

# estos ECSTA for Mitel 100

---

6.0.14.1079

1	Introduction.....	4
2	Mitel 100.....	6
2.1	General Settings.....	6
2.2	Advanced Settings.....	6
2.3	TAPI Lines .....	7
2.4	Configure telephone system .....	8
2.5	Supported TAPI operations.....	8
3	Set of rules for filtering out SIP lines.....	13
4	Licenses.....	15
5	Location Settings.....	16
6	Phone Number Format .....	17
7	Driver Management.....	19
8	PBX systems.....	20
9	Info about estos ECSTA for Mitel 100 .....	21

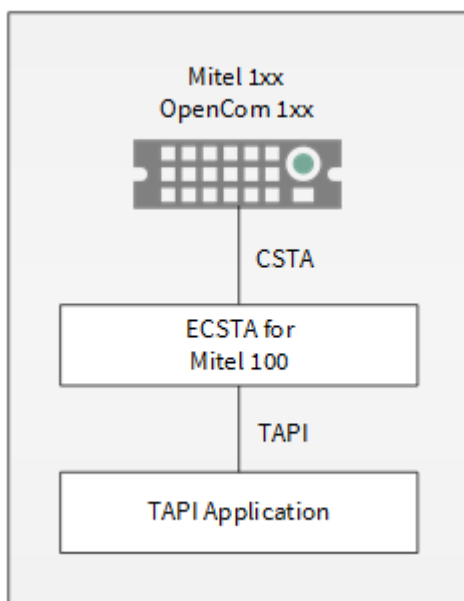


## 1 Introduction

estos ECSTA for Mitel 100 makes available several Telephony Service Providers for Microsoft® TAPI 2.1 (also 2.2 and 3.0). These TAPI drivers produce a central communication between a PC and the telephone system. On this server all devices on the telephone system are available as TAPI lines.

All estos ECSTA for Mitel 100 drivers use the efficient interface CSTA in order to communicate with the telephone system.

For the installation and management of the driver see Driver Management.



### Supported Telephone Systems

Driver	PBX
ECSTA for Mitel 100	Mitel 1xx
ECSTA for Mitel 100	Mitel OpenCom 1xx
ECSTA for Mitel 100	Mitel OpenCom 510
ECSTA for Mitel 100	Mitel OpenCom X320
ECSTA for Mitel 100	Aastra 800

### Using the driver in the network

After installing the driver, you may use all configured line devices on the server as TAPI-lines. If you are using a server based CTI Software, you may install it now. To have the TAPI-devices available on all workstations, the TAPI-driver must be distributed in the network. There are the following options:

1. **estos UCServer**

The estos UCServer is a CTI/UCC server software which distributes the telephony information across

the network using a graphical client or a multi-line TAPI driver. Workgroups, domains, Active Directory® and Terminal Server are supported.

2. **estos CallControlGateway**

The estos CallControlGateway for Lync/OCS is a gateway that connects the TAPI-lines with the Lync/OCS server. This enables the Lync/OCS client to use traditional telephone devices via Remote Call Control (RCC).

3. **Windows® Telephony Server**

The Windows® Telephony Server is a component of Windows® server operating systems since Windows® NT. You will need a domain. The service is limited to domain users.

## 2 Mitel 100

When setting up the driver, follow these steps:

1. **Check network connection**  
If the connection is established using TCP/IP, check if you can reach the PBX system (ping hostname)
2. **Test CSTA License in the telephone system**  
An OpenCSTA 100 license is required in the telephone system. This license can be activated as a demo license in the system.
3. **Test device ID in the telephone system**  
If several telephones have the same phone number, then this device ID must be obtained. See Configure Telephone system.
4. **Driver installation**  
See also Settings

### 2.1 General Settings

The connection between PBX and Server can be established using a TCP/IP connection to the LAN interface of the PBX.

#### Connection using a network (TCP/IP)

Input here the host name or the IP address of your telephone system. The standard TCP Port for the CSTA connection is 7001.

With the Button 'Find...' you can localize the telephone system in the network.

