

InterNaqs

Version 1.00

User Guide

**Nanometrics Inc.
Kanata, Ontario
Canada**

© 2003–2004 Nanometrics Inc. All Rights Reserved.

InterNaqs Version 1.00 User Guide

The information in this document has been carefully reviewed and is believed to be reliable for Version 1.00.xx. Nanometrics, Inc. reserves the right to make changes at any time without notice to improve the reliability and function of the product.

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Nanometrics Inc.

This product uses the Java™ 2 Runtime Environment (J2RE) Standard Edition, which includes code licensed from RSA Security, Inc. Some portions licensed from IBM are available at <http://oss.software.ibm.com/icu4j/>

Nanometrics, Inc.
250 Herzberg Road
Kanata, Ontario, Canada K2K 2A1
Tel (613)592-6776
Fax (613)592-5929
Email info@nanometrics.ca

Part number 15142R3

Release date 2004-09-13

Contents

Tables	iii
Chapter 1	
Getting Started	1
1.1 About InterNaqs	1
1.1.1 Inputs and outputs	1
1.1.1.1 Input files	1
1.1.1.2 Output files	2
1.2 Installing InterNaqs	2
1.2.1 Requirements	2
1.2.2 Install InterNaqs	2
Chapter 2	
NaqsToUdp v1.00	3
2.1 Running NaqsToUdp	3
2.1.1 Starting NaqsToUdp manually	3
2.1.2 Starting NaqsToUdp from NmxWatchdog	3
2.1.3 Stopping NaqsToUdp	4
2.1.4 Using the NaqsToUdp run-time commands	4
2.1.5 Monitoring NaqsToUdp operation	4
2.2 Configuring NaqsToUdp	5
2.2.1 [Connections]	5
2.2.2 [Log]	5
2.2.3 [Channels]	6
2.3 Example configuration file (NaqsToUdp.ini)	6
Chapter 3	
SeedlinkToUdp v1.00	9
3.1 Running SeedlinkToUdp	9
3.1.1 Starting SeedlinkToUdp manually	9
3.1.2 Starting SeedlinkToUdp from NmxWatchdog	9
3.1.3 Stopping SeedlinkToUdp	10
3.1.4 Using the SeedlinkToUdp run-time commands	10
3.1.5 Monitoring SeedlinkToUdp operation	10
3.2 Configuring SeedlinkToUdp	11
3.2.1 [Log]	11
3.2.2 [Connections]	11
3.2.3 [Station]	12
3.2.4 Naqs.stn [ChannelPrototype]	12
3.3 Example configuration file (SeedlinkToUdp.ini)	13

Tables

2-1	NaqsToUdp run-time commands	4
2-2	[Connections] section parameters	5
2-3	[Log] section parameters	6
3-1	SeedlinkToUdp run-time commands	10
3-2	[Log] section parameters	11
3-3	[Connections] section parameters	12
3-4	[Station] section parameters	12

InterNaqs is a set of utilities for the sharing of data among Nanometrics Naqs Server installations via TCP/IP, and subscription to Seedlink server data.

This user guide provides instructions for installing, configuring, and running the InterNaqs components. See also:

- ◆ NaqsServer
- ◆ General Utilities – Watchdog
- ◆ Online data access – Private Data Streams (in the Nanometrics Data Formats reference guide)

1.1 About InterNaqs

InterNaqs utilities include NaqsToUdp and SeedlinkToUdp:

- ◆ NaqsToUdp enables data to be transferred reliably from one Nanometrics acquisition system to another, via TCP/IP. It connects to the source NaqsServer and subscribes to one or more data channels using the NaqsServer datastream service. Packets are then relayed to the receiving NaqsServer via UDP. NaqsToUdp also relays retransmission requests from the receiving NaqsServer back to the source.
- ◆ SeedlinkToUdp enables data to be transferred from a Seedlink server system to a NaqsServer acquisition system. SeedlinkToUdp connects to the Seedlink server via TCP/IP, and subscribes to one or more data channels using the Seedlink protocol.

1.1.1 Inputs and outputs

1.1.1.1 Input files

NaqsToUdp and SeedlinkToUdp require that their configuration file be present in the current directory:

- ◆ `NaqsToUdp.ini` – Defines basic operating characteristics for NaqsToUdp (for example, IP addresses, ports, channels).
- ◆ `SeedlinkToUdp.ini` – Defines basic operating characteristics for SeedlinkToUdp (for example, IP addresses, ports, stations).

