

KIX Professional InventorySync Backend "opsi" Installation and configuration

c.a.p.e. IT® GmbH

Version: 17.0.0

Imprint

Title:	KIX Professional InventorySync Backend "opsi"
Topic:	Installation and configuration
Author:	c.a.p.e. IT® GmbH
Keywords:	KIX, Inventory, opsi, KIX Professional
Comments:	
Filename:	InventorySyncOpsiBackend_installation_and_configuration_en.odt
Version:	17.0.0
Set and Layout:	LibreOffice.org Version 5.2.x

Table of contents

1 General advices.....	4
1.1 Document information.....	4
1.1.1 Objectives.....	4
1.2 Changes.....	4
1.2.1 Change history.....	4
1.2.2 Authors.....	4
1.2.3 Document owner.....	4
1.2.4 Authorized staff.....	4
1.3 List of abbreviations.....	4
2 Installation.....	5
2.1 Requirements.....	5
2.2 Package Installation.....	5
3 Configuration.....	6
3.1 SysConfig Options.....	6
3.1.1 Sources.....	6
3.1.2 Source-Backend-Mapping.....	6
3.1.3 Source Parameters.....	6
3.1.4 SSL verification.....	7
3.2 InventoryContentXPath for backend "opsi".....	7
4 Appendix.....	8
4.1 Example CI class "Computer" for backend "opsi".....	8

1 General advices

1.1 Document information

1.1.1 Objectives

This document contains all the information to install and configure the KIX Professional InventorySync backend "opsi".

1.2 Changes

1.2.1 Change history

Version	Date	Changed chapters	Short description	Changed by
1.0.0	29.08.11	all	document creation	Rene Boehm
1.0.1	16.01.13	all	Completely reworked	Rene Boehm
1.0.2	04.07.13	Changes	Format fixes	Rene Boehm
1.0.3	30.10.13	1.1.1	Minor changes	Rene Boehm
1.0.4	21.05.15	2.1	Upgrade for OTRS 4.0	Ricky Kaiser
1.0.5	27.01.16	2.1 / 3.1.4	Upgrade for OTRS 5.0	Rene Boehm
17.0.0	05.02.17	All	Updated for KIX Professional 17.0.0	Rene Boehm

1.2.2 Authors

Surname, first name	Organisation	Function
Boehm, Rene	c.a.p.e. IT® GmbH	Head of Development

1.2.3 Document owner

Organisation	Surname, first name	Address	Contacts
c.a.p.e. IT® GmbH		Schoenherrstr. 8 09113 Chemnitz	info@cape-it.de +49 371 27095 - 620

1.2.4 Authorized staff

Surname, first name	Organisation	Function	Auth.
Boehm, Rene	c.a.p.e. IT® GmbH	Head of Development	rw
Boehm, Ralf	c.a.p.e. IT® GmbH	Developer	rw
Kaiser, Ricky	c.a.p.e. IT® GmbH	Developer	rw

1.3 List of abbreviations

CI	Config Item
CLI	Command Line Interface
GUI	Graphical User Interface

2 Installation

2.1 Requirements

To install and use this data backend for the KIX Professional function "InventorySync" the following is required:

- a KIX Professional 17.0.x installation
- a opsi server

2.2 Package Installation

Just install the „opsi“ backend module via your operating systems package management. The package name is „kixpro-inventorysync-opsi“. Please make sure you have fulfilled the necessary requirements (see 2.1).

3 Configuration

3.1 SysConfig Options

After you have installed all the needed packages, you have to configure all the parameters necessary for this backend to communicate with one or more opsi servers.

To configure the extension just open the SysConfig in the admin area. Select the SysConfig group "KIX Professional". After the page reload select the subgroup "ITSMConfigItem::InventorySync".

An example configuration is already included in the package. You should change it to your needs.

The following section lists all the important options needed to get the InventorySync function working.

3.1.1 Sources

SysConfig Option: ITSMConfigItem::InventorySync###Sources

A source represents a named communication point. A SysConfig hash option is used for its configuration. The hash-key is the internal name of the source, i.e. "opsisrv1". The hash-value is the display name of this source, i.e. "opsi server 1".

