.

2.4. N - Node Overview

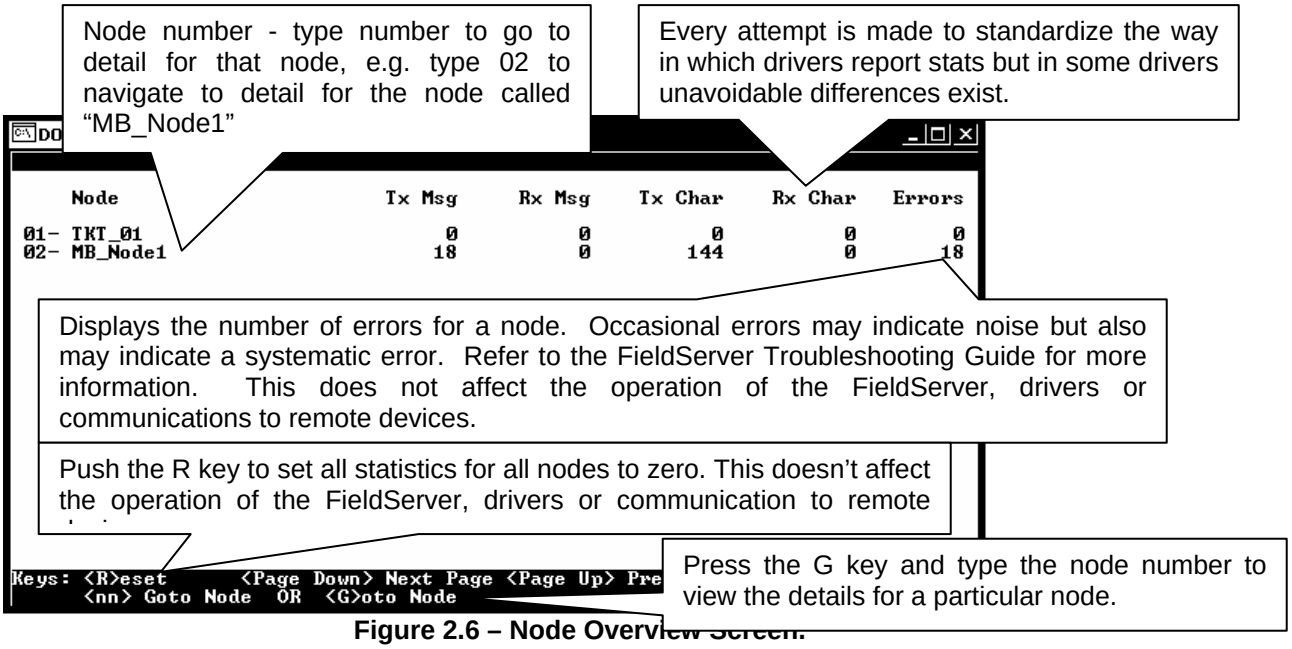


Figure 2.6 – Node Overview Screen

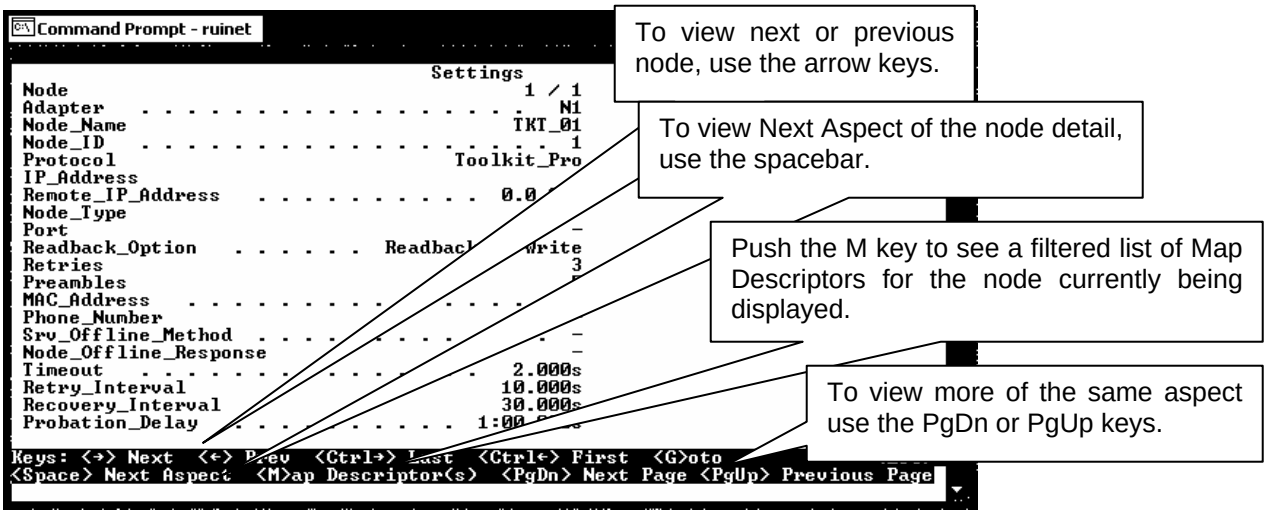


Figure 2.7 – Node Overview – Settings Aspect Page 1

2.4.1. Node Overview - Settings Aspect

Parameter	Description
Node	The current node being displayed out of the total.
Adapter	The adapter name where a node is connected to an adapter.
Node_Name	The node name specified in the CSV file.
Node_ID	The station number or address of the node. The actual meaning is dependent on the driver and protocol – refer to the Driver Manual.
Protocol	The protocol being used to update the data for that node. Refer to the Driver Manual
IP_Address	The IP address used by this node.
Remote_IP_Address	The remote IP address used by this node
Node_Type	Specified in the configuration file as the PLC_Type. - Consult the driver manual for additional information.
Port	Port number for a serial connection.
Readback_Option	After a write has been executed in a write through operation, the FieldServer schedules the read Map Descriptor to poll again in order to immediately read back the data that was written. The default is 'Readback_on_Write". Other options are "None" and "Expire_Current_Data"
Retries	Tells the driver how many times to retry a poll before considering the node to be offline. The default is 3.
Preambles	Counts data bytes received before a valid message, but not forming part of a valid message, e.g. a message fragment.
MAC_Address	Currently this field is not used at all. It is intended to allow drivers to resolve an IP address by giving a MAC address.
Phone_Number	This field is intended for modem support. Currently disabled.
Srv_Offline_Method	The method used on the node to decide if it is to be considered offline.
Node_Offline_response	The type of response the server side of the driver sends when it finds the downstream node to be offline.
Timeout	The timeout specified for the node. Refer to **Error! Reference source not found.**.
Retry_Interval	The amount of time in seconds that the driver should wait before retrying a poll after a timeout has occurred
Recovery_Interval	The time in seconds after a node goes off-line before the driver tries to poll the device again. – default 30seconds.
Probation_Delay	The length of time communication needs to be re-established for before an offline Client node is marked on-line again. Default 1 minute.
Network_Number	Network station number used on this node.
Server_Name	An alternate to specifying the IP address. Typically used when the user wants two nodes to talk to each other. When specified, the FieldServer sends out a broadcast with the server name and uses the reply to fill in the IP address for the node. Until the reply has been received all polling for the node is disabled. The server name given should correspond to the pair_name specified in the remote FieldServer's bridge settings.
Alias_Node_ID	This is used to distinguish between different nodes connected to the FieldServer when a PLC does not support the allocation of different None_ID's. Each node is given a different alias. Upstream devices poll the Alias_Node_ID and the FieldServer routes the poll to the correct PLC which is polled using the Node_ID.
Ports_on_PLC	For hot standby operation. This field is used to control which port on a

Parameter	Description
	PLC to poll.

2.4.2. Node Overview - Status Aspect

Parameter	Description
Node	The current node being displayed out of the total.
Node Status	For a client node. Online, offline, disabled or probation. Probation means that the node is in transition from offline to online. The node was offline, a poll has succeeded but the probation timeout has not expired so the node has not been returned to online yet. If the node isn't a client node then it is reported as server.
Node Mode	Client or Server
Retry State	The state number of the node retry state engine – for FieldServer developers.
Used Retries	The total number of retries since start-up
Recoveries	The number of times that the driver has gone from offline to online.
Active R/W on Startup	This is an important indication if the driver appears not to be polling. Displays yes or no. If a Map Descriptor with function = "ARS" (Active Read on Startup) is found then this field reports as 'yes'. ARS Map Descriptors are scheduled to occur when a node is still offline and are only used once. They are intended to establish a connection or to log into a remote device. No other Map Descriptors are polled until the ARS Map Descriptors complete normally.

2.4.3. Node Overview - Operating Statistics Aspect

Node statistics are a roll-up of all the statistics maintained by the Map Descriptors which belong to the node. For example, if three Map Descriptors belong to a single node, then each time a message is sent for each Map Descriptor, the statistics for the Map Descriptor, the node and the connection are updated.