You can edit the configuration files in any text editor.

1.1.1.2 Output files

NaqsToUdp and SeedlinkToUdp both produce a startup log file and daily log files, contain diagnostic messages generated by the program.

The daily log files provide a summary of the system operation. The date is encoded in the log file name (for example, `NaqsToUdp_20031114.log`). Each log message has an associated type:

- I – Informational messages tracing the normal operation of the system.
- V – Detailed informational messages tracing the normal operation of the system.
- D – Debug, or verbose trace, messages generated during normal operation.

Log verbosity can be configured to show only messages at or above a specified severity level, by adjusting the verbosity setting. The verbosity of the log on startup is set in the `.ini` file. During operation, verbosity can be set to a different level by using the runtime commands.

1.2 Installing InterNaqs

1.2.1 Requirements

- ◆ Windows or Solaris operating system
- ◆ Java Runtime Environment 1.4.1 or higher
- ◆ Patches as may be recommended for the OS and JRE
- ◆ Nanometrics DLLs/Libraries 1.8 or higher
- ◆ NaqsServer 1.80 or higher

1.2.2 Install InterNaqs

Install InterNaqs on the same machine as the receiving NaqsServer.

On Windows:

1. From either a command prompt or Windows Explorer, open the installation CD directory `Win32\InterNaqs\version number`
2. Copy all files from the `bin` directory into the `c:\nmx\bin` directory, and all files from the `user` directory into the `c:\nmx\user` directory (these directories are created on your computer during DLL installation).
3. Check the system path, and add the directory `c:\nmx\bin` if it is not already included.
4. Adjust the configuration parameters as outlined in Chapter 2, “NaqsToUdp v1.00” and Chapter 3, “SeedlinkToUdp v1.00”.

On Solaris and Linux:

- ▶ See the installation instructions for the acquisition system workstation.

NaqsToUdp is a communications program which enables data to be transferred reliably from one Nanometrics acquisition system to another, via TCP/IP. It connects to the source NaqsServer and subscribes to one or more data channels using the NaqsServer Datastream service. Packets received from the source NaqsServer are then relayed to the receiving NaqsServer via UDP. NaqsToUdp also relays retransmission requests from the receiving NaqsServer back to the source, to ensure full recovery of data which are missed due to any communications outage.

NaqsToUdp is installed as an InterNaqs utility; typically, it is run on the same machine as the receiving NaqsServer (see Section 1.2, “Installing InterNaqs,” on page 2).

2.1 Running NaqsToUdp

NaqsToUdp would normally be set up to run on the same machine as NaqsServer on the receiving system.

2.1.1 Starting NaqsToUdp manually

- ▶ To start NaqsToUdp from the command line, enter either:

```
NaqsToUdp
```

or

```
NaqsToUdp infile
```

where *infile* is the path to the NaqsToUdp configuration file. If no *infile* parameter is specified, NaqsToUdp looks for a file named `NaqsToUdp.ini` in its working directory.

2.1.2 Starting NaqsToUdp from NmxWatchdog

In Windows, NaqsToUdp can be started and monitored automatically by the Nanometrics watchdog program:

- ▶ Add an entry to your `watchdog.ini` file:

```
[ WatchEntry n ]  
ProgramTitle = NaqsToUdp  
ProgramPathname = "java -jar c:\nmx\bin\NaqsToUdp.jar [infile]"  
WorkingDirectory = "c:\nmx\user"  
ExitAction = Restart
```

```
PingsSemaphore = true
StartDelay = 6s
```

2.1.3 Stopping NaqsToUdp

NaqsToUdp must be shut down properly in order for it to release its system resources.

- ▶ To stop NaqsToUdp, enter `quit` in the NaqsToUdp command window.

2.1.4 Using the NaqsToUdp run-time commands

NaqsToUdp supports a basic keyboard interface for entering run-time commands, with the options described in Table 2-1.

- ▶ To enter run-time commands in the NaqsToUdp terminal window, enter *command*.

