ActiveTAPI-DyNAV

CTI for Microsoft Dynamics NAV

|  |  |
| --- | --- |
| **Author:** | **Manfred Gertz** |
| **Last Major Update:** | **15.01.2017** |
| **Relates to:** | **activeTAPI.COM-Dynav-v2017** |
| **Last Print Date:** | **15.01.2017** |
| **Source:** | [**http://www.activeTAPI.net**](http://www.activeTAPI.net) |
| **Pages:** | **40** |

# Table of Contents

[ActiveTAPI-DyNAV 1](#_Toc472342993)

[CTI for Microsoft Dynamics NAV 1](#_Toc472342994)

[Table of Contents 2](#_Toc472342995)

[The Basics 5](#_Toc472342996)

[activeTAPI-Runtime components 5](#_Toc472342997)

[Navision-Objects 5](#_Toc472342998)

[Other components 5](#_Toc472342999)

[Other components 6](#_Toc472343000)

[System requirements 7](#_Toc472343001)

[Supported Navision Versions 7](#_Toc472343002)

[Upgrade considerations 7](#_Toc472343003)

[Installation 9](#_Toc472343004)

[Quickstart 10](#_Toc472343005)

[Installation the activeTAPI-DyNAV Runtime components 10](#_Toc472343006)

[Prepare the Microsoft Dynamics NAV Database 10](#_Toc472343007)

[Changes in the standard objects of NAV 2017 10](#_Toc472343008)

[Table 18 - Customer 10](#_Toc472343009)

[Table 23 - Vendor 11](#_Toc472343010)

[Table 91 - User Setup 12](#_Toc472343011)

[Table 5050 - Contact 12](#_Toc472343012)

[Page 21 - Customer Card 13](#_Toc472343013)

[Page 22 - Customer List 14](#_Toc472343014)

[Page 26 - Vendor Card 14](#_Toc472343015)

[Page 27 - Vendor List 15](#_Toc472343016)

[Page 119 - User Setup 15](#_Toc472343017)

[Page 5050 - Contact Card 15](#_Toc472343018)

[Page 5052 - Contact List 16](#_Toc472343019)

[Codeunit 5055 - CustVendBank-Update 17](#_Toc472343020)

[Codeunit 5056 - CustCont-Updates 18](#_Toc472343021)

[Codeunit 5057 - VendCont-Updates 19](#_Toc472343022)

[New Navision-Objects 21](#_Toc472343023)

[Table 56780 - Tapi Logfile / TAPI Logbuch 21](#_Toc472343024)

[Table 56781 - Phone Index / Telefon-Index 21](#_Toc472343025)

[Table 56782 - TAPI Setup/TAPI Einrichtung 22](#_Toc472343026)

[Table 56783 - TAPI User / TAPI Benutzer 24](#_Toc472343027)

[Table 56784 - TAPI Line / TAPI Leitung 25](#_Toc472343028)

[Table 56785 - Note for Phone Index / Hinweis Telefon-Index 25](#_Toc472343029)

[Table 56786 - Phone Index temp / Telefon-Index temp 25](#_Toc472343030)

[Page 56780 - Menu Phone No. found / Auswahl Telefonnr. gefunden 26](#_Toc472343031)

[Page 56781 - Incoming call / Ankommender Anruf 26](#_Toc472343032)

[Page 56782 - TAPI Setup Card / TAPI Einrichtung Karte 26](#_Toc472343033)

[Page 56783 - TAPI User / TAPI Benutzer 27](#_Toc472343034)

[Page 56784 - TAPI Logs / TAPI Protokolle 27](#_Toc472343035)

[Form 56785 - Phoning Menu / Telephony (old) 27](#_Toc472343036)

[Page 56785 – Page with Timer 28](#_Toc472343037)

[Page 56786 - Menu Phone No. not found / Auswahl Telefonnr. nicht gefunden 28](#_Toc472343038)

[Page 56787 - TAPI Lines 29](#_Toc472343039)

[Page 56788 - End of call in 29](#_Toc472343040)

[Page 56789 - End of Call out 29](#_Toc472343041)

[Page 56790 Phone Indexes / Telefon Indices 30](#_Toc472343042)

[Page 56791 Notes for Phone Index 30](#_Toc472343043)

[Page 56792 Find Phone No. 30](#_Toc472343044)

[Report 56780 – Fill Phone Index 30](#_Toc472343045)

[Report 56781 – Update TAPI Lines 31](#_Toc472343046)

[Report 56782 – Delete TAPI Logfile 31](#_Toc472343047)

[Codeunit 56780 - TAPI Functions 31](#_Toc472343048)

[Codeunit 56781 - TAPI Event Publisher 35](#_Toc472343049)

[Codeunit 56782 – TAPI Functions SI 35](#_Toc472343050)

[Codeunit 56783 – TAPI Event Handler 35](#_Toc472343051)

[MenuSuite 1054 Dept – Add-on 4 36](#_Toc472343052)

[Upgrade information 37](#_Toc472343053)

[Important notes at the end 38](#_Toc472343054)

[Notes for developers - FAQ 38](#_Toc472343055)

[Contact Information 40](#_Toc472343056)

Introduction

activeTAPI-DyNAV is a solution that allows you to integrate “the Telephone” with Microsoft Dynamics NAV (formerly known as “Navision”). activeTAPI-DyNAV has been developed with activeTAPI, a basic set of CTI components, that allows you to integrate CTI with almost every application. The activeTAPI encapsulate the Windows Telephony API (known as TAPI), to gain access to the Telephony.

This means: activeTAPI-DyNAV is based on the Windows Standard – TAPI.

For that reason you don’t need any additional/special hardware to make it work. Of course, you need a Telephony hardware that is TAPI-compliant (has a TAPI-Device driver)[[1]](#footnote-1). That’s the only prerequisite you must fulfill.

The promise we make**:**

**If you can dial and receive telephone calls with your Windows Dialer, then you’ll be able to use Telephony with Microsoft Dynamics NAV (inbound and outbound)!**

# The Basics

activeTAPI-DyNAV consists of several components that will be explained here.

## activeTAPI-Runtime components

activeTAPI is a set of components that encapsulates the direct access to the Windows TAPI. These components are necessary to make TAPI available for high-level programming languages like C/SIDE. The activeTAPI components are well documented (CHM-Help-File) and you are allowed to use these components with Microsoft Dynamics NAV.[[2]](#footnote-2)

## Navision-Objects

The Navision-Objects are a set of objects that realize the Telephony functionality for a specific Navision version. Every set of Navision-Objects is available as a FOB-File.

Actually there exists different version of activeTAPI:

### Versions for older NAV Versions (before NAV 2017

Versions start with activeTAPI for NAV 3.70 up to NAV 2016.

### Version activeTAPI-DyNav for NAV 2017

In this version we made a break: We integrated events, so this version cannot be downgraded any more.

It consist of

AT-v2017-RTM-modified-standard-objects (fob and txt)

AT-v2017-RTM-new objects (fob and txt)

## Other components

activeTAPI-DyNAV contains some more additional component that should be mentioned here:

* Additional applications like activeTAPI-Dialer
* Documentation
* activeTAPI Online help for the Automation Objects
* Detailed documentation in the Navision objects
* Release notes for the actual version
* “Getting started” document
* This Document for the integration of activeTAPI-DyNAV into Navision including the “Quickstart”.

## Other components

activeTAPI-DyNAV contains some more additional component that should be mentioned here:

* Additional applications like activeTAPI-Dialer
* Documentation
* activeTAPI Online help for the Automation Objects
* Detailed documentation in the Navision objects
* Release notes for the actual version
* “Getting started” document
* This Document for the integration of activeTAPI-DyNAV into Navision including the “Quickstart”.

