

# CCONFSRV(8)

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>SYNOPSIS</b>	<b>1</b>
<b>2</b>	<b>DESCRIPTION</b>	<b>2</b>
<b>3</b>	<b>VARIABLE SUBSTITUTION</b>	<b>3</b>
<b>4</b>	<b>API - get section</b>	<b>4</b>
<b>5</b>	<b>API - list sections</b>	<b>6</b>
<b>6</b>	<b>EXIT STATUS</b>	<b>7</b>
<b>7</b>	<b>EXAMPLE</b>	<b>8</b>
<b>8</b>	<b>BUGS</b>	<b>9</b>
<b>9</b>	<b>SEE ALSO</b>	<b>10</b>
<b>10</b>	<b>COPYING</b>	<b>11</b>

## Chapter 1

# SYNOPSIS

**cconfsrv**

## Chapter 2

# DESCRIPTION

This is sepcial XATMI server shipped with Enduro/X in order to provide common-configuration (ini file) parsing facilities to all programming languages which are provided by Enduro/X. The idea behind the configuration server is that it uses XATMI API (exposed as service) in order to get configuration section or list all configuration that is used by application.

This server is optional and can be used by third party application in oder to read the configuration files via standard XATMI service calls.

---

## Chapter 3

# VARIABLE SUBSTITUTION

Several parameters in the `ndrxconfig.xml` file are processed via substitution engine. Engine processes puts the environment variables or special functions variables and `$(PARAMETER)` for functions. The value can be escaped with

## Chapter 4

# API - get section

Get mode (*EX\_CC\_CMD* UBF buffer set to *g* (default)). In this mode the request for configuration is made for particular section. The config files (or directories) for which to do the lookup can be set in *EX\_CC\_RESOURCE* field, you may use multiple occurrences. If field is not set, then lookup is done in all files which are loaded into configuration library. This includes Enduro/X own config files, which are set in **NDRX\_CCONFIG** environment variables (see *ex\_env(5)*). The get request must contain *EX\_CC\_LOOKUPSECTION* field, which indicates the section to lookup.

*EX\_CC\_LOOKUPSECTION* can be set to lookup the sub-sections, for example if you have in config files:

/paht/to/test1.ini:

```
[TEST1]
PARAM1=value1
PARAM2=value2

[TEST1/SUB1]
PARAM3=value3
PARAM5=value5
```

/path/to/test2.ini:

```
[TEST1/SUB1/SUB2]
PARAM3=value4
```

Then by doing lookup to *TEST1/SUB1/SUB2* section, you will get following result:

Request buffer:

```
EX_CC_CMD      g
EX_CC_LOOKUPSECTION TEST1/SUB1/SUB2
EX_CC_RESOURCE  /paht/to/test1.ini
EX_CC_RESOURCE  /paht/to/test2.ini
```

Response buffer:

```
EX_CC_SECTION   test1
EX_CC_KEY       PARAM1
EX_CC_VALUE      value1
EX_CC_SECTION   test2
EX_CC_KEY       PARAM2
EX_CC_VALUE      value2
EX_CC_SECTION   TEST1/SUB1/SUB2
EX_CC_KEY       PARAM3
EX_CC_VALUE      value4
EX_CC_SECTION   TEST1/SUB1
EX_CC_KEY       PARAM5
EX_CC_VALUE      value5
```

Mention that *PARAM3* value is set to *value4* and not *value3*, because *TEST1/SUB1/SUB2* closer matched the lookup request. The principle is that all parent variables are gathered. If any variable is common between parent and child, then child element will be returned, as it is with higher priority.

When doing requests to configuration server it is possible to request for mandatory field keys checking. The must-have keys are loaded into *EX\_CC\_MANDATORY* field occurrences. If during section lookup key is not found, then error 6 is returned.

It is possible request format verification. The key-format fields are used into *EX\_CC\_FORMAT\_KEY* + *EX\_CC\_FORMAT\_FORMAT* occurrences. If the format does not match field, error code 7 is returned.

***EX\_CC\_FORMAT\_FORMAT* field can have following values:**

**t** - Boolean field, valid values starts with on of the following symbols: *TtFfYyNn10*.

**sX..Y** or **s..Y** (e.g. s10..20, s..5) - String field, min length X and max length Y. If X is not set (second format), then X is assumed to be 0.

**iX..Y** or **i..Y** (e.g. i1..2, i..3) - Integer field, min len (measured as string) X and max len (measured as string Y).

**nX..Y** or **n..Y** (e.g. n2..5, n..3) - Numeric field (can be floating point format). Min length X (measured as string) and max length Y (measured as string).

---

## Chapter 5

# API - list sections

It is possible to enter in conversational mode in request sections which begins with certain phrase or it is possible to list all sections for given resources. To enter is list mode you need to call the configuration server with *EX\_CC\_CMD* UBF buffer field set *l*. If this is the only field then all sections from default Enduro/X configuration files will be returned.

The sections which starts with phrase can be set in *EX\_CC\_LOOKUPSECTION* UBF field occurrences. Resource to lookup can be specified in *EX\_CC\_RESOURCE*.

## Chapter 6

# EXIT STATUS

When error occurs, the configuration service returns **TPESVCFAIL**, and *EX\_NERROR\_CODE* is set to corresponding error code. *EX\_NERROR\_MSG* have some additional information about error condition.

**EX\_NERROR\_CODE** values:

- 1** Invalid INI file(s).
  - 2** Malloc failed (system have low of memory or some kind of other system related problem).
  - 3** Unix error occurred.
  - 4** Invalid arguments passed.
  - 5** System failure.
  - 6** Mandatory field is missing. *EX\_NERROR\_MSG* specifies the exact problematic field.
  - 7** Field value format error. *EX\_NERROR\_MSG* specifies the exact problematic field.
-

## Chapter 7

# EXAMPLE

See `atmitest/test030_cconfsrv/atmict30.c` for sample code.

## Chapter 8

# BUGS

Report bugs to [support@mavimax.com](mailto:support@mavimax.com)

## Chapter 9

## SEE ALSO

`ex_env(5)`, `ndrxconfig.xml(5)`

## **Chapter 10**

# **COPYING**

© Mavimax, Ltd