Table 2-1 NaqsToUdp run-time commands

To do this...	Enter this command...
List the channels available for subscription.	C
Display all log messages in the log file; set the log verbosity to DEBUG	D
Suppress debug messages in the log file; set the log verbosity to VERBOSE	V
Suppress debug and verbose messages in the log file; set the log verbosity to INFO	I
Move the log file (close the current log and start a new file)	M
Report the number of packets received from and sent to source.	R
Stop NaqsToUdp and exit.	quit

2.1.5 Monitoring NaqsToUdp operation

NaqsToUdp generates log messages that trace the operation of the program. It displays these messages in the terminal window, and writes them to the NaqsToUdp log file. You can set the level of detail (the verbosity) of the information to be displayed and recorded.

- ▶ To view the log, open the file *LogFilename_date.log* (for example, `NaqsToUdp_20031110.log`) in a text editor. The log file name and location are set in the [Log] section of the `NaqsToUdp.ini` file.
- ▶ To set the verbosity of log messages on startup, edit the [Log] section of the `NaqsToUdp.ini` file.
- ▶ To change the verbosity of log messages while NaqsToUdp is running, use the run-time commands.

2.2 Configuring NaqsToUdp

The operating configuration for the program is specified in the `NaqsToUdp.ini` file. This file contains these three sections; all entries are required:

- ♦ [Connections]
- ♦ [Log]
- ♦ [Channels]

The format for each `naqstoudp.ini` entry is *Parameter = Value*. For example, `DestinationPort = 32000`. See also the example configuration file in Section 2.3 on page 6.

2.2.1 [Connections]

The [Connections] section defines the hostname and datastream port of the data source, and the hostname and UDP port of the receiving NaqsServer. It contains the parameters described in Table 2-2.

Table 2-2 [Connections] section parameters

Parameter	Definition
<i>SourceHost</i>	The host name or IP address of the machine on which the source NaqsServer is running. Example: <code>SourceHost = naqs.source.net</code>
<i>SourcePort</i>	The NaqsServer Datastream port of the source acquisition system. Example: <code>SourcePort = 28000</code>
<i>UseCallback</i>	Specifies whether to connect to the source NaqsServer using a callback socket. This should be set to <code>Yes</code> only if the [Datastream] <i>SocketType</i> parameter on the source NaqsServer is set to <code>Callback</code> . • Permitted values: <code>Yes</code> , <code>No</code> . Example: <code>UseCallback = No</code>
<i>DestinationHost</i>	The host name or IP address of the machine on which the receiving NaqsServer is running. • If NaqsToUdp and the receiving NaqsServer are running on the same machine, you can set this to <code>localhost</code> . • If you wish to make the data available to more than one acquisition system on your local network, set this to a multicast address. Example: <code>DestinationHost = localhost</code>
<i>DestinationPort</i>	The UDP receive port on the receiving NaqsServer. • This should match the [NetworkInterface] <i>Port</i> parameter in the <code>Naqs.ini</code> file. Example: <code>DestinationPort = 32000</code>

2.2.2 [Log]

The [Log] section defines the location, name and verbosity of the NaqsToUdp log file. It contains the parameters described in Table 2-3.

Table 2-3 [Log] section parameters

Parameter	Definition
<i>LogFilename</i>	The base filename for the NaqsToUdp log file. NaqsToUdp creates a new log file every day; it builds the log file name by inserting the date (<i>yyyymmdd</i>) between the base name and the file extension. For example, <code>NaqsToUdp_20031114.log</code> <ul style="list-style-type: none"> • Permitted values: Any valid file name, with no spaces. Example: <code>LogFilename = NaqsToUdp.log</code>
<i>LogDirectory</i>	The pathname for the directory in which to store the NaqsToUdp log file. Names are treated as relative pathnames (relative to the directory in which NaqsToUdp is running), unless they are specified as absolute names (with a leading slash). <ul style="list-style-type: none"> • Permitted values: Any valid pathname, with no spaces. Do <i>not</i> include the trailing slash. Example: <code>LogPath = logs</code>
<i>Verbosity</i>	The startup verbosity of the log file, to show all messages (DEBUG, resulting in very large log files), suppress debug messages (VERBOSE, resulting in large log files), or suppress debug and detailed messages (INFO, the setting typically used to monitor operation). <ul style="list-style-type: none"> • Permitted values: DEBUG, VERBOSE, INFO. Example: <code>Verbosity = INFO</code>

2.2.3 [Channels]

The [Channels] section defines the channels that you want this program to receive. This section may contain any number of entries, one per line. Each channel should be identified using the dotted station/channel name by which the channel is identified on the source acquisition system; for example, `STN02.bhz`. See also Section 2.3, “Example configuration file (`NaqsToUdp.ini`)”.

