

# estos ECSTA for Mitel MiVoice Office 400

---

7.0.1.1179

1	Introduction.....	4
2	Software requirements .....	5
3	Driver administration.....	6
4	Installation.....	7
5	PBX Configuration .....	8
6	Mitel MiVoice Office 400.....	9
6.1	Connection settings.....	9
6.2	Line Settings .....	9
6.3	Advanced Settings.....	10
6.3.1	Phone number formatting.....	10
6.4	Licenses .....	12
7	Supported TAPI operations.....	13
8	Set of rules for filtering SIP lines.....	16
9	Info about estos ECSTA for Mitel MiVoice Office 400 .....	19

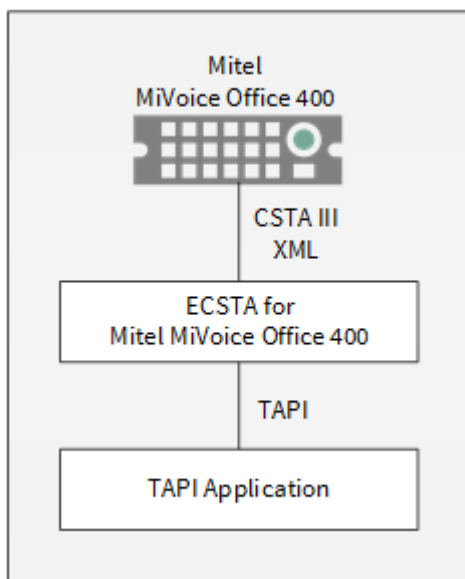


## 1 Introduction

estos ECSTA for Mitel MiVoice Office 400 offers a Telephony Service Provider (TSP) for Microsoft® TAPI 2.1 (also 2.2 and 3.0). This TAPI driver implements a central communication between a PC and the PBX system. estos ECSTA for Mitel MiVoice Office 400 uses the CSTA protocol for connecting to the PBX system.

For installation of the driver see Driver Installation.

For administration of the driver see Driver Administration.



### Using in the network

Once the driver has been installed, you can use all extensions set up on the server as TAPI lines. If you are using server-supported CTI software, you can now install this to be able to control the telephones from all workstations. For example the estos UCServer.

- **estos UCServer**

The estos UCServer is a CTI/UCC server software which distributes the telephony informations across the network using a graphical client or a multiline TAPI driver. Workgroups, domains, Active Directory® and Terminal Server are supported.

## 2 Software requirements

An Mitel MiVoice Office 400 PBX system version 3.1 or higher is required.

The estos ECSTA for Mitel MiVoice Office 400 can be installed on all Windows® systems that support Microsoft® TAPI 2.0 or higher. The following systems are supported in 64-bit versions.

- Windows® 10 64bit
- Windows® 11 64bit
- Windows Server® 2016
- Windows Server® 2019
- Windows Server® 2022

## 3 Driver administration

### Installation

Using the Windows® Installer (msi) packet, the driver will be installed on the system.

### Driver instance

The driver can be used to connect one or more PBX systems. For each PBX system you want to connect, you must configure one instance of the driver.

### Register at Tapi System

During the installation, an instance in the already selected driver is registered at the Tapi System. For this purpose you indicate 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 via *Telephone and Modem options* in the Control Panel or via the provided program *Advanced Telephone Driver options*, that can be found in the Control Panel or Start menu.

### Updates

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

If an instance of the driver is configured, it will be removed during the update process. It will be automatically re-added afterwards.

### Deinstallation

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

## 4 Installation

For installation and administration of the driver see Driver management.

When setting up the driver, follow these steps:

1. **PBX Configuration**  
Please check the PBX Configuration.
2. **Run driver setup**  
Launche the MSI installation package to install the driver. There is an installer for 64bit operating systems.
3. **Establish the connection to the PBX system**  
Configure the connection to the PBX system. Each instance of the driver supports the connection to one PBX system.
4. **Line configuration**  
The driver automatically offers all currently available devices from the PBX system as Tapi lines. No configuration is required.

## 5 PBX Configuration

The PBX system configuration can be accessed with a Web Browser.

### **CSTA-Service**

The CSTA-Service in the PBX system must be enabled. The configuration can be found in **Configuration - IP-Network - CSTA-Service**. The default port for the CSTA-Service is 7001.

### **User account**

The driver requires a user account to connect to the PBX system. The user account is configured in **Configuration - System - Access control - User account**.

### **Authorization profile**

The Authorization profile that is assigned to the User profile needs the Interface access right for CTI third party.

### **Special settings for correct signalling of redirects**

Under **Configuration - System - Extended - Signallin** set option **Call transfer without prior notice to Ring back tone**



## 6 Mitel MiVoice Office 400

estos ECSTA for Mitel MiVoice Office 400 can be used to control and monitor telephone devices attached to a Mitel MiVoice Office 400 PBX system. The driver can be installed in multiple instances. Each driver instance supports the connection to one PBX system.

