Casebox Documentation

Release 1.9.1

KETSE

December 01, 2015

Contents

Casebox is Content Management System for record, file and task management.

The code is open source, and available on github.

The main documentation is organized into a three sections:

- User Manual
- Administration Manual
- Developer Manual

CHAPTER 1

Introduction

Casebox is an open source Content Management System. You can define data models of different complexity and implement your own workflows (see Developer Docs). Casebox provides a safe and secure solution for all your data on a server that you control (hosted in your own office, a VPS, or even your personal computer).

Casebox is accessed from a web browser. User Experience is similar to a desktop application.

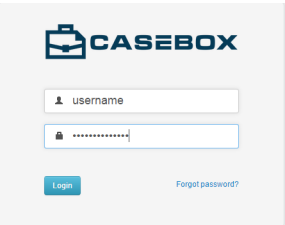
Casebox is your one stop shop for records, tasks and files.

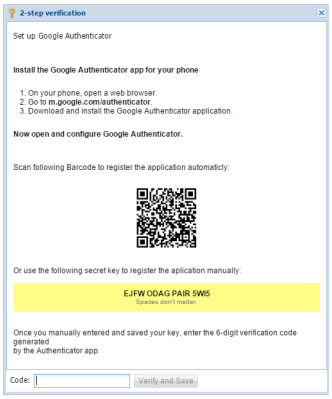
CHAPTER 2

Two Step Verification

Usually called Two Factor Authentication, it is highly recommended to enable it. You'll protect your account with both your password and your phone.

Read Google tutorial about this.





CHAPTER 3

Features

File Management

- File Repository: Ability to maintain files centrally with permission control for users
- File Versioning: Multiple version of the file can be maintained on the server
- File Preview: Known document types can be previewed on Casebox without needing to download the image/document.
- Document Editing: Word documents can be edited remotely using office client with changes updated directly to server

Task Management

- Create/Manage Tasks: Users can create and manage tasks
- Task Assignment: Lets you assign a user of the system to a task
- Customizable Task Details: Lets you define custom fields for a task
- Attach files to the tasks and to the comments

Case Workflow

- Intake Process: The intake form and workflow are customizable
- Caseworker Assignment: Lets you assign a caseworker to a particular client
- Case Notes: Lets you record case notes for a particular client on the client record. These notes can be customized with multiple field types, including drop-down menus and free text fields

Work Management

- Comment/Track Objects or Tasks: Users can comment on any task or object and maintain a discussion history. Users can also respond to comments by email.
- Notifications: Subscribe to any task/object you want to monitor, the system shows the notifications in the interface to items you have subscribed to or have commented on. Additionally notification emails are sent.

Client Data Tracking

- Customizable Demographic Tracking: Tracking client name, phone, address, and whatever fields required to track.
- Files: Casebox allows you to attach one or more files to a client record
- Searching: Lets you search for a client on any user-defined "searchable" fields. Users can define any field in the client record as searchable.
- Export: create a word document of any given list of files or records

Relationships

- Relationship Demographics: Lets you store related objects such as client relatives, witnesses, perpetrators through custom fields.
- Relationship Count: Lets you track how many people are related to the primary object/entity.

Contact Management and Scheduling

- Contact Database: Can be set up to store all contacts related to the organization.
- Manual Communications Log: Allows for logging of communication updates, such as phone calls or personal meetings, using the case notes functionality on a client record.

Customization and Integration

- Custom Fields: Lets you add an unlimited number of custom fields, which can be placed on most screens in the system.

Security and Access Management

- User Permissions: Allows for different permission for folders within the system
- Remote Access: As a web-based system, users can enter data remotely in real time.
- Multi-User Access: Multiple users can view client profiles simultaneously, but changes made to the profile may be overwritten.
- Multilingual Support: The system can display information in multiple languages and supports non-Latin characters and right-to-left languages. Language settings can be applied at both the field and form level.

Reporting and Querying

- Graphs: Building automatic pie charts, bar charts and pivot tables on the fly based on search results.
- Faceted Searches: On the fly filtering of results based on search facets defined by user.

CHAPTER 4

Introduction

Administrator Manual describes what's possible to configure directly from Casebox front-end without the need for server level access or development.

This includes a set of HOW-TOs for simple form creation and administration as well as other configurations that help customize Casebox to your organization specific needs.

In order to configure new objects here are a few quick steps to help you get started.

To start adding a new template to input data records, you need to create a new Form/Template.

Once you've added a template you can configure the template, and add new fields to it.