- ▶ To obtain a list of the channels available for subscription by NaqsToUdp, at the NaqsToUdp command window enter `c`.

2.3 Example configuration file (`NaqsToUdp.ini`)

This section contains an example `NaqsToUdp.ini` file. You can edit this file in any text editor.

The inifile reader ignores white space and blank lines, so white space can be added anywhere within the file if desired to improve readability. Also, the inifile reader recognizes the double-slash “//” as a comment delimiter. You can use this to add comments anywhere in the file, and to temporarily remove parameters or sections from the file.

```
// NaqsToUdp.ini
// Configuration file for NaqsToUdp version 1.00

[ Connections ]
SourceHost = naqs.source.net
SourcePort = 28000
UseCallback = No
DestinationHost = localhost
```

```
DestinationPort = 32000

[ Log ]
LogFilename = NaqsToUdp.log
LogDirectory = logs
Verbosity = verbose

[ Channels ]
STN02.bhz
STN02.bhn
STN02.bhe
STN02.soh
```


SeedlinkToUdp

v1.00

SeedlinkToUdp is a communications program which enables data to be transferred from any Seedlink server system to a Nanometrics NaqsServer acquisition system. SeedlinkToUdp connects to the Seedlink server via TCP/IP, and subscribes to one or more data channels using the Seedlink protocol. The current version accepts data using Steim (1) compression.

SeedlinkToUdp is installed as an InterNaqs utility; typically, it is run on the same machine as NaqsServer (see Section 1.2, “Installing InterNaqs,” on page 2).

3.1 Running SeedlinkToUdp

SeedlinkToUdp would normally be set up to run on the same machine as NaqsServer.

3.1.1 Starting SeedlinkToUdp manually

- ▶ To start SeedlinkToUdp from the command line, enter either:

```
SeedlinkToUdp
```

or

```
SeedlinkToUdp infile
```

where *infile* is the path to the SeedlinkToUdp configuration file. If no *infile* parameter is specified, SeedlinkToUdp looks for a file named `SeedlinkToUdp.ini` in its working directory.

3.1.2 Starting SeedlinkToUdp from NmxWatchdog

In Windows, SeedlinkToUdp can be started and monitored automatically by the Nanometrics watchdog program:

- ▶ Add an entry to your `watchdog.ini` file:

```
[ WatchEntry n ]  
ProgramTitle = SeedlinkToUdp  
ProgramPathname = "java -jar c:\nmx\bin\SeedlinkToUdp.jar [infile]"  
WorkingDirectory = "c:\nmx\user"
```

```
ExitAction = Restart
PingsSemaphore = true
StartDelay = 6s
```

3.1.3 Stopping SeedlinkToUdp

SeedlinkToUdp must be shut down properly in order for it to release its system resources.

- ▶ To stop SeedlinkToUdp, enter `quit` in the SeedlinkToUdp command window.

3.1.4 Using the SeedlinkToUdp run-time commands

SeedlinkToUdp supports a basic keyboard interface for entering run-time commands, with the options described in Table 3-1.

- ▶ To enter run-time commands in the SeedlinkToUdp terminal window, enter *command*.

Table 3-1 SeedlinkToUdp run-time commands

To do this...	Enter this command...
Display all log messages in the log file; set the log verbosity to DEBUG	D
Suppress debug messages in the log file; set the log verbosity to VERBOSE	V
Suppress debug and verbose messages in the log file; set the log verbosity to INFO	I
Move the log file (close the current log and start a new file)	M
Stop SeedlinkToUdp and exit.	<code>quit</code>

3.1.5 Monitoring SeedlinkToUdp operation

SeedlinkToUdp generates log messages that trace the operation of the program. It displays these messages in the terminal window, and writes them to the SeedlinkToUdp log file. You can set the level of detail (the verbosity) of the information to be displayed and recorded.

- ▶ To view the log, open the file *LogFilename_date.log* (for example, *SeedlinkToUdp_200311211.log*) in a text editor. The log file name and location are set in the [Log] section of the *SeedlinkToUdp.ini* file.
- ▶ To set the verbosity of log messages on startup, edit the [Log] section of the *SeedlinkToUdp.ini* file.
- ▶ To change the verbosity of log messages while SeedlinkToUdp is running, use the run-time commands.