#### User name and password

To register at the telephone system, you employ the User **Administrator**.

Input the password

### 2.2 Advanced Settings

#### Read parallel devices / line keys

If this option is activated the driver will query all physical devices for a line number and map them as TAPI lines. Please observe the following precautions when using line keys and parallel devices simultaneous.

- You do not use line keys or parallel devices  
  
Disable the function.
- Line keys are used:  
  
Disable the function.  
Line keys are not supported in this configuration. The driver can not distinguish between the primary telephone number and a line key. With configured line keys the driver will create a TAPI line for each key which is configured on a phone. Each line will have the same number. Only the primary device the pbx is reporting for a line number will be mapped.
- Parallel devices are used  
  
Activate the function.
- Line keys and parallel devices are used.

The driver can not distinguish between parallel devices and a line key. You may activate the function but you will need to map the correct line to a user by hand. To facilitate this you may configure Device IDs within the phone system. The Device ID should have the same id as the line number. Within the line list of the TAPI subsystem you can now see which line is referenced on which device. (E.G. line 11 on device with primary line 11 will be shown with name 11;11 (1. numeral is the number, 2. numeral is the Device ID). With parallel devices add ascending numbers to the DeviceID, 1101 for the first device with number 11, 1102 for the 2. Device with number 11. (In the line list you will find lines named 11;1101 and 11;1102)

#### **Readout line names**

If this option is active, the names of the telephones from the telephone system are requested.

#### **Node numbers**

If your PBX system is using node numbers you can enter these numbers semicolon separated. These node numbers are removed from numbers presented by the pbx.

#### **Emulate blind transfer**

If the PBX system does not offer blind transfer the driver can emulate this feature. The driver will setup a consultation transfer which is merged as soon as the called destination is rings.

#### **Blind transfer timeout**

Timeout for the blind transfer emulation. The consultation call must be answered within this timeout. Otherwise the consultation call is canceled and the initial call is retrieved.

#### **Rules for detecting SIP lines**

Choosing "Edit Rules" takes you to a dialog box that allows you to specify whether SIP lines should be detected by the ECSTA via a set of rules, filtered out and thus not forwarded to the TAPI's line management (See Set of Rules for Filtering Out SIP Lines).

#### **Snapshot for active calls**

The driver can validate existing calls. This prevents that calls are displayed on the PC that do no more exist in the PBX system.

Enter a time interval in seconds.

The higher this value the longer it takes for the driver to detect a hanging call. The lower this value the higher is the load on the PBX.

#### **Retry MonitorStart**

In the event that the driver cannot start monitoring an extension because, for example, the extension has not been connected to the PBX, the driver will attempt to start monitoring at periodic intervals.

Enter a time interval in seconds.

## **2.3 TAPI Lines**

Enter the lines that the driver should make available through TAPI. The numbers are the internal phone numbers for your devices. The names will be used to display the TAPI lines.

The device ID is necessary, in case several telephones have the same number.

#### **Phone Number Format**

The lines can optionally be indicated as an internal phone number, for example 123 or canonical +49(30)12345-123.

You should use canonical phone numbers if you use several instances of TAPI driver and thus switch on several telephone systems at the same time. A canonical phone number consists as follows:

*+Country code (prefix) Local prefix-extension*

In the dialog box Location You can configure the country code, area code and local dialing prefix.

### Load Lines

The available lines are determined automatically from the telephone system.

All phone numbers from the phone system that can be operated from the PC are selected. If several phones have the same phone number, possibly only one of these phones is selected. All other phones must be configured manually.

### Manual addition of a line

Give the internal phone number of the telephone. If necessary, give the device ID of the telephone. Optionally a name can also be given.

### Device ID

The device ID serves to identify the telephone clearly. This is required if several telephones have the same phone number.

### Import text

You can import a list of lines from a text file. The file must begin every line with the phone number. Optionally, the name can also be split with a comma.

### Export text

You may export the actual line configuration to a text file.

### Attention

After the installation of the driver it may be necessary to restart the computer.

