NmxToCSS3

Version 1.00

User Guide

Contents

Tables

NmxToCSS3

1 About NmxToCSS3

NmxToCSS3 maintains a copy of the time series data from a Nanometrics acquisition system in CSS3.0 format. It connects to NaqsServer and subscribes to one or more data channels using the NaqsServer Datastream service, then converts and stores the data for local analysis. The design is based on the standard specified in "CSS Version 3 Database: Schema Reference Manual" J.Anderson, W.E. Farrell, K. Garcia, J.Geven, H. Swanger, Rev 1.2 January 1995.

NmxToCSS3 should be run on the same machine as NaqsServer, or on a machine with TCP/IP access to the NaqsServer machine and network access to the Naqs ringbuffers, Naqs station file, and to the NmxToCD1(1) configuration and calibration history files.

1.1 Typical operation

NmxToCSS3 stores continuous time series data in binary waveform files, indexed by a `wfdisc` flat database file. Each waveform file stores data for a single seismic channel for a configurable time period (1 to 24 hours) at a specified sample rate. Data are stored in uncompressed s4 format.

NmxToCSS3 subscribes to real-time data from NaqsServer and writes data to the CSS3.0 files as packets are received. For each data segment, a `wfdisc` record is created containing station and channel information and pointers to the binary data. On a periodic maintenance cycle, `wfdisc` records for contiguous data segments are consolidated into a single record, provided that those segments are contained within the same binary file.

On startup NmxToCSS3 checks the waveform disk file records for missing data, and uses DataServer to request any missing data from Naqs. Waveform data files are created for the current scheduled interval, but will include data only from the actual startup time forward. The period from the scheduled interval to startup time will be empty. Use [Parameters] *BackFillData* to set the number of hours of data to fill prior to actual startup time. NmxToCSS3 will create and fill the current and previous interval files as required. (See also Figure 1-1.)

Figure 1-1 Example of back-filling data for [Parameters] *FileSize* = 4 and *BackFillData* = 8
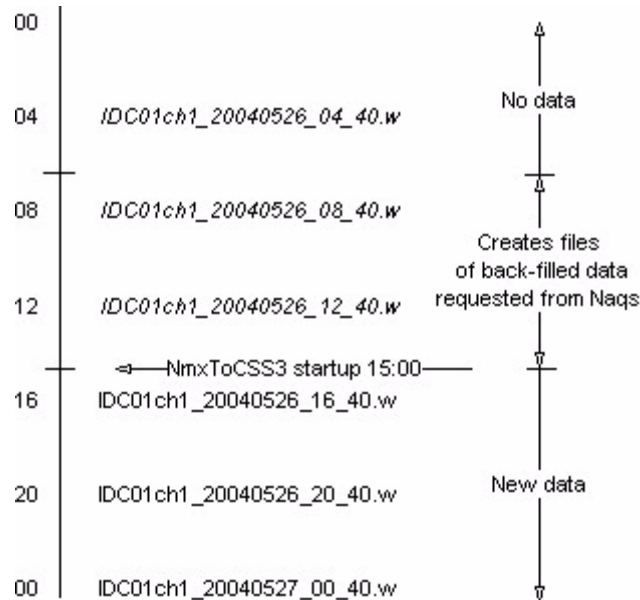
In addition to the `wfdisc` table and waveform files, NmxToCSS3 creates and maintains these tables in CSS Version 3.0 format: `affiliation`, `instrument`, `lastid`, `network`, `sensor`, `site`, and `sitechan`. (The `remark` table is also included, but is always empty; the *commid* attribute is not supported.) See also Appendix B, "CSS 3.0 Tables". Following acceptance of updated calibration values broadcast by the Calibration software, NmxToCSS3 updates the `sensor` table and uses that value in future `wfdisc` entries.

There must be one instance of NmxToCSS3 for each CD1.*x* sender.

The NmxToCSS3 operating configuration (for example, NaqsServer host address, instrument nominal calibration values) is maintained in the `NmxToCSS3.ini` file. Any changes to the configuration are implemented the next time the software is restarted. Program activity is recorded in a log file with a configurable verbosity level.