3.2 Configuring SeedlinkToUdp

The operating configuration for the program is specified in the SeedlinkToUdp.ini file, and requires additional [ChannelPrototype] sections to the Naqs station file.

SeedlinkToUdp.ini contains three sections; all entries are required:

- ◆ [Log]
- ◆ [Connections]
- ◆ [Station]

The format for each SeedlinkToUdp.ini entry is *Parameter = Value*. For example, `DestinationNaqsPort = 32000`. See also the example configuration file in Section 3.3, “Example configuration file (SeedlinkToUdp.ini),” on page 13.

3.2.1 [Log]

The [Log] section defines the location, name, and verbosity of the SeedlinkToUdp log file. It contains the parameters described in Table 3-2.

Table 3-2 [Log] section parameters

Parameter	Definition
<i>LogFile</i>	The base filename for the SeedlinkToUdp log file. SeedlinkToUdp creates a new log file every day; it builds the log file name by inserting the date (<i>yyyymmdd</i>) between the base name and the file extension. For example, <code>SeedlinkToUdp_20031215.log</code> . <ul style="list-style-type: none"> • Permitted values: Any valid file name, with no spaces. Example: <code>LogFile = SeedlinkToUdp.log</code>
<i>LogPath</i>	The pathname for the directory in which to store the SeedlinkToUdp log file. Names are treated as relative pathnames (relative to the directory in which SeedlinkToUdp is running), unless they are specified as absolute names (with a leading slash). <ul style="list-style-type: none"> • Permitted values: Any valid pathname, with no spaces. Do <i>not</i> include the trailing slash. Example: <code>LogPath = logs</code>
<i>Verbosity</i>	The startup verbosity of the log file, to show all messages (DEBUG, resulting in very large log files), suppress debug messages (VERBOSE, resulting in large log files), or suppress debug and detailed messages (INFO, the setting typically used to monitor operation). <ul style="list-style-type: none"> • Permitted values: DEBUG, VERBOSE, INFO. Example: <code>Verbosity = INFO</code>

3.2.2 [Connections]

The [Connections] section defines the hostname and UDP port of the receiving NaqsServer on which SeedlinkToUdp is running, and the hostname and TCP port of the source Seedlink server. It contains the parameters described in Table 3-3.

Table 3-3 [Connections] section parameters

Parameter	Definition
<i>DestinationNaqsAddress</i>	The host name or IP address of the NaqsServer machine. <ul style="list-style-type: none"> If SeedlinkToUdp and the receiving NaqsServer are running on the same machine, this may be set to <code>localhost</code>. Example: <code>DestinationNaqsAddress = localhost</code>
<i>DestinationNaqsPort</i>	The UDP receive port on the NaqsServer. <ul style="list-style-type: none"> This should match the [NetworkInterface] <i>Port</i> parameter in the <code>Naqs.ini</code> file. Example: <code>DestinationNaqsPort = 32000</code>
<i>SeedlinkServerAddress</i>	The host name or IP address of the machine on which the Seedlink server is running. Example: <code>SeedlinkServerAddress = seedlink.source.net</code>
<i>SeedlinkServerPort</i>	The Seedlink server TCP port number. Example: <code>SeedlinkServerPort = 18000</code>

3.2.3 [Station]

The [Station] section defines the channels that you want this program to receive from the Seedlink server. It contains the parameters listed in Table 3-4. There can be any number of [Station] sections.

Table 3-4 [Station] section parameters

Parameter	Definition
<i>Network</i>	The Seedlink network name. Example: <code>Network = RO</code>
<i>Station</i>	The Seedlink station name. Example: <code>Station = ST31</code>
<i>Channel</i>	The Seedlink channel names, as a comma-separated list, or use * to list all channels for the station. Example: <code>Channel = BHZ</code>
<i>Model</i>	The instrument model as defined in the NaqsServer <code>Naqs.stn</code> file. Example: <code>Model = HRD</code>
<i>SerialNumber</i>	The instrument serial number as defined in the NaqsServer <code>Naqs.stn</code> file. Example: <code>SerialNumber = 181</code>

3.2.4 `Naqs.stn` [ChannelPrototype]

For data to be received correctly, the Naqs station file must have [ChannelPrototype] sections defined for the Seedlink channels, with *BundlesPerPacket* set to 27. (See also the NaqsServer user guide.)