The fields can be of various types, number, date, or an object you can look up from a thesaurus or from other objects you've created the tree or from a list of user.

You can also set up conditional fields that rely on a value chosen by the user in the form.

Once you have your form set up, you can choose where it appears by configuring your menus. When a form is created it is not added to the menu by default, you must explicitly add it to menu. The form can appear globally, or under a specific folder or template.

You may wish to configure your form so that some fields are easily used for search, this is called faceting. You can define your facets by assigning the field to the SOLR database which is used for indexing your data. This is a simple task that involves adding a SOLR column name to your field. Next you would configure your facets in your casebox configuration, and add that to your filters or smart folders.

Now that you have configured your forms in the database, you may choose to navigate your data using the facets you created by creating smart folders. These folders navigate your data based on search values, think of it as a constant running search.

You may also want to set up search templates if you want to search for specific items that match specific fields in your form.

These are simple guidelines to creating your own forms and administering your Casebox.

CHAPTER 5

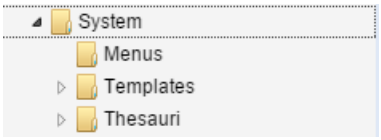
Forms

These are the steps required to be able to use a new form/object in Casebox

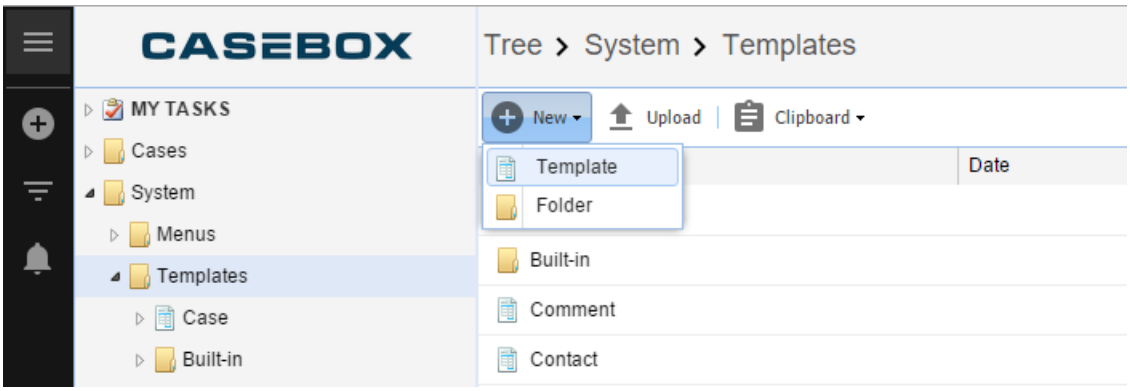
1. Create a new form
2. Add fields to the newly created form
3. Make the form available in menus

5.1 Create New Form

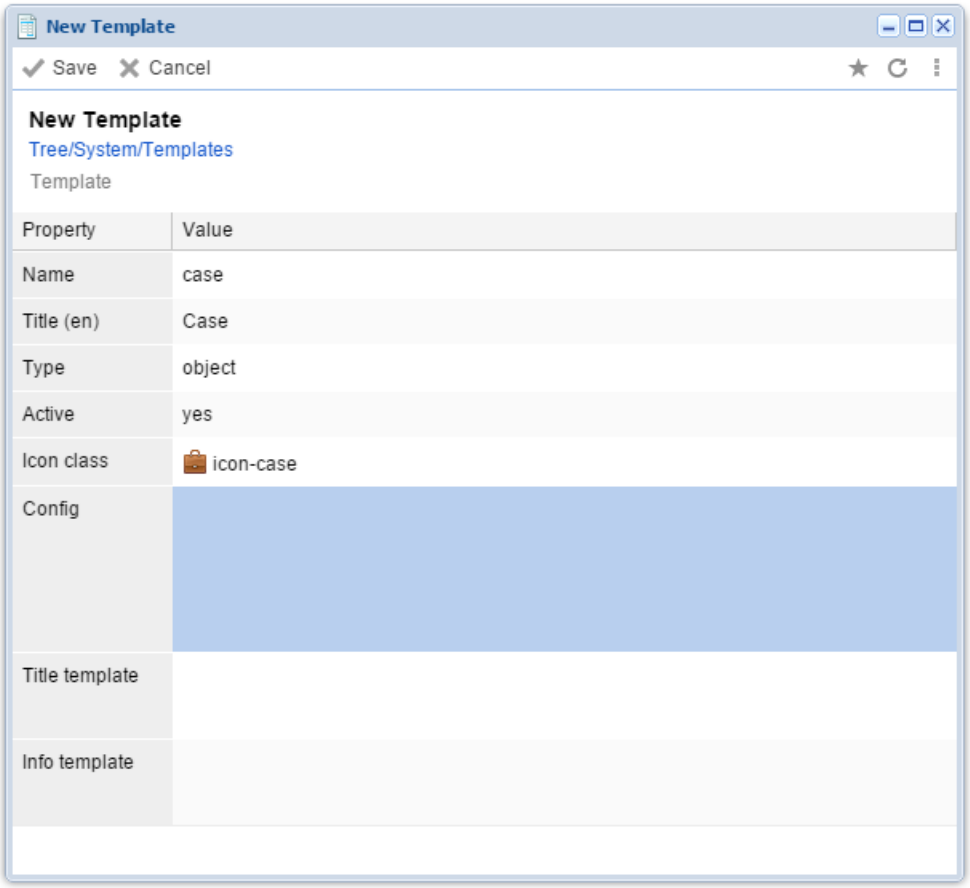
Use the tree window to browse to System/Templates folder