1.2 Summary of inputs and outputs

1.2.1 Inputs

- ◆ `NmxToCSS3.ini` – This configuration file defines operating characteristics for the NmxToCSS3 program, such as source file locations.
- ◆ Naqs server:
  - • Data from NaqsServer – NmxToCSS3 receives online data from the NaqsServer Datastream service via a TCP connection. See the Nanometrics Data Formats reference guide for information on private data streams, and the NaqsServer manual.

- NmxToCSS3 requires read access to the Naqs station file and ringbuffers:
  - `Naqs.stn` – The Naqs station file `Naqs.stn` is used for outage recovery on startup, and is the source of some attribute values (see Appendix B). It must either be stored on the same machine as NmxToCSS3 or be accessible over a LAN on a shared drive.
  - Naqs ringbuffer files – The ringbuffers are used to backfill data, which are acquired via DataServer.
  - DataServer – Forwards data from the Naqs ringbuffers in response to requests, via TCP (see also the DataServer manual).
  - Calibration data – When the user accepts calibration changes, NmxToCSS3 writes the changes to records in the appropriate tables (for example, to the `sensor` table, where a new record is created and the old record updated with an endtime).

> **Note** NmxToCSS3 requires access to the `CalHistory.cd`*x* file, which must exist before NmxToCSS3 is started. Either have the `CalHistory.cd`*x* file created automatically by starting NmxToCD*x*, or manually place a file in the location defined by [Parameters] *CalHistoryFile*.

- `CalHistory.cd`*x* – The calibration history file is created by NmxToCD1(1). It contains the starting calibration information for the `sensor` table (and the `wfdisc` table entries) and the mapping between the *site.channel* name and the Nanometrics *station.channel*. The source of the values depends on which CD sender is creating the file:
  - For systems receiving CD1.1 subframes, the Nanometrics channel name is set using information from the NmxToCD11 configuration file. The *sta, chan, calib,* and *calper* fields are set using information from the CD1.1 subframe headers.
  - For systems receiving CD1.0 data, all of the fields are set using information from the NmxToCD1 configuration file.

1.2.2 Outputs

- *StationChannel_Date_Time_sampleRate*`.w` (waveform files) – Each of the waveform files (for example, `IDC02ch2_20040521_16_120.w`) stores 1 to 24 hours of binary waveform data for a single seismic channel. Data are stored in uncompressed s4 format.
- CSS Version 3.0 tables:
  - *Date*`.wfdisc` (waveform index file) – A waveform disc file (for example, `20040521.wfdisc`) provides an index for a set of waveform binary files. It contains records with station and channel information, and pointers to the binary data. Waveform disc records for contiguous data segments are consolidated into a single record, provided that those segments are all contained within the same binary file.
  - *CSSPrefix.CSS3Table* (additional tables) – The CSS3.0 tables used in addition to `wfdisc` tables include `affiliation`, `instrument`, `lastid`, `network`, `remark` (always empty), `sensor`, `site`, and `sitechan`. See Appendix B for a summary of the tables used in NmxToCSS3 and the sources of attribute values.

◆ NmxToCSS3_*yyyymmdd*.log – The log file contains diagnostic messages gener-
ated by NmxToCSS3 and provides a summary of the program operation. Each log
message has an associated type, ranked by severity (Table 1-1). Log verbosity can
be configured to show only messages at or above a specified severity level.

The verbosity of the log on startup is set in the [Log] section of the
NmxToCSS3.ini file (Table 3-2 on page 5). While NmxToCSS3 is running, you
can set verbosity to a different level by using the run-time commands (Section 4.3
on page 10).

Table 1-1 NmxToCSS3 log message types

Label	Description
F	Fatal errors – Serious errors which cause immediate system shutdown.
E	Errors – Abnormal occurrences which will likely affect data integrity.
W	Warnings – Less serious abnormal occurrences.
I	Informational messages – Messages tracing the normal operation of the system.
V	Verbose messages – Detailed informational messages tracing the normal operation of the system.
D	Debug messages – Additional verbose trace messages.

2 Installing NmxToCSS3

NmxToCSS3 typically is installed on the NaqsServer machine. To simplify the config-
uration, install and run NmxToCSS3 in the same working directory as NaqsServer.

▸ See the installation instructions for the acquisition system workstation.

3 Definition of configuration file sections and parameters

The NmxToCSS3.ini file contains mandatory sections, for which all entries are
required, and optional sections. Mandatory sections include:

◆ [NmxToCSS3]
◆ [Log]
◆ [Connections]
◆ [Parameters]
◆ [Instrument]

Optional sections include:

◆ [Site]
◆ [SiteChan]

The format for each NmxToCSS3.ini entry is *Parameter = Value*. For example,
StationCode = WRA. See also the example configuration file in Appendix A.

