# bintec elmeg be.IP Series v10

XCAPI voiceoverIP

May 17, 2018

www.xcapi.de e-mail sales@te-systems.de phone +49 5363 8195-0

TE-SYSTEMS competence in e-communications.

## Introduction

This document is intended to support you with the integration of XCAPI into an existing environment of the bintec elmeg be.IP. In the following sections we describe the essential configuration steps for SIP trunking to allow optimal interworking of both, the XCAPI and the be.IP.
Though being based on the bintec elmeg be.IP v10 series, this document is applicable with other versions given a few adjustments.

At this point we suppose that the be.IP environment and the physical or virtual application server is available and accessible through the network. Application server in this context mean, a server with a recent available Microsoft Windows operating system with latest updates and patches included. Further, that the XCAPI and the CAPI 2.0 voice or fax application is properly installed.

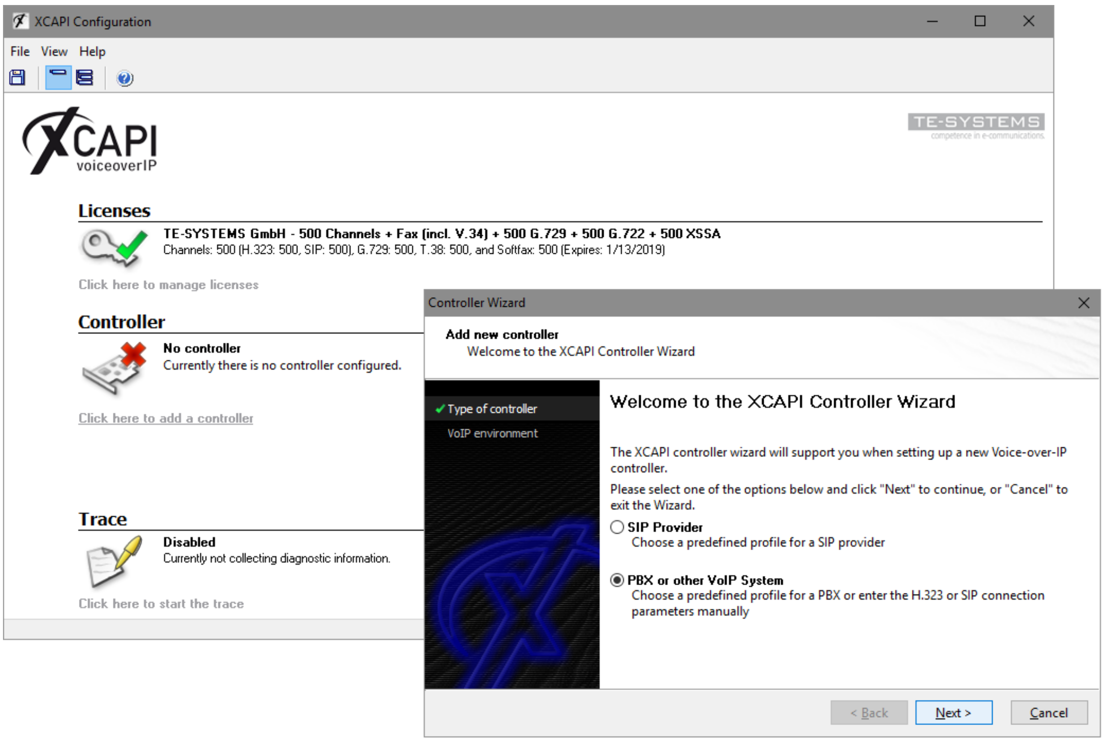
It is also supposed that the public network access via ISDN and/or SIP is given and properly working, also in context with the custom and country dependent numberings and call routings. The same goes for the networking (LAN, WAN, DMZ, NAT, Firewall) itself as such topics are beyond the scope of this document and thus not shown here at all. Please refer to the respective manufacturer documentations, manuals and examples in such cases.

However, independent of the deployed application, the SIP connection can be tested with the XCAPI's included test application (xtest.exe) that is available within the XCAPI's installation folder (by default `\\Program Files (x86)\TE-SYSTEMS\XCAPI\`). This test tool allows to check with inbound and outbound calls, fax and testing several supplementary services.

We recommend to visit our YouTube channel frequently for XCAPI related tutorials about licensing, the test tool, line monitor, tracing, analyzing and others. Registered community users can check about latest XCAPI documents, TechNotes and versions.

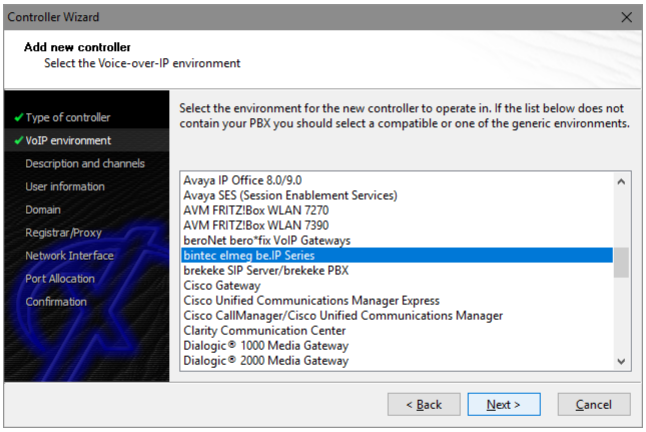
## XCAPI Configuration

Please start up the XCAPI configuration to create a new controller assigned to the bintec elmeg be.IP.
If you've just installed the XCAPI and start the configuration tool for the first time or no controller is available at all, the XCAPI controller wizard will pop up automatically. To start up the XCAPI controller wizard manually, the hyperlink labeled **Click here to add a controller** on the main page has to be clicked.
However, select **PBX or other VoIP System** in the initial **Type of controller** dialog and proceed with the **Next** button.



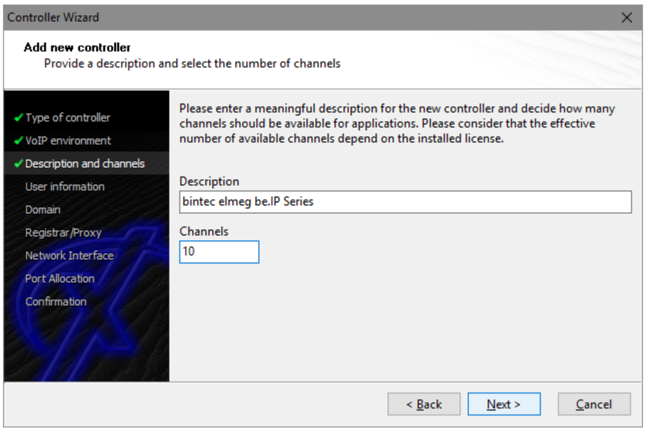
## 2.1 VoIP Environment

The next dialog lists some common Voice-over-IP environments. Selecting one of those will setting up the XCAPI controller with a selection of near-optimal presets and sparing you manual configurations. Note that the **bintec elmeg be.IP IP Series** entry is selectable from XCAPI version 3.6.73.