The driver requires at least the PBX system version 3.1. Please check the PBX system configuration.

The connection to the PBX system is established using CSTA Phase III XML(ed5).

The PBX system requires CSTA licenses.

The driver automatically offers all currently available devices from the PBX system as Tapi lines. The line devices are updated automatically.

For Installation of the driver see Driver Installation.

For administration and configuration of the driver see Driver Administration.

### 6.1 Connection settings

#### Connection

The connection to the PBX system is established using TCP. Enter the host name or IP address of the PBX system. The default port für TCP is 7001.

#### Login

Enter the user name and password. The user must be configured as a system user account. See also PBX system configuration.

### 6.2 Line Settings

Here you see all lines, the driver offers as Tapi Lines. The phone numbers are the internal callnumbers of your devices, the names are used for display only.

#### Limitations

Line Keys on the phones are not supported. The CSTA Interface of the Mitel MiVoice Office 400 does not provide information about line keys.

#### Automatic Update

The lines are automatically read and updated from the PBX system. Configuration changes as well as login and logouts at phones are automatically adopted.

#### Extras - Query available lines

The currently available lines are automatically read from the PBX system.

#### Extras - 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.

#### Extras - 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.

## 6.3 Advanced Settings

### Phone Number Format:

- **Phone number formatting**  
You may change the phone number format, that the driver reports to the applications. You may also change the phone numbers that are sent from the PC to the PBX. See Phone number formatting.

### Advanced Parameters:

- **Rules for recognising SIP lines**  
Via "Edit rules" you can access a dialogue that can be used to specify whether SIP lines should be detected automatically or via a set of rules by the ecsta, filtered out and thus not forwarded to the Tapi line management or not. When Sip lines are automatically filtered out, all lines that have been configured in the system for the device type "Sip" are filtered out. When filtering out SIP lines using a set of rules, the administrator must create a set of rules regarding the phone numbers, names or IDs (See set of rules for filtering out SIP lines), via which determines which lines are filtered out.
- **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.

### 6.3.1 Phone number formatting

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 match the phone number.

Hint: The Caret Sign (^) is located on the german keyboard on top left.

Here is a short description of the available expressions:

character	description
^	The beginning of the search string (phone number). The expression "^o" matches "o" only at the beginning of the search string.
^	The caret (^) immediately following the left square 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 the target character should not be o
\$	The dollar sign (\$) will match the end of the string. The expression "152\$" will match the sub-string "152" only if it is at the end of the string.
	The alternation character ( ) allows either expression on its side to match the target string. The expression "1 2" will match "1" as well as "2".
.	The dot (.) will match any character.
*	The asterix (*) indicates that the character to the left of the asterix in the expression should match 0 or more times.
+	The plus (+) is similar to asterix but there should be at least one match of the character to the left of the + sign in the expression.
?	The question mark (?) matches the character to its left 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.
[]	Brackets ([]) enclosing a set of characters indicates that any of the enclosed characters may match the target character.

**Replace with:**

Enter the expression how the output of the phone number should look like.

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

\2 the second...

**Check:**

You may check your expressions right here by entering a phone number in the indicate field. The resulting output will be displayed. If the expression from the search pattern is not found, the phone number will be send to the output without modification.

**Examples:**

Effect	search pattern	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	<code>^80(.*)</code>	<code>0\1</code>
Remove a private pin number that may be identified by a 50 followed by a 3 digit pin.	<code>^50[0-9][0-9][0-9](.*)</code>	<code>\1</code>
Suppress all phone numbers that are internal (not more than 3 digits).	<code>^[0-9][0-9][0-9]\$</code>	
Add an access code (leading 0) to all numbers with more than 3 digits (e.g. all external numbers).	<code>^[0-9][0-9][0-9].+</code>	<code>0\1</code>
Add an area code and prefix (03012345) to all internal numbers (1 to 3 digits).	<code>^[0-9][0-9]?[0-9]?\$</code>	<code>03012345\1</code>
Adding an area code to all numbers not beginning with 0 and containing at least 4 digits (thus not internal).	<code>^(^0)[0-9][0-9][0-9].*</code>	<code>08151\1</code>

## 6.4 Licenses

The driver can be licensed in 2 different ways:

### About the estos UCServer:

The estos UCServer assigns a licence for each line activated in the UCServer. The number of lines licensed in estos UCServer is divided between the SIP lines and TAPI lines entered in UCServer. If additional line licences are required, the licences in estos UCServer must be increased.

### With a registered licence

A licence can be entered in the driver and/or a test licence can be downloaded. With a test licence, 25 lines are available without restriction for 45 days. Several licences can be entered to increase the number of line licences.

The following applies to both types of licences:

If the existing licences are used up, opening additional lines will fail.

If you have added several driver instances, all instances share the licences entered. You can allocate the lines to different instances as required.