Note Path names are treated as relative (relative to the directory in which
NmxToCSS3 is running), unless they are specified as absolute (with a
leading slash). The current directory can be indicated with a dot (.). Do
not include the trailing slash in path name parameters.

3.1 [NmxToCSS3]

The [NmxToCSS3] section defines the station name (Table 3-1). This must be the first section in the file.

Table 3-1 [NmxToCSS3] section parameter

Parameter	Definition
StationCode	The IDC station name for this instance of NmxToCSS3 (where a station can include up to 100 channels from Nanometrics stations). • Permitted values: a valid IDC station code. Example: `StationCode = WRA`

3.2 [Log]

The [Log] section defines the name, location, and verbosity of the NmxToCSS3 log file. It contains the parameters described in Table 3-2.

Table 3-2 [Log] section parameters

Parameter	Definition
LogFile	The base filename for the NmxToCSS3 log file. NmxToCSS3 creates a new log file every day. The log file name is built by inserting the date (*yyyymmdd*) between the base name and the file extension; for example, `NmxToCSS3_20040521.log` • Permitted values: a valid file name, with no spaces. Example: `LogFile = NmxToCSS3.log`
LogPath	The path name for the directory in which to store the NmxToCSS3 log file. • Permitted values: a valid path name, with no spaces. Example: `LogPath = logs`
Verbosity	The startup verbosity (level of detail of messages) of the log file. • Permitted values: DEBUG, VERBOSE, INFO. Example: `Verbosity = INFO`

3.3 [Connections]

The [Connections] section defines NmxToCSS3 connections to other parts of the system. It contains the parameters described in Table 3-3.

Table 3-3 [Connections] section parameters

Parameter	Description
NaqsAddress	The IP address or host name of the NaqsServer machine. • Permitted values: a valid host name or dotted decimal IP address. Example: `NaqsAddress = 199.71.138.213`
NaqsPort	The port number of the Naqs Datastream service. • Permitted values: a valid port number, typically 28000.
DataServerAddress	The IP address or host name of the DataServer machine. • Permitted values: a valid host name or dotted decimal IP address.

Table 3-3 [Connections] section parameters (Continued)

Parameter	Description
DataServerPort	The port number of the DataServer. • Permitted values: a valid port number, typically 28002.
DataServerUser	The DataServer user name (see also the DataServer manual). • Permitted values: a valid entry in `dataserver.ini` [Users].
DataServerPwd	The DataServer password for the user (see also the DataServer manual). • Permitted values: a valid entry in `dataserver.ini` [Users].
CalibrationAddress	The multicast IP address that the Nanometrics Calibrate software is sending to. • Permitted values: a multicast IP address in dotted decimal format.
CalibrationPort	The port number of the Calibrate multicast address. • Permitted values: a valid port number.

3.4 [Parameters]

The [Parameters] section defines waveform and table file characteristics, and the path to Naqs and Calibration input files. It contains the parameters described in Table 3-4.

Notes:

1) Changing file size during operation is not recommended. A change of *FileSize* will overwrite the existing waveform files for that day (edit *BackFillData* to populate the new files). However, the records for the previous waveform files will still exist in the `wfdisc` table.

2) A large *BackFillData* (for example, 72 hours) will require peak CPU usage for an extended period (for example, half an hour).

Table 3-4 [Parameters] section parameters

Parameter	Definition
FileSize	The size of each waveform file in hours. • Permitted values: 1, 2, 3, 4, 6, 12, 24. If a value other than a permitted value is entered, NmxToCSS3 uses the default value 4. Example: `FileSize = 4`
BackFillData	The number of hours to backfill data from the start of the NmxToCSS3 program (see also Section 1.1, "Typical operation," on page 1). • Permitted values: a positive integer. Example: `BackFillData = 24`
CSSPath	The root directory of the CSS tables (except for `wfdisc`). • Permitted values: a valid path name, with no spaces. Example: `CSSPath = cssTables`
WaveformPath	The root directory of the waveform data files and `wfdisc` files. • Permitted values: a valid path name, with no spaces. Example: `WaveformPath = waveforms`