# System requirements

As mentioned earlier, all you need is a TAPI compliant telephone hardware. This can be any telephone hardware, like a modem, an ISDN card, a telephone, a PBX, a VoIP network, … - TAPI is hardware independent!

All you need is a TAPI device driver for your hardware (a TSP). Please refer to your telephone hardware manual or ask your telephone hardware vendor if your device supports TAPI, which additional requirements this device may have and how these devices connect to your PC or network (serial, USB, LAN, …).

When you can dial a telephone number with your Windows Dialer, then it is time to install and launch activeTAPI-DyNAV.

activeTAPI Windows runs on Windows 7, 8 or 10.

## Supported Navision Versions

If you need versions of activeTAPI for Navision versions older than NAV 2017, please contact us.

## Upgrade considerations

There are two parts of upgrades:

* Objects: There is no possibility to upgrade from older versions, because activeTAPI v2017 integrates new possibilities like events.
* Data: The data structure is the same as for activeTAPI v2016, v2015, …

### Changes in Navision-Objects v2015

The changes were:

* The Startpage (56785 Page with Timer) is no longer started with the properties of the NAV-Desktop-Icon. It is now started in Codeunit 40, Trigger “LogInStart”, to avoid issues with “Change Company”.
* In Report 56780 "Fill Phone Index" there was no Request Page. It was added now.
* .NET-Check in Codeunit 40 (thanks to Cajus)
* In Page 56784 "TAPI Logs" there is a page action "Dial" (Anrufen). The Code was changed. Now always the Remote ID is dialed (if not empty)
* In the Table 56782 TAPI Setup there is new field "Runs in Developer Mode". This field can only be changed directly in the table (developers should know how to do this ...). If it is activated, activeTAPI automation object "TAPI Functions SI" and “Page with Timer” is not started in Codeunit 40, LoginStart.  
  So Developers, that have not installed activeTAPI, will not get automation errors anymore. But take care: You should also deactivate "Format Phone Numbers" and "Format Fax Numbers" in the TAPI SetUp. Otherwise you will get automation errors, if you change any Phone No.  
  AND: Don't forget, to deactivate "Runs in Developer Mode" at the end of your work.

### Changes in Navision-Objects v2016

* No changes, only adapted from NAV 2015 to NAV 2016 (CU07).

### Changes in Navision-Objects v2017

Upgrade Info

* Events were introduced, to minimize the changes to be made in NAV standard objects.

# Installation

Run activeTAPI.COM-DyNav\_v2017\_Setup.msi to install and register the activeTAPI-Runtime components on the local machine. This installation step is mandatory, because the activeTAPI-Runtime components are the basis for the Navision-Objects.

Optionally you can install (copy) the delivered Navision-Objects into the installation directory. You can then import these Navision-Objects into your Microsoft Dynamics NAV application. The Navision-Objects are not locked and can be modified and/or enhanced the way you like. Normally you’ll install the Navision-Objects on the machine that is used to implement new Navision functionalities, like Telephony in that case.

Again, the activeTAPI-Runtime-components must be installed on EVERY workstation, on which telephone functionality will be used with Navision. This concerns also workstations where "only" telephone numbers of customers, vendors, contacts are entered/changed (ref. Phone Index) or formatted. But good news: If a user does not telephone with activeTAPI in NAV, he needs no license.

Upgrade Info: Before you can install the activeTAPI-Runtime components, you must uninstall an existing version of activeTAPI! Previous version of activeTAPI won’t run with actual version.

Upgrade Info

And after deinstallation, it is a good idea, to clean the registry with tools like CCleaner or others

# Quickstart

No matter, whether you are working in a 1st-Party-situation (your local telephone is connected to your local PC) or in a 3rd-Party-situation (your telephone system is connected with your network server): Try whether you can make a call with the standard Windows Dialer (Accessories, Telephone). If this does not work, your telephone hardware is not correct or not correctly connected to your telephone (system) or the setup is not correct (the following informations are based on the newest version v2017).

## Installation the activeTAPI-DyNAV Runtime components

With a standard installation by call of activeTAPI.COM-DyNav\_v2017\_Setup.msi.

## Prepare the Microsoft Dynamics NAV Database

This topic is described in the document *activeTAPI.COM-DyNav v2017 - Getting Started.doc.*

# Changes in the standard objects of NAV 2017

The comments in the objects are at least in English (partly English/German) and the objects are multi-language enabled (US-English (ENU) and German (DEU)).

Changes of the standard objects are based NAV 2017 RTM Build 13682.

The Version History about older versions of activeTAPI was deleted. If you need is, you can find it in the older document “activeTAPI-DyNAV\_v2016\_ENU.doc.

## Table 18 - Customer

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.12.2016** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers 4 Events implemented** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

In the OnValidate trigger of the field 2 Name:

IF ("Search Name" = UPPERCASE(xRec.Name)) OR ("Search Name" =

'') THEN

"Search Name" := Name;

// AT2017 BEGIN

IF Name <> xRec.Name THEN

ATEvPub.OnCustomerNameChanged(Rec);

// AT2017 END

In the OnValidate trigger of the field 9 Phone No.

// AT2017 BEGIN

ATEvPub.OnCustomerPhoneNoChanged(Rec);

GET("No.");

// AT2017 END

In the OnValidate trigger of the field 39 Blocked

// AT2017 BEGIN

IF Blocked <> xRec.Blocked THEN

ATEvPub.OnCustomerBlockedChanged(Rec);

// AT2017 END

In the OnValidate-Trigger of the field 84 Fax No.

// AT2017 BEGIN

ATEvPub.OnCustomerFaxNoChanged(Rec);

GET("No.");

// AT2017 END

## Table 23 - Vendor

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.12.2016** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **4 Events implemented** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

In the OnValidate trigger of the field 2: Name

IF ("Search Name" = UPPERCASE(xRec.Name)) OR ("Search Name" =

'') THEN

"Search Name" := Name;

// AT2017 BEGIN

IF Name <> xRec.Name THEN

ATEvPub.OnVendorNameChanged(Rec);

// AT2017 END

In the OnValidate trigger of the field 9 Phone No.

// AT2017 BEGIN

ATEvPub.OnVendorPhoneNoChanged(Rec);

GET("No.");

// AT2017 END

In the OnValidate trigger of the field 39 Blocked

// AT2017 BEGIN

IF Blocked <> xRec.Blocked THEN

ATEvPub.OnVendorBlockedChanged(Rec);

// AT2017 END

In the OnValidate-Trigger of the field 84 Fax No.

// AT2017 BEGIN

ATEvPub.OnVendorFaxNoChanged(Rec);

GET("No.");

// AT2017 END

## Table 91 - User Setup

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **01.12.2016** | **New field 56780 (with Code and local Var.)**  **Additional Tests**  **new field 56781** |

### Changes

New field

56780 Use activeTAPI, Boolean

DEU: activeTAPI verwenden

local Var: TAPIUser, record, 56783 (TAPI User)

56781 See TAPI Logfile completely, Boolean

DEU: TAPI Logbuch komplett ansehen

In the OnValidate trigger of the field 56780 Use active TAPI

// AT2017 BEGIN

IF "Use activeTAPI" THEN BEGIN

IF TAPIUser.GET("User ID") THEN BEGIN

TAPIUser.Blocked:= FALSE;

TAPIUser.MODIFY;

END;

END ELSE BEGIN

IF TAPIUser.GET("User ID") THEN BEGIN

TAPIUser.Blocked:= TRUE;

TAPIUser.MODIFY;

END;

END;

// AT2017 END

## Table 5050 - Contact

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.12.2016** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **4 Events implemented** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

In the OnValidate trigger of the field 2: Name:

NameBreakdown;

ProcessNameChange;

// AT2017 BEGIN

IF Name <> xRec.Name THEN

ATEvPub.OnContactNameChanged(Rec);

// AT2017 END

In the OnValidate trigger of the field 9 Phone No.