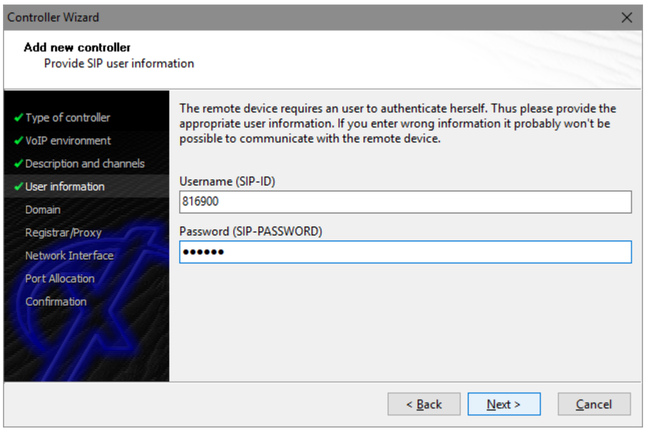

## 2.2 Description and Channels

When the VoIP environment was selected, the next dialog allows to set a meaningful description for the controller. Also the number of channels that the new controller will be able to provide can be set. So enter how many simultaneous connections the XCAPI controller should handle when communicating with the be.IP and the bound CAPI 2.0 application.



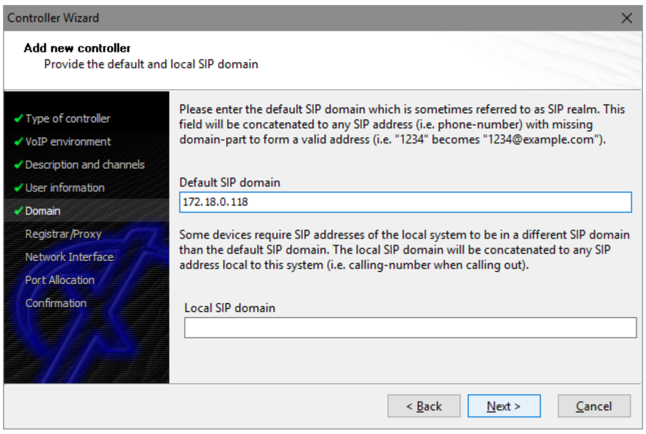
## 2.3 User Information

Next, please provide the SIP user information for the XCAPI extension you are about to create in the be.IP. The according be.IP configurations will be described in the chapter **XCAPI as SIP Extension** from page 9.



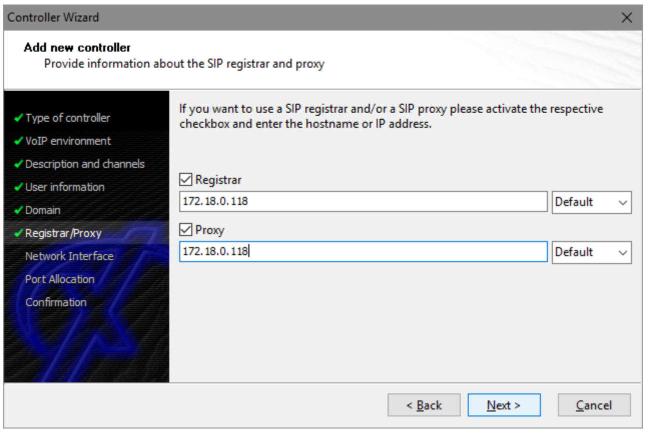

## 2.4 Domain

Next the IP address or host name of the be.IP gateway must be provided, in this example the gateway Ethernet address is using IP 172.18.0.118.



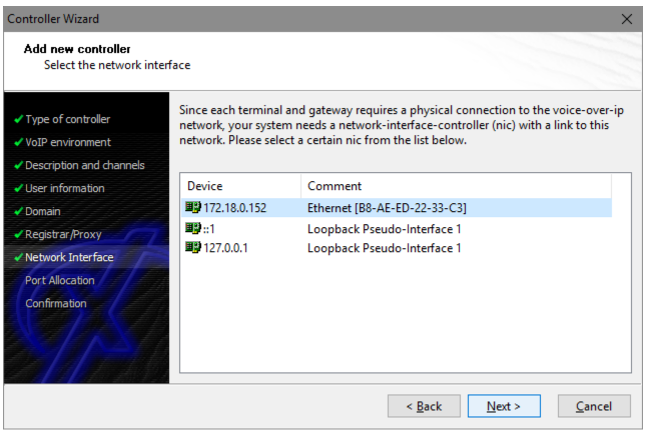
## 2.5 Registrar Proxy

In this dialog the be.IP referring SIP registrar and proxy must be specified as IP address or host name. Please note, port 5060 is used as default for SIP via UDP or TCP which can be changed on demand. Also the local listening port is always set to port 5060 for UDP/TCP which can be changed after the controller wizard configuration steps.



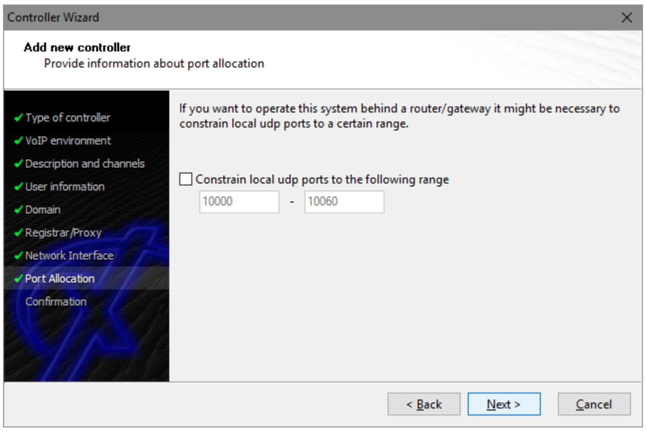

## 2.6 Network Interface

Afterwards, select the network interface that will be used for the inbound and outbound communications for this controller. Note that this is the XCAPI controllers used Ethernet interface which is used for the SIP communication with the be.IP.



