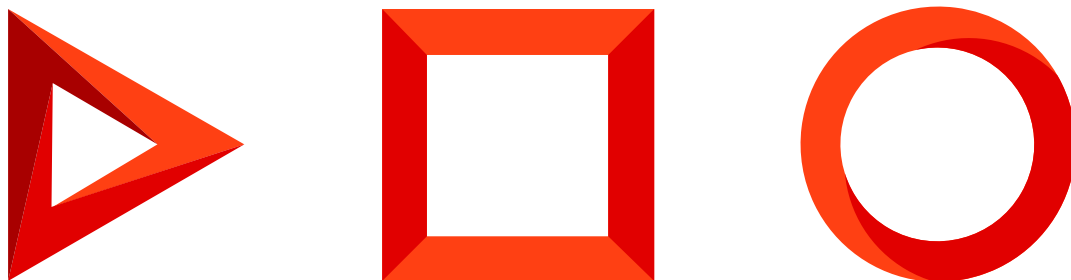


# Phone integration connectors

Version 7.17



This documentation is provided under restrictions on use and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this documentation, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

# Table of Contents

<b>Set up integration with Webitel</b>	<b>5</b>
Available Webitel base phone integration settings	5
Setting up Webitel internal phone parameters	5
<b>Set up integration with Asterisk</b>	<b>6</b>
1. Prepare Asterisk for integration	6
2. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)	7
3. Set up the message exchange library	13
4. Set up the Asterisk parameters	14
<b>Set up integration with Avaya</b>	<b>15</b>
1. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)	16
2. Set up the message exchange library	17
3. Set up the Avaya parameters	18
<b>Set up integration with Cisco Finesse</b>	<b>18</b>
Select the message exchange library	19
Set up the Cisco Finesse parameters	19
Set up the common parameters	19
Set up the individual parameters	19
<b>Set up integration with TAPI</b>	<b>20</b>
1. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)	20
2. Set up the message exchange library	21
3. Set up the TAPI parameters	21
<b>Set up integration with CallWay</b>	<b>22</b>
1. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)	22
2. Set up the message exchange library	24
3. Set up the CallWay parameters	24
<b>Set up integration with Infinity X</b>	<b>25</b>
1. Prepare Infinity X	25
2. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)	25
3. Set up the message exchange library	26
4. Set up the Infinity parameters	27
<b>Configure a WSS phone service connection</b>	<b>27</b>
<b>Creatio phone integration FAQ</b>	<b>28</b>
Which telephone systems can be integrated with Creatio?	29
How can I integrate cloud telephone providers?	29
Which features do the supported connectors have?	29
How can I update Creatio Messaging Service (formerly Terrasoft Messaging Service)?	29

Can I dial an external phone number with an extension external number, bypassing the secretary/answering machine?	
How can I add a custom process to the CTI panel?	30 <sup>29</sup>
<b>Feature comparison for supported phone systems</b>	<b>31</b>
Notes	32

# Set up integration with Webitel

PRODUCTS: [ALL CREATIO PRODUCTS](#)

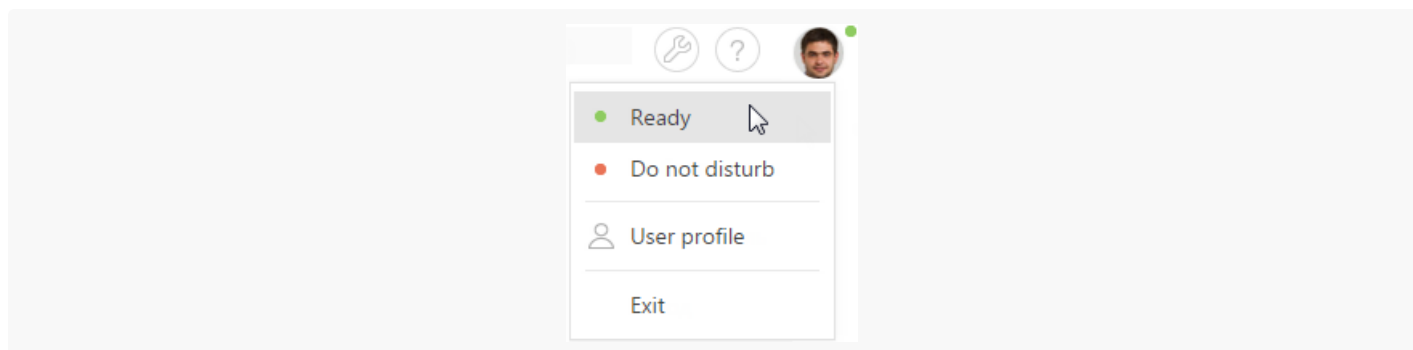
## Available Webitel base phone integration settings

Webitel phone integration is a built-in service and is available for internal calls without the need for additional setup. To make and receive external calls in Webitel, you will need to install the Webitel connector from the Creatio marketplace and set up the phone integration. More information about the installation of the marketplace applications is available in the [separate article](#).

**Note.** If you purchase Webitel Call Manager cloud, all setup will be performed by Webitel support. If you purchase Webitel Call Manager on-site, you can order setup service from Webitel or study the [requirements](#) and perform the setup according to the [Webitel setup guide](#).

A green indicator at the top right corner of the application indicates that phone integration runs correctly in Creatio ([Fig. 1](#)).

Fig. 1 Agent status indicator



To work with the service, populate the “Webitel users” lookup. You have to register your company's employees in Creatio to enable them to make internal calls directly from the application.

**Note.** See more details on how to register Creatio users in the [“Add a new user in Creatio”](#) article.

When adding a Creatio user, the system will automatically assign a Webitel extension number. It will be displayed on the **Communication options** detail of the contact page. By default, the phone number assignment starts from 100. Upon the next user registration, the following ordinal numbers will be assigned, for example, 101, 102, 103, etc.

**Note.** The telephone number is generated automatically based on the specified template. You can change the auto-numbering using the “Webitel user number mask” system setting.

## Setting up Webitel internal phone parameters

1. Open the user profile page by clicking the [ *Profile* ] image button on the main page of the application.

1. Click the [ *Call Center parameters setup* ] button.
1. Select or remove the required checkboxes:
  - a. [ *Disable Contact Center integration* ] - select to disable a built-in Webitel integration. The call button will not be displayed on the communication panel of the application.
  - b. [ *Enable debugging* ] - this checkbox allows you to display troubleshooting information within the browser console. This troubleshooting information can be used when the phone integration runs into problems and the customer addresses the service team.
  - c. [ *Use Web phone* ] - this checkbox enables you to use a webphone. Clear the checkbox if you need to use the down-line phone.
  - d. [ *Use video* ] - this checkbox enables making video calls on internal numbers.
1. Click [ *Save* ].
1. Refresh the browser page to apply the changes.