## 2.4 Configure telephone system

### Several telephones with the identical phone number

The telephone system supports the parallel connection of several telephones with identical phone numbers. Control of single telephones must be distinguishable. A device ID is required for this. If no device IDs are configured, only one of the telephones with an identical phone number can be controlled.

### Device ID

The Device ID configuration is located in the PBX settings under Configurator - Telephony - Extended - Device ID.

All multiple occurring phone numbers must obtain a device-ID. Only then are they usable from the PC.

Example of sensible device ID assignment: The call number 33 is assigned to three terminals. Terminal 1 receives the device ID 3301, terminal 2 the ID 3302 and terminal 3 the ID 3303.

Ph.No.	Description	Type (configured)	Port	Device ID
33		Aastra 6773ip (OP73IP)	0030420CEE8E	3301
33		Aastra 6775ip (OP75IP)	0030420CEDC5	3302

### CSTA Licenses

An OpenCSTA 100 license is required in the telephone system. This license can be activated as a demo license in the system.

The license settings in the configuration of the telephone system can be found under configurator - System - Licenses.

## 2.5 Supported TAPI operations

The driver offers the following TAPI call control operations.

SIP devices do not support all the features listed here.



Action:	Corresponding TAPI function:
Dial	TSPI_lineMakecall
Hang up	TSPI_lineDrop
Answer call	TSPI_lineAnswer
Hold call <sup>2</sup>	TSPI_lineHold
Retrieve call <sup>2</sup>	TSPI_lineUnhold
Redirect call	TSPI_lineRedirect
Blind transfer <sup>23</sup>	TSPI_lineBlindTransfer
Setup transfer <sup>2</sup>	TSPI_lineSetupTransfer
Swap hold <sup>2</sup>	TSPI_lineSwapHold
Complete transfer <sup>2</sup>	TSPI_lineCompleteTransfer (LINETRANSFERMODE_TRANSFER)
Create conference <sup>2</sup>	TSPI_lineCompleteTransfer (LINETRANSFERMODE_CONFERENCE)
Add to conference <sup>2</sup>	TSPI_lineAddToConference
Remove from conference <sup>2</sup>	TSPI_lineRemoveFromConference
Pickup <sup>2</sup>	TSPI_linePickup
DTMF dialing <sup>2</sup>	TSPI_lineGenerateDigits
Dial in dial tone mode	TSPI_lineDial
Forwarding	TSPI_lineForward TSPI_lineGetAddressStatus <ul style="list-style-type: none"> <li>• LINEFORWARDMODE_UNCOND</li> <li>• LINEFORWARDMODE_UNCONDEXTERNAL<sup>2</sup></li> <li>• LINEFORWARDMODE_UNCONDINTERNAL<sup>2</sup></li> <li>• LINEFORWARDMODE_BUSY<sup>2</sup></li> <li>• LINEFORWARDMODE_BUSYEXTERNAL<sup>2</sup></li> <li>• LINEFORWARDMODE_BUSYINTERNAL<sup>2</sup></li> <li>• LINEFORWARDMODE_NOANSW<sup>2</sup></li> <li>• LINEFORWARDMODE_NOANSWEXTERNAL<sup>2</sup></li> <li>• LINEFORWARDMODE_NOANSWINTERNAL<sup>2</sup></li> </ul>

Do not disturb <sup>2</sup>	TSPI_lineForward TSPI_lineGetAddressStatus
Message waiting <sup>2</sup>	TSPI_lineSetLineDevStatus TSPI_lineGetLineDevStatus

<sup>2</sup>These functions are not supported by SIP phones.

<sup>3</sup>Emulated by the driver (System does not natively support this feature)

The following TAPI functions are implemented due to requirements of the TAPI subsystem.