// AT2017 BEGIN

ATEvPub.OnContactPhoneNoChanged(Rec);

GET("No.");

// AT2017 END

In the OnValidate-Trigger of the field 84 Fax No.

// AT2017 BEGIN

ATEvPub.OnContactFaxNoChanged(Rec);

GET("No.");

// AT2017 END

In the OnValidate-Trigger of the field 5061 Mobile Phone No.

// AT2017 BEGIN

ATEvPub.OnContactMobilePhoneNoChanged(Rec);

GET("No.");

// AT2017 END

## Page 21 - Customer Card

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
|  |  |  |
| **AT2017** | **02.11.2016** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **Event added (Phone No. OnAssistEdit)** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

OnAssistEdit trigger of Phone No.

// AT2017 BEGIN

ATEvPub.OnCustomerPagesDialPhoneNo(Rec);

// AT2017 END

## Page 22 - Customer List

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
|  |  |  |
|  |  |  |
| **AT2017** | **02.11.2017** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **Event added (Phone No. OnLookup)**  **New action (Dial / Anrufen) and event added OnAction** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

OnLookup trigger of Phone No.

// AT2017 BEGIN

ATEvPub.OnCustomerPagesDialPhoneNo(Rec);

// AT2017 END

In Page Actions, ActionContainer, ActionItems new action:

ID=1000056780, Caption DEU=Anrufen, ENU=Dial; Promoted Yes, PromotedCategory Process, Image Calls, Scope Repeater

OnAction trigger

// AT2017 BEGIN

ATEvPub.OnCustomerPagesDialPhoneNo(Rec);

// AT2017 END

## Page 26 - Vendor Card

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.11.2017** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **Event added (Phone No. OnAssistEdit)** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

OnLookup trigger of Phone No.

// AT2017 BEGIN

ATEvPub.OnCustomerPagesDialPhoneNo(Rec);

// AT2017 END

## Page 27 - Vendor List

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
|  |  |  |
| **AT2017** | **02.11.2017** | **Event added (Phone No. OnLookup)**  **New action (Dial / Anrufen) and event added OnAction** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

OnLookup trigger of Phone No.

// AT2017 BEGIN

ATEvPub.OnVendorPagesDialPhoneNo(Rec);

// AT2017 END

In Page Actions, ActionContainer, ActionItems new action:

ID=1000056780, Caption DEU=Anrufen, ENU=Dial; Promoted Yes, PromotedCategory Process, Image Calls, Scope Repeater

OnAction trigger

// AT2017 BEGIN

ATEvPub.OnVendorPagesDialPhoneNo(Rec);

// AT2017 END

## Page 119 - User Setup

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **01.11.2016** | **2 fields added"** |

## Page 5050 - Contact Card

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
|  |  |  |
| **AT2017** | **02.11.2017** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **2 Events added (OnAssistEdit on Phone No. and Mobile Phone No)** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

OnAssistEdit trigger of Phone No.

// AT2017 BEGIN

ATEvPub.OnContactPagesDialPhoneNo(Rec);

// AT2017 END

OnAssistEdit trigger of Mobile Phone No.

// AT2017 BEGIN

ATEvPub.OnContactPagesDialMobilePhoneNo(Rec);

// AT2017 END

## Page 5052 - Contact List

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.11.2017** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **2 Event added (OnLookup Phone No. and Mobile Phone No.)**  **OnAction MakePhone “TAPIManagement …” outcommented**  **New event OnContactPagesDialPhoneNo** |

### Changes

New global variable

ATEvPub, Codeunit, 56781 (TAPI Event Publishers)

OnLookup trigger of Phone No.

// AT2017 BEGIN

ATEvPub.OnContactPagesDialPhoneNo(Rec);

// AT2017 END

OnLookup trigger of Mobile Phone No.

// AT2017 BEGIN

ATEvPub.OnContactPagesDialMobilePhoneNo(Rec);

// AT2017 END

In Page Actions, ActionContainer, ActionItems, MakePhoneCall:

// AT2017

{ ORIGINAL

TAPIManagement.DialContCustVendBank(Database::Contact,“No.”, GetDefaultPhoneNo,‘’);

END ORIGINAL }

ATEvPub.OnContactPagesDialPhoneNo(Rec);

// AT2017 END

ID=1000056780, Caption DEU=Anrufen, ENU=Dial; Promoted Yes, PromotedCategory Process, Image Calls, Scope Repeater

OnAction trigger

// AT2017 BEGIN

ATEvPub.OnVendorPagesDialPhoneNo(Rec);

// AT2017 END

## Codeunit 5055 - CustVendBank-Update

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.11.2017** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **UpdateCustomer event added**  **UpdateVendor event added** |

### Changes

In the UpdateCustomer trigger

WITH Cust DO BEGIN

GET(ContBusRel."No.");

NoSerie:= "No. Series ";

VatRegNo := “VAT Registration No.”;

TRANSFERFIELDS(Cont);

"No." := ContBusRel."No.";

"No. Series ":= NoSerie;

MODIFY;

// AT2017 BEGIN

ATEvPub.OnUpdateCustomerFromContact(Cust, Cont);

// AT2017 END

IF ("VAT Registration No." <> '') AND ("VAT Registration

No." <> VATRegNo) THEN

VATRegistrationLogMgt.LogCustomer(Cust);

END;

In the UpdateVendor trigger

WITH Vend DO BEGIN

GET(ContBusRel."No.");

NoSerie:= "No. Series ";

PurchaserCode:=   Vend."Purchaser code";

VATRegNo := "VAT Registration No.";

TRANSFERFIELDS(Cont);

"No." := ContBusRel."No.";

"No. Series ":= NoSerie;

"Purchaser Code":= PurchaserCode;

MODIFY;

// AT2017 BEGIN

ATEvPub.OnUpdateVendorFromContact(Vend, Cont);

// AT2017 END

IF ("VAT Registration No." <> '') AND ("VAT Registration

No." <> VATRegNo) THEN

VATRegistrationLogMgt.LogVendor(Vend);

END;

## Codeunit 5056 - CustCont-Updates

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.11.2016** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **3 events added** |

### Changes

In the OnModify trigger  
local Var PhoneNoChanged, Boolean  
At the end:

...

ContNo:= Cont."No.";

NoSeries:= Cont."No. Series ";

Cont.VALIDATE("E-Mail",Cust."E-Mail");

Cont.TRANSFERFIELDS(Cust);

Cont."No." := ContNo;

Cont."No. Series ":= NoSeries;

Cont.VALIDATE(Name);

Cont.OnModify(OldCont);

// AT2017 BEGIN

ATEvPub.OnUpdateContactFromCustomer(Cont,Cust);

// AT2017 END

Cont.MODIFY(TRUE);

In the InsertNewContact (in the center);

WITH Cont DO BEGIN

INIT;

TRANSFERFIELDS(Cust);

Cont.VALIDATE(Name);

"No." := '';

"No. Series ":= '';

RMSetup.TESTFIELD("Contact Nos.");

NoSeriesMgt.InitSeries(RMSetup."Contact Nos.",'',0D," No."," No. Series");

Type:= Type::Company;

TypeChange;

SetSkipDefault(FALSE);

INSERT(TRUE);

// AT2017 BEGIN

ATEvPub.OnUpdateContact(Cont);

// AT2017 END

END;

In the InsertNewContactPerson trigger (at the end):

IF ContBusRel.FINDFIRST THEN

IF ContComp.GET(ContBusRel."Contact No.") THEN

WITH Cont DO BEGIN

INIT;

"No." := '';

INSERT(TRUE);

"Company No." := ContComp."No.";

Type:= Type::Person;

VALIDATE(Name, Cust.Contact);

InheritCompanyToPersonData(ContComp, FALSE);