## Set up integration with Asterisk

PRODUCTS: [ALL CREATIO PRODUCTS](#)

To set up an Asterisk integration, take the following steps:

1. Prepare Asterisk for integration. [Read more >>>](#)
2. Set up Creatio Messaging Service. [Read more >>>](#)
3. Set up the message exchange library. [Read more >>>](#)
4. Set up the Asterisk parameters. [Read more >>>](#)

In Creatio, the Asterisk integration functionality requires a separate license. You need to generate a license request, send it to our service team, upload the received license file into the system, and distribute the licenses among the users. Read more: [Creatio licensing](#) and [Manage user licenses](#).

The integration is only possible if complete preliminary Asterisk setup was performed by the phone integration administrator.

**Attention.** If you set up the telephony for a Creatio production environment, deploy Creatio Messaging Service on a separate node rather than on the Creatio application server. To ensure the fault tolerance of your phone integration, we recommend setting up at least two nodes with Creatio Messaging Service, as well as a balancer that would redirect users in case of lost connection with one of the nodes.

### 1. Prepare Asterisk for integration

The AMI (Asterisk Manager Interface) interface is used to interact with Asterisk. Use AMI to connect to Asterisk servers, configure and manage client programs.

1. Create an AMI user for Creatio, specify the user's parameters in the "manager.conf" file, for example:

```
[ creatio ]

secret = creatio

deny=0.0.0.0/0.0.0.0

permit=0.0.0.0/0.0.0.0

read = system,call,log,verbose,command,agent,user,originate

write = system,call,log,verbose,command,agent,user,originate
```

Replace the “deny” and “permit” values with the corresponding addresses.

2. Check the parking feature activity and the “features.conf” file parameters, for example:

```
parkext => 700

parkpos => 701-720

context => parkedcalls

parkingtime => 45
```

**Attention.** The “parkingtime” value determines the maximum time a call can be on hold. Upon the expiry of this time the subscriber will resume the conversation with the agent. The “parkingtime” value should be sufficiently long to avoid early call return.

## 2. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)

The messaging service lets you connect Creatio to Asterisk via AMI protocol and distribute call events between Creatio users. Some of the settings differ depending on the platform on which the Asterisk telephony service is deployed - .NET Framework or .NET Core.

### Set up Creatio Messaging Service on the .NET Framework platform

1. Before installing Creatio Messaging Service (CMS), make sure that your computer runtime environment has:

- A .NET Framework package version 4.7.2 or later on the server where you are going to install Creatio Messaging Service.
- At least 2 Gb of RAM and 20 Gb of free drive space.

2. Contact Creatio support to receive the messaging service installation files or download the files via the URL:

[Download Creatio Messaging Service](#). Unpack the archive to a folder to ensure a smooth installation. If you run the installation directly from the archive, the archiver application may interfere with the install wizard.

**Attention.** Deploy CMS on the server connected to both the Creatio application server and the PBX.  
Read more: [Telephony integration basics](#).

3. Run the Creatio Messaging Service Install.msi file on the machine intended as the message exchange server and proceed with the installation.
4. Make sure that the “TerrasoftMessagingService” service is running in the Windows Services application. If the “TerrasoftMessagingService” service is not running, start it manually.
5. Open the folder with the service files: ~\BPOnline Messaging Service. Specify the following parameters for Asterisk connector in the “Terrasoft.Messaging.Service.exe.config” configuration file:

```
<asterisk filePath="" url="Asterisk_name_or_server_address" port="Asterisk_server_port" userN
```

See the **list of Asterisk connector parameters** in the following table.



Parameter caption	Parameter function
FilePath	Use this parameter for diagnostics of the system. It allows to repeat a set of events from the file. The default value should be empty.
URL	Asterisk server IP address parameter.
protocol	The parameter enables selecting the protocol type: SIP or PJSIP. Contact your PBX administrator to find out the needed protocol type.
Port	AMI protocol port. By default, "5038."
OriginateContext	The command is used to initialize the call from Creatio phone number. The parameter contains the caption of the context from which the call will be made to the user phone number. The default value for FreePBX is "from-internal"
parkingLotContext	The context for call initialization to receive parking line. The default value is "originateContext."
AutoPauseOnCommutationStart	The checkbox is used for the correct work with Asterisk queues. If the checkbox is selected, the system will put the agent on a pause in all queues after answering the call. The feature is used to avoid the second call during the handling of the first one and/or putting the first call on hold.
queueExtensionFormat	The call channel format while receiving the call from the queue. If using LocalChannel, the default value in FreePBX is "Local/{0}@from-queue."
sendRingStartedOnRingingState	The checkbox stands for the correct handling of the call from the queue. If this checkbox is selected, the system will display the call with the user after receiving the "NewState" event with the Ringing parameter by AMI. The default value is "On."
traceQueuesState	This setting is used to define the agent's status in the queues. If the agent receives the second call from the queue while handling the first one in Creatio, it is used for debugging. Information about the status of agents is being written to the log file of the connector. The default value is "Off."

Configuration example:

```
<asterisk filePath="" url="10.0.15.185" port="5038"
```

```

userName="bpm" secret="bpm" originateContext="from-internal"

parkingLotContext="from-internal" autoPauseOnCommutationStart="true"

queueExtensionFormat="Local/{0}@from-queue/n" asyncOriginate="true"

sendRingStartedOnRingingState="true" traceQueuesState="false"

protocol="SIP/" packetInfoConfig=""/>

```

## 6. Test the phone integration.

**Note.** Follow this [instruction](#) if you need to update Creatio Messaging Service.

## Set up Creatio Messaging Service on the .NET Core platform

**Attention.** You can set up Messaging Host on the .NET Core platform for Creatio version 7.16.3 and later.

1. Install Docker. Installing Docker on Linux platforms is covered in the [Docker guide](#). Run the “docker --version” command on Linux machine to verify the installed Docker version.
2. Install Docker Compose. Installing Docker Compose for Linux OS is covered in the [Docker documentation](#). Run the “docker-compose --version” command on Linux machine to verify the installed Docker Compose version.
3. Install and set up the Docker Compose components. The container of the messaging service is deployed via the Docker Compose utility. Download the [archive](#). Unzip the archive with the configuration files and scripts to a custom directory, for example, /opt/messaging.host.

**Note.** The configuration files contain all necessary default settings for a Linux based server.

The archive structure of the configuration files and scripts:

```

/etc/

...\appsettings.json - service configuration

...\nlog.config - setup of the service log level

docker-compose.yml - utility configuratiton Docker Compose

.env - file containing the environment variables to run the components

```

4. Using the Linux terminal, go to the /docker-compose catalog of the unzipped archive, for example, /opt/messaging.host/docker-compose.
5. Run the “sudo docker-compose pull” command in the terminal. Wait until the download of the required service component images from the [Docker Hub](#) is complete.