Parameter	Description
Node	The current node being displayed out of the total number of nodes.
Client Read Msg sent	Refer to Section 2.3.1 for a description. On this screen the statistic count applies to the node only.
Client Read Msg recd	
Client Write Msg sent	
Client Write Msg recd	
Client Passthru Msg sent	
Client Passthru Msg recd	
Client Broadcast msg	
Client Bytes Sent	
Client Bytes Recd	
Server Msg recd	
Server Msg sent	
Server Bytes Sent	
Server Bytes Recd	
Cache - Hits	
Cache - Misses	
Cache - Created	
Cache - Bumped	
PEX Write thru	
Server Response Max	
Server Response Avg	
Link Control	
Messages Reconstructed	
Unsupported Property	
Unsolicited Messages Recd	
Single Write	
Single Item Read	
Block Write	
Block Read	
Sequence Error	
Data Object Startup	
Expedite Read	
Expedite Write	
Fasttrack Read	
Fasttrack Write	
Fasttrack Overrun Read	
Fasttrack Overrun Write	
Max Read Response Time	
Min Read Response Time	
Avg Read Response Time	
Max Write Response Time	
Min Write Response Time	
Avg Write Response Time	
Max Passthru Response Time	
Min Passthru Response Time	
Avg Passthru Response Time	
TCP Conn Lost	

Parameter	Description
TCP Send Failed	

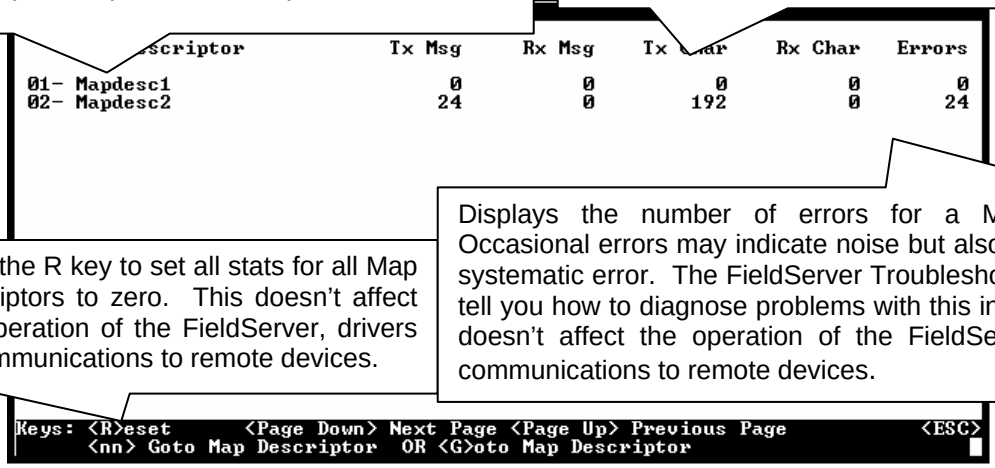
2.4.4. Node Overview - Error Statistics Aspect

Parameter	Description
Node	The current node being displayed out of the total number of nodes.
PEX No slave	Refer to Section 2.3.3 for a description. On this screen the count applies to the node only.
Server Overruns	
Server Hold Timeouts	
Timeouts	
Checksum Errors	
Protocol Errors	
Noise	
Bad Length	
Bad Node	
Bad Function	
No Start	
PLC exception	
NAK	
Streaming	
Premature	
Preamble	
IC Timeouts	
Address Errors	
Data Object offline	
Node Offline	
Msg Ignored	
Sys Cleared	
Squelch TX	
Squelch RX	
Segmentation Not Supported	
Passthru Retries	
Passthru Overruns	
Passthru Early Retries	
Passthru Normal Retries	
Passthru Early Overruns	
Passthru Normal Overruns	
Passthru Early Overrun Fails	
Passthru Normal Overrun Fails	
PWT Expired	

2.5. M – Map Descriptor Overview

Type number to go to detail for that Map Descriptor - e.g. Type 02 to navigate to detail for the Map Descriptor called "MapDesc2"

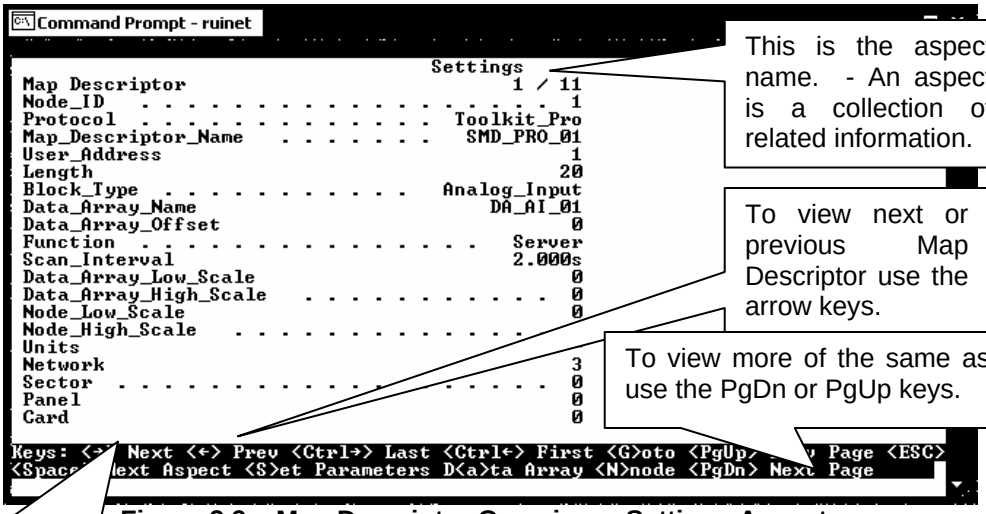
Every attempt is made to standardize the way in which drivers report stats but in some drivers unavoidable differences exist.



Displays the number of errors for a Map Descriptor. Occasional errors may indicate noise but also may indicate a systematic error. The FieldServer Troubleshooting Guide will tell you how to diagnose problems with this information. This doesn't affect the operation of the FieldServer, drivers or communications to remote devices.

Push the R key to set all stats for all Map Descriptors to zero. This doesn't affect the operation of the FieldServer, drivers or communications to remote devices.

Figure 2.8 – Map Descriptor Overview Screen



This is the aspect name. - An aspect is a collection of related information.

To view next or previous Map Descriptor use the arrow keys.

To view more of the same aspect use the PgDn or PgUp keys.

Figure 2.9 – Map Descriptor Overview – Settings Aspect

To view Next Aspect of the node detail, use the spacebar.

2.5.1. Map Descriptor Overview - Settings Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total
Node_ID	The Node ID used by this Map Descriptor when the driver builds read or write messages..
Protocol	The protocol used by this Map Descriptor.
Map_Descriptor_Name	Used to identify a Map Descriptor by name.
User_Address	Allows a Map Descriptor to address remote device data at a specific start memory location.
Length	Alows a Map Descriptor address a number of remote device data locations from the start address.
Block_Type	Used by some drivers to indicate the data format used to pack a block of data, e.g. when reading a block of 4 bytes from a PLC and the Block_Type is Word, the incoming data will be interpreted as 2 words.
Data_Array_Name	The name of the Data Array where information will be stored to and retrieved from by the Map Descriptor.
Data_Array_Offset	The offset into the Data Array where data should be stored on reads or retrieved from on writes.
Function	The Map Descriptor function can be mainly read or write with a number of variations of each. Refer to the FieldServer Manual for other functions.
Scan_Interval	When using continuous Map Descriptor functions such as RDBC, this is the time a Map Descriptor will wait before polling for data again.
Data_Array_Low_Scale	Used in the scaling of data values before storing them or before sending them in write messages. Consult the Driver manual to determine whether the driver supports scaling.
Data_Array_High_Scale	
Node_Low_Scale	
Node_High_Scale	
Units	Used to specify engineering units to interpret data if used. Will display a dash if not used.
Network	Used by some drivers as a network number.
Sector	Used by some drivers as a sector number for rack addressing.
Panel	Used by some drivers as a panel number for rack addressing.
Card	Used by some drivers as a card number for rack addressing.

2.5.2. Map Descriptor Overview – Status Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total.
Scan Timer	Value of scan timer used by driver for this Map Descriptor. This is for debugging purposes only and has no specific meaning.