MODIFY(TRUE);

Cust."Primary Contact No." := "No.";

// AT2017 BEGIN

ATEvPub.OnUpdateContact(Cont);

// AT2017 END

END;

## Codeunit 5057 - VendCont-Updates

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **02.11.2016** | **Global Var ATEvPub Codeunit 56781 TAPI Event Publishers**  **3 events added** |

### Changes

In the OnModify trigger  
local Var PhoneNoChanged, Boolean  
At the end:

ContNo:= Cont."No.";

NoSeries:= Cont."No. Series ";

SalespersonCode := Cont."Salesperson Code";

Cont.VALIDATE("E-Mail",Vend."E-Mail");

Cont.TRANSFERFIELDS(Vend);

Cont."No." := ContNo;

Cont."No. Series ":= NoSeries;

Cont.VALIDATE(Name);

Cont.OnModify(OldCont);

Cont.MODIFY(TRUE);

// AT2017 BEGIN

ATEvPub.OnUpdateContactFromVendor(Cont,Vend);

// AT2017 END

In the InsertNewContact (in the center);

WITH Cont DO BEGIN

INIT;

TRANSFERFIELDS(Vend);

VALIDATE(Name);

VALIDATE("E-Mail");

"No." := '';

"No. Series ":= '';

RMSetup.TESTFIELD("Contact Nos.");

NoSeriesMgt.InitSeries(RMSetup."Contact Nos.",'',0D," No."," No. Series");

Type:= Type::Company;

TypeChange;

SetSkipDefault;

INSERT(TRUE);

// AT2017 BEGIN

ATEvPub.OnUpdateContact(Cont);

// AT2017 END

END;

In the InsertNewContactPerson trigger (at the end):

IF ContBusRel.FINDFIRST THEN

IF ContComp.GET(ContBusRel."Contact No.") THEN

WITH Cont DO BEGIN

INIT;

“No." := '';

INSERT(TRUE);

"Company No." := ContComp."No.";

Type:= Type::Person;

VALIDATE(Name, Vend.Contact);

InheritCompanyToPersonData(ContComp, FALSE);

MODIFY(TRUE);

Vend."Primary Contact No." := "No.";

// AT2017 BEGIN

ATEvPub.OnUpdateContact(Cont);

// AT2017 END

END;

# New Navision-Objects

## Table 56780 - Tapi Logfile / TAPI Logbuch

In this table all in- and outgoing telephone calls are logged and can be used for further statistics and reports. The page for this table is the page 56784 "TAPI Logs".

### Version History

|  |  |  |
| --- | --- | --- |
| **Documenta-tion Trigger** | **Date** | **Note / Changes** |
| **AT01.20** | **08.09.2003** | **New Fields: 70 Remote ID Text 30 71 Remote ID Name Text 30 80 Local ID Text 30 81 Local ID Name Text 30** |
| **AT05.00** | **16.06.2007** | **Length of all name-fields changed from 30 to 50** |
| **AT2009** | **06.11.2009** | **New Index User,Entry No. for List per User** |
| **AT2016** | **30.06.2016** | **Field User from len 20 to len 50 (thanks to Werner)** |  |  |

## Table 56781 - Phone Index / Telefon-Index

This table contains all telephone numbers, which are stored in Navision in a defined format (Plain) and an allocation to the Navision table, in which the telephone number was registered.

This way was chosen, so that for the caller identification of incoming calls it is not necessary to scan several tables for the telephone number.

The table is always maintained, if telephone number is entered, changed or deleted or if a customer, vendor or contact is deleted.

The report 56780 “Fill Phone index” (see below) can be used, to fill the Phone Index from the existing telephone numbers at the beginning of the work with activeTAPI-DyNAV.

You can see error messages from the report (double numbers, etc..) in the table 56785 “Note for Phone Index” or in the Page 56791 “Notes for Phone-Index”.

You can use Page 56790 “Phone Indexes” to change the Phone Index. This is usually necessary after running the report, because in the database there are senseless telephone numbers like "."," 0 "," 99999 ", etc.. These should be removed from the Phone Index (and naturally also from the telephone number fields).

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
|  |  |  |
| **AT05.00** | **16.06.2007** | **Length of name-field changed from 30 to 50** |
| **AT2009** | **06.11.2009** | **New Field "Blocked"; Field "Blocked" added to all Index-Keys** |

## Table 56782 - TAPI Setup/TAPI Einrichtung

This table contains the basic setup for the Telephony functionality (treatment with Page 56782 TAPI Setup Card). These are in detail:

### 10 length of internal phone No. / Länge interne Rufnummer

In order to be able to differentiate internal from external calls, you enter the (max.) length of internal call numbers, e.g. 2 (or 3).

### 20 Action for incoming call / Aktion eingehende Anrufe

This is a general default, if there is no entry in the TAPI User table.

### 30 Create call Data / Anrufdaten erstellen (for forwarding/for redirection)

Some TAPI drivers support the possibility, beside a GUID for each individual telephone call, provide in addition a 2. GUID for the entire call run.

Example: Reception accepts a call from a customer and forwards it to employee B. Actually this are 2 calls (customer > Reception, Reception > Employee B), which have in each case their own GUID. With the ability of the TAPI driver "Create call data" this two calls get the same second GUID, so that the calls can be recognized as coherently.

If your TAPI driver supports this feature, say Yes. If you are not sure, say No.

### 40 TAPI line name / Name TAPI Leitung

This is a general default, if there is no entry in the TAPI User table.

On condition that lines were read in (see MenuSuite 1054 Telephony (CTI), Setup, Update lines).

### 50 Trace level / Trace level

Can be set only internally (0 = no Tracing, 1 (Error) to 4 (Verbose). See also documentation in the code of the Codeunit 56781 TAPI Main.

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT01.20** | **08.09.2003** | **New Fields: 60 Min. length for Phoneindex (Integer) 70 Format Telephone numbers (Boolean)** |

### 60 Min. length für Phoneindex / Mindestlänge für Telefonindex

With the entry in the field 60 “Min. length for Phoneindex" can be defined, starting from which length telephone numbers are to be registered into the telephone index. Example: Internal telephone numbers (length 3) are not to be taken up to the telephone index: Entry in the field 60: 3.

### 70 Format Telephone numbers / Telefon-Nummern formatieren

If you choose Yes in the field 70 "Format Telephone numbers”, the telephone numbers of customers, suppliers and contacts are automatically converted after the input into the canonical format: +49 (211) 123456.

### 80 Format Fax numbers / Fax-Nummern formatieren

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT04.00** | **04.11.2005** | **New Fields: 80 Format Fax Numbers** |

If you input yes in field 80 „Format Fax Numbers, the Fax numbers of customers, vendors and contacts will be formatted directly after input to canonic format: +49 (211) 123456.

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2009** | **06.11.2009** | **New Fields: 90 Check of double numbers (in Phone Index) 100 Create Interaction Log Entry** |

### 90 Check of double numbers / Prüfung doppelte Nummern

If you activate this option, Navision will check after an input of a phone no., if this phone no. does already exist.

### 100 Create Interaction Log Entry / Aktivitäts-Protokollposten erstellen

If you activate this option, you will have the possibility, to create an Interaction Log Entry at the end of a phone call. The known fields are already filled

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2011** | **05.04.2011** | **New Field: 110 Log internal calls** |

### 110 Log internal calls / Interne Anrufe in Logbuch

If you activate this option, also the internal calls (see also field 10 Length of internal phone No.) will be stored to the Tapi Logfile.