You can add more sources for each external inventory server.

3.1.2 Source-Backend-Mapping

SysConfig Option: ITSMConfigItem::InventorySync###Backend

The system needs to know which backend should be used for a source to communicate with the external inventory tool. The hash-key is the Source's identifier (i.e. "opsisrv1" in our example) and the hash-value has to be "opsi" to tell InventorySync to use the opsi backend for this data source.

3.1.3 Source Parameters

SysConfig Option: ITSMConfigItem::InventorySync###Parameters

All the parameters have to be entered in the following form:

<parameter>=<value>, <parameter>=<value>, ...

You have to configure the following parameters to use the "opsi" backend:

Parameter	Required	Description
URL	X	the URL to the rpc interface of the opsi server (usually http://<host>:4447/rpc)
User	X	username to authenticate to the opsi server
Password	X	password to authenticate to the opsi server

3.1.4 SSL verification

SysConfig Option: InventorySync::Backend::opsi###UserAgentVerifyHostname

This option can be found in the subgroup "ITSMConfigItem::InventorySync::opsi" and allows you to disable the SSL certificate validation if necessary

3.2 InventoryContentXPath for backend "opsi"

The basic extension of CI classes and the usage of the InventoryContentXPath parameter are described in the documentation of the KIX Professional function „InventorySync“.

The structure and contents of the inventory data (hardware and software) returned by opsi is defined on the opsi server in the files under /etc/opsi/hwaudit/.

The Xpath for opsi has the following structure:

Hardware Xpath:

HW/<opsi device class>/<attribute>

Software Xpath:

SW/<software registration key pattern>/<attribute>

Typical Xpaths for opsi may be:

HW/PROCESSOR/name
to get the name attribute of each processor

HW/HARDDISK_DRIVE/model
to get the model attribute of each harddisk drive

SW/*.*/displayVersion
to get the name of the software, regardless of its Windows registration key

SW/netmeeting/displayVersion
to always get the name of the software registered in Windows with the key „netmeeting“

Please contact your opsi consultant to clarify this if you can't figure out how the opsi Xpath works.

4 Appendix

4.1 Example CI class “Computer” for backend “opsi”

The following example shows a CI class definition for class “Computer” with a couple of changes for the opsi backend. Please note that these are no mandatory changes! You are absolutely free to make you own changes. You have just to make sure that mandatory CI attributes will get some value during the sync process. Otherwise KIX Professional will show an error message.

```
[
  {
    Key => 'Vendor',
    Name => 'Vendor',
    Searchable => 1,
    Input => {
      Type => 'Text',
      Size => 50,
      MaxLength => 50,
    },
  },
  {
    Key => 'Model',
    Name => 'Model',
    InventoryContentXPath => 'HW/COMPUTER_SYSTEM/model',
    Searchable => 1,
    Input => {
      Type => 'Text',
      Size => 50,
      MaxLength => 50,
    },
  },
  {
    Key => 'Description',
    Name => 'Description',
    Searchable => 1,
    Input => {
      Type => 'TextArea',
    },
  },
  {
    Key => 'Type',
    Name => 'Type',
    Searchable => 1,
    Input => {
      Type => 'GeneralCatalog',
      Class => 'ITSM::ConfigItem::Computer::Type',
      Translation => 1,
    },
  },
  {
    Key => 'Owner',
    Name => 'Owner',
    Searchable => 1,
    Input => {
      Type => 'Customer',
    },
  },
  {
    Key => 'SerialNumber',
    Name => 'Serial Number',
    InventoryContentXPath => 'HW/BASE_BOARD/serialnumber',
    Searchable => 1,
    Input => {
      Type => 'Text',
    },
  },
]
```



```

        Size => 50,
        MaxLength => 100,
    },
},
{
    Key => 'Software',
    Name => 'Software',
    InventoryContentXPath => 'SW/.*/displayName',
    InventoryContentEvalString => '$Inventory{displayVersion}',
    Input => {
        Type => 'Text',
        Size => 50,
        MaxLength => 100,
    },
    CountMax => 1000,
    Sub => [
        {
            Key => 'Version',
            Name => 'Version',
            InventoryContentXPath => 'SW/.*/displayVersion',
            Input => {
                Type => 'Text',
                Size => 20,
                MaxLength => 20,
            },
        },
    ],
},
},
{
    Key => 'CPU',
    Name => 'CPU',
    InventoryContentXPath => 'HW/PROCESSOR/name',
    Input => {
        Type => 'Text',
        Size => 50,
        MaxLength => 100,
    },
    CountMax => 16,
},
},
{
    Key => 'Ram',
    Name => 'Ram',
    InventoryContentXPath => 'HW/MEMORY_MODULE/memorytype',
    Input => {
        Type => 'Text',
        Size => 50,
        MaxLength => 100,
    },
    CountMax => 10,
},
},
{
    Key => 'HardDisk',
    Name => 'Hard Disk',
    InventoryContentXPath => 'HW/HARDDISK_DRIVE/name',
    Input => {
        Type => 'Text',
        Size => 50,
        MaxLength => 100,
    },
    CountMax => 10,
    Sub => [
        {
            Key => 'Capacity',
            Name => 'Capacity',
            InventoryContentXPath => 'HW/HARDDISK_DRIVE/size',
            Input => {
                Type => 'Text',
                Size => 20,
                MaxLength => 10,
            },
        },
    ],
},
},
{
    Key => 'CD-Rom',