2.5.3. Map Descriptor Overview - Operating Statistics Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total number of Map Descriptors.
Client Read Msg sent	Refer to Section 2.3.1 for a description. On this screen the statistic count applies to the Map Descriptor only.
Client Read Msg recd	
Client Write Msg sent	
Client Write Msg recd	
Client Passthru Msg sent	
Client Passthru Msg recd	
Client Broadcast msg	
Client Bytes Sent	
Client Bytes Recd	
Server Msg recd	
Server Msg sent	
Server Bytes Sent	
Server Bytes Recd	
Cache - Hits	
Cache - Misses	
Cache - Created	
Cache - Bumped	
PEX Write thru	
Server Response Max	
Server Response Avg	
Link Control	
Messages Reconstructed	
Unsupported Property	
Unsolicited Messages Recd	
Single Write	
Single Item Read	
Block Write	
Block Read	
Sequence Error	
Data Object Startup	
Expedite Read	
Expedite Write	
Fasttrack Read	
Fasttrack Write	
Fasttrack Overrun Read	
Fasttrack Overrun Write	
Max Read Response Time	
Min Read Response Time	
Avg Read Response Time	
Max Write Response Time	
Min Write Response Time	
Avg Write Response Time	
Max Passthru Response Time	
Min Passthru Response Time	
Avg Passthru Response Time	

Parameter	Description
TCP Conn lost	
TCP Send Failed	

2.5.4. Map Descriptor Overview - Error Statistics Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total number of Map Descriptors.
PEX No slave	Refer to Section 2.3.3 for a description. On this screen the statistic count applies to the Map Descriptor only.
Server Overruns	
Server Hold Timeouts	
Timeouts	
Checksum Errors	
Protocol Errors	
Noise	
Bad Length	
Bad Node	
Bad Function	
No Start	
PLC exception	
NAK	
Streaming	
Premature	
Preamble	
IC Timeouts	
Address Errors	
Data Object offline	
Node Offline	
Msg Ignored	
Sys Cleared	
Squelch TX	
Squelch RX	
Segmentation Not Supported	
Passthru Retries	
Passthru Overruns	
Passthru Early Retries	
Passthru Normal Retries	
Passthru Early Overruns	
Passthru Normal Overruns	
Passthru Early Overruns Fails	
Passthru Normal Overruns Fails	
PWT Expired	

2.6. A – Data Array Overview

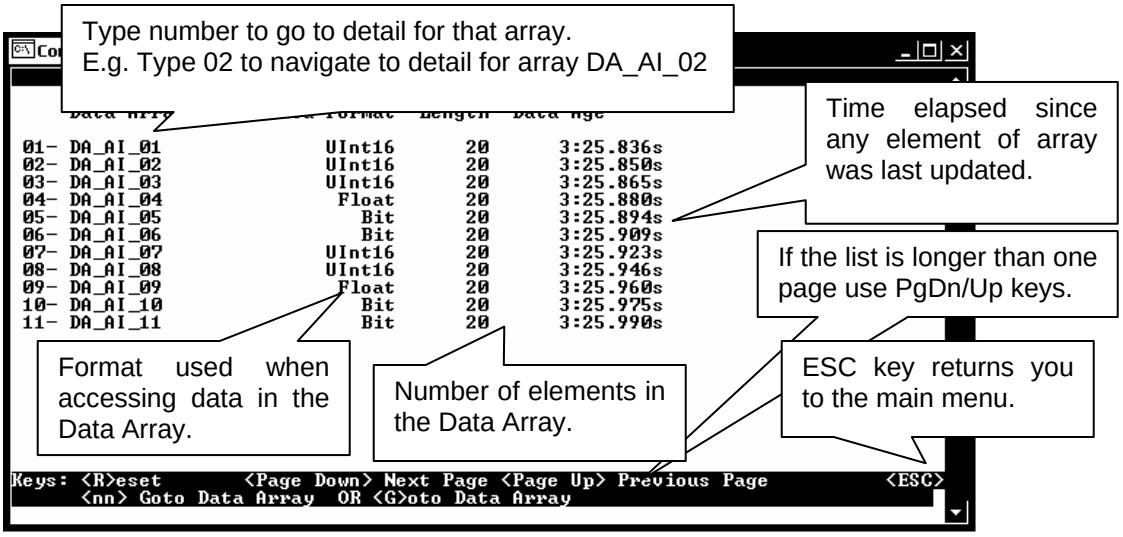


Figure 2.10 – Data Array Overview Screen

2.6.1. Data Array Detail Screen

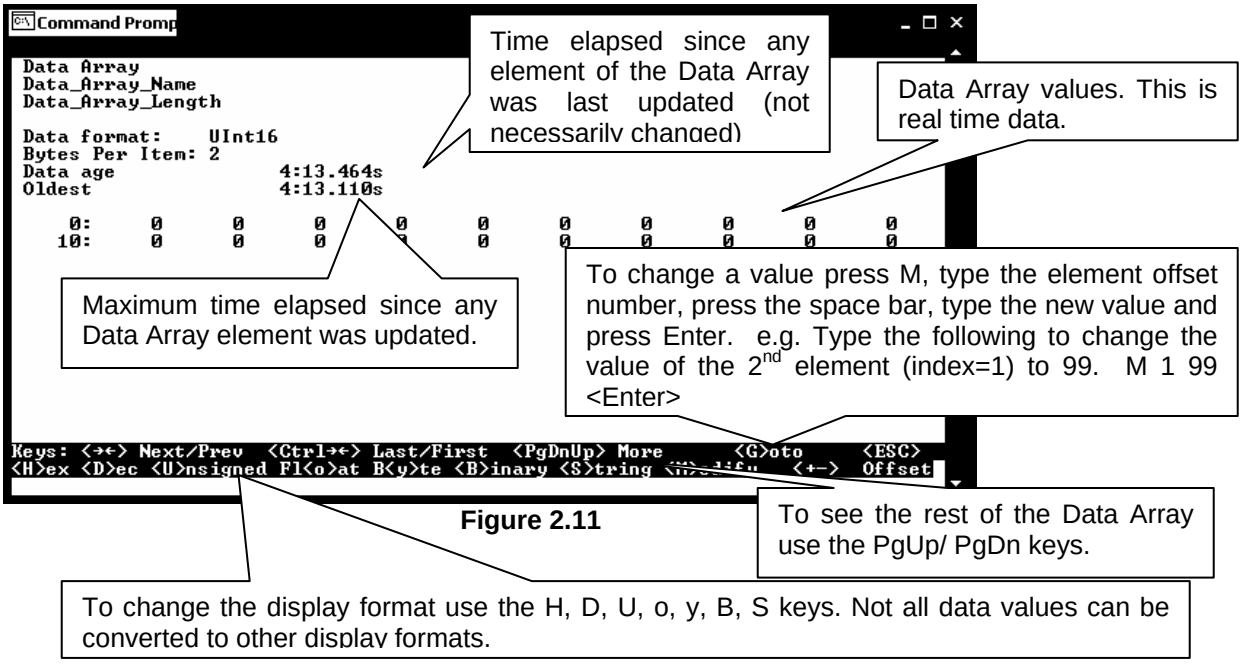


Figure 2.11

2.7. E – Error Messages

The error screen contains messages printed by the FieldServer kernel as well as messages from many drivers. Starting late in 2002, drivers started sending their messages to the F Screen.

The error screen is a circular buffer which can hold a limited number of lines of information. Once full, as each new line is added to the bottom of the buffer, the top line is removed.