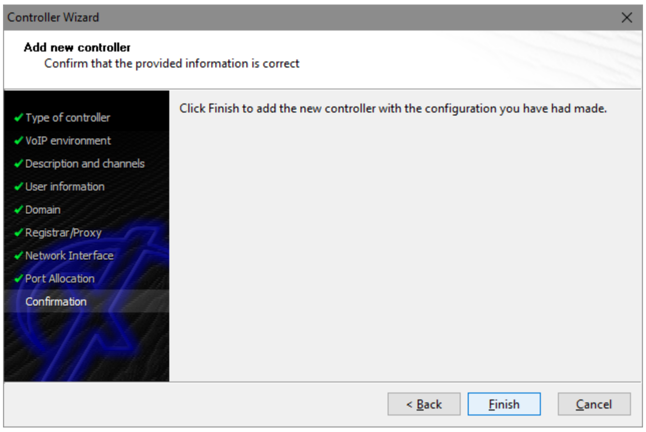
## 2.7 Port Allocation

On demand and in the case of any router or firewall restrictions for UDP (RTP/T.38) a port range can be specified. In this example no range will be set which allows the XCAPI controller using a random port range between 1024 and 65535.

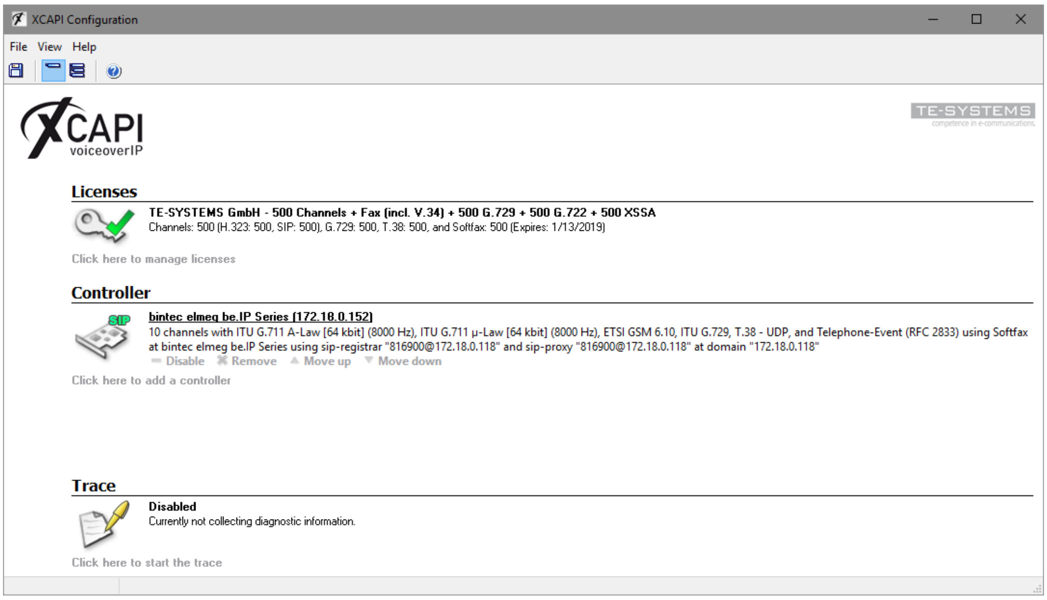



## 2.8 Confirmation

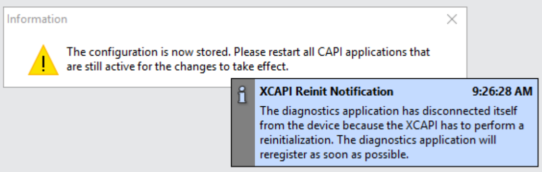
The final wizard dialog performs some checks on the configuration parameters you've made. If errors will be detected, use the **Back** button to the respective erroneous dialogs and correct them. Use the **Finish** button in order to finally create the new controller.



The newly created XCAPI controller for the be.IP is now listed on the main page of XCAPI configuration. Push the **Save** button and exit the configuration tool.




Note that the bound CAPI 2.0 application with its services must always be restarted to take effect on the XCAPI controller changes. Restarting any of the XCAPI services won't help at all. If enabled, the XCAPI diagnostic monitor pop-up with a re-initialization notification on success.