Table 3-4 [Parameters] section parameters (Continued)

Parameter	Definition
CSSPrefix	The prefix for all CSS table filenames (except for `wfdisc`). For example, *CSSprefix*.`affiliation`. • Permitted values: an alphanumeric string. Example: `CSSPrefix = global`
NaqsStnFile	The full path to the Naqs station (`Naqs.stn`) file. • Permitted values: a valid path name, with no spaces. Example: `NaqsStnFile = Naqs.stn`
CalHistoryFile	The full path to the calibration history file (from CD1.*x* sender). • Permitted values: a valid path name, with no spaces. Example: `CalHistoryFile = CalHistory.cd11`

3.5 [Instrument]

The [Instrument] section defines instrument characteristics and the location of response information. It contains the parameters described in Table 3-5.

▸ Create one [Instrument] section for each instrument.

Table 3-5 [Instrument] section parameters

Parameter	Definition
Type	The type of the instrument, identified by an FDSN channel code. • Permitted values: an alphanumeric string. Example: `Type = SRO`
Name	The name of the instrument. This must match the sensor type in the `Naqs.stn` file. • Permitted values: an alphanumeric string. Example: `Name = EU101`
Dir	The directory containing the response file. • Permitted values: a valid path name, with no spaces. Example: `Dir = responses`
ResponseFile	The response file name. • Permitted values: a valid file name, with no spaces. Example: `ResponseFile = sro.fap`
SampleRate	The sample rate for this response. • Permitted values: a positive integer. Example: `SampleRate = 100`
NCalib	The nominal calibration factor, typically provided by the sensor manufacturer. • Permitted values: a positive float number. Example: `NCalib = 0.356`
NCalper	The nominal calibration period, typically provided by the sensor manufacturer. • Permitted values: a positive float number. Example: `NCalper = 1.0`

Table 3-5 [Instrument] section parameters (Continued)

Parameter	Definition
ResponseType	The format of the response file. • Permitted values: paz, fap, fir, paxfir. Example: `ResponseType = fap`

3.6 [Site]

The optional [Site] sections define station characteristics such as array member locations. Any unspecified value is set to NA (not applicable). They contain the parameters defined in Table 3-6.

Table 3-6 [Site] section parameters

Parameter	Description
Site	The site name as an NIEC code. • Permitted values: an alphanumeric string of 3 to 5 characters. Example: `Site = wra1`
StationType	The type of station, either a single station or an array. • Permitted values: ss, ar. Example: `StationType = ar`
RefSite	The reference station for array members as an NIEC code. • Permitted values: an alphanumeric string of 3 to 5 characters; use a hyphen (-) to indicate not applicable. Example: `RefSite = -`
DNorth	Offset North from the array reference in kilometres. • Permitted values: a positive float number. Example: `DNorth = 0.0`
DEast	Offset East from the array reference in kilometres. • Permitted values: a positive float number. Example: `DEast = 0.0`

3.7 [SiteChan]

The optional [SiteChan] sections define channel names and positions. They contain the parameters defined in Table 3-7.

Table 3-7 [SiteChan] section parameters

Parameter	Description
Channel	The channel name in dotted format (*stationCode.channelIdentifier*). This must match the value that is in the `CalHistory.cd`*x* file. • Permitted values: any channel name in dotted format. Example: `Channel = wra1.ch3`
EmplacementDepth	The position of the sensor relative to the elevation of the site. • Permitted values: any positive float number. If not specified, it is set to 0.0. Example: `EmplacementDepth = 1.5`

4   Running NmxToCSS3

NmxToCSS3 would normally run on the same machine as NaqsServer on the receiving system. On a typical system, the software is started automatically (for example, with scripts on Solaris). If you start NmxToCSS3 manually, ensure that NaqsServer and DataServer are started before you start NmxToCSS3, and that a `CalHistory.cd`*x* file exists in the expected location.

There must be one instance of NmxToCSS3 for each CD1.*x* sender.

▸ After changing a station configuration (for example, adding a new site), restart the system applications in this order:

a) NaqsServer

b) NmxToCD1(1)

c) NmxToCSS3

4.1   Starting and stopping NmxToCSS3 manually

1. Before starting NmxToCSS3:

a) Start NaqsServer and DataServer.

b) Ensure that the calibration history file exists. Either place a file of the correct format in the location defined in the [Parameters] section as *CalHistoryFile*, or create it automatically by starting NmxToCD*x*:

▸ For NmxToCD11, `CalHistory.cd11` is created once sufficient packets have been received. This may take a few minutes.