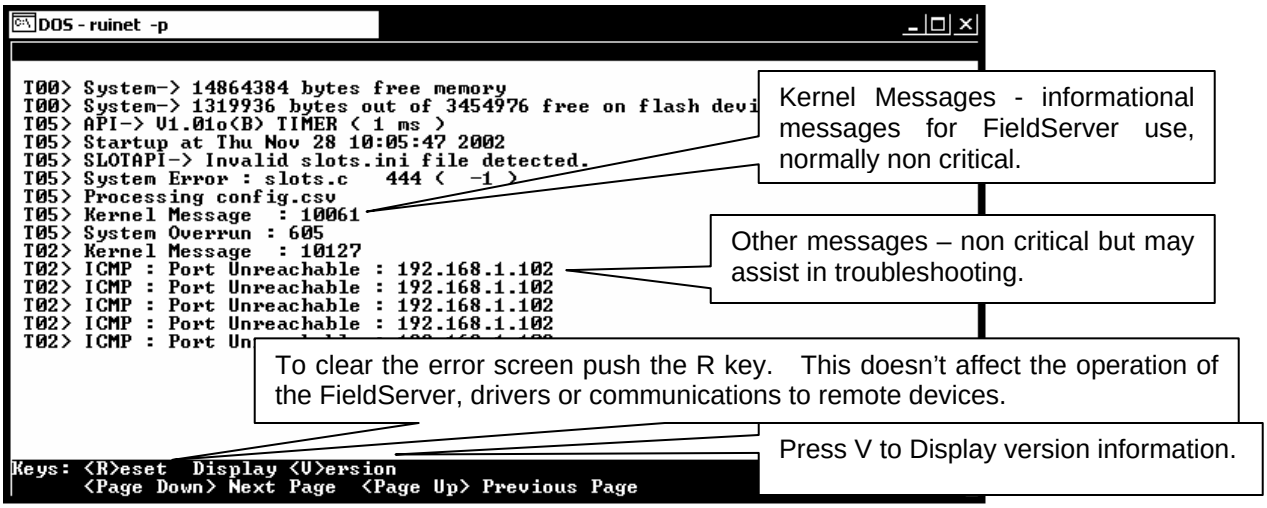


Figure 2.12

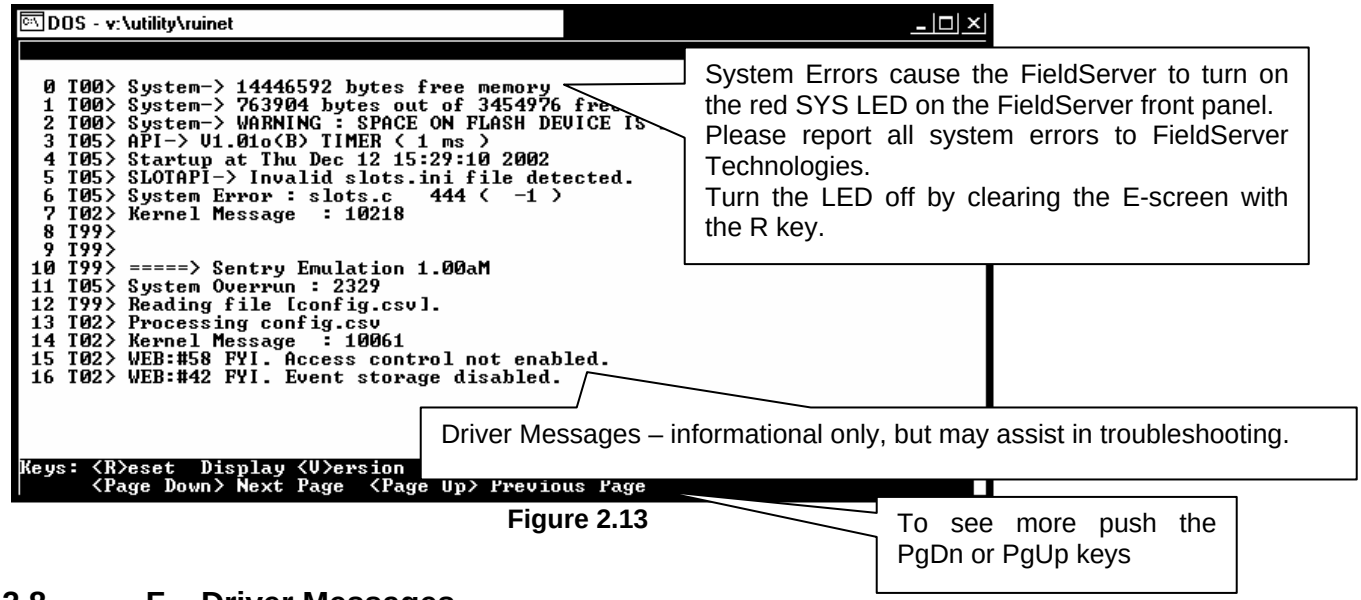


Figure 2.13

2.8. F – Driver Messages

The F screen is similar to the E-Screen. Newer type drivers send their error or informational messages to the F screen. In the past all messages were sent to the E screen.

2.9. D – Download Configuration to FieldServer

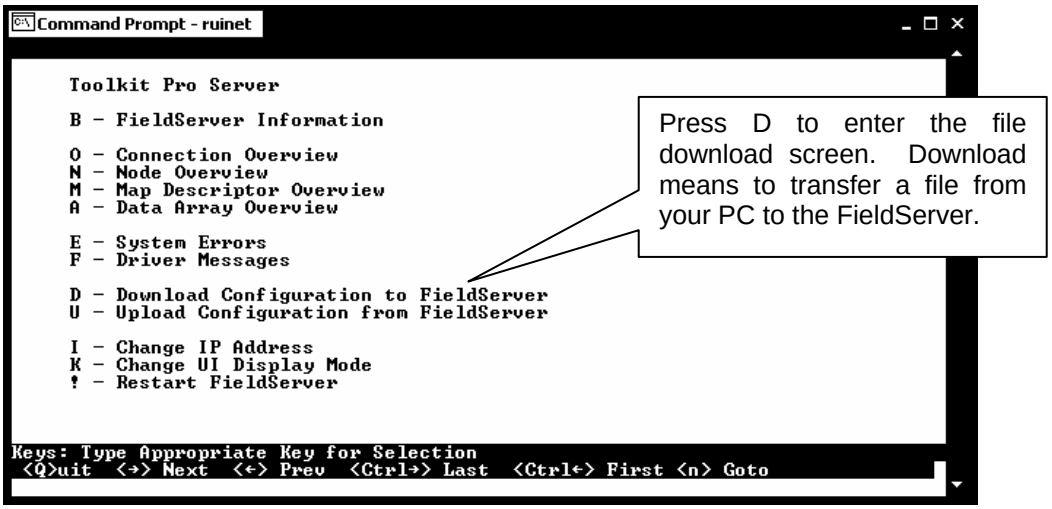


Figure 2.14

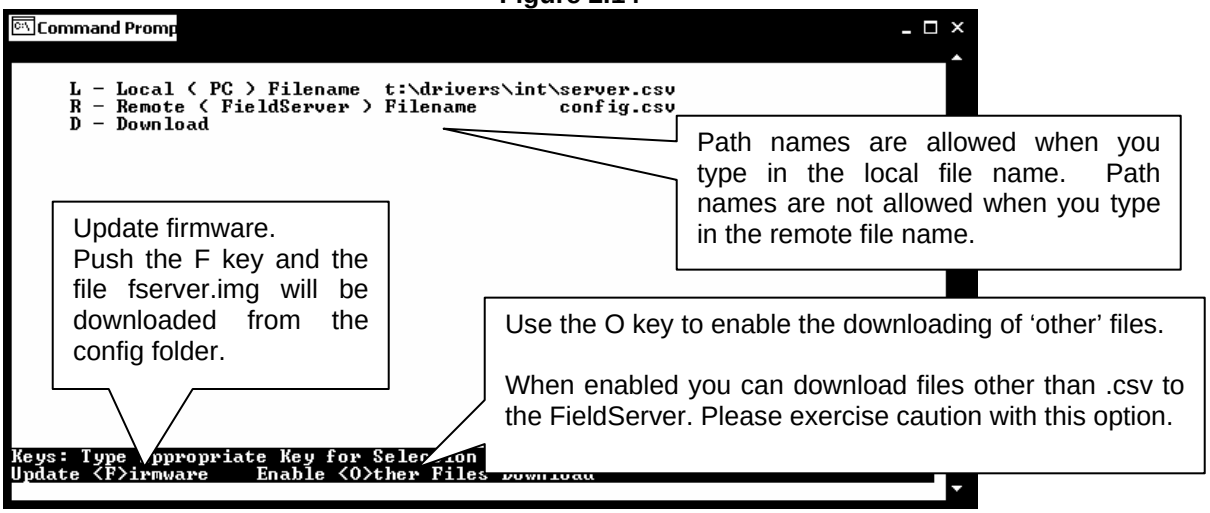


Figure 2.15

2.9.1. Procedure for Downloading a File from a PC to a FieldServer

- From the main menu select ***"D"*** to go to the download screen.
- Specify the local file name by selecting ***"L"***, type the file name and press ***<Enter>***
- If the remote filename is incorrect, select ***"R"***, type the file name and press ***<Enter>.***
- Begin the download by selecting ***"D".***