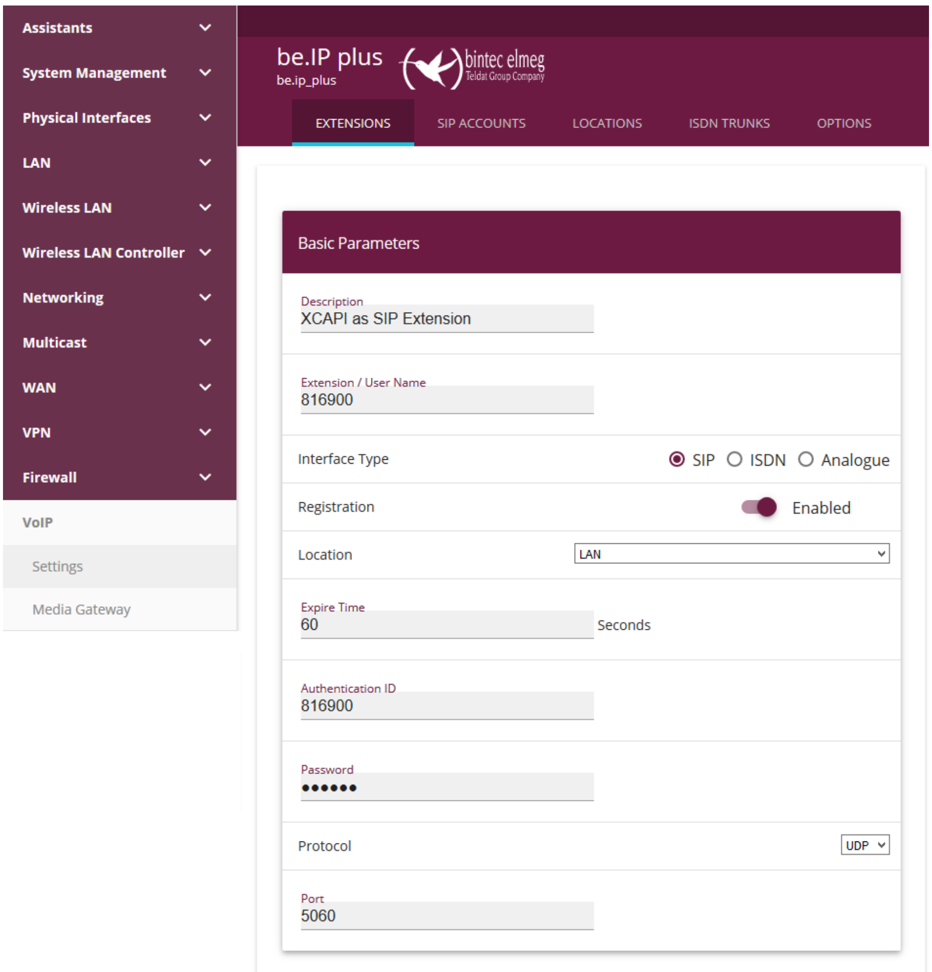


# Configuring the be.IP

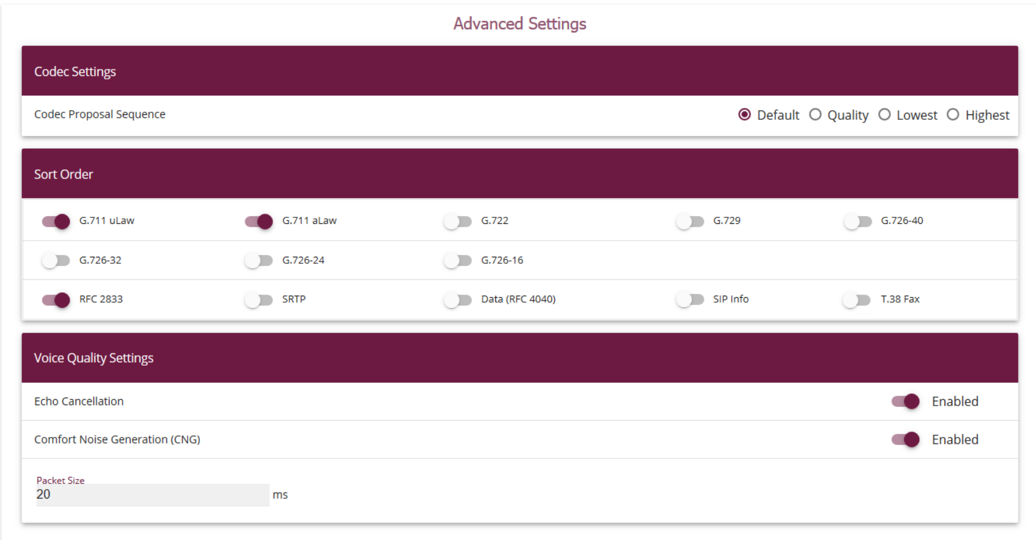
This chapter gives an overview about the be.IP related configurations for the SIP extension usage together with XCAPI. For a better overview some additional be.IP configurations will be shown. This examples configurations are reviewed via the be.IP frontend in the **Full Access** view and may not be shown in their order of configuration steps.

## 3.1 XCAPI as SIP Extension

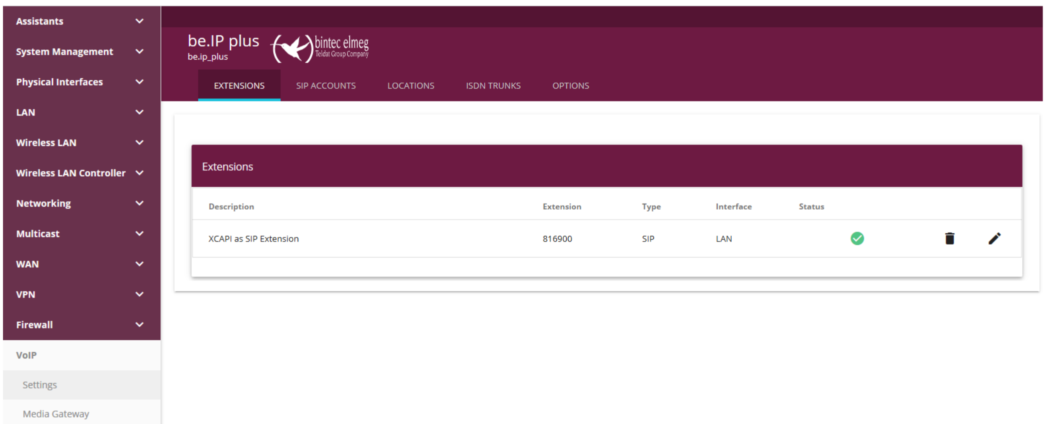
The extension credentials are here used as shown on the screenshot below. As described in the XCAPI configuration chapter **User Information** from page 5, this credentials has to be used by the XCAPI controller for an appropriate registration and authentication process towards the be.IP. The **Protocol**, **Port** and **Timer Expiry** are used with the given defaults. They are also set by default through the XCAPI controller wizard if selecting be.IP as VoIP environment as described on page 4.



The codec related configurations can be made in the extensions **Advanced Settings**. Here they are used in accordance with the given XCAPI controller wizard defaults. T.38 Fax is disabled as the G.711 fax pass through (SoftFax) method is basically preferred, hence only the G.711 and RFC 2833 codecs are enabled.
For additional information about voice, DTMF and fax please check with the according **Appendix** chapters of this document from page 14.

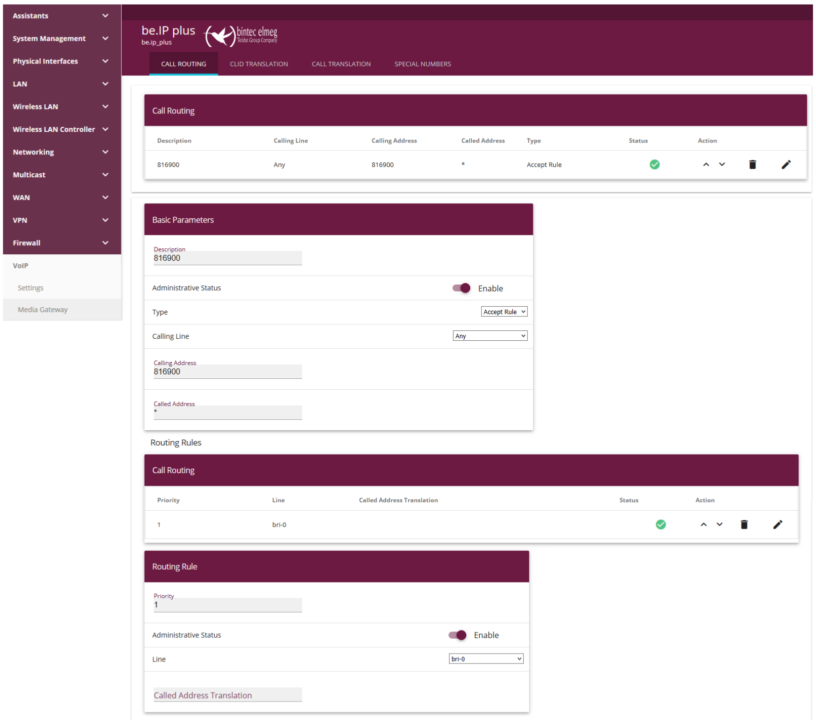



On registration success of the XCAPI controller the according state will be displayed for the SIP extension.