```

```

Name => 'CD-Rom',
Searchable => 1,
InventoryContentXPath => 'HW/OPTICAL_DRIVE/name',
Input => {
  Type => 'Text',
  Size => 50,
  MaxLength => 100,
},
},
{
  Key => 'FQDN',
  Name => 'FQDN',
  Searchable => 1,
  InventoryContentXPath => 'HW/COMPUTER_SYSTEM/name',
  Input => {
    Type => 'Text',
    Size => 50,
    MaxLength => 100,
  },
},
{
  Key => 'NIC',
  Name => 'Network Adapter',
  InventoryContentXPath => 'HW/NETWORK_CONTROLLER/name',
  InventoryContentEvalString => '$Inventory[ipaddress]',
  Input => {
    Type => 'Text',
    Size => 50,
    MaxLength => 100,
  },
  CountMin => 0,
  CountMax => 10,
  CountDefault => 1,
  Sub => [
    {
      Key => 'IPoverDHCP',
      Name => 'IP over DHCP',
      Input => {
        Type => 'GeneralCatalog',
        Class => 'ITSM::ConfigItem::YesNo',
        Translation => 1,
      },
    },
    {
      Key => 'IPAddress',
      Name => 'IP Address',
      Searchable => 1,
      InventoryContentXPath => 'HW/NETWORK_CONTROLLER/ipaddress',
      Input => {
        Type => 'Text',
        Size => 40,
        MaxLength => 40,
      },
      CountMin => 0,
      CountMax => 20,
      CountDefault => 0,
    },
  ],
},
{
  Key => 'GraphicAdapter',
  Name => 'Graphic Adapter',
  InventoryContentXPath => 'HW/VIDEO_CONTROLLER/name',
  InventoryContentEvalString => '$Inventory{deviceId}',
  Input => {
    Type => 'Text',
    Size => 50,
    MaxLength => 100,
  },
},
{
  Key => 'OtherEquipment',
  Name => 'Other Equipment',
  Input => {
    Type => 'TextArea',

```

```
    },
    CountMin => 0,
    CountDefault => 0,
  },
  {
    Key => 'WarrantyExpirationDate',
    Name => 'Warranty Expiration Date',
    Searchable => 1,
    Input => {
      Type => 'Date',
    },
  },
  {
    Key => 'InstallDate',
    Name => 'Install Date',
    Searchable => 1,
    Input => {
      Type => 'Date',
    },
    CountMin => 0,
    CountDefault => 0,
  },
  {
    Key => 'Note',
    Name => 'Note',
    Searchable => 1,
    Input => {
      Type => 'TextArea',
    },
    CountMin => 0,
    CountDefault => 0,
  },
];
```