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2015** | **02.12.2014** | **New field: 120 Runs in Developer Mode** |

### 120 Runs in Developer Mode / Läuft im Entwickler Modus

If you activate this option (you can only do that directly in the table, developers should know how to do this ...), activeTAPI automation object "TAPI Functions SI" and Startpage 56785 is not started in Codeunit 40, LoginStart. Developers, that have not installed activeTAPI, will not get automation errors anymore. But take care: You should also deactivate "Format Phone Numbers" and "Format Fax Numbers" in the TAPI SetUp. Otherwise you will get automation errors, if you change any Phone No.  
AND: Don't forget, to deactivate "Runs in Developer Mode" at the end of your work.

## Table 56783 - TAPI User / TAPI Benutzer

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2009** | **06.11.2009** | **New Field:** **Action for end of income. call Reference/Note.none Action for end of outgo. call Reference/Note.none** |
| **AT2016** | **30.06.2016** | **Field User from len 20 to len 50 (thanks to Werner)** |

Here you’ll find the setup for the individual TAPI user. On condition that in the “User Setup” for this user the field "Use active TAPI" was activated. These are in detail:

### 1 User

Select or enter User ID.

### 10 Action for incoming call (option)

Determines which action takes place when the caller conveys the call number and the number is found in Navision.

|  |  |
| --- | --- |
| **Show caller for short time** | **Number, name and origin will be shown for a short time (e.g. 3 Sec.)** |
| **Show card** | **The Card Page (customer, vendor, contact) is shown.** |
| **Show menu** | **Show a menu with different options (see Page 56780 Menu Phone No. found)** |

### 11 Action for new external telephone number

Determines which action takes place when the caller does not convey a call number or the telephone number was not found in Navision.

|  |  |
| --- | --- |
| **Open Contact Card** | **Opens the contact card** |
| **Show menu** | **Show a menu with different options (see Page 56786 menu Phone No. not found)** |

### 20 Start activeTAPI Dialer / activeTAPI Dialer starten

Automatically start the activeTAPI-Dialer with Navision.

### 21 Dialer Directory

Drive, path and filename of the dialer (selectable); with standard installation it is:  
<Program Files>\activeTAPI\Dialer\activeTAP-Dialer.exe

### 30 TAPI-LineName

Allocation of a line (device name) to the user (select); on condition that lines were read in (see Menu Suite 1054 " Telephony (CTI)", "Setup”, "Update lines").

### 90 Blocked (not editable)

If in the User Setup the field "Use active TAPI" is deactivated for the user, the user will not be deleted as activeTAPI-user, but he will be blocked.

### 100 Action for end of incom. call

Some people don’t like any action at the end of a call. So they can choose None here.

### 110 Action for end of outgo. call

Some people don’t like any action at the end of a call. So they can choose None here.

### 200 HelpField

For internal use only

### 201 GUIDTemp

For internal use only

### 202 BLOBTemp

For internal use only

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2009** | **06.11.2009** | **New Fields: Action for end of incom. Call Reference/Note,none Action for end of outgo. Call Reference/Note,none** |

## Table 56784 - TAPI Line / TAPI Leitung

Internal table: If the lines (Line Device Names) were read in (ref. Menu Suite 1054 " Telephony (CTI)", "Setup", "Update lines"), they are stored here; shown with Page 56787 "TAPI Lines”.

## Table 56785 - Note for Phone Index / Hinweis Telefon-Index

This table contains error messages from the report 56780 "Fill Phone Index". Contents can be looked at and worked on with Page 56791 "Notes for Phone Index".

## Table 56786 - Phone Index temp / Telefon-Index temp

This table is equal to table 56781 Phone Index / Telefon-Index. Field 7 Table Name / Tabellen-Name (Text 50) was added. This table is used temporarily for the search of phone numbers.

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT05.00** | **16.06.2007** | **Length of all name fields changed from 30 to 50** |

## 

## Page 56780 - Menu Phone No. found / Auswahl Telefonnr. gefunden

Menu selection for incoming calls (number found) with the possibility, to jump into the respective card, into the item list, for customers into quotes, orders, invoices, credit notes (open and posted).

Appears, if in the table "TAPI User" in the field 10 "Action for incoming call / Aktion bei eingehendem Anruf” the option "Show menu / Auswahl anzeigen" was selected.

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT05.00** | **16.06.2007** | **Length of all name fields changed from 30 to 50** |
| **AT05.00.01** | **13.03.2008** | **New connection with automation objects** |
| **AT2011** | **05.04.2011** | **Changes for RTC: CREATE, Order of buttons** |
| **AT2011.31** | **11.10.2011** | **Reassignment of Automation Variables** |
| **AT2013** | **11.05.2013** | **FORM replaced by PAGE** |

## Page 56781 - Incoming call / Ankommender Anruf

Short announcement of the caller (Name, Phone No., ...);

Appears, if in the table "TAPI user" in field 10 "Action for incoming call / Aktion bei eingehendem Anruf” the option "Show caller for short time / Anrufer kurz anzeigen” was selected.

In the Properties of the Page (Timer Interval) the time, how lang the Page should be shown, can be set. (default is 4000 = 4 Millisec.).

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT05.00** | **16.06.2007** | **Length of all name fields changed from 30 to 50** |

## Page 56782 - TAPI Setup Card / TAPI Einrichtung Karte

Page for Table 56782 "TAPI Setup" (see there); called from the Menu Suite 1054, " Telephony (CTI)", "Setup”, "TAPI Setup”.

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2017** | **15.01.2017** | **Action: Create Setup Data** |

## Page 56783 - TAPI User / TAPI Benutzer

Page for the Table 56783 "TAPI User" (see there); called from the Menu Suite 1054, “Telephony (CTI)”, "TAPI User".

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2009** | **06.11.2009** | **New fields: "Action for end of incom. call", "Action for end of outgo. call"** |
| **AT2009.01** | **07.07.2010** | **Changes for RTC: IMPORT bei Directory of dialer** |

## Page 56784 - TAPI Logs / TAPI Protokolle

Page for the Table 56780 "Tapi Logfile"; called from the Menu Suite 1054, " Telephony (CTI)", "Journal TAPI".

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT01.20** | **08.09.2003** | **Fields added: Local ID (Name), Remote ID (Name)**  **Fields deleted: User info., UserInfo1** |
| **AT2009** | **06.11.2009** | **Global Var UserSetup Rec 91 Filter according to User SetUp SourceTableView changed from SORTING(Entry No.) to SORTING(User,Entry No.) Global Vars: PhoneNo Text 30 TAPIFu CU 56780 Button Dial** |
| **AT2013** | **11.05.2013** | **Button Dial: Promoted Yes, PromotedCategory Process, Image Calls** |
| **AT2015** | **06.11.2014** | **Dial Remote ID** |

## Form 56785 - Phoning Menu / Telephony (old)

Menu for the Telephony, inserted in Form 330 "Main Menu" (bottom left: Telephony). The menu options in detail:

### TAPI User

Setup of the TAPI Users (precondition: In the User Setup the field “Use active TAPI” was activated for the user).  
Call of the Form 56783 "TAPI User" (Table 56783 "TAPI User")

### Reports

Actually there are three reports:

Report 56780 "Fill Phone Index" (see there), which can be called also by “Periodic Activities", "Fill Phone Index”.