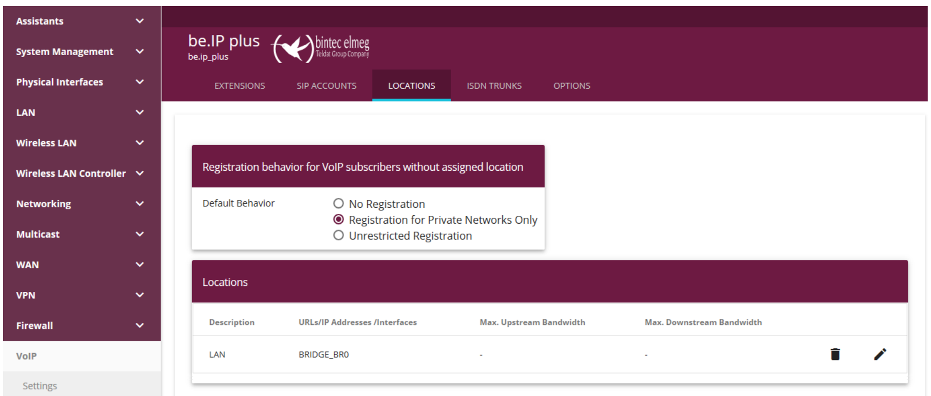
### 3.1.1 Call Routing

For the **Call Routing** exact or common rules can be used. The example below shows that the call routing for the XCAPI SIP extension is used with an exact rule. The related number is forwarded to the bri-0 interface. Any called address is allowed and no **Called Address Translation** is set.



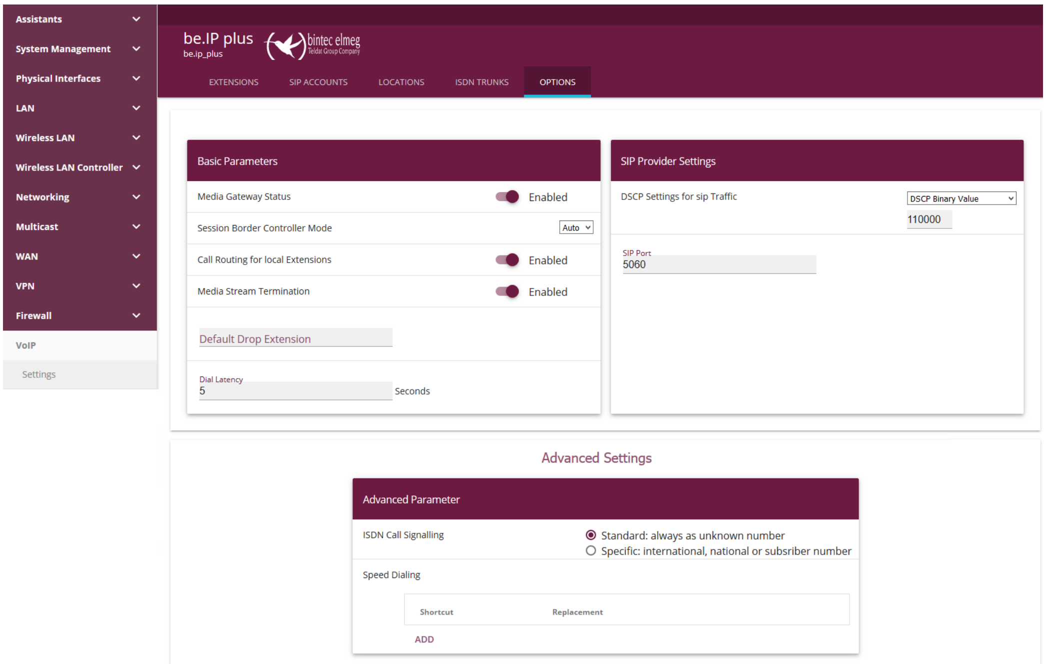

### 3.1.2 VoIP Locations

The **Registration behavior for VoIP subscribers without assigned location** of the **Locations** tab is here used with its default.



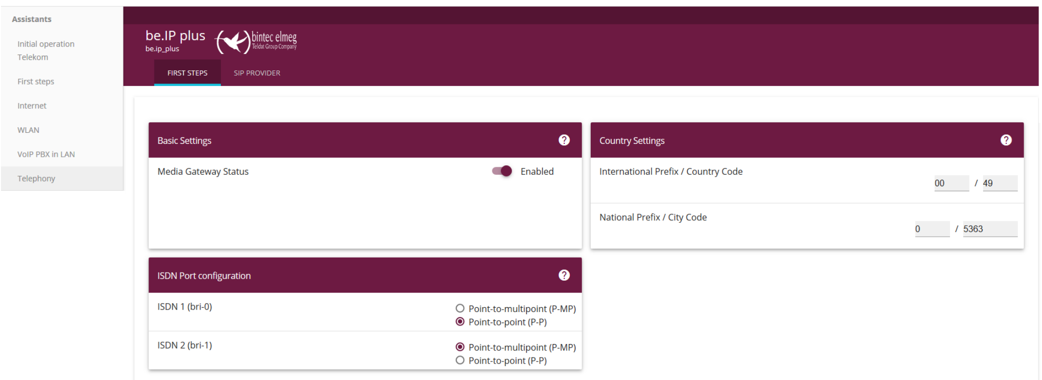
### 3.1.3 VoIP Options

The **Options** of the **VoIP Settings** are used as follows.



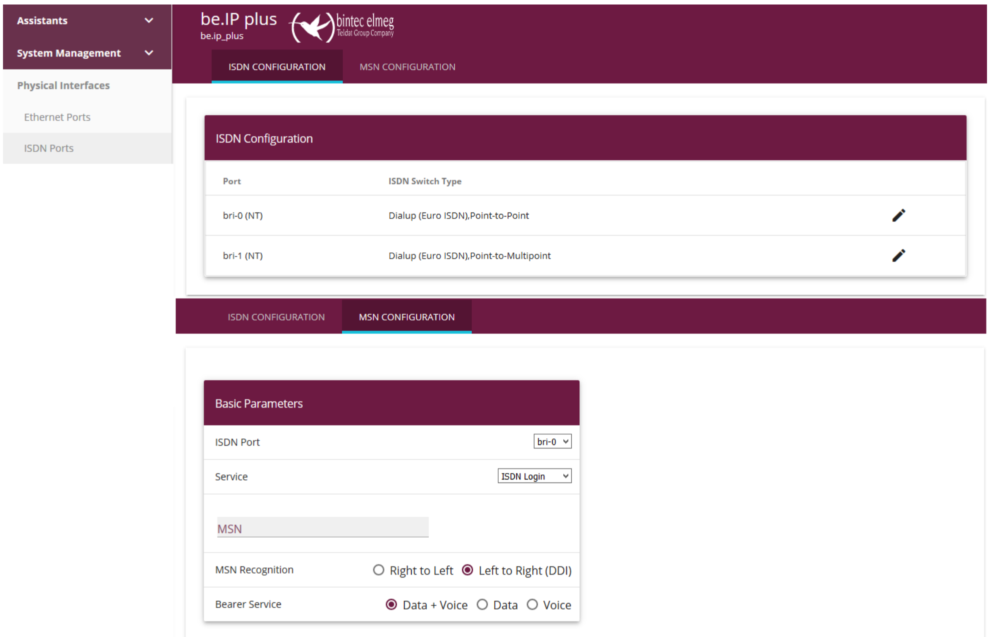

## 3.2 Telephony Basic Settings

The **Telephony** settings of this bintec elmeg be.IP environment are here used as follows and shown below. Ensure that the **Media Gateway Status** is enabled for allowing VoIP connections. The **Country Settings** are custom dependent and has to be taken into account for the numbering and call routing behavior.