2.10. U - Upload Configuration from FieldServer

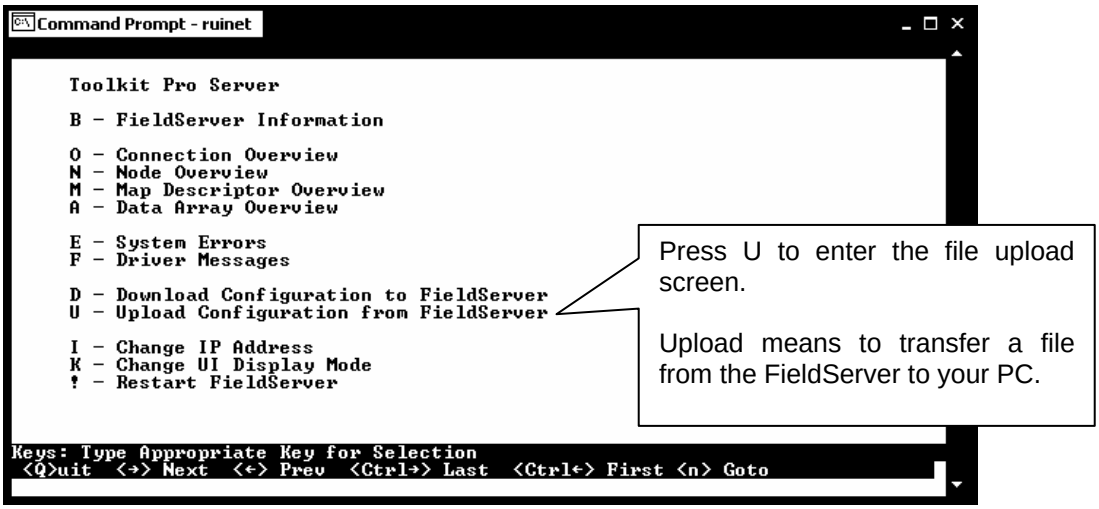


Figure 2.16

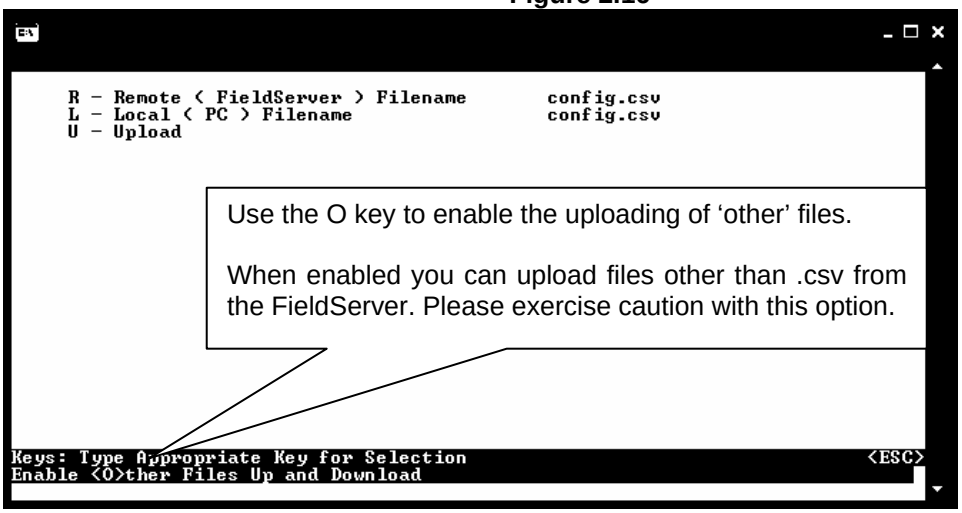


Figure 2.17

2.10.1. Procedure to Upload a File from a FieldServer to a PC

- From the main menu, select ***"U"*** to go to the Upload screen.
- Select ***"R"***, type in the remote file name (File on the FieldServer) and press ***<Enter>***
- If the local file name is incorrect, select ***"L"***, type the name and press ***<Enter>.***
- Begin the upload by pressing ***"U".***
- When the upload is completed, the uploaded file may be opened with one of the listed editors. Note that the editors are not supplied with RuiNet and must be loaded on your machine to work effectively.
- See Appendix B.10 for more information

2.11. I - Change IP Address

From the main menu, press ***"I"*** to enter the Edit IP Address Settings menu.

2.11.1. Procedure to Change a FieldServer's IP Address

- Press ***"1"*** to modify the IP address of the N1 Ethernet adapter or ***"6"*** to modify the IP address of the N2 Ethernet adapter (if available on the FieldServer).
- Type in a new IP address such as 192.168.0.35 and press ***<Enter>.***
- If necessary, press***"2"*** or ***"7"*** and change the netmask.
- Restart the FieldServer.

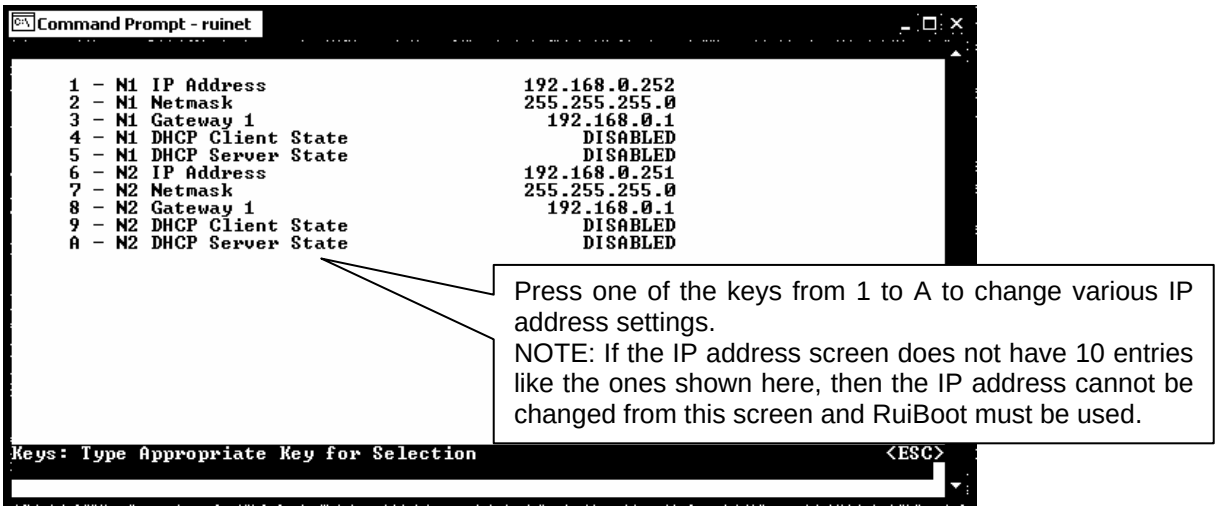


Figure 2.18

Common IP address ranges and netmasks:
Class A – IP (1.0.0.1 to 126.255.255.254), Netmask (255.0.0.0)
Class B – IP (128.0.0.1 to 191.255.255.254), Netmask (255.255.0.0)
Class C – IP (192.0.0.1 to 223.255.255.254), Netmask (255.255.255.0)

Do not use broadcast IP address ending on 255. This IP address is reserved as a target IP address when a device wants to send a message to all other devices on the network.

2.11.2. Obtain the IP address using the DHCP Client

The FieldServer can obtain its IP address from a DHCP Server using the built-in DHCP Client. Note that the network's DHCP Server must be setup correctly for this to work. Please contact your network administrator for assistance.

- Enable the DHCP Client on Ethernet port N1 and/or N2 by pressing ***"4"*** or ***"9"***
- Press ***"Y" <Enter>*** to enable or ***"Esc"*** to escape. ***"N"*** will disable the DHCP client
- Press any key to continue and escape back to the main menu. Press***"!"*** to restart the FieldServer
- Press ***"Q"*** twice to quit from RuiNet and the discovery screen
- Run RuiNet again to connect to the FieldServer

If the FieldServer cannot contact a DHCP Server, the FieldServer will use the last good IP address it was configured with.

2.11.3. Obtain the IP address using the FieldServer's DHCP Server

The FieldServer can be used as a DHCP Server to allocate an IP address to a PC or Laptop that you want to use to connect with RuiNet to the FieldServer.

Enable the DHCP Server on the FieldServer.

- Enable the DHCP Server on Ethernet port N1 and / or N2 by pressing ***"5"*** or ***"A"***
- Press ***"Y" <Enter>*** to enable or ***"Esc"*** to escape. ***"N"*** will disable the DHCP client
- Press any key to continue and escape back to the main menu. Press***"!"*** to restart the FieldServer

When connecting a PC or Laptop to the FieldServer on port N1 (or N2 for X40), be sure to enable the DHCP Client function on the PC or Laptop. The following screen-shot from the Network Connection's properties on a Windows XP PC shows the TCP/IP settings needed to enable the PC's DHCP Client:

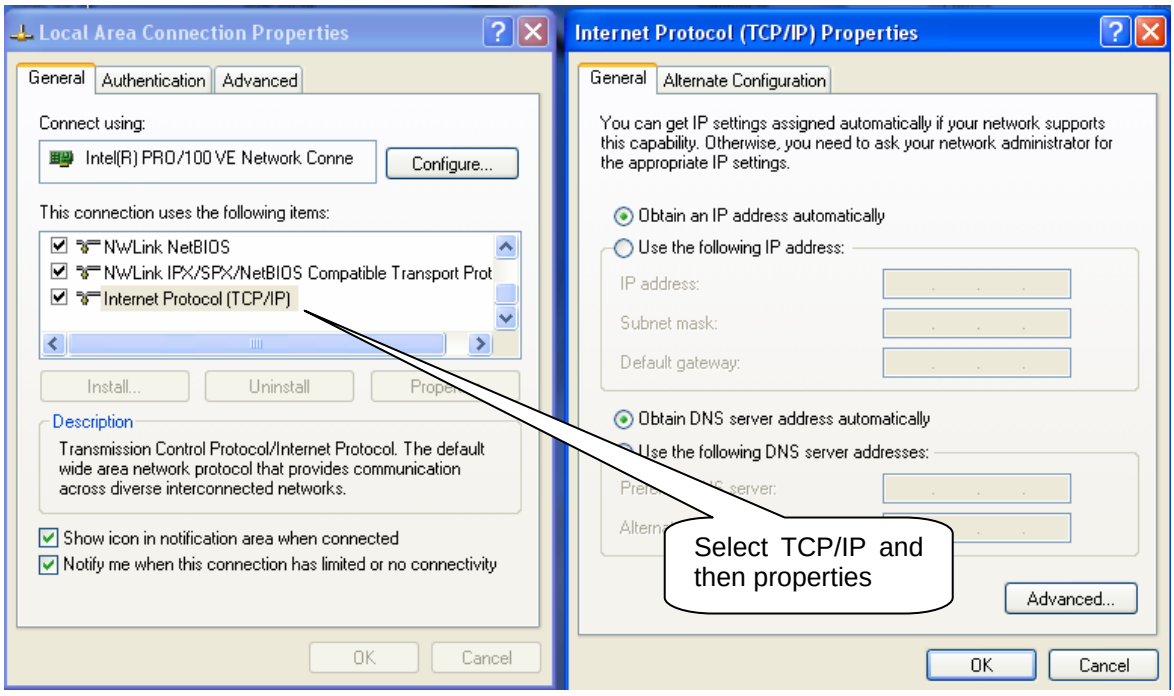


Figure 2.19

Simply restart the PC while connected to the FieldServer and it will obtain an IP address from the FieldServer.

NOTE: Some implementations of the Microsoft DHCP Client will not use an IP address lease offered by the FieldServer until its previous lease has been released. The way to release and renew a lease on a PC is to open a command prompt window and use the following commands (syntax may be different on Windows other than XP):

ipconfig /release *
ipconfig /renew *

The FieldServer checks every few minutes for the existence of other DHCP Servers and will disable itself if any are detected. A message on the RuiNet E screen will indicate when the DHCP Server has been disabled.

2.12. K - Change UI Display Mode

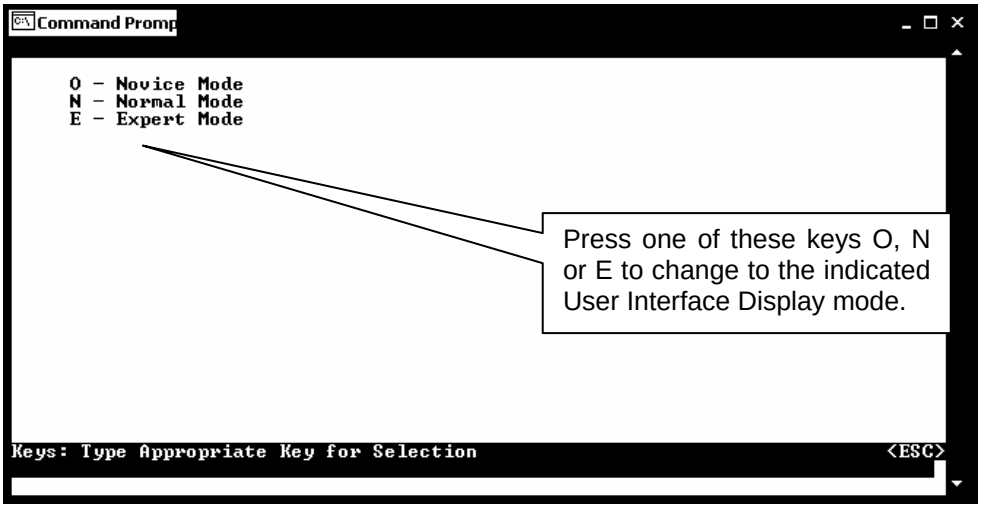


Figure 2.20

The Expert Mode allows viewing of the different FieldServer tiers. The Novice and Normal modes currently have the same functionality in that it prevents viewing the FieldServer tiers.

NOTE: Normal and Expert modes are intended for the use of FieldServer personnel only. They provide no improved application functionality, and the added functions in these modes are neither documented, nor supported by FieldServer Technical Support. Consequently, it is strongly advised that these modes are not enabled.

2.13. Restart FieldServer

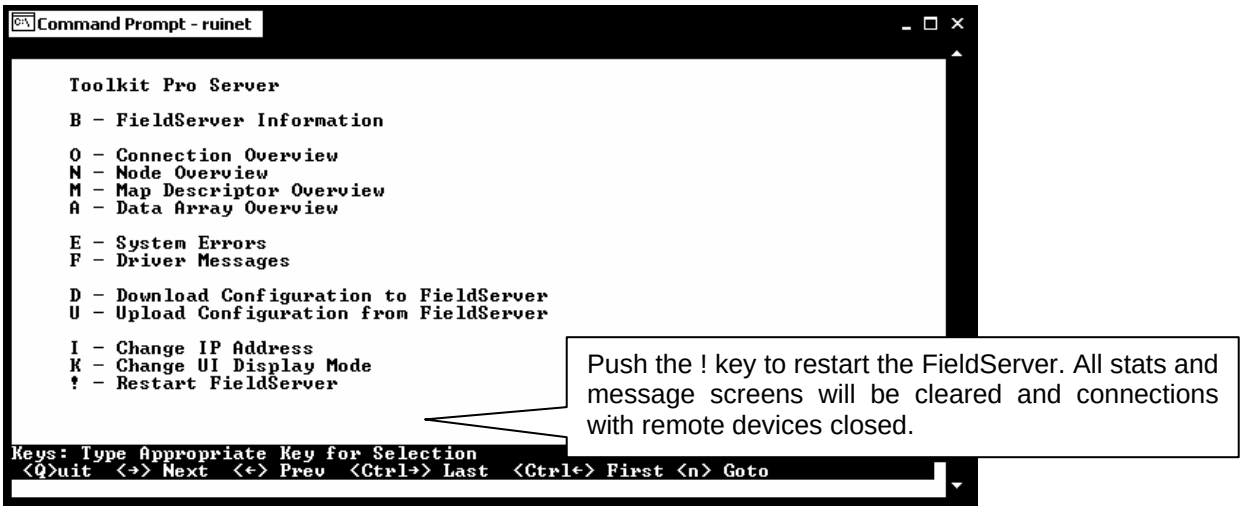


Figure 2.21

RuiNet will display the main interactive menu again after the FieldServer has restarted.

Appendix A. Troubleshooting

Appendix A.1. Connection to a FieldServer

The following situations can prevent Ruinet from connecting to a FieldServer:

- Bad or incorrect Ethernet cable. In this case the connection lights on the FieldServer and/or PC will not be illuminated either
- Incorrect Ethernet card and/or protocol setup. In this case Ruiping probably won't work either. Contact the Systems Administrator to have an Ethernet port set up correctly.
- If two FieldServers with different names have the same IP address then these are shown in red in the list of FieldServers. Ruinet cannot be used with these FieldServers except to restart them. Disconnect one FieldServer from the network and change the other FieldServer's IP address to a different value using Ruinet which will connect now that there is only one FieldServer.
- If a FieldServer is not on the same subnet as the PC, and there is more than one FieldServer on the network, then it will not be possible to use broadcast mode to cross subnets. Disconnect the other FieldServers and use broadcast mode, or change the IP address of the "orphan" FieldServer to put it in the same subnet and then reconnect the other FieldServers.
- Ruinet uses UDP port 1024. Ensure that the firewall or router is not blocking this port.
- If the FieldServer is connected to the host computer's network on adapter N2 then the SMT protocol must be defined for the N2 adapter in the config.csv file. Add the following lines to the file only if it is not possible to connect to N1 instead:

```
Adapters
Adapter,                    Protocol
N2,                         SMT
```

Appendix A.2. File Download

- Unless a path is specified for the local file name then Ruinet expects to find the local file in the same folder as that in which Ruinet was launched or in the folder set as the working directory (as specified in the Windows shortcut). If the file cannot be found then an error is reported.
- The local and remote file names must meet the DOS 8.3 file name format requirements.
- There must be sufficient flash disk space on the FieldServer.
- If the download of files like fserver.img (the firmware file) or config.csv (the configuration file) fails before completion then the next time the FieldServer is restarted the FieldServer will try and run with corrupt firmware or a corrupt configuration. If the firmware is corrupt then future downloads may not work at all and the user may need to follow a recovery procedure. The procedure is available on the FieldServer website as an application note.