**Attention.** If the server Internet access is restricted, download the needed images manually to a machine with free access (see the docker-compose.yml configuration file) and transfer the images as files to the target machine using the [sudo docker export](#) and [sudo docker import](#) commands.

6. Specify the following parameters for Asterisk connector in the “etc/appsettings.json” configuration file:

```
{
  "url": "Asterisk_server_name_or_address",

  "port": "Asterisk_server_port",

  "userName": "Asterisk login",

  "secret": "Asterisk password",

  "originateContext": "Outbound context",

  "parkingLotContext": "Parking context",

  "autoPauseOnCommutationStart": "true",

  "queueExtensionFormat": "Local/{0}@from-queue/n",

  "asyncOriginate": "true",

  "sendRingStartedOnRingingState": "true",

  "traceQueuesState": "false",

  "protocol": "The used protocol type - SIP/ or PJSIP/",

  "packetInfoConfig": "Additional package parameters for processing in configuration",

  "filePath": ""
}
```

7. Run the sudo docker-compose up -d command to launch the service. A “logs” folder will be created in the current catalog.

See the **list of Asterisk connector parameters** in the following table.

Parameter caption	Parameter function
FilePath	Use this parameter for diagnostics of the system. It allows to repeat a set of events from the file. The default value should be empty.
URL	Asterisk server IP address parameter.
protocol	The parameter enables selecting the protocol type: SIP or PJSIP. Contact your PBX administrator to find out the needed protocol type.
Port	AMI protocol port. By default, "5038."
OriginateContext	The command is used to initialize the call from Creatio phone number. The parameter contains the caption of the context from which the call will be made to the user phone number. The default value for FreePBX is "from-internal"
parkingLotContext	The context for call initialization to receive parking line. The default value is "originateContext."
AutoPauseOnCommutationStart	The checkbox is used for the correct work with Asterisk queues. If the checkbox is selected, the system will put the agent on a pause in all queues after answering the call. The feature is used to avoid the second call during the handling of the first one and/or putting the first call on hold.
queueExtensionFormat	The call channel format while receiving the call from the queue. If using LocalChannel, the default value in FreePBX is "Local/{0}@from-queue."
sendRingStartedOnRingingState	The checkbox stands for the correct handling of the call from the queue. If this checkbox is selected, the system will display the call with the user after receiving the "NewState" event with the Ringing parameter by AMI. The default value is "On."
traceQueuesState	This setting is used to define the agent's status in the queues. If the agent receives the second call from the queue while handling the first one in Creatio, it is used for debugging. Information about the status of agents is being written to the log file of the connector. The default value is "Off."

Configuration example:

```
"asterisk": {
```

```

"url": "Asterisk_server_name_or_address",

"port": "Asterisk_server_port",

"userName": "Asterisk login",

"secret": "Asterisk password",

"originateContext": "Outbound context",

"parkingLotContext": "Parking context",

"autoPauseOnCommutationStart": "true",

"queueExtensionFormat": "Local/{0}@from-queue/n",

"asyncOriginate": "true",

"sendRingStartedOnRingingState": "true",

"traceQueuesState": "false",

"protocol": "The used protocol type - SIP/ or PJSIP/",

"packetInfoConfig": "Additional package parameters for processing in configuration",

"filePath": ""
}


```

8. Test the phone integration.

### 3. Set up the message exchange library

Message exchange library selection and setup is performed once by the system administrator. Some of the settings differ depending on the platform that the Asterisk telephony service is deployed on – .NET Framework or .NET Core.

#### Set up the library on the .NET Framework platform

1. Open the system designer by clicking  in the top right corner of the application window.
2. Click “System settings” in the “System setup” block.
3. In the “Default messages exchange library” (“SysMsgLib” code) system setting, set the default value to “Telephony integration library based on Asterisk (AMI) protocol.”
4. In the “Message exchange server” (“SysMsgServerNode” code) system setting, select the connection parameters of the system messages service. In the [ *Default value* ] field, specify the message exchange


network address in the following format: “ws://[ server ]:2013” for sites served over HTTP or “wss://[ server ]:2013” for sites served over HTTPS, where:

- [ server ] – domain name of the server that hosts the message exchange service. Using IP addresses or “localhost” is not recommended.
- “2013” – the port used by default for connecting to the messaging service. You can change the port number in the “Terrasoft.Messaging.Service.exe.config” file. Using “localhost” is not recommended, as it may cause errors when connecting to the phone integration server. When using a wss connection, make sure that server address matches the address in the SSL certificate.

**Note.** If your website is served over HTTPS and secure (WSS) connection is used for WebSockets, you will need to install a security certificate on the message exchange server and specify it in the configuration files of the message service. For more information about the setup process, contact Creatio technical support at support@creatio.com.

## Set up the library on the .NET Core platform

**Attention.** You can set up Messaging Host on the .NET Core platform for Creatio version 7.16.3 and up.

1. Open the system designer by clicking  in the top right corner of the application window.
2. Click “System settings” in the “System setup” block.
3. In the “Default messages exchange library” (“SysMsgLib” code) system setting, set the default value to “Phone integration library based on Asterisk (AMI) protocol.”
4. In the “Message exchange server” (“SysMsgServerNode” code) system setting, select the connection parameters of the message service. In the [ *Default value* ] field, specify the message exchange network address in the following format: “http://[ server ]:2013” for sites served over HTTP or “https://[ server ]:2014” for sites served over HTTPS, where:
  - [ server ] – domain name of the server that hosts the message exchange service. Using IP addresses or “localhost” is not recommended.
  - “2013 or 2014” – the port used by default for connecting to the messaging service. You can change the port number in the “docker-compose.yml” configuration file. Using “localhost” is not recommended, as it may cause errors when connecting to the phone integration server. When using an HTTPS connection, make sure that server address matches the address in the SSL certificate.

**Note.** If your website is served over HTTPS and secure connection is used for WebSockets, you will need to install a security certificate on the message exchange server and specify it in the docker-compose configuration. For more information about the setup process, contact Creatio technical support.

## 4. Set up the Asterisk parameters

These settings should be applied to every Creatio user who received Asterisk integration license. To do this:

1. Open the user profile page by clicking the [ *Profile* ] image button on the main page of the application.
  2. Click the [ *Call Center parameters setup* ] button.
  3. On the displayed page, fill out the required fields:
    - a. [ *Disable call Center integration* ] - this checkbox allows you to disable Creatio integration with the phone integration. The call button will not be displayed on the communication panel of the application.
    - b. [ *Number* ] - Asterisk user line number. It matches the phone number by default. For example, to track the SIP/305 user line, specify the "305" value, and to track the SIP/office line, specify the "office" value.
- Attention.** A separate line is used for each user. It is not recommended to specify the same line for several users, as it may cause errors.
- c. [ *Outgoing call context* ] - specify the outgoing call context if it differs from the system outgoing call context specified in the "Terrasoft.Messaging.Service.exe.config" file for this particular user.
  - d. [ *Enable debugging* ] - this checkbox allows you to display troubleshooting information within the browser console. This troubleshooting information can be used when the phone integration runs into problems and the customer addresses the service team.
4. Click [ *Save* ].
  5. Refresh the browser page to apply the changes.