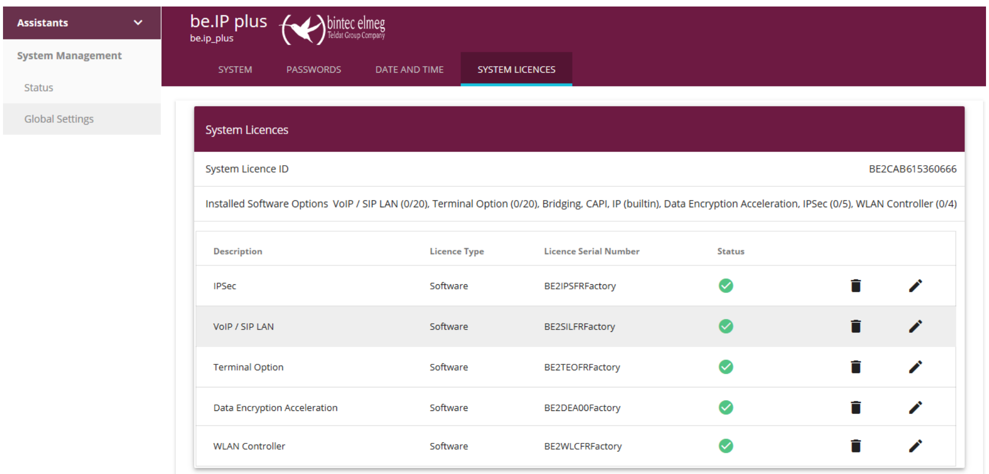
## 3.3 Physical Interfaces

The screenshot below is just shown for completeness and refers to the mentioned call routings between XCAPI and the bri-0 interface of this example.




## 3.4 System Licenses

Please ensure that the **VoIP / SIP LAN** status is in operation.



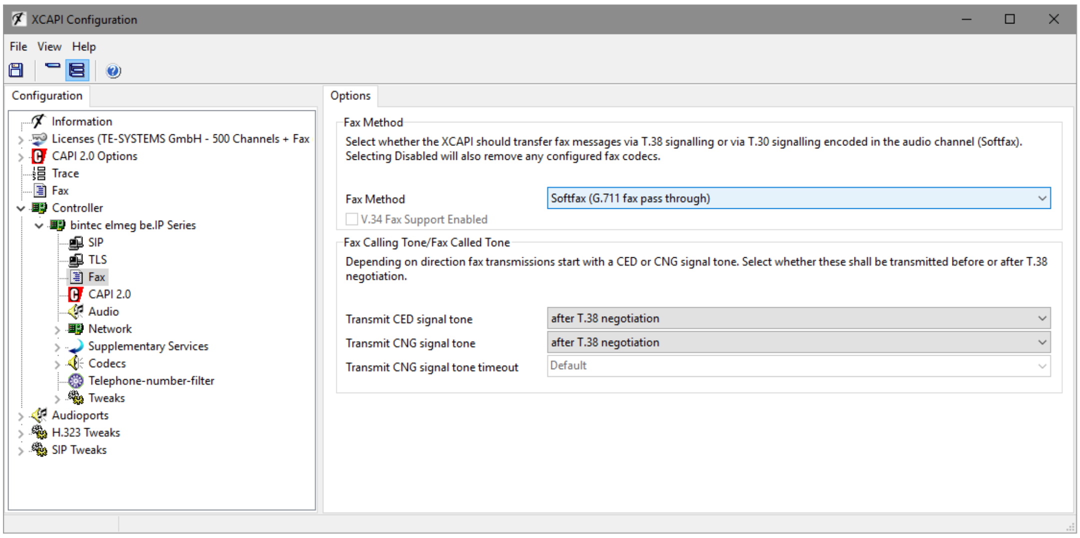
# Appendix

The appendix gives several information and configuration hints as well as some considerations. If using the XCAPI controller wizard with its be.IP template most of the shown settings are used by default. Nevertheless, the next topics and the shown configurations must be reviewed, checked and tested, especially with the participating extensions, VoIP or ISDN boards and SIP or ISDN provider.

## 4.1 SoftFax (G.711 fax pass through)

In the **SoftFax** mode, the XCAPI simulates an analog Fax device by transmitting modulated Fax-signals modem-like through the established G.711 audio channels

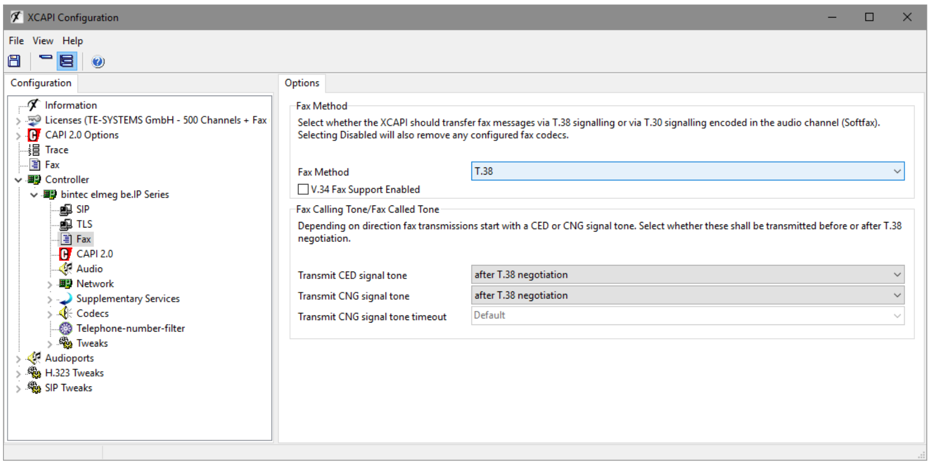
The **Fax** method within the gateway parameters **Media** settings are used by default with **G.711**. However, for appropriate facsimile interworking it has been ensured that those Codec, Framing, Bandwidth and DTMF settings (as shown in the chapter **XCAPI as SIP Extension** starting on page 9) are set conform to the ones of the XCAPI controller configuration and other participating SIP instances.