▸ For NmxToCD1, `CalHistory.cd1` is created on startup.

2. Start NmxToCSS3:

- Solaris and Linux:

▸ To start NmxToCSS3, enter `nmxtocss3 start` in any terminal window.

▸ To stop NmxToCSS3, enter `nmxtocss3 stop` in any terminal window.

- Windows:

▸ To start NmxToCSS3 from the command line, enter either:

`nmxtocss3`

or

`nmxtocss3` *inifile*

where *inifile* is the path to the NmxToCSS3 configuration file. If *inifile* is not specified, NmxToCSS3 looks for a file named `NmxToCSS3.ini` in its working directory.

▸ To stop NmxToCSS3, enter `quit` in the NmxToCSS3 command window.

4.2   Starting NmxToCSS3 from NmxWatchdog

In Windows, NmxToCSS3 can be started automatically and monitored by the Nano-metrics watchdog program.

▸ Add this entry to the `watchdog.ini` file:

```
[ WatchEntry n ]
ProgramTitle = NmxToCSS3
ProgramPathname = "java -jar c:\nmx\bin\NmxToCSS3.jar [inifile]"
WorkingDirectory = "c:\nmx\user"
ExitAction = Restart
PingsSemaphore = true
StartDelay = 6s
```

4.3   Using the NmxToCSS3 run-time commands

NmxToCSS3 supports a basic keyboard interface for entering run-time commands, with the options described in Table 4-1. The commands are case-insensitive.

▸ On Solaris and Linux, you can run these commands from any terminal window: Enter `NmxToCSS3` *command*

▸ On Windows, run these commands from the NmxToCSS3 console window: Enter *command*

Table 4-1 NmxToCSS3 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 NmxToCSS3 and exit	QUIT

4.4   Monitoring NmxToCSS3 operation

NmxToCSS3 generates log messages that trace the operation of the program. It displays these messages in the terminal window, and writes them to the NmxToCSS3 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 log file *LogFilename_date*`.log` (for example, `NmxToCSS3_20040521.log`) in a text editor. The log file name and location are set in the [Log] section of the NmxToCSS3 configuration file.

▸ To set the verbosity of log messages on startup, edit the [Log] section of the NmxToCSS3 configuration file.

▸ To change the verbosity of log messages while NmxToCSS3 is running, use the run-time commands.

4.5 Removing old data

You can remove old data manually, or automatically via a cron job or a scheduled task.

▸ If you change the *WaveformPath* (in the `.ini` file [Parameters] section), update the cron job to access the new location, and manually delete the old data file.