## Set up integration with Avaya

PRODUCTS: ALL CREATIO PRODUCTS

To set up an Avaya integration, take the following steps:

1. Set up Creatio Messaging Service. [Read more >>>](#)
2. Set up the message exchange library. [Read more >>>](#)
3. Set up the Avaya parameters. [Read more >>>](#)

In Creatio, the Avaya integration functionality requires a separate license. You need to generate a license request, send it to our service team, upload the received license file into the system, and distribute the licenses among the users. Read more: [Creatio licensing](#) and [Manage user licenses](#).

The integration is only possible if complete preliminary Avaya setup was performed by the phone integration administrator.

**Attention.** If you set up the telephony for a Creatio production environment, deploy Creatio Messaging Service on a separate node rather than on the Creatio application server. To ensure the fault tolerance of your phone integration, we recommend setting up at least two nodes with Creatio Messaging Service, as well as a balancer that would redirect users in case of lost connection with one of the nodes.

# 1. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)

The messaging service allows you to connect Creatio to Avaya via the DMCC .NET API integration protocol and distribute the call events between Creatio users.

**Attention.** Install Creatio Messaging Service on a .NET Framework Windows server to integrate Avaya PBX.

DMCC .NET API integration protocol requires licenses. The number of licenses should correspond to the number of Creatio users who simultaneously use Avaya phone integration. See [Avaya documentation](#) for more information.

**Note.** Your PBX must include Avaya Application Enablement Services (AES) component for Avaya phone integration. The integration is available for AES server version 5.2 and later.

1. Before installing Creatio Messaging Service (CMS), make sure that your computer runtime environment has:
  - A .NET Framework package version 4.7.2 or later on the server where you are going to install Creatio Messaging Service.
  - At least 2 Gb of RAM and 20 Gb of free drive space.
2. Contact Creatio support to receive the messaging service installation files or download the files via the URL: [Download Creatio Messaging Service](#). Unpack the archive to a folder to ensure a smooth installation. If you run the installation directly from the archive, the archiver application may interfere with the install wizard.

**Attention.** Deploy CMS on the server connected to both the Creatio application server and the PBX. Read more: [Telephony integration basics](#).

3. Run the Creatio Messaging Service Install.msi file on the machine intended as the message exchange server and proceed with the installation.
4. Make sure that the "TerrasoftMessagingService" service is running in the Windows Services application. If the "TerrasoftMessagingService" service is not running, start it manually.
5. Open the folder with the service files: ~\BPMonline Messaging Service. Specify the following parameters for Avaya connector in the "Terrasoft.Messaging.Service.exe.config" configuration file:

```
<avaya serverIp="" port="4721" useSecureSockets="False" ctiUser="" psw="" protocolVersion="ht
```

See the **list of configuration file parameters** in the table below.




Parameter caption	Parameter function
avaya serverIp	AES server address.
port	Connection port to AES server. Default value: “4721” for the unsecured connection or “4722” for the secured connection.
useSecureSockets	The checkbox for the encrypted connection usage requires adding a certificate. By default, “False.”
ctiUser	Avaya AES (Avaya AES user login) username.
psw	Avaya AES user password.
protocolVersion	The protocol used to connect to AES server. the default value: “http://www.ecma-international.org/standards/ecma-323/csta/ed3/priv6.”
switchName	Avaya (hostname Avaya switch) hostname switch.

## 6. Test the phone integration.

**Note.** Follow this [instruction](#) if you need to update the Creatio Messaging Service.

## 2. Set up the message exchange library

Message exchange library selection and setup is performed once by the system administrator.

1. Open the system designer by clicking  in the top right corner of the application window.
2. Click “System settings” in the “System setup” block.
3. In the “Default messages exchange library” (“SysMsgLib” code) system setting, set the default value to “Telephony integration library based on Avaya (DMCC) protocol.”
4. Specify the message service connection parameters. To do this, open the “Message exchange server” (“SysMsgServerNode” code) system setting. In the [ *Default value* ] field, specify the message exchange network address in the following format: “ws://0.0.0.0:2013” if your website is served over HTTP or “wss://0.0.0.0:2013” if your website is served over HTTPS, where:
  - “0.0.0.0” – IP address that your Creatio users use to access your message exchange server.
  - “2013” – the port used by default for connecting to the messaging service. You can change the port number in the “Terrasoft.Messaging.Service.exe.config” file.

**Note.** If your website is served over HTTPS and secure (WSS) connection is used for WebSockets, you will need to install a security certificate on the message exchange server and specify it in the configuration files of the message service. For more information about the setup process, contact

Creatio technical support.

5. Click [ Save ].

### 3. Set up the Avaya parameters

These settings should be applied for each Creatio user who received Avaya integration license. Use the user login credentials to access the system.

1. Open the user profile page by clicking the [ *Profile* ] image button on the main page of the application.
2. Click the [ *Call Center parameters setup* ] button.
3. On the opened page, fill out the required values:
  - a. [ *Disable Call Center integration* ] - this checkbox allows you to disable Creatio integration with the phone integration. The call button will not be displayed on the communication panel of the application.
  - b. [ *Agent's Id* ], [ *Password* ] - agent's data on Avaya server.
  - c. [ *Number* ] - agent's number on the Avaya server.
  - d. [ *Enable debugging* ] - this checkbox allows you to display troubleshooting information within the browser console. This troubleshooting information can be used when the phone integration runs into problems and the customer addresses the service team.
4. Click [ Save ].
5. Refresh the browser page to apply the changes.

## Set up integration with Cisco Finesse

PRODUCTS: [ALL CREATIO PRODUCTS](#)

This article describes integration with Cisco Finesse version 11.5 and later. We recommend using new phone service versions to ensure secure and reliable operation. If you need to set up integration with Cisco Finesse version 11.0 and earlier for testing purposes or to look for errors, follow the [instructions for Creatio version 7.16](#).

To set up integration with Cisco Finesse, take the following steps:

1. Set up the service message exchange library. [Read more >>>](#)
2. Set up the Cisco Finesse parameters [Read more >>>](#)

In Creatio, the Cisco Finesse integration functionality requires a separate license. To license the phone service, generate a license request, send it to our support team, upload the received license file to Creatio, and distribute the licenses among users. Learn more in separate articles: [Creatio licensing](#), [Manage user licenses](#).

Before you set up the integration, make sure that the phone service administrator has already configured Cisco Finesse.

**Attention.** To ensure the integration operates as intended, enable the [CORS technology](#) support in Cisco Finesse settings. You can do this using the following commands


```
utils finesse cors enable_all
```

```
utils service restart Cisco Finesse Tomcat
```

```
utils service restart Unified CCX Notification Service
```

## Select the message exchange library


The setup is performed once by the system administrator.