- ▶ If a [ChannelPrototype] is already defined in the `Naqs.stn` file for a type of channel that will also be received via SeedlinkToUdp, add a new [ChannelPrototype] section for the Seedlink channel type with *BundlesPerPacket* set to 27.

For example:

```
[ ChannelPrototype ] // predefined channel - all fields mandatory

TypeName = BHZ-1 // label for this type
Name = BHZ // channel name
Component = 1 // digitiser component
Sensor = STS-2 // pointer to predefined [Sensor] characteristics
Azimuth = 0 // azimuth in degrees clockwise from North
Dip = 90 // dip in degrees (positive down)
Depth = 0 // has to be defined for each channel
BundlesPerPacket = 15 // number of bundles per data packet
RingBufferSize = 50 // file size in MB
RingBufferPath = d:\nmx // where files are located
ResponseFile = none // name of SEED response file

[ ChannelPrototype ] // predefined channel - all fields mandatory

TypeName = BHZ-seedlink // label for this type
Name = BHZ // channel name
Component = 1 // digitiser component
Sensor = STS-2 // pointer to predefined [Sensor]
// characteristics
Azimuth = 0 // azimuth in degrees clockwise from North
Dip = 90 // dip in degrees (positive down)
Depth = 0 // has to be defined for each channel
BundlesPerPacket = 27 // number of bundles per data packet
RingBufferSize = 50 // file size in MB
RingBufferPath = d:\nmx // where files are located
ResponseFile = none // name of SEED response file
```

3.3 Example configuration file (SeedlinkToUdp.ini)

This section contains an example SeedlinkToUdp.ini file. You can edit this file in any text editor.

The inifile reader ignores white space and blank lines, so white space can be added anywhere within the file if desired to improve readability. Also, the inifile reader recognizes the double-slash “//” as a comment delimiter. You can use this to add comments anywhere in the file, and to temporarily remove parameters or sections from the file.

```
// SeedlinkToUdp.ini
// Configuration file for SeedlinkToUdp Version 1.00.00

[ Log ]

LogFile = SeedlinkToUdp.log // base name for the log file
LogPath = logs // directory in which to store the log file
Verbosity = INFO // start-up verbosity (normally use INFO)

[ Connections ]

DestinationNaqsAddress = 199.71.138.183 // Naqs server IP address
DestinationNaqsPort = 32000 // Naqs server port number
SeedlinkServerAddress = korf.ethz.ch // Seedlink server IP address
// or host name
SeedlinkServerPort = 18000 // Seedlink server port number
```

```
[ Station ]

Network = RO          // The Seedlink Network Name
Station = ST31        // The Seedlink Station Name
Channel = BHN, BHE    // Seedlink channels as a comma-separated list,
                    // or * for all channels
Model = HRD           // The NaqsServer Instrument Model
SerialNumber = 181    // The NaqsServer Instrument SerialNumber

[ Station ]

Network = RO          // The Seedlink Network Name
Station = ST32        // The Seedlink Station Name
Channels = *          // Seedlink channels as a comma-separated list,
                    // or * for all channels
Model = HRD           // The NaqsServer Instrument Model
SerialNumber = 182    // The NaqsServer Instrument SerialNumber

[ Station ]

Network = RO          // The Seedlink Network Name
Station = ST33        // The Seedlink Station Name
Channels = *          // Seedlink channels as a comma-separated list,
                    // or * for all channels
Model = HRD           // The NaqsServer Instrument Model
SerialNumber = 183    // The NaqsServer Instrument SerialNumber

[ Station ]

Network = RO          // The Seedlink Network Name
Station = ST34        // The Seedlink Station Name
Channels = *          // Seedlink channels as a comma-separated list,
                    // or * for all channels
Model = HRD           // The NaqsServer Instrument Model
SerialNumber = 184    // The NaqsServer Instrument SerialNumber

[ Station ]

Network = CH          // The Seedlink Network Name
Station = BOURR       // The Seedlink Station Name
Channels = *          // Seedlink channels as a comma-separated list,
                    // or * for all channels
Model = HRD           // The NaqsServer Instrument Model
SerialNumber = 201    // The NaqsServer Instrument SerialNumber

[ Station ]

Network = CH          // The Seedlink Network Name
Station = DAVOX       // The Seedlink Station Name
Channels = *          // Seedlink channels as a comma-separated list,
                    // or * for all channels
Model = HRD           // The NaqsServer Instrument Model
SerialNumber = 202    // The NaqsServer Instrument SerialNumber
```