Appendix A Configuration File Example

This section contains an example NmxToCSS3 configuration file `NmxToCSS3.ini`. You can edit the `NmxToCSS3.ini` 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, so comments can be added anywhere in the file.

```
// NmxToCSS3.ini
// Simple ini-format configuration file for NmxToCSS3 Version 1.00.00

[ NmxToCSS3 ]

StationCode = IDC

[ Log ]
LogFile   = NmxToCSS3.log   // base name for the log file
LogPath   = logs            // directory in which to store the log file
Verbosity = VERBOSE         // start-up verbosity (normally use INFO)

[ Connections ]

NaqsAddress        = 199.71.138.183 // NaqsServer IP address
NaqsPort           = 28000          // NaqsServer port number
DataServerAddress  = 199.71.138.183 // DataServer IP address
DataServerPort     = 28002          // DataServer port number
DataServerUser     = rob            // DataServer user name
DataServerPwd      = rob            // DataServer password
CalibrationAddress = 224.1.1.1      // Calibration multicast IP Address
CalibrationPort    = 28004          // Calibration port number

[ Parameters ]   // The values listed are the defaults if invalid
                 // values are entered, the defaults are used
FileSize      = 4  // Size waveform of files in hours excepted values
                   // are: 1, 2, 3, 4, 6, 12, or 24; if an invalid
                   // value entered then 4 is used
BackFillData  = 24 // Number of hours to backfill data from the
                   // start of the NmxToCSS3 program
CSSPath       = cssTables  // root-directory of css tables (except
                           // wfdisc)
```

```
WaveformPath   = waveforms   // root-directory of waveform data files
                             // and wfdiscs
CSSPrefix      = global      // prefix for all css tables (except
                             // wfdisc)
NaqsStnFile    = c:\nmx\user\Naqs.stn  // the full path to the Naqs.stn
                                       // file
CalHistoryFile = calHistory          // the full path to the calibration
                                     // history file (from cd sender)


//////////////////////////////////////////////////////////////////////
[ Instrument ]

Type         = STS-2    // The type of the instrument
Name         = STS-2    // The name of the instrument
Dir          = responses // The directory containing the response file
ResponseFile = sro.fap   // The response file
SampleRate   = 100       // SampleRate for this response
NCalib       = 0.456     // Nominal Calibration
NCalper      = 1.0       // Nominal Calibration Period
ResponseType = fap       // Format of response file (paz, fap, fir,
                         // paxfir)


[ Instrument ]

Type         = HRD     // The type of the instrument
Name         = name2   // The name of the instrument
Dir          = responses // The directory containing the response file
ResponseFile = hrd.fap   // The response file
SampleRate   = 100       // SampleRate for this response
NCalib       = 0.556     // Nominal Calibration
NCalper      = 1.0       // Nominal Calibration Period
ResponseType = fap       // Format of response file (paz, fap, fir,
                         // paxfir)


[ Instrument ]

Type         = SRO     // The type of the instrument
Name         = name3   // The name of the instrument
Dir          = responses // The directory containing the response file
ResponseFile = sro.fap   // The response file
SampleRate   = 100       // SampleRate for this response
NCalib       = 0.356     // Nominal Calibration
NCalper      = 1.0       // Nominal Calibration Period
ResponseType = fap       // Format of response file (paz, fap, fir,
                         // paxfir)


//////////////////////////////////////////////////////////////////////
// Site blocks are optional
// If not specified then all the values are set to NA

[ Site ]

Site        = IDC01  // The site name
StationType = ar     // Type of station. (ss-single station, ar-array)
RefSite     = -      // The reference station for array members
                     // (- is NA)
DNorth      = 0.0    // Offset North from array reference in km
DEast       = 0.0    // Offset East from array reference in km


[ Site ]
```

```
        Site        = IDC02   // The site name
        StationType = ar      // Type of station (ss-single station, ar-array)
        RefSite     = IDC01   // The reference station for array members
                              // (- is NA)
        DNorth      = 1.5     // Offset North from array reference in km
        DEast       = 2.5     // Offset East from array reference in km


        [ Site ]


        Site        = IDC03   // The site name
        StationType = ar      // Type of station (ss-single station, ar-array)
        RefSite     = IDC01   // The reference station for array members
                              // (- is NA)
        DNorth      = 3.5     // Offset North from array reference in km
        DEast       = 4.5     // Offset East from array reference in km


        ////////////////////////////////////////////////////////////////////
        // SiteChan blocks are optional
        // If not specified the Edepth is set to 0.0


        [ SiteChan ]


        Channel         = IDC01.ch3  // The Channel dotted Name
        EmplacementDepth = 1.5        // Relative position to the elevation of
                                      // the site


        [ SiteChan ]


        Channel         = IDC01.ch2  // The Channel dotted Name
        EmplacementDepth = 2.5        // Relative position to the elevation of
                                      // the site


        [ SiteChan ]


        Channel         = IDC01.ch1  // The Channel dotted Name
        EmplacementDepth = 3.5        // Relative position to the elevation of
                                      // the site
```

Appendix B CSS 3.0 Tables

Table B-1 shows a summary of the CSS Version 3.0 tables used in NmxToCSS3, and the source of attribute values where applicable (for example, the `NmxToCSS3.ini` [Instrument] section *Name* parameter as the source for the `instrument` table *insname* attribute). For more information on the tables and attributes, see "CSS Version 3 Database: Schema Reference Manual" J. Anderson, W.E. Farrell, K. Garcia, J. Geven, H. Swanger, Rev 1.2. January 1995.

The files are written to the directory structure shown in Section B.2.

B.1 Summary of tables

Notes:

1) Attribute source `.ini` refers to the `NmxToCSS3.ini` configuration file.

2) The lddate format allows at most 17 chars (*yy*:*mm*:*dd hh*:*mm*:*ss*), therefore NmxToCSS3 uses digits to represent the month. For example, June is encoded as 06.