1. Open the System Designer, e. g., by clicking  in the top right.
2. Click “System settings” in the “System setup” block.
3. Set the “Default messages exchange library” (“SysMsgLib” code) system setting to “Cisco Finesse telephony integration library (Finesse 11.5+).”
4. Click [ Save ].

## Set up the Cisco Finesse parameters

### Set up the common parameters

The setup is performed once by the system administrator for all Creatio users.

1. Open the System Designer, e. g., by clicking  in the top right.
2. Click “System settings” in the “System setup” block.
3. Specify the address of your Cisco Finesse server in the “Finesse 11.5+ server address” (“FinesseServerAddress” code) system setting. Use the following format: https://:8445. For example, https://hq-uccx.abc.inc:8445.
4. Click [ Save ].
5. Specify the WSS protocol address of your Cisco Finesse server in the “Finesse 11.5+ websocket address” (“FinesseWebsocketAddress” code) system setting. Use the following format: wss://:8445. For example, wss://hq-uccx.abc.inc:8445.
6. Click [ Save ].

### Set up the individual parameters

The setup is performed for each Creatio user who received the Cisco Finesse integration license. Use the user login credentials to log in to Creatio.

1. Open the user profile page, e. g., by clicking [ *Your profile* ] on the Creatio homepage.
2. Click [ *Call Center parameters setup* ]. This opens a page.
3. Fill out the required fields:

- a. Select the [ *Disable Contact Centre integration* ] checkbox to disable Creatio integration with the phone service. This hides the call button from the Creatio communication panel.
  - b. Specify the Cisco Finesse user parameters in the [ *Agent Id* ], [ *Extension* ], [ *Password* ] fields.
  - c. Select [ *Enable debugging* ] checkbox to display troubleshooting information within the browser console. For example, you can use the information if you encounter phone service and customer contact issues.
4. Click [ *Save* ].
  5. Refresh the browser page to apply the changes.

# Set up integration with TAPI

PRODUCTS: [ALL CREATIO PRODUCTS](#)

To set up a TAPI integration, take the following steps:

1. Set up Creatio Messaging Service. [Read more >>>](#)
2. Set up the message exchange library. [Read more >>>](#)
3. Set up the TAPI parameters. [Read more >>>](#)

In Creatio, the TAPI integration functionality requires a separate license. You need to generate a license request, send it to our service team, upload the received license file into the system, and distribute the licenses among the users. Read more: [Creatio licensing](#) and [Manage user licenses](#).

The integration is only possible if complete preliminary TAPI driver setup was performed by the phone integration administrator.

**Attention.** If you set up the telephony for a Creatio production environment, deploy Creatio Messaging Service on a separate node rather than on the Creatio application server. To ensure the fault tolerance of your phone integration, we recommend setting up at least two nodes with Creatio Messaging Service, as well as a balancer that would redirect users in case of lost connection with one of the nodes.

## 1. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)

The messaging service allows you to connect Creatio to TAPI via the integration protocol and distribute the call events between Creatio users.

**Attention.** Install Creatio Messaging Service on a .NET Framework Windows server to integrate TAPI PBX.

1. Before installing Creatio Messaging Service (CMS), make sure that your computer runtime environment has:
  - A .NET Framework package version 4.7.2 or later on the server where you are going to install Creatio Messaging Service.
  - At least 2 Gb of RAM and 20 Gb of free drive space.

2. Contact Creatio support to receive the messaging service installation files or download the files via the URL: [Download Creatio Messaging Service](#). Unpack the archive to a folder to ensure a smooth installation. If you run the installation directly from the archive, the archiver application may interfere with the install wizard.


**Attention.** Deploy CMS on the server connected to both the Creatio application server and the PBX. Read more: [Telephony integration basics](#).

3. Run the Creatio Messaging Service Install.msi file on the machine intended as the message exchange server and proceed with the installation. The TAPI driver must be installed and configured on the same computer by the phone integration administrator. To ensure that the TAPI driver runs correctly, use the “Conference.exe” utility you can receive from the support team.
4. Test the phone integration.

**Note.** Follow this [instruction](#) if you need to update the Creatio Messaging Service.

## 2. Set up the message exchange library

Message exchange library selection and setup is performed once by the system administrator.

1. Open the system designer by clicking  in the top right corner of the application window.
2. Click “System settings” in the “System setup” block.
3. In the “Default messages exchange library” (“SysMsgLib” code) system setting, set the default value to “Telephony integration library based on Tapi protocol.”
4. In the “Message exchange server” (“SysMsgServerNode” code) system setting, select the connection parameters of the system messages service. In the [ *Default value* ] field, specify the message exchange network address in the following format: “ws://0.0.0.0:2013” if your website is served over HTTP or “wss://0.0.0.0:2013” if your website is served over HTTPS, where:
  - “0.0.0.0” – IP address that your Creatio users use to access your message exchange server.
  - “2013” – the port used by default for connecting to the messaging service. You can change the port number in the “Terrasoft.Messaging.Service.exe.config” file.

**Note.** If your website is served via HTTPS and secure (WSS) connection is used for WebSockets, you will need to install a security certificate on the message exchange server and specify it in the configuration files of the message service. For more information about the setup process, contact Creatio technical support at [support@creatio.com](mailto:support@creatio.com).

5. Click [ Save ].

## 3. Set up the TAPI parameters

These settings should be applied for each Creatio user who received TAPI integration license. Use the user login credentials to access the system.

1. Open the user profile page by clicking the [ *Profile* ] image button on the main page of the application.
  2. Click the [ *Call Center parameters setup* ] button.
  3. On the opened page, fill out the required values:
    - a. [ *Disable Call Center integration* ] - this checkbox allows you to disable Creatio integration with the phone integration. The call button will not be displayed on the communication panel of the application.
    - b. [ *Line* ] - user line number in TAPI.
- Attention.** A separate line is used for each user. It is not recommended to specify the same line for several users, as it may cause errors.
- c. [ *Enable debugging* ] - this checkbox allows you to display troubleshooting information within the browser console. This troubleshooting information can be used when the phone integration runs into problems and the customer addresses the service team.
4. Click [ *Save* ].
  5. Refresh the browser page to apply the changes.

## Set up integration with CallWay

PRODUCTS: ALL CREATIO PRODUCTS

To set up a CallWay integration, take the following steps:

1. Set up Creatio Messaging Service. [Read more >>>](#)
2. Set up the message exchange library. [Read more >>>](#)
3. Set up the CallWay parameters. [Read more >>>](#)

In Creatio, the CallWay integration functionality requires a separate license. You need to generate a license request, send it to our service team, upload the received license file into the system, and distribute the licenses among the users. Read more: [Creatio licensing](#) and [Manage user licenses](#).