Appendix B. Advanced Command Line Option Switches

Appendix B.1. Connecting to a FieldServer using IP Address -i

Specify the IP address using the ***-i*** switch.

```
RuiNet –i192.168.0.45
```

RuiNet will only connect to the FieldServer with IP address 192.168.0.45

Appendix B.2. Connecting to a FieldServer using Name -y

Specify a FieldServer's name with the ***-y*** switch

```
RuiNet –yModbus_Test
```

RuiNet will connect to the FieldServer with the specified name.

Appendix B.3. Broadcast Mode -ib

RuiNet can be used in broadcast mode to poll all IP addresses on a network. This feature is not recommended under normal operating conditions and should only be used when there is just one FieldServer on the network.

```
RuiNet –ib
```

RuiNet will use broadcasting to connect to a FieldServer.

Appendix B.4. The Most Recently Connected FieldServer -p

Use the ***-p*** command line switch and RuiNet will connect to the same FieldServer that was connected previously. The most recently connected to FieldServer will be shown on top of the list of "Recently connected to FieldServers" on the RuiNet list of FieldServers screen. This command line switch is useful in batch files.

```
RuiNet –p
```

RuiNet will connect to the FieldServer it connected to previously.

Sample batch file:

```
RuiNet –zdefault.htm –m0
RuiNet –p –zzone1.htm
RuiNet –p –zzone2.htm
RuiNet –p –zzone3.htm
RuiNet –p –zzone4.htm
```

No FieldServer targeted and no auto connect so user must select a FieldServer even if there is only one on the network.

No FieldServer targeted but –p ensures that the FieldServer selected by the user when the previous command was executed is the FieldServer that is connected to with this command.

Appendix B.5. Disable Auto Connect Mode –m0

If the target FieldServer is not specified with the *–i* switch then RuiNet either auto connects to the only FieldServer found if there is only one, or presents a list of FieldServers and auto connects to the most recently connected to FieldServer after 10 seconds. If there was no

previously connected FieldServer, RuiNet connects to the first FieldServer in the list after 10 seconds. This behavior can be overridden by using the ***–m0*** switch. When specifying this switch on the command line then RuiNet does not auto connect to any FieldServer.

RuiNet –m0
Ensures presentation of a list and auto connection does not occur.

Appendix B.6. Restart a FieldServer -b

Tells RuiNet to restart a FieldServer. No confirmation to restart is requested.

RuiNet –b
Restarts the FieldServer that RuiNet auto connects to.

RuiNet –i192.168.1.81 –b
Restarts a specific FieldServer.

Appendix B.7. Startup Screen -x

Directs RuiNet to display a particular screen on connection to a FieldServer. Specify the switch and the start-up screen number.

System Errors	(E Screen)	-x1
Driver Messages	(F Screen)	-x2
Connection Overview	(O Screen)	-x3
Node Overview	(N Screen)	-x4
Map Descriptor Overview	(M Screen)	-x5
Data Array Overview	(A Screen)	-x6
FieldServer Information	(B Screen)	-x7

RuiNet –x3
When RuiNet connects the connection overview screen will be displayed.

Appendix B.8. Transferring files –l, -f

When a file is transferred, its local file name (on host computer) and its FieldServer-side file name must be known for RuiNet to work correctly. The local file name is always specified with the ***–l*** switch whereas the remote or FieldServer-side file name is always specified with the ***–f*** switch. RuiNet will fill in the default target file name if possible and thus one of the file name switches may be omitted in some circumstances.

If transferring a file with the extension .CSV from a host computer to a FieldServer then RuiNet will assume that the FieldServer-side file name is config.csv unless using the *–**f*** command line switch to specify a different target file name.

RuiNet -fconfig.csv –lc:\temp\config.csv –u1
Transfers the file config.csv from the FieldServer to the host computer where it will be saved as config.csv in the temp folder of the C drive

Appendix B.9. Downloading Files –u0

This command is used to transfer files from a PC to a FieldServer. In the following examples the target FieldServer is not specified. If there is a single FieldServer on the network then the files will be transferred to that FieldServer. If more than one FieldServer is connected to the network then RuiNet will provide a list and the transfer will begin once a FieldServer is selected from the list.

RuiNet –ldelivery.csv –u0 -o

Transfers the file delivery.csv from your computer to the FieldServer where it will be saved as config.csv. In this example the ***–o*** option forces RuiNet to download the file even if delivery.csv (on host computer) and config.csv (on the FieldServer) are the same.

RuiNet -ldefault.htm –u0

Transfers the file default.htm from host computer to the FieldServer where it will be saved as default.htm.

RuiNet –lc:\temp\delivery.csv –fdelivery.csv –u0

Transfers the file delivery.csv from host computer's temp folder on the C drive to the FieldServer where it will be saved as delivery.csv.

RuiNet –lconfig.csv –u0 –n3

Transfers the file config.csv from host computer to the FieldServer where it will be saved as config.csv. RuiNet expects that the file config.csv exists on the host computer and is located in the same folder as the folder where RuiNet is executed. In this example the ***–n3*** switch tells RuiNet, that if the transfer fails, to attempt the transfer a maximum of 3 times before giving up.

Wildcards are allowed on download filenames. RuiNet will present a list of the files marked for download and a list of the files already on the FieldServer for you to decide whether to proceed with the download. The "list download files" prompt can be overridden with a ***–c*** switch.

RuiNet –ltest*.csv –u0

Transfers all files with filenames starting with test and ending with .csv to the FieldServer.

Appendix B.10. Uploading Files –u1

This command is used to transfer a file from the FieldServer to the PC. In the following example the target FieldServer is not specified. If there is a single FieldServer on the network then the files will be transferred with that FieldServer. If more than one FieldServer is connected to the network then RuiNet will provide a list and the transfer will begin once the required FieldServer is selected.

RuiNet -fconfig.csv –u1

Transfers the file config.csv from the FieldServer to the host computer where it will be saved as config.csv. It will be saved in the folder from which RuiNet was executed.

Appendix B.11. Forcing a Download -o

When downloading a file that already exists on the FieldServer, RuiNet checks if the file to be downloaded is identical to the file already on the FieldServer. If the files are identical, RuiNet responds that a download is not necessary.

This behavior can be overridden and the download forced to happen by using the ***–o*** command line switch.

```
RuiNet –lconfig.csv –u0 –o
```

RuiNet downloads config.csv regardless of whether the file on the FieldServer is identical.

Appendix B.12. Number of File Transfer Tries -n

If a file is not transferred correctly then RuiNet will retry the transfer. RuiNet continues to retry until the transfer is successful. This can be avoided by specifying the maximum number of retries that may be attempted.

```
RuiNet –lconfig.csv –u0 –n10
```

RuiNet will try up to ten times to successfully download the file config.csv to a FieldServer.

Appendix B.13. Set FieldServer Side File Name for a transfer -f

If transferring a file with the same name from host computer to a FieldServer, omit this switch. RuiNet will always use the FieldServer file name of config.csv for any CSV file downloaded unless an altenate file name is specified using this switch. If a file is downloaded with the name ”fserver.img”, RuiNet will automatically convert this to “cb8menu.exe” on the FieldServer.
A path in the FieldServer Side file name cannot be set.

```
RuiNet –lconfig.csv –fbackup.csv –u0
```

RuiNet will download config.csv from host PC to the FieldServer and save it as backup.csv on the FieldServer.

Appendix B.14. Set Local File Name for a Transfer -l

If transferring a file with the same name from host computer to a FieldServer, omit this switch. If the file is to have a different name or if the file is to be saved to a specific drive and/or folder then use this switch.

```
RuiNet –lconfig.csv –fbackup.csv –u1
```

RuiNet will upload backup.csv from the FieldServer and save it as config.csv on the host PC.

Appendix B.15. Help -h

Prints a list of command line switches. If this command line switch is used then all others are ignored.

```
RuiNet –h
```

RuiNet prints out all options and exits.

Appendix B.16. Create a Log File -a

Instructs RuiNet to make a log file containing progress information.

```
RuiNet –lconfig.csv –u0 –alog.txt
```

RuiNet creates a log file called log.txt on host PC and logs the progress and results of the config.csv file download to the log file.

The log file contains time and date stamped entries that describe the actions performed by RuiNet. The log file is used by FieldServer technical support and developers for debugging purposes. An excerpt from a typical log file is shown below:

```
07/30/03 20:34:09 Doing rui_send_command with cmd = 182
07/30/03 20:34:09 Rui_version = 41610400
07/30/03 20:34:29 Calling Download_with_repeat (config.csv, config.csv, 1, 0, 0, 1024
07/30/03 20:34:29 Setting tx_rui = 3000
07/30/03 20:34:29 Getting Rui_get_file_size for = config.csv
07/30/03 20:34:29 Restoring tx_rui = 2000
07/30/03 20:34:29 Rui_get_file_size = 4259
```

Appendix B.17. Version Information -v, -ve

Prints RuiNet version information.

```
RuiNet –v
```

RuiNet prints version information and exits.

```
RuiNet –ve
```

RuiNet prints extended version information and exits.