Table B-1 CSS Version 3.0 tables and attributes

Table	Description	Attributes
affiliation	The `affiliation` table provides the relation for clustering stations into networks. Entries are created whenever network or station configuration changes are made and the software is restarted.	• *net* – the network name, from `.ini` [NmxToCSS3] *StationCode*. • *sta* – station code name, from the CD1.*x* sender calibration history file (`calhistory.cdx`). • *lddate* – the date and time when the entry is created.

Table B-1 CSS Version 3.0 tables and attributes (Continued)

Table	Description	Attributes
instrument	The instrument table holds these values: • Nominal one-frequency calibration factors for each instrument. • Pointers to the nominal frequency-dependent calibration for an instrument. • Pointers to the exact calibrations obtained by direct measurement on a particular instrument. Entries are created whenever a user accepts calibration changes broadcast by the Calibration software.	• *inid* – a unique key generated by the table manager. • *insname* – from .ini [Instrument] *Name.* • *instype* – from .ini [Instrument] *Type.* • *band* – set to NA (-). • *digital* – recording system, set to "d" (digital). • *samprate* – from .ini [Instrument] *SampleRate.* • *ncalib* – nominal calibration factor, from .ini [Instrument] *NCalib.* • *ncalper* – nominal calibration period, from .ini [Instrument] *NCalper.* • *dir* – directory to store the frequency response file, from .ini [Instrument] *Dir.* • *dfile* – frequency response file, from .ini [Instrument] *ResponseFile.* • *rsptype* – response file format: fap. • *lddate* – the date and time when the entry is created.

lastid	Lastid is a reference table from which programs may retrieve the last sequential value of one of the table keys. Entries are created whenever a new entry is created in the corresponding table. Entries are updated if they already exist.	• *keyname* – chanid | inid | wfid
		  • chanid is used when an entry is created in .sitechan.
		  • inid is used when an entry is created in .instrument.
		  • wfid is used when an entry is created in .wfdisc.
		• *keyvalue* – the last assigned value of the counter for the specified keyname.
		• *lddate* – the date and time when the entry is created.

network	The network table gives general information about seismic networks. Entries are created when NmxToCSS3 detects network changes on startup.	• *net* – the network name, from .ini [NmxToCSS3] *StationCode.* • *netname* – set to NA (-). • *nettype* – set to NA (-). • *auth* – set to NA (-). • *commid* – set to NA (-1). • *lddate* – the date and time when the entry is created.
remark	The remark table is empty.	not applicable

Table B-1 CSS Version 3.0 tables and attributes (Continued)

Table	Description	Attributes
sensor	The sensor table provides specific calibration information for physical channels, including a record of updates in the calibration factor or clock error of each instrument, and links a sta/chan/time to a complete instrument response in the .instrument table. An entry is created for each instrument, and for calibration changes.	• *sta* – from the CD1.*x* sender calibration history file (CalHistory.cd*x*). • *chan* – from the CD1.*x* sender calibration history file (CalHistory.cd*x*). • *time* – the time when a record is created, in seconds and fractions of a second since January 1, 1970. • *endtime* – set to NA initially (+9999999999.999), and set to the new time whenever a new record is created. • *inid* – corresponds to instrument id record. • *chanid* – corresponds to a sitechan id record. • *jdate* – the time (see above), in Julian format (*yyyyddd*). • *calratio* – calibration ratio, of *ncalib* from .instrument to the current calibration factor. • *calper* – the current calibration period, as broadcast by the Calibrate software (typically 1 Hz). • *tshift* – set to zero. • *instant* – set to y. • *lddate* – the date and time when the entry is created.
site	The site table provides information about the location of a station. Entries are created when NmxToCSS3 detects station configuration changes on startup.	• *sta* – from the CD1.*x* sender calibration history file (CalHistory.cd*x*). • *ondate* – date the station began operating; initially set to 24 hours before the start of the backfill date, unless the record already exists then uses the date when the record was created. • *offdate* – set to NA (-1) initially. Change to the date when creating a new record. • *lat* – Naqs.stn [Station] *Latitude.* • *lon* – Naqs.stn [Station] *Longitude.* • *elev* – Naqs.stn [Station] *Elevation.* • *staname* – Naqs.stn [Station] *Description.* • *statype* – set to NA (-). • *refsta* – array reference site for beams, from .ini [Site] *RefSite*, or set to NA (-). • *dnorth* – offset North from the array reference, from .ini [Site] *DNorth*, or set to NA (0.0). • *deast* – offset East from the array reference, from .ini [Site] *DEast*, or set to NA (0.0). • *lddate* – the date and time when the entry is created.