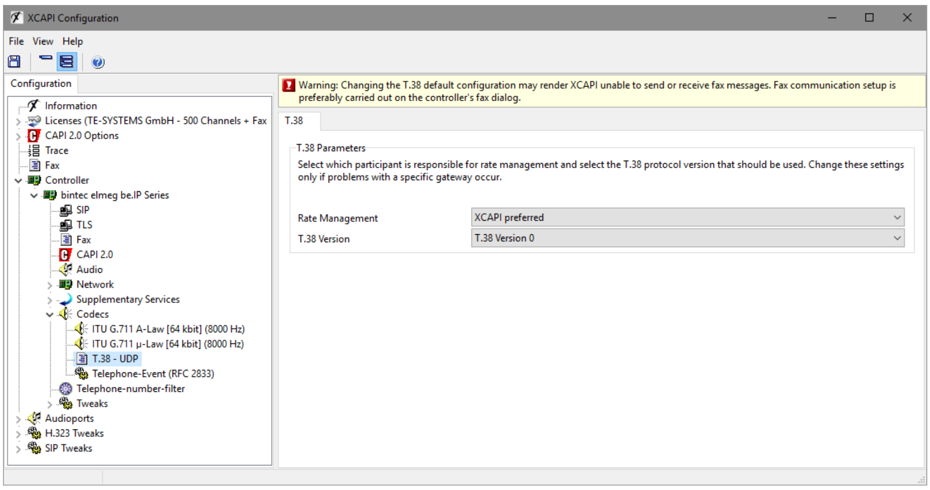

Avoid using the **Softfax (G.711 fax pass through)** method if the be.IP is connected to a SIP instance (provider, gateway, session border controller etc.) which is restricted to T.38 only support.

## 4.2 T.38

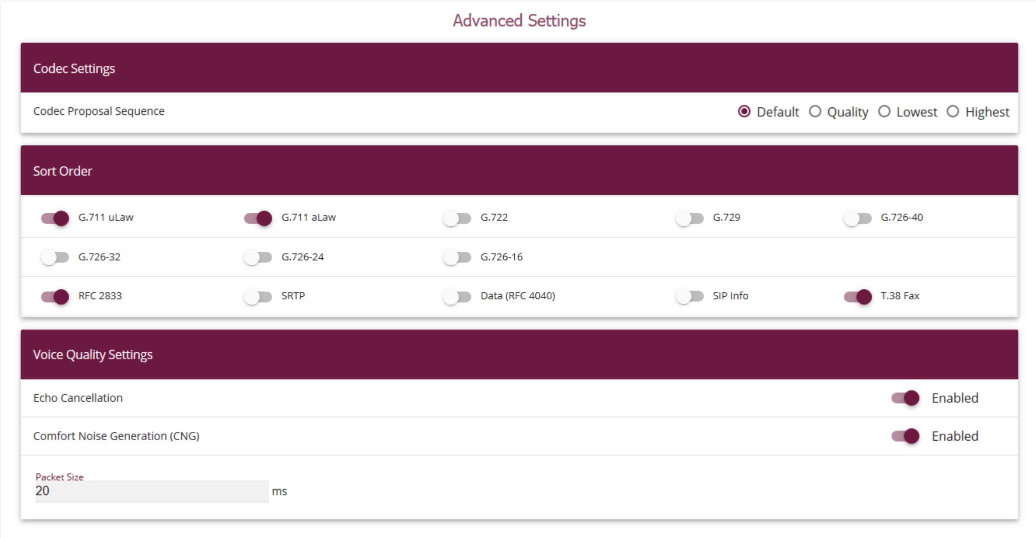
In the case of T.38 usage, this protocol has also to be supported by the be.IP related PSTN trunk (SIP gateway, SIP provider). It is recommended to avoid unnecessary transcoding (G.711 to T.38 or vice versa) and using conform fax methods for all participating instances. For enabling T.38 this **Fax Method** must be selected as shown next.




Unless suggested or required, do not change any of the T.38 parameters. Adjusting any of these values commonly fall off in quality for T.38 interworking.

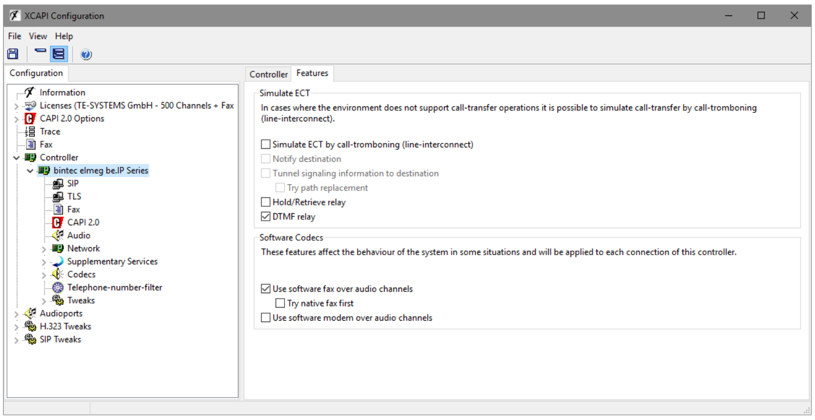


Further T.38 has to be enabled for the XCAPI SIP extension too. In the case of a connected SIP provider with T.38 support, this codec must also be set for the provider related SIP account.



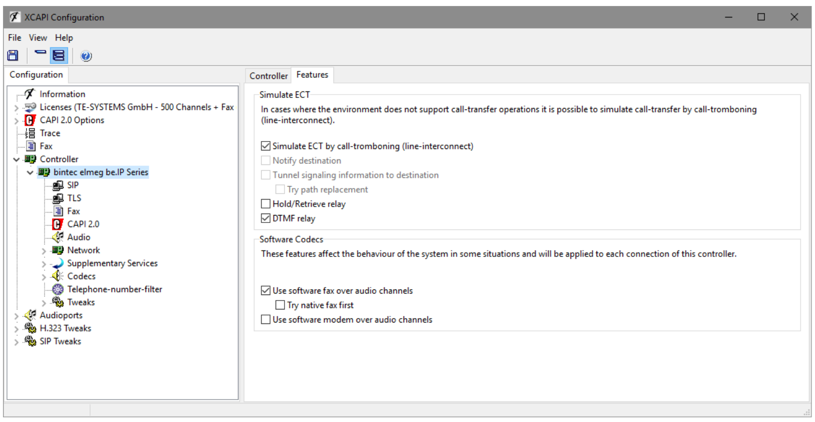
## 4.3 Call Transfer

For enabling call transfer via SIP refer **simulated ect by call-tromboning (line-interconnect)** has to be disabled within the XCAPI controller **features** tab. While our tests we observed that the CAPI 2.0 application has to set one channel on hold and the call transfer via SIP refer has to be initiated on the other channel. The call transfer via SIP refer should be tested as the behavior might vary according to circumstances (SIP provider, ISDN Port, Firmware etc.).
In the case of interoperability issues simulated call transfer has to be used as described as in the next chapter.