Journal TAPI

List of all stored telephone calls  
Call of Form 56784 "TAPI Logs" (Table 56780 "Tapi Logfile")

### Periodic Activities

|  |  |
| --- | --- |
| **Fill Phone Index** | **The report should be run one time at the start of the work with activeTAPI-DyNAV, to fill the Phone Index with the telephone numbers already stored in Navision. Call of Report 56780 "Fill Phone Index** |
| **Notes for telephone index** | **List and modification of error messages from filling the telephone index Call of Form 56791 "Notes for Phone index" (Table 56785 "Note for Phone index")** |
| **Modify Phone Index** | **After filling the Phone Index it can modified, to delete senseless telephone numbers like "."," 0 "," 999999 ", etc. Call of Form 56790 "Phone Indexes" (Table 56781 "Phone Index")** |

### Setup

|  |  |
| --- | --- |
| **Update of lines** | **Before the allocation of lines (Line Device Names) in the menu "TAPI Setup" or "TAPI User" the lines must be read in into Navision.** |
| **TAPI Setup** | **TAPI Basic Adjustments (see Table 56782 "TAPI Setup")** |

## Page 56785 – Page with Timer

What Codeunit 56789 TAPI Main did together with Classic Client, does Page with Timer for RT-Client. It is always started with the RT-Client (see Getting started) and is must always run, to handle the telephony events. The page contains the Timer-AddIn, that transfers the events from activeTAPI to the RT-Client. The Page has PageType “ComfirmationDialog”. Therefore it is not visible. If you want to see it, for example for development purposes, change the PageType to Card.

## Page 56786 - Menu Phone No. not found / Auswahl Telefonnr. nicht gefunden

Menu selection for incoming calls (number not found) with the following options:

* Search (customer list)
* Insert (customer or contact)
* Information (item list)

After inserting a new customer / contact you can search again for the telephone number. If the telephone number is found now, the card 56780 "Menu Phone No. Found" is shown.

The menu selection appears, if in the table "TAPI User" in the field 10 "Action for incoming call " the option "Show menu" was selected.

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT01a** | **13.03.2008** | **New Connection with Automation objects** |
| **AT2011** | **05.04.2011** | **Changes for RTC: CREATE** |
| **AT2011.3** | **11.10.2011** | **Reassignment of Automation Variables** |
| **AT2013** | **11.05.2013** | **FORM replaced by PAGE** |

## Page 56787 - TAPI Lines

List Page for the lines (Line Device Names), that were read in (Table 56784 "TAPI Line")

## Page 56788 - End of call in

Final Page for incoming calls; appears after completion of an incoming call.

Here the user can input a Reference (Text 30) and/or a Note (Text 80). With the button "Finished" reference and note are written to the "TAPI Log". With the button "Done" these informations are not written to the "TAPI Log".

### Version History

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT05.00** | **16.06.2007** | **Length of all name fields changed from 30 to 50** |
| **AT2009** | **06.11.2009** | **Button "Call back", if found** |
| **AT2011** | **05.04.2011** | **Changes for RTC (Order of buttons)** |
|  | **08.04.2011** | **Old Code deleted, Text found corrected and transferred to OnOpenPage** |

## Page 56789 - End of Call out

Final Page for outgoing calls; appears after completion of an outgoing call.

Here the user can input a reference (Text 30) and/or a note (Text 80). With the button "Done" the reference and note are written into the "TAPI Log". With the button "Done" these informations are not written into the "TAPI Log".

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT04.00** | **04.11.2005** | **Dial again if not connected** |
| **AT05.00** | **16.06.2007** | **Length of all name fields changed from 30 to 50** |
| **AT2009** | **06.11.2009** | **Option Create Interaction Log Entry** |
| **AT2011** | **05.04.2011** | **Changes for RTC (Order of buttons)** |
|  | **08.04.2011** | **TextFound transferred to OnOpenPage** |

## Page 56790 Phone Indexes / Telefon Indices

List- and modification-Page for the Phone Index (Table 56781 "Phone Index").

## Page 56791 Notes for Phone Index

List- and modification-Page for the error messages from filling the Phone Index from the report 56780 "Fill Phone Index" (Table 56785 "Note for Phone-Index").

## Page 56792 Find Phone No.

In this Page you can search – like Navigate – for a telephone number in Navision. The search takes place in the Phone-Index. Results are shown with Table-Name, Customer-/Vendor-/Contact-Number with the possibility to go to the card (Button Show).

You can also search for parts of the numbers like 171 or 3546. If you input a number with leading zero or special signs like -+()/ (Example: 0172-1234567) the leading zero and the special signs will be deleted (1721234567). Otherwise the number cannot be found in the Phoneindex -the number there looks like 00491721234567).

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT04.00** | **04.11.2005** | **Created** |
| **AT2009** | **26.08.2010** | **New compiled (change in table 56786 Phone Index Temp)** |
| **AT2013** | **11.05.2013** | **Form replaced by Page**  **Function renamed UpdatePageAfterFindRecords** |

## Report 56780 – Fill Phone Index

If you start working with activeTAPI-DyNAV, you can fill the Phone Index with the telephone numbers (of customers, vendors, contacts, …) , that are already stored in Navision.

Called by “Reports” (if registered as report) or by menu " Telephony (CTI)", "Periodic activities", "Fill Phone Index".

Results of the report can be shown and modified:

* Telephone Index: Menu "Telephony (CTI)", "Periodic activities", "Update Phone Index"
* Error messages from the production of Phone Index: Menu "Telephony", "Periodic activities", "Notes for Phone Index"

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT01.20** | **25.09.2003** | **New (second) parameter for Function ConvToPlain** |
| **AT01.20a** | **13.03.2008** | **New Connection with Automation objects** |
| **AT2009** | **06.11.2009** | **Check double Numbers options; Fill new field "Blocked"** |
| **AT2011** | **05.04.2011** | **Option Format existing numbers Phone / Fax Check Min. length for Phoneindex** |
| **AT2013** | **11.05.2013** | **Adapted to 2013** |
| **AT2015** | **06.11.2014** | **Adapted to 2015, Request Page added** |

## Report 56781 – Update TAPI Lines

As it is not possible to program actions directly in the Menu Suite, this report „Update TAPI Lines“, which was performed directly in the menu „TAPI Setup, had to be transferred to the Codeunit 56780 TAPI Functions. As it is also not possible to call a subfunction of a Codeunit out of the Menu Suite, the call of the subfunction is performed in this report (Processing only).

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT04.00** |  | **New report to be used in the Menu Suite** |
| **AT2013** | **11.05.2013** | **To be used in Departments**  **Caption added, Adapted to 2013** |

## Report 56782 – Delete TAPI Logfile

After a time the TAPI Logfile becomes greater and greater and greater. With this report you can delete old entries.

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT2011** | **05.04.2011** | **Created** |
| **AT2013** | **11.05.2013** | **Adapted to 2013** |

## Codeunit 56780 - TAPI Functions

Function collection for the Telephony:

The Phone-Index functions are called from the OnValidate-Trigger of a telephone number. If there are several telephone numbers in a table (e.g. Contact), it is checked whether the telephone numbers are the same. If the Telephone number does already exist in another address of the same table, a warning is shown.

Example: With a (new) telephone number in a contact it is checked whether there is already another contact with this telephone number. It is not checked whether there exists a customer with this telephone number, because there is always a customer with this telephone number, if Business Relationship Management integration is activated.

### PhoneIndexCust

Provides the Phone Index for a customer

### PhoneIndexVend

Provides the Phone Index for a vendor

### PhoneIndexContAtt

Provides the Phone Index for a contact

Note: For each telephone number of each address a record is written to the Phone Index. Example: With a contact "Phone No." and "Mobile Phone No." are entered. Then 2 Phone Index entries are provided:

|  |  |  |
| --- | --- | --- |
| **Table of origin** | **5050** | **5050** |
| **Number of origin** | **No. (contact)** | **No. (contact)** |
| **Number 2 of origin** | **empty** | **empty** |
| **Type of origin** | **0 (business)** | **2 (mobile)** |
| **Name of origin** | **Name (contact)** | **Name (contact)** |
| **Phone No.** | **Phone No.** | **Mobile Phone No.** |

No. 2 of origin is only necessary for Navision 2.x, because the contact persons are stored in a separate table (5002) with the key "Prospect No.” (= No. of origin) and “Nr”. (= No. 2 of origin).