Table B-1 CSS Version 3.0 tables and attributes (Continued)

Table	Description	Attributes
sitechan	The sitechan table describes the orientation of a recording channel at the site referenced by station. Entries are created when site channel configuration changes are detected.	• *sta* – from the CD1.*x* sender calibration history file (calhistory.cd*x*). • *chan* – from the CD1.*x* sender calibration history file (calhistory.cd*x*). • *ondate* – date the sensor began operating; initially set to 24 hours before the start of the back-fill date, unless the record already exists then uses the date when the record was created. • *chanid* – a unique key generated by the table manager. • *offdate* – set to NA (-1) initially. Change to the date when creating a new record. • *ctype* – channel type, set to "n" (normal). • *edepth* – relative position to the elevation of site. Set to .ini [SiteChan] *ElevationDepth*, or to 0.0 if the parameter *ElevationDepth* isn't in the .ini file. • *hang* – converted from Naqs.stn [ChannelPrototype] *Azimuth.* • *vang* – converted from Naqs.stn [ChannelPrototype] *Dip.* • *descrip* – channel description; set to NA (-). • *lddate* – the date and time when the entry is created.

Table B-1 CSS Version 3.0 tables and attributes (Continued)

Table	Description	Attributes
wfdisc	The wfdisc table provides an index to the waveform binary files stored on disk. Records for contiguous data segments are consolidated into a single record, provided that those segments are all contained within the same binary file.	• *sta* – from the CD1.*x* sender calibration history file (calhistory.cd*x*). • *chan* – from the CD1.*x* sender calibration history file (calhistory.cd*x*). • *time* – start time of the data, in seconds and fractions of a second since January 1, 1970. • *wfid* – a unique identifier for a segment of digital waveform data, generated by the table manager. • *chanid* – corresponds to sitechan id record. • *jdate* – start date of the data, in Julian format (*yyyyddd*). • *endtime* – the time of the last sample in the waveform file. • *nsamp* – the number of samples in a waveform segment. • *samprate* – the current sample rate, updated with data broadcast by the Calibrate software. • *calib* – the current calibration factor, updated with data broadcast by the Calibrate software; valid at the oscillation period specified by *calper*. • *calper* – the period for which *calib*, *ncalib*, and *calratio* are valid, updated with data broadcast by the Calibrate software. • *instype* – from .ini [Instrument] *Type*. • *segtype* – waveform indexing method; always set to "o" (original). • *datatype* – s4 uncompressed data. • *clip* – c	n, to indicate whether the data were clipped or not. • *dir* – directory to find the waveform data files, from .ini [Parameters] *WaveformPath*. • *dfile* – waveform file name. • *foff* – the byte offset of a waveform segment within a data file, used when data are multiplexed. • *commid* – set to NA (-1). • *lddate* – the date and time when the entry is created.

B.2 Table directory structure

The CSS tables are written to the directory *CSSPath*, and the waveform files are written to subdirectories by date under the directory *WaveformPath* (the paths are defined in the [Parameters] section of the `NmxToCSS3.ini` file). For example:

—*CSSPath*/
——*CSSPrefix*`.affiliation`
——*CSSPrefix*`.instrument`
——*CSSPrefix*`.lastid`
——*CSSPrefix*`.network`
——*CSSPrefix*`.remark`
——*CSSPrefix*`.sensor`
——*CSSPrefix*`.site`
——*CSSPrefix*`.sitechan`

—*WaveformPath*/
——*StationCode*/
———`2004/`
————`05/`
—————`26/`
——————`20040526.wfdisc`
——————`IDC01ch1_20040526_04_120.w`
——————`IDC01ch1_20040526_16_120.w`
——————`IDC01ch2_20040526_04_120.w`
——————`IDC04ch1_20040526_04_40.w`
——————`…`
—————`27/`
——————`20040527.wfdisc`
——————`IDC01ch1_20040527_16_120.w`
——————`IDC01ch2_20040527_16_120.w`
——————`…`