Appendix B.18. Skip date and time check on connect -m1

If the system date and time of the machine running RuiNet differs by more than 5 minutes from that of the FieldServer you will be prompted to update the FieldServer. This behavior can be overridden using the ***–m1*** switch.

```
RuiNet –m1
```

Ensures update FieldServer date and time prompt does not occur.

Appendix B.19. Delete a File -z

Delete a file on the FieldServer. Use this switch with extreme caution as no opportunity for confirmation to delete is provided. If RuiNet connects, the file will be deleted.

RuiNet –zdefault.htm
Deletes the file default.htm on the FieldServer that RuiNet connects to.

RuiNet –i192.168.1.81 –zdefault.htm
Deletes the file default.htm on a specifically targeted FieldServer.

Appendix B.20. Synch FieldServer Time with the Computer's Time -g

The FieldServer's real time clock is maintained with a super-capacitor. The longest that the time can be maintained for is about 30 days before the capacitor completely discharges. A consequence of this is that often when a user first receives their FieldServer the time is not correctly set.

The FieldServer date and time can be synchronized with host computer's date and time by using the ***–g*** switch.

RuiNet –g
Sets a FieldServer's date and time to that of the host computer.

Appendix B.21. Set Timeout -k

Sets the timeout in milliseconds. This is the time RuiNet waits for a response before deciding that 1) a FieldServer connection has been lost or 2) A FieldServer cannot be connected to.

Normally the default timeout never needs to be changed but when connecting to FieldServers on remote networks or over the internet it might be necessary to make the timeout longer. The default timeout value is currently 2 seconds.

RuiNet –k10000 –i66.24.1.192
Tells RuiNet to try and connect to a specific FieldServer and sets the timeout to 10 seconds.

Appendix B.22. Run in Test Mode

This option is for FieldServer testing purposes only. It activates various test actions when RuiNet is used. It currently causes checksum errors during a file download to test if retries are working.

Appendix C. Advanced Topics

Appendix C.1. The Ruinet Log File

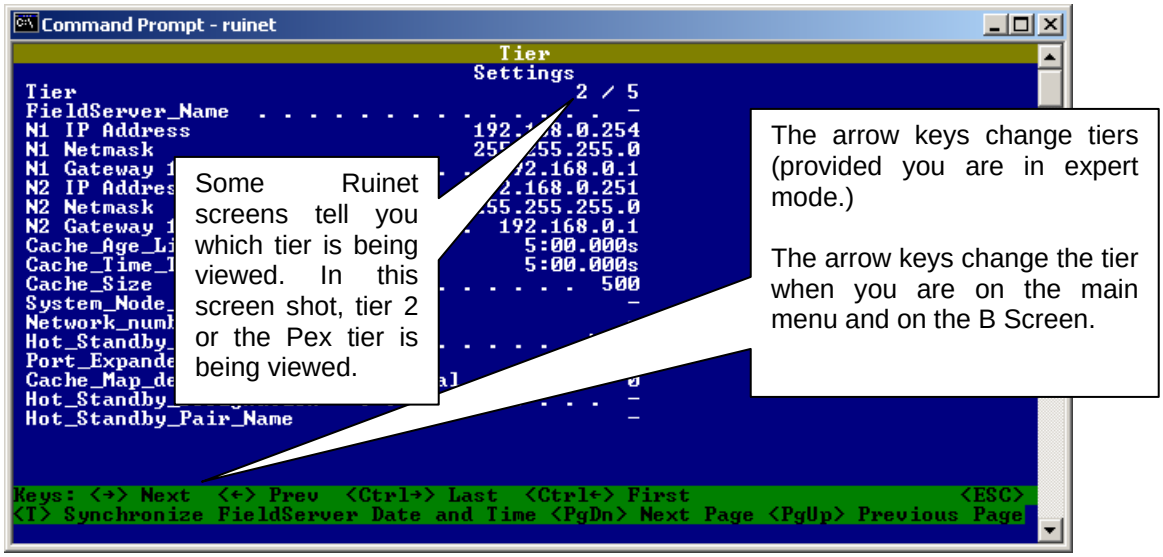
The Ruinet log file is created when the –a switch has been specified on the Ruinet command line. The log file contains time and date stamped entries that describe the actions performed by Ruinet. The log file is used by FieldServer technical support and developers for debugging purposes. An excerpt from a typical log file is shown below:

```
07/30/03 20:34:09 Doing rui_send_command with cmd = 182
07/30/03 20:34:09 Rui_version = 41610400
07/30/03 20:34:29 Calling Download_with_repeat (config.csv, config.csv, 1,
0, 0, 1024
07/30/03 20:34:29 Setting tx_rui = 3000
07/30/03 20:34:29 Getting Rui_get_file_size for = config.csv
07/30/03 20:34:29 Restoring tx_rui = 2000
07/30/03 20:34:29 Rui_get_file_size = 4259
```

Appendix C.2. FieldServer Tiers

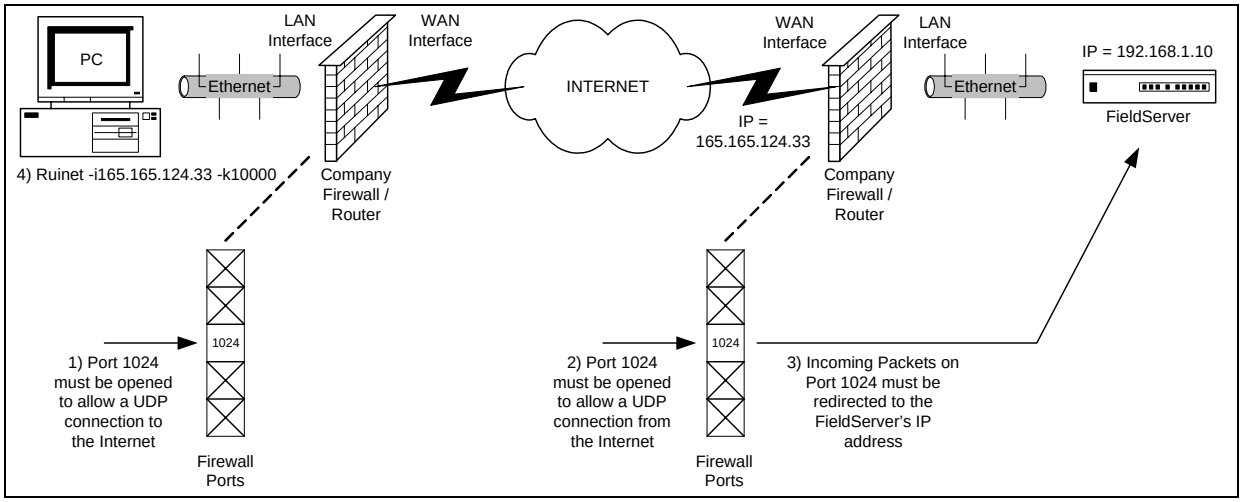
A tier is a virtual FieldServer – a software construction on which the various drivers are executed. Most applications only use one tier (called “Tier 2” or “Tier B” or “The Pex Tier”) but some emulations and complex configurations use multiple tiers.

When Ruinet connects to a FieldServer it will always connect to the Pex Tier (Tier 2). The arrow keys allow the user to change tier but only if in the expert mode. If not in expert mode then the arrow keys have no effect.



Appendix C.3. Connecting to a FieldServer over the Internet

The diagram below shows how this is typically done.



Note the following:

1. On the machine running Ruinet, the firewall must allow UDP connections to port 1024 on the WAN (Wide Area Network) interface / Internet.
2. On the network where the FieldServer is situated, the firewall must allow UDP connections from the WAN interface / Internet to port 1024.
3. On the network where the FieldServer is situated, the firewall must redirect network packets from the internet on port 1024 to the FieldServer's IP address.
4. To connect to the FieldServer, run:

```
Ruinet –i165.165.124.33 –k10000
```

Note that the –k switch sets the timeout to 10 seconds or 10,000 ms. This is a timeout value that should work well over the Internet, but it might be necessary to increase this further on a very slow connection if timeouts keep occuring.

Appendix D. Default settings for parameters

Parameter	Default Setting
Default response timeouts	2000 ms = 2 sec
Inter character timeout	500 ms
SCADA hold	2000 ms = 2 sec
Data cache age limit for acceptable data	20000 ms = 20 sec
Cache	80
Retry Interval	10000 ms = 10 sec
Recovery Interval	30000 ms = 30 sec
Probation Delay	60000 ms = 1 min
Scan Interval	1000 ms = 1 sec
Poll Delay	50 ms
Retries	3
Activity Timer	120000 ms = 2 hour
Parity	None
Baud	9600
Data Bits	8
Stop Bits	1
Handshake Timeout	2000 ms = 2 sec