The integration is only possible if complete preliminary CallWay setup was performed by the phone integration administrator.

**Attention.** If you set up the telephony for a Creatio production environment, deploy Creatio Messaging Service on a separate node rather than on the Creatio application server. To ensure the fault tolerance of your phone integration, we recommend setting up at least two nodes with Creatio Messaging Service, as well as a balancer that would redirect users in case of lost connection with one of the nodes.

### 1. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)

The messaging service allows you to connect Creatio to CallWay via the integration protocol and distribute call events between Creatio users.

1. Before installing Creatio Messaging Service (CMS), make sure that your computer runtime environment has:
  - A .NET Framework package version 4.7.2 or later on the server where you are going to install Creatio Messaging Service.
  - At least 2 Gb of RAM and 20 Gb of free drive space.
2. Contact Creatio support to receive the messaging service installation files or download the files via the URL: [Download Creatio Messaging Service](#). Unpack the archive to a folder to ensure a smooth installation. If you run the installation directly from the archive, the archiver application may interfere with the install wizard.

**Attention.** Deploy CMS on the server connected to both the Creatio application server and the PBX. Read more: [Telephony integration basics](#).

3. Run the Creatio Messaging Service Install.msi file on the machine intended as the message exchange server and proceed with the installation.
4. Make sure that the “TerrasoftMessagingService” service is running in the Windows Services application. If the “TerrasoftMessagingService” service is not running, start it manually.
5. Open the folder with the service files: ~\BPMonline Messaging Service. Specify the CallWay server address in the “Terrasoft.Messaging.Service.exe.config” file:

```
<callway url="CallWay_server_address" login="XXX" password="XXX" port="34600" restoreConnecti
```

See the **list of configuration file parameters** in the table below.


Parameter caption	Parameter function
url	CallWay server IP address parameter.
login	Integration protocol login and password.
password	
port	Integration protocol port. By default, “34600.”
restoreConnectionInterval	The time interval to reconnect the integration messaging service to the CallWay phone integration server if a failure occurs. The value is specified in milliseconds and is “2000” - by default.

6. Test the phone integration.

**Note.** Follow this [instruction](#) if you need to update the Creatio Messaging Service.

## 2. Set up the message exchange library

Message exchange library selection and setup is performed once by the system administrator.

1. Open the system designer by clicking  in the top right corner of the application window.
2. Click “System settings” in the “System setup” block.
3. In the “Default messages exchange library” (“SysMsgLib” code) system setting, set the default value to “Telephony integration library based on Callway protocol.”
4. Specify the message service connection parameters. To do this, open the “Message exchange server” (“SysMsgServerNode” code) system setting. In the [ *Default value* ] field, specify the message exchange network address in the following format: “ws://0.0.0.0:2013” if your website is served over HTTP or “wss://0.0.0.0:2013” if your website is served over HTTPS, where:
  - “0.0.0.0” - IP address that your Creatio users use to access your message exchange server.
  - “2013” - the port used by default for connecting to the messaging service. You can change the port number in the “Terrasoft.Messaging.Service.exe.config” file.

**Note.** If your website is served over HTTPS and secure (WSS) connection is used for WebSockets, you will need to install a security certificate on the message exchange server and specify it in the configuration files of the message service. For more information about the setup process, contact Creatio technical support at [support@creatio.com](mailto:support@creatio.com).

5. Click [ *Save* ].

## 3. Set up the CallWay parameters

These settings must be applied for each Creatio user who received CallWay integration license. Use the user login credentials to access the system.

1. Open the user profile page by clicking the [ *Profile* ] image button on the main page of the application.
2. Click the [ *Call Center parameters setup* ] button.
3. On the opened page, fill out the required values:
  - a. [ *Disable Call Center integration* ] - this checkbox allows you to disable Creatio integration with the phone service. The call button will not be displayed on the communication panel of the application.
  - b. [ *Agent internal number* ] - CallWay user phone number.
  - c. [ *Routing rule* ] - specify the routing rule if it is required for the CallWay configured server.
  - d. [ *Use CallWay client* ] - select the checkbox if an employee uses CallWay internal client. While using the CallWay internal client, the answer button in the application becomes available.
  - e. [ *Enable debugging* ] - this checkbox allows you to display troubleshooting information within the browser console. This troubleshooting information can be used when the phone integration runs into problems and the customer addresses the service team.
4. Click [ *Save* ].
5. Refresh the browser page to apply the changes.



# Set up integration with Infinity X

PRODUCTS: [ALL CREATIO PRODUCTS](#)

To set up an Infinity X integration, take the following steps:

1. Prepare Infinity for integration. [Read more >>>](#)
2. Set up Creatio Messaging Service (CMS). [Read more >>>](#)
3. Set up message exchange library. [Read more >>>](#)
4. Set up Infinity X parameters. [Read more >>>](#)

In Creatio, the Infinity integration functionality requires a separate license. You need to generate a license request, send it to our service team, upload the received license file into the system, and distribute the licenses among the users. Learn more in separate articles: [Creatio licensing](#), [Manage user licenses](#).

The integration is only possible if complete preliminary Infinity setup was performed by the phone integration administrator.

**Attention.** If you set up the telephony for a Creatio production environment, deploy Creatio Messaging Service on a separate node rather than on the Creatio application server. To ensure the fault tolerance of your phone integration, we recommend setting up at least two nodes with Creatio Messaging Service, as well as a balancer that would redirect users in case of lost connection with one of the nodes.

## 1. Prepare Infinity X

Create an account in Infinity X with the role “Third-Party integration connection”. This account will be used on the integration service level. It is not recommended to use this user account for other purposes, or log in under this user account, as this may cause integration errors.

## 2. Set up Creatio Messaging Service (formerly Terrasoft Messaging Service)

The messaging service connects to Infinity and distributes call events between Creatio users.

**Attention.** Install Creatio Messaging Service on a .NET Framework Windows server to integrate Infinity X PBX.

1. Make sure that your computer runtime environment has the following before installing Creatio Messaging Service (CMS):
  - a .NET Framework package version 4.7.2 or later on the server where you are going to install Creatio Messaging Service
  - at least 2 Gb of RAM and 20 Gb of free drive space


2. Send the following files from the “Client” folder created during the Infinity X setup to Creatio support:
  - Cx.Integration.BaseConnector.dll
  - Cx.Integration.AgatInfinityConnectorInterfaces.dll
  - Cx.Integration.AgatInfinityConnectorFactory.dll
3. Creatio support will send you an archive with CMS files to upload and install on the server. Unpack the archive to a folder to ensure a smooth CMS installation. If you run the installation directly from the archive, the install wizard might not operate as intended.

**Attention.** Deploy CMS on the server connected to both the Creatio application server and the PBX. Learn more in a separate article: [Telephony integration basics](#).