With “Type of origin” the following options are used:

0 business / dienstlich

1 private / privat

2 mobile / mobil

3 second / zweite

4 extension / Durchwahl

### ConvToPlain

Formatting function for telephone number; Input: telephone number, Output: Telephone number in the plain format

### Dial

Dialing function; called from the telephone Bitmap of every telephone number; the key fields are handed over, to identify the data record and the Phone No. clearly (Table No., No., may be No. 2, Type of origin)

### PhoneNoFindAgain

This function is called, if after an incoming call, with which the telephone number was not found, the button “Phone No. find again” in the Page 56786 "Menu Phone No. not found" is pressed. This makes however sense only if the caller was inserted as customer or contact.

### TAPIRights

### TAPILineReady

### TAPINotReadyMessage

With these functions it is examined whether - and if which - rights the user has for the Telephony. If not, an error message is spent. Thus it is prevented that automation functions are called, although these are not installed with the user.

### DeletePhoneIndex

Is called from the OnDelete trigger of each table with telephone numbers, in order to remove the telephone numbers from the Phone Index.

### ConvToNumber

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |
| **AT01.20** | **08.09.2003** | **Bug Fix: Return value of Function "TAPILineReady" changed from text 30 to text 50 (possibility of Field Overflow)**  **Integration of new field in TAPI Setup: "Min. length for PhoneIndex"**  **New Function ConvertToNumber : Gives back input number without any special signs like./ - etc.**  **New Function ConvertToCanonic: Gives back canonic f formatted input number; can be used for Customer/Vendor/Contact Phone Numbers, if "TAPI Setup"."Format Telephone number" is set to Yes.**  **New Function EndOfTAPI for closing actions like closing Dialer** |
| **AT04.00** | **04.11.2005** | **New Function Update Lines**  **New Function Dial No**  **Error Correction in DeletePhoneIndex: MODIFY after DELETEALL deleted**  **Error Correction in EndOfTapi: Error in changing Company was fixed.** |
| **AT04.00.02** | **15.09.2006** | **Error Correction: 1) In Functions ConvToCanonic, ConvToPlain, ConvToNumber: Add ATTelephony-Object and initialize it. 2) In Functions for PhoneIndex: Even if a user is not a TAPI User, the input/changes in Phone-No-Fields should be stored in the Phone Index.** |
| **AT05.00** | **16.06.2007** | **Length of all name fields changed from 30 to 50** |
| **AT05.00.01** | **13.03.2008** | **New connection with automation objects** |
| **AT2009** | **06.11.2009** | **ConvToCanonic transferred to SI-CU 56782 New Global Var TAPIFuSI Codeunit 56782 Field "Blocked" in Phone Index (Customer / Vendor) Check for double numbers defined in Setup Option: Create Interaction Log Entry** |
| **AT2011** | **05.04.2011** | **Changes for RTC: CREATE, ISSERVICETIER (Dialer) Old Code deleted, AppTrace Transfer from TAPI Main: Handling of Events BeginOfTAPI - called from CU 1** |
|  | **08.04.2011** | **Automation new defined for Handle functions** |
| **AT2011.3** | **11.10.2011** | **In TAPILineReady added \_\_ to the TAPI-Drivername Function EndOfTAPI transferred to the end of the function list** |
|  | **06.02.2015** | **UpdateLines changed** |

### Update Lines

This function was called directly from the Menu TAPI Setup. It was transferred to the TAPI Functions and is called out of the MenuSuite with Report 56781 Update TAPI Lines.

### DialNo

This function directly calls a telephone number. It is used in “Try again” in the Page 56789 End of Call out.

### CreateInteractionLogEntry

Is used, if field *Create Interaction Log Entry* is activated in TAPI Setup.

### HandleCallNew HandleCallFinished HandleLineMessage HandleCallUserInfo

These are the new functions for the event handling, calls from CU *TAPI Main* for Classic Client and from Page *Page with Timer* in RT-Client.

### BeginOfTAPI

Starts the Dialer, if wanted.

### EndOfTAPI

Ends the dialer, if started, and clears Telephony Object.

Error correction: If you are working with 2 (or more) Companies in Navision together with Telephony, you sometimes had some error messages while changing company.This was corrected.

## Codeunit 56781 - TAPI Event Publisher

In this Codeunit, you find all the event publishers for activeTAPI (new in v2017)

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |

|  |  |  |
| --- | --- | --- |
| **AT2017** | **02.11.2016** | **Event Publishers for activeTAPI** |

## Codeunit 56782 – TAPI Functions SI

This is a separate Single Instant Codeunit for formatting of Phone- and Fax No.’s.

It was transferred from TAPI Functions to here.

The reason is, that this solution runs quicker and using this Codeunit does not consume an activeTAPI-user-license.

So you can install activeTAPI for ALL NAV-Users, that they are able to format numbers, even if they are no active-TAPI-Users.

The new variable ATPhoneRunning was introduced, to avaoid Crash of the RT-Client, if activeTAPI is not installed (thanks to Heiko).

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |

|  |  |  |
| --- | --- | --- |
| **AT2009** | **06.11.2009** | **ConvToCanonic transferred from CU 56780** |
| **AT2011** | **05.04.2011** | **Changes for RTC: CREATE** |
| **AT2015** | **11.02.2015** | **New Var ATPhoneRunning to avoid RT-Client-Crash, if activeTAPI is not installed (thanks to Heiko)** |

## Codeunit 56783 – TAPI Event Handler

This Codeunit is totally new. It contains the event handlers – most of the code, that was part of NAV standard objects. The name describes, what the event handler does.

For example: DeletePhoneIndexOnCustomerDelete: This event handler deletes the Phone Index of a customer, if the customer is deleted.

**The list of ebent handlers:**

DeletePhoneIndexOnCustomerDelete

PhoneIndexCustOnCustomerValidateName

FormatPhoneNoOnCustomerValidatePhoneNo

PhoneIndexCustOnCustomerValidateBlocked

FormatFaxNoOnCustomerValidateFaxNo

SetUpTAPIOnLogInStart

DeletePhoneIndexOnVendorDelete

PhoneIndexCustOnVendorValidateName

FormatPhoneNoOnVendorValidatePhoneNo

PhoneIndexCustOnVendorValidateBlocked

FormatFaxNoOnVendorValidateFaxNo

DeletePhoneIndexOnContactDelete

PhoneIndexContOnContactValidateName

FormatPhoneNoOnContactValidatePhoneNo

FormatFaxNoOnContactValidateFaxNo

FormatPhoneNoOnContactValidateMobilePhoneNo

DialCustomerPhoneNoOnCustomerPages

DialVendorPhoneNoOnVendorPages

DialContactPhoneNoOnContactPages

DialContactMobilePhoneNoOnContactPages

UpdateCustomerFromContact

UpdateVendorFromContact

UpdateContactFromCustomer

UpdateContactFromVendor

UpdateContact

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |

|  |  |  |
| --- | --- | --- |
| **AT2017** | **02.11.2016** | **Codeunit for TAPI Event Handler** |

## MenuSuite 1054 Dept – Add-on 4

The Menu-Suite includes the new Menu „Telephony (CTI)“ / „Telephony“. The number was changed from 1055 to 1054, because 1055 is used for OPPlus. If you need another object number, then use the TXT-Objekt, to change the number.

|  |  |  |
| --- | --- | --- |
| **Documentation  Trigger** | **Date** | **Note / Changes** |

|  |  |  |
| --- | --- | --- |
| **AT2017** | **15.01.2017** | **Transferred from No. 1055 to 1054** |

# Upgrade information

The activeTAPI-Runtime components of activeTAPI.COM-DyNav\_v2017 were changed and extended. Therefore you must deinstall the old version **in each case** on all PC's concerned (Start, Settings, Control panel, Software, activeTAPI, Deinstall) and install the new version (SETUP). The new version in the control panel/software is called: activeTAPI.COM-DyNAV v2017.