**Other functions exported by the driver:**

TSPI\_lineSendUserUserInfo

TSPI\_lineClose

TSPI\_lineCloseCall

TSPI\_lineConditionalMediaDetection

TSPI\_lineDevSpecific

TSPI\_lineDevSpecificFeature

TSPI\_lineGetDevConfig

TSPI\_lineSetDevConfig

TSPI\_lineGetAddressCaps

TSPI\_lineGetAddressStatus

TSPI\_lineGetAddressID

TSPI\_lineGetCallAddressID

TSPI\_lineGetCallInfo

TSPI\_lineGetCallStatus

TSPI\_lineGetDevCaps

TSPI\_lineGetExtensionID

TSPI\_lineGetIcon

TSPI\_lineGetID

TSPI\_lineGetNumAddressIDs

TSPI\_lineNegotiateExtVersion

TSPI\_lineNegotiateTSPIVersion

TSPI\_lineOpen

TSPI\_lineSelectExtVersion

TSPI\_lineSetDefaultMediaDetection

TSPI\_lineSetStatusMessages

TSPI\_lineSetAppSpecific

TSPI\_lineSetCallData

TSPI\_providerCreateLineDevice

TSPI\_providerEnumDevices

TSPI\_providerFreeDialogInstance

TSPI\_providerGenericDialogData

TSPI\_providerInit

TSPI\_providerShutdown

TSPI\_providerUIIdentify

TSPI\_lineGetCallIDs

TUISPI\_lineConfigDialog

TUISPI\_lineConfigDialogEdit

TUISPI\_providerConfig

TUISPI\_providerInstall

TUISPI\_providerRemove

TUISPI\_providerGenericDialog

TUISPI\_providerGenericDialogData

### 3 Set of rules for filtering out SIP lines

You can enter rules that determine whether lines should be filtered out during readout or not. If "no detection of SIP lines" is selected, all read-in lines will be included in the line management. If "Rules for detecting SIP lines" is selected, a set of rules is applied when reading in lines, which determines whether a line should be filtered out or included in the line management. This set of rules consists of individual rules created by the user.

For each rule it can be defined whether a configurable character string can be searched for in the name or call number, in order to filter out the line in the case of a match. Alternately, a regular expression can also be defined via which a search pattern is determined which is then applied in the search in the line name or the line number. If the search is successful the line will be filtered out.

The following rules can be configured:

- Filtering out of lines via a character string in line names  
 When reading in the lines, those lines are not included in the line management whose line name contains the character string entered in the "Search for:" column.  
 Example: If lines whose line name contains the character string "Paul" should be filtered out, all lines having the line names "Paul Sample", "Pauline Sample", "Lukas Paulsen"... will be filtered out.
- Filtering out of lines via a character string in the line call number  
 When reading in the lines, those lines are not included in the line management whose line call number contains the character string entered in the "Search for:" column.  
 Example: If lines whose line number contains the character string "897" should be filtered out, lines with the line numbers "897", "089123456897", "0897123456", would be filtered out.
- Filtering out of lines via a search pattern in line names which is configured via a regular expression  
 The character string in the "Search for:" column must be a regular expression. When reading in the lines, those lines for which the search was successful by means of the regular expression in the line name are not included in the line management.  
 Examples of regular expressions are listed further below.
- Filtering out of lines via a search pattern in the line call number which is configured via a regular expression  
 The character string in the "Search for:" column must be a regular expression. When reading in the lines, those lines for which the search was successful by means of the regular expression in the line call number are not included in the line management.  
 Examples of regular expressions are subsequently listed.

Here is a brief overview about the syntax of some regular expressions: A brief overview of the permitted expressions:

Character	regular expression	Description
-----------	--------------------	-------------