4. Run the Creatio Messaging Service Install.msi file on the machine intended as the message exchange server and proceed with the installation.
5. Make sure that the “TerrasoftMessagingService” service is running in the Windows Services application. If the “TerrasoftMessagingService” service is not running, start it manually.
6. Copy the “Client” folder (created during the Infinity X setup) to a separate folder, e.g., the folder with the CMS service.
7. Open the folder with the service files: ~\BPMonline Messaging Service.
8. Specify the following parameters in the Infinity block of the Terrasoft.Messaging.Service.exe.config file:
  - a. infinityXClientPath. The path to the “Client” folder. If the “Client” folder is located in the CMS service folder, the parameter value will be as follows: infinityXClientPath=“\Client”
  - b. thirdPartyIntegrationLogin. The login of the user with the “Third-Party integration connection” role.
  - c. thirdPartyIntegrationLogin. The password of the user with the “Third-Party integration connection” role.
9. Restart the “TerrasoftMessagingService” service.
10. Test the phone integration.

### 3. Set up the message exchange library

Message exchange library selection and setup is performed once by the system administrator.

1. Open the system designer by clicking  in the top right corner of the application window.
2. Click “System settings” in the “System setup” block.
3. Set the default value of the “Default messages exchange library” (“SysMsgLib” code) system setting to “Telephony integration library based on Infinity protocol”.
4. Specify the message service connection parameters. To do this, open the “Message exchange server” (“SysMsgServerNode” code) system setting. In the [ *Default value* ] field, specify the network address of the message exchange server in the following format: “ws://0.0.0.0:2013” if your website is served over HTTP or “wss://0.0.0.0:2013” if your website is served over HTTPS, where:
  - “0.0.0.0” is the IP address that your Creatio users use to access your message exchange server.
  - “2013” is the port used by default for connecting to the messaging service. You can change the port

number in the “Terrasoft.Messaging.Service.exe.config” file.

**Note.** If your website is available by HTTPS and secure (WWS) connection is used for WebSockets, you will need to install a security certificate on the message exchange server and specify it in the configuration files of the message service. For more information about the setup process, contact Creatio technical support at support@creatio.com.

5. Click [ Save ].

## 4. Set up the Infinity parameters

Apply these settings for each Creatio user who received Infinity integration license. Use the user login credentials to access the system.

1. Open the user profile page by clicking the [ Profile ] image button on the main page of the application.
2. Click the [ Call Center parameters setup ] button. This opens a page.
3. Fill out the required values on the page that opens:
  - a. Select the [ Disable Contact Centre integration ] checkbox to disable Creatio integration with the telephony. The call button will not be displayed on the communication panel of the application.
  - b. Specify application IP address and port in the [ Infinity server address ] field. Learn more about the ports required for implementing the full functionality of Infinity X in [Creatio Community](#).
  - c. Specify the telephone number of the line which will be used for the call in the [ Line ] field.

**Attention.** A separate line is used for each user. It is not recommended to specify the same line for several users, as it may cause errors.

- d. Select [ Enable debugging ] checkbox to display troubleshooting information within the browser console. This troubleshooting information can be used when the phone integration runs into problems and the customer addresses the service team.
4. Click [ Save ].
5. Refresh the browser page to apply the changes.

# Configure a WSS phone service connection

PRODUCTS: [ALL CREATIO PRODUCTS](#)

You can configure a WSS phone service connection. This procedure is relevant for Asterisk, Avaya, TAPI, CallWay, and Infinity phone service users.

**Note.** Configure the [WebSockets technology](#) to ensure the correct operation of the phone service in Creatio.

Modify the **Terrasoft.Messaging.Service.exe.config** configuration file on the phone server to configure the secure connection with the phone service:

1. Delete or comment out the following configuration block in the <servers> section:

```
<server name="ClientWebSocketService"

serviceName="ClientWebSocketService" ip="Any" port="2013" mode="Tcp"

idleSessionTimeout="10000" maxCommandLength="4096"

maxConnectionNumber="10000">

</server>
```

2. Contact an official certification center to receive a PFX certificate. A system administrator must obtain the certificate.
3. Remove the comment from the following block in the <servers> section:

```
<server name="ClientWebSocketService"

serviceName="ClientWebSocketService" ip="Any" port="2013" mode="Tcp"

idleSessionTimeout="10000" maxCommandLength="4096" security="tls" >

<certificate filePath="certificate.pfx" password="111"></certificate>

</server>
```

4. Specify the PFX digital certificate you received from the certification center, as well as the access password in the <certificate> section. Place the certificate in the folder with the Creatio Messaging Service binary files.

**Attention.** It is not recommended to use self-signed certificates, because this may violate security conditions.

5. After modifying the configuration file, change the address of the connection to the phone server in the "Message exchange server" ("SysMsgServerNode" code) [system setting](#) from ws:// to **wss://**.

## Creatio phone integration FAQ

PRODUCTS: [ALL CREATIO PRODUCTS](#)

## Which telephone systems can be integrated with Creatio?

By default, Creatio is integrated with Webitel telephone service. If necessary, you can integrate other telephone systems with Creatio. The list of supported telephone systems is available in the [Feature comparison for supported phone systems](#) article.

## How can I integrate cloud telephone providers?

Creatio interacts with phone providers through an IP-PBX. To integrate with a cloud telephone system, set up one of the [supported IP-PBX](#). The setup procedures for integration with different phone systems are available in separate articles.

If you are not using connectors listed in the comparison table for phone integration features, you can use any of the connectors available on [Creatio Marketplace](#).

## Which features do the supported connectors have?

The feature comparison for integration with different telephone systems, as well as their integration requirements, are available in the [Feature comparison for supported phone systems](#) article.

## How can I update Creatio Messaging Service (formerly Terrasoft Messaging Service)?

1. Contact Creatio support to receive the new version of Creatio Messaging Service installation files or download them via the URL: [Download Creatio Messaging Service](#).
2. Save the Terrasoft.Messaging.Service.exe.config settings file.
3. Stop the "TerrasoftMessagingService" service in the Windows Services application.
4. Uninstall Creatio Messaging Service.
5. Restart the phone integration server.
6. Install the new version of Creatio Messaging Service. Unpack the archive to a folder to ensure a smooth installation. If you run the installation directly from the archive, the archiver application may interfere with the install wizard.
7. Compare the settings in the new and the old version. Transfer the settings to the new version if needed.
8. Make sure that the "TerrasoftMessagingService" service is running in the Windows Services application. If the "TerrasoftMessagingService" service is not running, start it manually.
9. Test the phone integration.