## 7 Supported TAPI operations

The driver offers the following TAPI call control operations.

Action:	Corresponding TAPI function:
Dial	TSPI_lineMakecall
Hang up	TSPI_lineDrop
Answer call	TSPI_lineAnswer
Hold call	TSPI_lineHold
Retrieve call	TSPI_lineUnhold
Redirect call	TSPI_lineRedirect
Blind transfer	TSPI_lineBlindTransfer
Setup transfer	TSPI_lineSetupTransfer
Swap hold	TSPI_lineSwapHold
Complete transfer	TSPI_lineCompleteTransfer (LINETRANSFERMODE_TRANSFER)
Create conference	TSPI_lineCompleteTransfer (LINETRANSFERMODE_CONFERENCE)
Add to conference	TSPI_lineAddToConference
Remove from conference	TSPI_lineRemoveFromConference
Pickup	TSPI_linePickup
DTMF dialing	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_BUSY</li> <li>• LINEFORWARDMODE_NOANSW</li> </ul>
Message waiting	TSPI_lineSetLineDevStatus TSPI_lineGetLineDevStatus

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

**Other exported functions:**

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

## 8 Set of rules for filtering SIP lines

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

For each rule, you can specify whether a configurable character string should be searched for in the name, in the phone number or in the ID in order to filter out the line in the event of a match. Alternatively, a regular expression can be defined to specify a search pattern that is then applied when searching in the line name or line number. If the search is successful, the line is filtered out.

The following rules can be configured:

- Filtering out lines via a character string in the line name  
 When importing the lines, those whose line name contains the character string entered in the "Search for:" column are not included in the line management.  
 Example: If lines whose line name contains the character string "Paul" are to be filtered out, all lines with the line names "Paul Muster", "Pauline Muster", "Lukas Paulsen"... are filtered out.
- Filtering out lines via a character string in the line number  
 When importing the lines, those whose line number contains the character string entered in the "Search for:" column are not included in the line management.  
 Example: If lines whose line number contains the character string "897" are to be filtered out, lines with the line numbers "897", "089123456897", "0897123456" would be filtered out.
- Filtering out lines via a character string in the ID  
 When importing the lines, those whose ID contains the character string entered in the "Search for:" column are not included in the line management.  
 Example: If lines whose ID contains the character string "!3>" are to be filtered out, all IDs with the identifiers "N", "N", "N"... filtered out.
- Filtering out lines using a search pattern in the line name that is configured using a regular expression  
 The character string in the "Search for:" column must be a regular expression. When importing the lines, the lines for which the search using the regular expression in the line name was successful are not included in the line management.  
 Examples of regular expressions are listed below.
- Filtering out lines via a search pattern in the line number that is configured using a regular expression  
 The character string in the "Search for:" column must be a regular expression. When the lines are read in, those for which the search using the regular expression in the line number was successful are not included in the line management.  
 Examples of regular expressions are listed below.



- Filtering out lines via a search pattern in the ID that is configured using a regular expression  
The character string in the "Search for:" column must be a regular expression. When importing the lines, those for which the search using the regular expression in the ID was successful are not included in the line management.  
Examples of regular expressions are listed below.

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

characters	regular expression	Description of the
^	^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 phone number or name. The regular expression "^o" is used to check whether the digit 'o' is at the beginning of the call number, or the regular expression "^Max" is used to check 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 phone number or name. The regular expression "15\$" checks whether the character string "15" is at the end of the phone number or the regular expression "mann\$" checks whether the name ends with the character string "mann". In both cases, the line would be filtered out.
	64 72 73 or Max mann lich	The vertical bar ( ) separates two or more alternative character strings in the search. A match is made if one of the alternatives applies. The regular expression "64 72 73" is used to search for the sequence of digits 64 or 72 or 73 anywhere in the phone number. The regular expression "Max mann lich" is used to search for the character string "Max" or "mann" or "lich" 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 at the position. Searching with the regular expression "5.6" will find the digit sequences 506, 516, 526... in the call number. Search with the regular expression "M..er" will find the character strings "Meyer", "Meier", "Mayer",... in the name. If the search is successful, the line is filtered out.
[]	[123] or [asd]	The square brackets ([]) specify a set of characters to be searched for in the phone number or name. Search with the regular expression "[123]" will filter out all phone numbers containing the digits 1, 2 or 3. Search with the regular expression "[asd]" will filter out all names containing the letter a, s or d.

**Check:**

You can check your set of rules directly in the "Rules for detecting sip lines" dialogue. In the Search text field, enter a character string that you want to test with the set of rules. The "Recognised as SIP line:" field shows you whether a rule for filtering could be successfully applied to the search text.

See also Advanced settings.

## 9 Info about estos ECSTA for Mitel MiVoice Office 400

estos ECSTA for Mitel MiVoice Office 400 is a product of estos GmbH.

Copyright (C) 2024 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.