For the new version activeTAPI.COM-DyNav\_v2017 Navision 2017 was used with RTM version (Build 13682).

The changes, extensions and error corrections are described in the chapter “Introduction” and in the documentation of the single objects.

As we started with events in v2017, an upgrade from or a downgrade to former versionS is not possible.

If you want to install ActiveTAPI together with older versions of NAV, please contact us.

# Important notes at the end

Naturally you can correct errors, that may be in the code of the activeTAPI-DyNAV objects, yourself directly.

However, we will appreciate if you’d send us a short description, so that we can also fix this error generally. Otherwise every developer has to fix this error himself. This is an unnecessary waste of time and waste of resources, in our opinion. Please help us to make to product better.

## Notes for developers - FAQ

What do you have to do, if …

**In a table/a page (e.g. customer) a new field "Mobile Phone No." shell be inserted. It should be available for activeTAPI-DyNAV.**

1. In the Codeunit 56781 TAPI Event Publishers create a Integration Event   
    OnCustomerMobilePhoneNoChanged(Cust : Record Customer).   
   The easiest way is to copy (and modify) the existing function   
    OnCustomerPhoneNoChanged(Cust : Record Customer).

In the Codeunit 56781 TAPI Event Publishers create a Integration Event Event   
 OnCustomerPagesDialMobilePhoneNo(Cust : Record Customer)   
The easiest way is to copy (and modify) the existing function   
 OnCustomerPagesDialPhoneNo(Cust : Record Customer).

1. Enter the new field into the Table 18 "Customer" (e.g. Field 50000).
2. n the OnValidate trigger of the new field write:  
   ATEvPub.OnCustomerPhoneNoChanged(Rec);  
   GET(“No.”);  
   (the global variable ATEvPub already exists).
3. Add the field into the Page 21 "Customer Card". In the OnAssistEdit-trigger input the code  
   ATEvPub.OnCustomerPagesDialMobilePhoneNo(Rec);
4. In the Codeunit 56783 TAPI Event Handler create 2 new functions:   
    FormatMobilePhoneNoOnCustomerValidateMobilePhoneNo   
   (Copy and modify FormatPhoneNoOnCustomerValidatePhoneNo)  
    DialCustomerMobilePhoneNoOnCustomerPages  
   (Copy and modify DialCustomerPhoneNoOnCustomerPages)
5. Test it: Input a value into the new field "Mobile Phone No.". Click on the three points (OnAssistEdit) of the new field.

**If your page is not a Card, but a List (f.e. Customer List)**

1. Add a Page Action, f.e. Dial Mobile
2. In the OnAction-Trigger input the Code:   
   ATEvPub.OnCustomerPagesDialMobilePhoneNo(Rec);   
   Global Var ATEvPub does already exist.
3. Test it: Input a value into the new field "Mobile Phone No.". Click on the new Action Button.

**A Table, which is so far not yet connected with activeTAPI-DyNAV, should be integrated (example: Employee).**

1. In the Codeunit 56780 TAPI Functions create a new procedure, e.g.:  
   PhoneIndexEmp.   
   As parameters you enter: Emp, record, 5200
2. Copy the contents of the procedure "PhoneIndexCust" into the new procedure and change every "Cust" against "Emp".  
   In the lines " TelIndex.SETRANGE("Table of origin", 18);" change 18 (Customer) against 5200 (Employee).  
   When inserting into the Phone Index (at the end of the procedure) leave the line:  
   ” TelIndex."Type of origin" := TelIndex."Type of origin"::business” unchanged;  
   Business means Office (option 0) or change the option to private (1 = private).
3. Add the new table to the functions “Dial” and “PhoneNoFindAgain”.
4. In the codeunit 56781 copy an modify the IntegrationEvents to employee:  
   OnCustomerNameChanged(Cust : Record Customer)  
   OnCustomerPhoneNoChanged(Cust : Record Customer)
5. In the codeunit 56783 copy and modify the EventSubscriber functions to employee:  
   PhoneIndexCustOnCustomerValidateName(Cust : Record Customer)  
   DialCustomerPhoneNoOnCustomerPages(Cust : Record Customer)
6. In the table 5200 Employee create a new global variable:   
   ATEvPub, Codeunit, 56781, TAPI Event Publisher
7. In the OnValidate-Trigger of the field 13 "Phone No." you enter:  
   ATEvPub.OnEmployeePhoneNoChanged(Rec);   
   GET("No.")   
   The other changes (Field Name, OnDeleteTrigger can be copied from the  
   customer table and modofied.
8. Copy the Code from the OnAssitEdit-trigger from the Page 21 "Customer Card" (behind "Phone No.”) into the Page 5200 “Employee Card” behind the field "Phone No.":   
   ATEvPub.OnCustomerPagesDialPhoneNo(Rec);   
   Instead of x you enter the number, which was used above under b).
9. Test it: For an employee enter a valid “Phone No.” and click on the three points (OnAssistEdit) of the new field.
10. Add the new table 5200 in the Trigger “DefVar” of the following Pages:   
     - 56780 Menu Phone No. Found  
     - 56781 Incoming call  
     - 56788 End to of call in  
     - 56789 End to of call out...   
      
    IF Found THEN BEGIN  
     CASE OriginTab OF  
    ...  
     //Employee / Mitarbeiter  
     5200:  
     BEGIN  
     Origin:= Text009;  
     OriginNr:= DELCHR(OriginKey1, '<>');  
     END;  
    and enter a new text constant (in the example Text009) like DEU=Mitarbeiter;ENU=Employee
11. In the Page 56780 "Menu Phone No. Found " add a new Page Action “Show Employee (Card)” (or copy and change the Page Action "Show Customer (Card)”).  
    Enter a new global variable "Emp, Record, 5200".  
    In the OnPush-Trigger of the button you enter:  
     Emp.RESET;  
     IF OriginTab = 5200 THEN  
     IF Emp.GET(OriginKey1) THEN;  
    and do the same in the corresponding page.  
     PAGE.RUN(5200, Emp, Emp.Name);
12. May be you want to expand Report 56780 “Fill Phone Index”.

**As action for incoming calls you have chosen: "Show caller for short time”. You want to change the time how long this Page is shown (default is 2 sec).**

The menu uses PAGE 56781 "Incoming call". In the OnOpenPage-in the variable “RTC-Timer”. Actual value is 20 (unit 1/10 sec = 2 sec.). Little trick: As the RT-Client is not able to close a page by command, we added the timer-AddIn (see also Page 56785) to the page and use the Windows Scripting Host (WSH): In OnInit a WSHShell (automation) is created and in the timer-OnControlAddIn we send ESC-Key to close the page.

# Contact Information

For all questions according to the **Navision objects** contact

EDV-Beratung (Mr.) **Manfred Gertz** \* Isarstrasse 41 \* D-40699 Erkrath

Fon: 02104-808383 \* Fax: 02014-808385 \* E-Mail: [mg@gertz-edv.de](mailto:mg@gertz-edv.de)

For all questions according to the **activeTAPI-Runtime and/or Telephony**:

Mr. **Markus Schmidt** \* Wiesentalstrasse 10a \* D-65207 Wiesbaden

Fon: 0611-9545577 \* Fax: 0611-9545578 \* E-Mail: [msc@activetapi.net](mailto:msc@activetapi.net)

1. Please ask your hardware vendor/manufacturer/supplier, if your Telephone hardware is TAPI-compliant and if you’re able to use TAPI in your LAN. Also, your hardware vendor will tell you, how to activate and how to configure TAPI. [↑](#footnote-ref-1)
2. If you’re planning to enhance/fix/modify the delivered Navision Objects, we suggest you to have the Windows SDK Documentation available to have access to the Windows TAPI Documentation. This will help you to better understand how TAPI works etc. [↑](#footnote-ref-2)