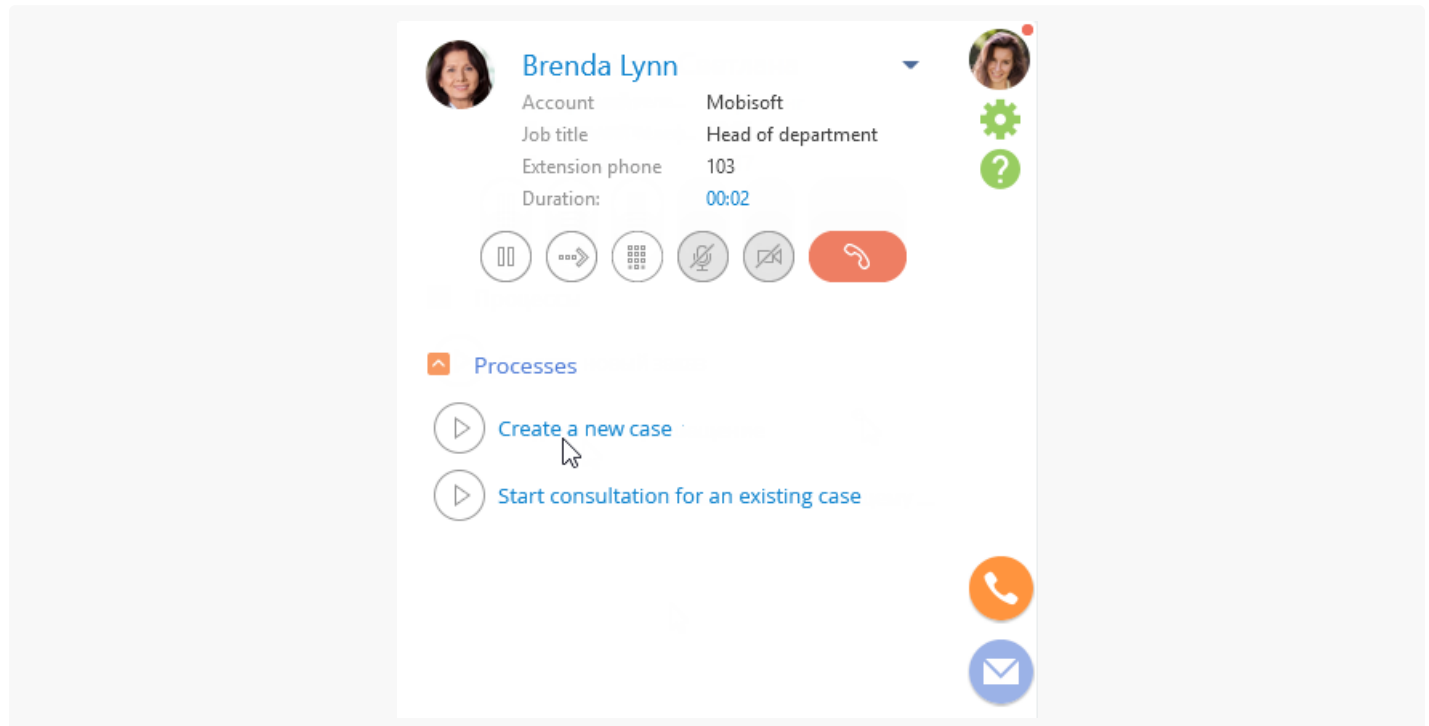
## Can I dial an external phone number with an extension external number, bypassing the secretary/answering machine?

This depends on the settings in the PBX API. Most APIs do not have this function. If the PBX API has this function, you will need to perform additional settings on the phone integration server.

## How can I add a custom process to the CTI panel?

Products that include the [Agent Desktop](#) functionality come with several [business processes](#) out-of-the-box. Agents can run these processes directly from the CTI panel during calls. For example, in service products, these are [ *Create new case* ] and [ *Start consultation for an existing case* ] processes (Fig. 1).

Fig. 1 Business processes in the CTI panel



To add a business process to the CTI panel, you need to add the process to the [ *CTI panel actions* ] [lookup](#). As a result, the process will become available in the [ *Processes* ] area of the CTI panel for the Contact Center agents.

**Note.** In Creatio, “contact center agents” are users who are members of the [organizational role](#) specified in the “Folder – Contact Center agents” (ContactCenterOperatorsFolder) [system setting](#). By default, the role is “CC agents.”

Add the following [process parameters](#) to bind the process to other Creatio records:

- **“CallId”** of the “Unique identifier” data type. This parameter is automatically populated with the unique identifier of the corresponding [ *Calls* ] section record.
- **“ContactId”** of the “Lookup” data type. Populate this parameter with the unique ID of the caller/call recipient’s contact.
- **“AccountId”** of the “Lookup” data type. Populate this parameter with the unique ID of the caller/call recipient’s account.
- **“PhoneNumber”** of the “Text (250 characters)” data type. Populate this parameter with the telephone number of the caller/call recipient.

You can use this data to automatically populate the fields of the new record created during the process execution.

# Feature comparison for supported phone systems

PRODUCTS: [ALL CREATIO PRODUCTS](#)

The telephony features in Creatio vary depending on the connected phone system. By default, Creatio is integrated with Webitel telephone service. If necessary, you can connect a different system.

Below is a feature comparison table for different phone systems:

TAPI	Cisco Finesse	Avaya	Webitel	Infinity	Asterisk	CallWay
<b>“Search by caller ID” phone system functionality</b>						
<a href="#">Note 1</a>	+	+	+	+	+	+
<b>“Make outgoing calls” phone system functionality</b>						
+	+	+	+	+	<a href="#">Note 2</a>	<a href="#">Note 3</a>
<b>“Receive incoming calls” phone system functionality</b>						
+	+	+	+	+	-	<a href="#">Note 4</a>
<b>“Place calls on hold, unhold calls” phone system functionality</b>						
+	+	+	+	+	+	+
<b>“End calls” phone system functionality</b>						
+	+	+	+	+	+	+
<b>“Manage agent status” phone system functionality</b>						
<a href="#">Note 5</a>	+	+	+	+	-	+
<b>“Transfer calls” phone system functionality</b>						
+	+	+	+	+	+	+
<b>“Save the information to [ Calls ] section” phone system functionality</b>						
+	+	+	+	+	+	+
<b>“Call from browser” phone system functionality</b>						

-	-	+	-	-	-	-
<b>“Replay calls” phone system functionality</b>						
-	-	-	+	-	-	-
<b>Telephone system versions</b>						
All phone systems that use TAPI 2.X	Cisco Finesse 11.5+	AES v5.2-7		X	13	

## Notes

1  
Due to TAPI limitations, caller identification is unavailable for calls routed through UCCX while using CUCM.

2  
The user might have to respond to an incoming system call to initiate an outgoing call from Creatio. The call flow depends on the software/hardware phone version/model.

3  
CallWay software phones are fully supported. If the agent uses a different IP or software phone, they have to respond to an incoming “system” call to initiate an outgoing call from Creatio.

4  
Only CallWay software phones are fully supported.

5  
The following 2 statuses are available: “Ready” and “Do not disturb” (DND). Cisco is currently not supported.