^	^o or ^Max	The circumflex character (^), followed by a character string in a regular expression, checks whether the character string is at the beginning of the call number or name. Using the regular expression "^o" causes the program to check whether the digit 'o' is at the beginning of the call number, or with the regular expression "^Max", whether the name begins with the character string "Max". In both cases the line would be filtered out.
\$	15\$ or mann\$	A character string followed by a dollar sign (\$) in a regular expression checks whether the character string is at the end of the call number or name. The regular expression "15\$" checks whether the character sequence "15" appears at the end of the call number, or the regular expression "mann\$" checks whether the name ends with the character string "mann". In both cases the line is filtered out.
	64 72 73 or Max mann lich	The vertical pipe character ( ) separates two or more alternative character strings in the search. A match occurs when one of the alternatives applies. With the regular expression "64 72 73" a search for the numerical sequence 64 or 72 or 73 is performed anywhere in the call number. With the regular expression "Max mann lich" a search in the character string "Max" or "mann" or "lich" is performed anywhere in the name. If the search is successful the line is filtered out.
.	5.6 or M.er	The dot (.) in the regular expression is a placeholder for any character in this position. A search with the regular expression "5.6" will find the digit sequences 506, 516, 526... in the call number. A search with the regular expression "M.er" will find the character sequences "Meyer", "Meier", "Mayer",... in the name. If the search is successful the line will be filtered out.
[]	[123] or [asd]	The square brackets ([ ] and (]) indicate a quantity of characters to be searched for in the call number or name. A search with the regular expression "[123]" will filter out all call numbers in which the digits 1, 2 or 3 occur. A search with the regular expression "[asd]" will filter out all names in which the letters a, s or d occur.

**Check:**

You can directly check your set of rules in the "Rules for detecting SIP lines" dialog box. Enter a character string in the Search Text field that you want to test with the set of rules. In the "Detected as SIP line:" field, you can see whether a rule for filtering was successfully applied to the search text.

See also Advanced Settings.

## 4 Licenses

You can test the driver for 45 days with 25 lines without a license and without other limitations. Once the trial period has expired, the lines will automatically be removed. You can purchase licenses codes and activate the driver permanently with them. To purchase licenses, please contact your dealer.

## 5 Location Settings

The phone numbers of the telephone system are defined in the Location Settings

### **Extension Range**

Indicate here which internal phone numbers exist in the phone system. This information is used to determine the available lines from the phone system. The internal phone numbers of the telephone system and the phone number information of a call are used to determine whether it is an internal or an external call. In addition, when lines are generated automatically, estos UCServer checks whether the number of the line to be generated automatically lies within the range of internal numbers. Only then is the line generated by estos ECSTA for Mitel 100 and the TAPI notified.

### **Location**

Indicate here which international phone numbers precede the internal extension numbers. The input of this location information should only be carried out if you use several driver instances and have connected several telephone systems. This location information can only be configured when the option Use Location is switched on. If you enter a location here, all extension phone numbers are entered fully international.

Example: With location information '49(89)1234' the extension '100' is formatted as '49(89)1234-100'.

### **Phone Number Format**

The phone number registered from the driver at the application can be changed with rules. Moreover, you can also change phone numbers sent by the PC to the telephone system. See Phone number formatting.



## 6 Phone Number Format

You may enter rules for formatting the phone numbers.

The phone numbers that are reported from the PBX system to the PC and the numbers that are send from the PC to the PBX system may be modified with a Search and Replace function.

The phone numbers that are sent from the Tapi Application to the PBX (make call) may be modified as well.

Each line in the list contains an option if direct text compare or a regular expression is used.

The Search and Replace function uses regular expressions. If a search pattern matches, the result from replace with will be used as output. If the search pattern does not match, the original number will be reported unchanged. The entries in the list are processed one after each other. If one match is found the remaining enties will be ignored.

There are three categories:

- **Incoming**  
This section is for phone number from incoming calls that are reported from the PBX system to the PC.
- **Outgoing**  
This section is for phone number from outgoing calls that are reported from the PBX system to the PC.
- **PC Dialing**  
This section is for phone numbers that are dialed on the PC and send to the PBX system

### Search for:

Enter the regular expression that will be used to find a phone number.

Hint: The caret (^) can be found on the upper left key of a keyboard with German layout.

An overview of allowed expressions:

Character	Description
^	The beginning of the phone number. The expression "^o" matches 'o' only at the beginning of a phone number.
^	The caret (^) immediately following the left-bracket has a different meaning. It is used to exclude the remaining characters within brackets from matching the target string. The expression "[^o-8]" indicates that only digits between o and 8 are allowed.
\$	The dollar sign (\$) will match the end of the string. The expression "152\$" will match the substring "152" only if it is at the end of the string.
	The ( ) character allows both characters between which it is located. The expression "8 9" allows '8' or '9'.
.	The dot (.) allows any character (or any digit).
*	The asterisk (*) indicates that the character to its left must be present o times or more.