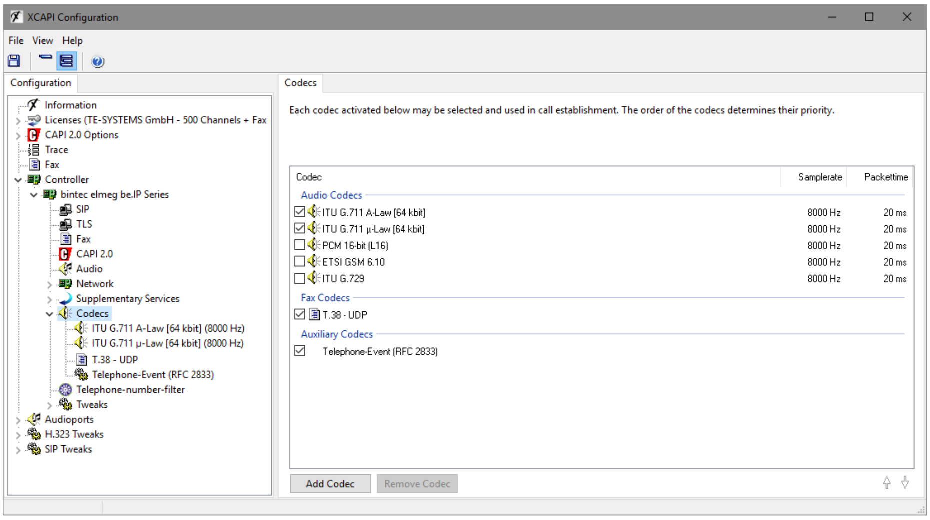
## 4.4 Simulated Call Transfer

Even though it is recommended to use the previously described call transfer via SIP refer, in some application specific cases or for interoperability reasons the **Simulated Call Transfer** has to be used. Whenever the CAPI application initiates a call transfer between two active participants, the XCAPI starts triggering the call transfer simulation. During this simulation two b-channels are occupied, but from application side the calls are released such as in a real call transfer scenario. So check the **Features** tab of the respective XCAPI controller and ensure that the **Simulate ECT by call-tromboning (line-interconnect)** parameter is enabled.

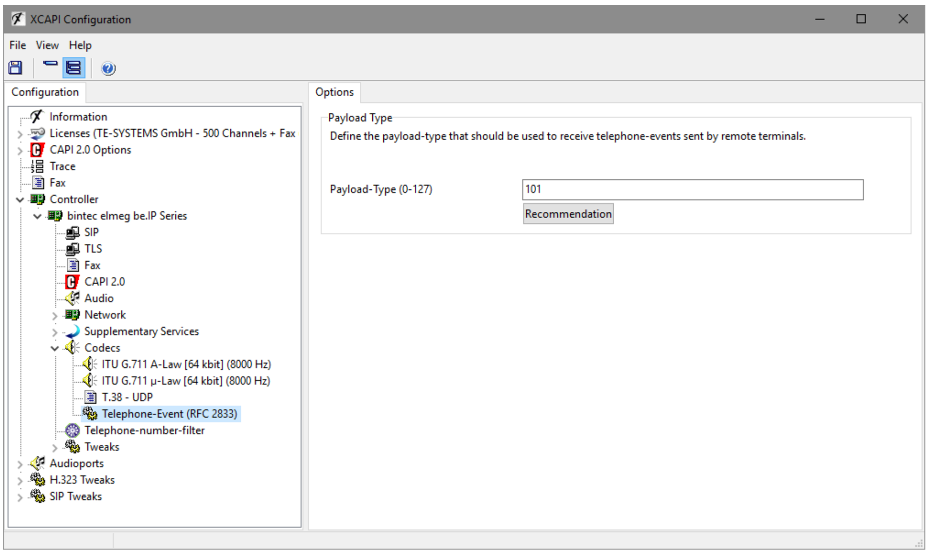


## 4.5 Codecs

As a general rule for codec usage and configurations for voice, fax and DTMF we recommend using conform selections and settings for all involved VoIP instances (Extensions, SIP Provider, Gateways etc.). The required codecs must be available and used with the same packet size. Any transcoding, especially in the case of fax, has to be avoided. For configuration details and hints about fax please check with the according **Appendix** chapters from page 14.




The RFC2833 payload is set by default to value 101.



### 4.6 Recommendations, Considerations and Restrictions

- It's recommended connecting XCAPI as SIP extension as described in this example. Nevertheless the XCAPI controller can also be connected via a **SIP Account** configuration of bintec elmeg be.IP with a few adjustments.
- Check the according TechNotes (XCAPI TechNote (en) - VMware Virtual Machines or XCAPI TechNote (en) - Microsoft Hyper-V) if XCAPI will be used in such a virtual environment.
- Ensure conform voice, fax and DTMF codec configurations for appropriate interworking between all be.IP SIP instances.
- Check if the be.IP numbering format and the required supplementary services are meeting the expectations for the bound CAPI 2.0 application.
- The be.IP seems not supporting any redirection number via **Diversion** or **History Info** yet.
- Also message waiting indications via **SIP Notify** seems not being supported by the be.IP yet.
- As fax is basically a real-time based protocol, it's strictly recommended to check the fax reliability in conjunction with the local and public network circumstances before XCAPI and the bound CAPI 2.0 application will be used in production.

# Exclusion of Liability



The information contained in this document was correct at the time of writing. TE-SYSTEMS GmbH reserves the right to make any alterations without prior notice.

The utmost care was applied during the compilation of texts and images, as well as during the creation of the software. Nevertheless, no responsibility can be taken for the content being accurate, up to date or complete, nor for the efficient or error-free operation of the software for a particular purpose. Therefore, TE-SYSTEMS GmbH cannot be held liable for any damages resulting directly or indirectly from the use of this document.

## Trademarks

All names of products or services used are trademarks or registered trademarks (also without specified indication) of the respective private or legal persons and are therefore subject to legal regulations.

## Third Party Disclaimer and Limitations

„OpenSSL", developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/), written by Eric Young (eay@cryptsoft.com) and written by Tim Hudson (tjh@cryptsoft.com).

„MD2, MD4 and MD5 Message Digest Algorithms" via source code derived from the RSA Data Security, Inc.

„RFC 4634 Secure Hash Algorithm", via source code derived from the RFC 4634.

## Copyright-Notices