+	The Plus sign (+) is similar to the asterisk, only the character to the left must be present at least once.
?	The question mark (?) indicates that the character to the left must be present 0 or 1 times.
()	The parenthesis affects the order of pattern evaluation and also serves as a tagged expression that can be used when replacing the matched sub-string with another expression.
[]	The corner brackets ([]) indicates the amount of signs that are permitted at this point.

**Replace with:**

Enter the expression that defines how the number is to be formatted.

\1 represents the first matched expression enclosed by parentheses '(')' from the *search pattern* field.

\2 the second...

**Check:**

You can check your input directly here. Enter a call number in the Call Number field that you wish to test with the Search/Replace. In the Output field the modified call number is displayed. If there is an error in the regular expression, it is indicated here. If the expression from Search for is not found, the call numbers will be output directly without any changes.

**Examples:**

Result	Search for	Replace with
Remove a leading 0 from the phone number	^0(.*)	\1
Replace a leading 80 at the beginning of the phone number by a 0	^80(.*)	0\1
Remove a private pin number that may be identified by a 50 followed by a 3 digit pin.	^50[0-9][0-9][0-9](.*)	\1
Suppress all internal numbers having a 3-digit extension.	^[0-9][0-9][0-9]\$	
Add an access code (leading 0) to all numbers with more than 3 digits (e.g. all external numbers).	^([0-9][0-9][0-9].+)	0\1
Add the PBX system root number (03012345) to all internal numbers (with 1 to 3 digits)	^([0-9][0-9]?[0-9]?)\$	03012345\1
Adding an area code to all numbers not beginning with 0 and containing at least 4 digits (thus not internal).	^[^0][0-9][0-9][0-9].*	08151\1

See also Location.

## 7 Driver Management

### Installation

The driver is installed on the system with the Windows® Installer (msi) package. If the set-up package contains several drivers, you can select which drivers should be installed during the set-up.

### Driver instance

The driver can be used to connect one or several phone systems. For this purpose one or several driver instances are registered in Windows® Tapi System.

### Register at Tapi System

During the installation, an instance from the selected driver is already registered at the Tapi system. Thereby, you select the necessary data in a Wizard in order to connect the driver with the telephone system.

### Configuration of Driver instance

The configuration of the driver instances takes place either over *Telephone and Modem option* in the Control Panel or over the standard program *Advanced Telephone Driver option* that can be found in the Control Panel or in the Start Menu.

### Add further driver instances

If you want to connect the computer with a further telephone system, you must add a further instance of the driver to the Tapi system. This takes place with Help in the standard program *Advanced Telephone Driver options* that can be found in the Control Panel or in the Start Menu.

### Updates

To install an Update start the Windows® Installer Package (msi).

If a driver instance is configured, this is deleted during the Update and is automatically added again after the Update is completed. If several driver instances are configured, the computer must be restarted after the Update.

### De-installation

The Deinstallation takes place via the Windows® Software Administration. In the Deinstallation, all instances of the driver are removed from the Tapi system and the software uninstalled.

## 8 PBX systems

Driver	PBX
ECSTA for Mitel 100	Mitel 1xx
ECSTA for Mitel 100	Mitel OpenCom 1xx
ECSTA for Mitel 100	Mitel OpenCom 510
ECSTA for Mitel 100	Mitel OpenCom X320
ECSTA for Mitel 100	Aastra 800

## 9 Info about estos ECSTA for Mitel 100

estos ECSTA for Mitel 100 is a product of estos GmbH.

Copyright (C) 2023 estos GmbH.

Product updates can be found under <https://www.estos.de/>

Please find FAQ's and support under <https://support.estos.de>

Active Directory®, Microsoft®, Windows Server®, Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

All brands and product names used in this document are for identification purposes only and may be trademarks or registered trademarks of their respective owners.