Use the “New” button and choose “Template”

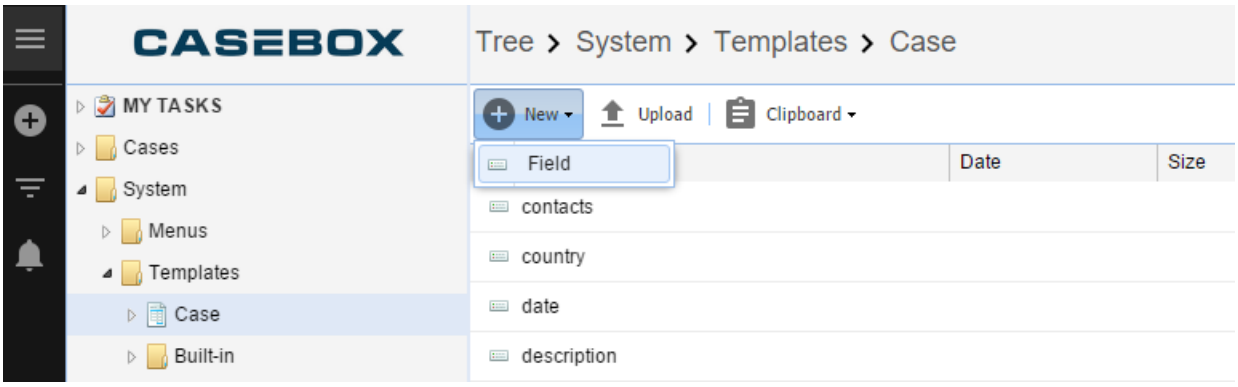


Create form/object name by filling out the form

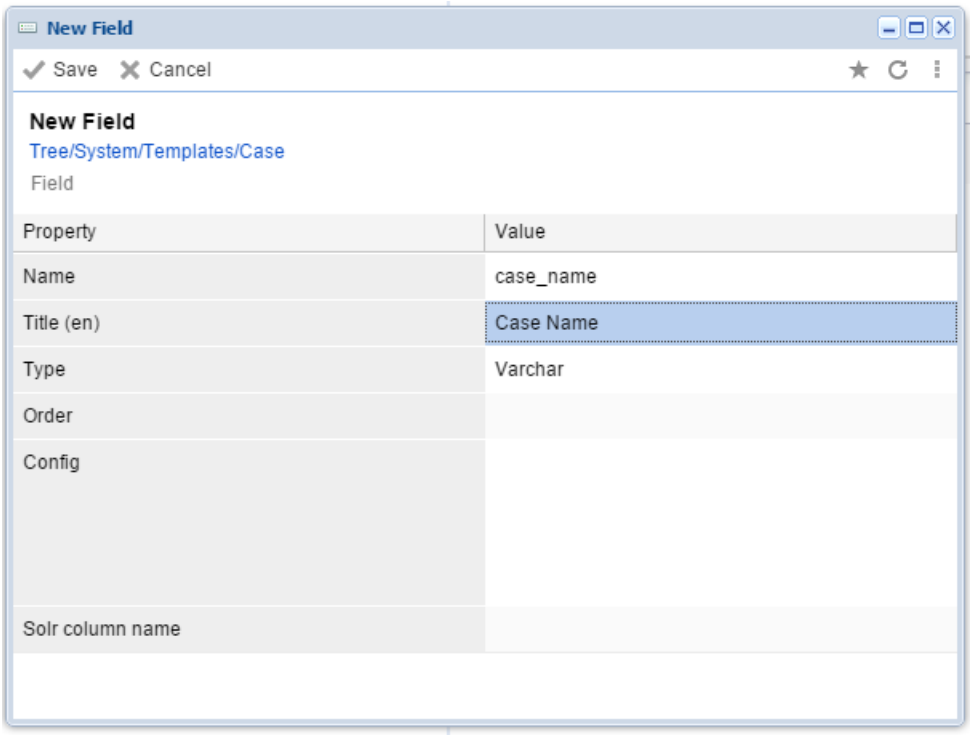


5.2 Add fields to form

Browse inside the newly created template (by double clicking newly created template) and adding different fields through the new "Field" button.



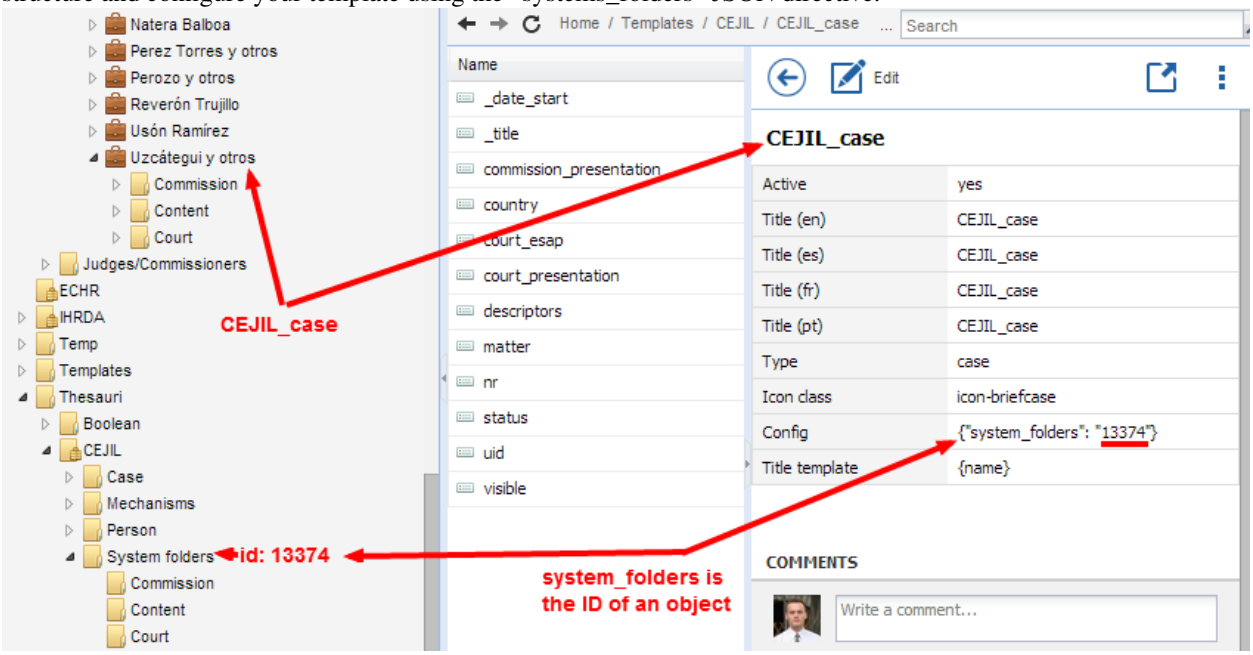
Add a new field



For more information on field creation visit: Fields

5.3 Auto-create a default set of subfolders

If you want to create a set of subfolders with the creation of each new object, all you have to do is create a folder structure and configure your template using the "systems_folders" JSON directive.



5.4 '_title' field name and '_auto_title' field type

Each template should have a field named _title, it tells CaseBox to use the value of the field for display purposes (in the tree, grid). Usually its of varchar type. When the title can be automatically generated out of other fields, then a special _auto_title field type is used.

_auto_title is specified in *templates* table. Examples:

{f34} {f35} {f36} - use field id's from *templates_structure* {template_title}: {where_submitted} {where_submitted_info} - or use field names

5.5 Advanced Template Configuration

For Template configuration

Property	Description	Example
id	a unique identificator of the template	Used in configuring the Menu and the filter in _objects field type.
Name	the name of a template	
$lang	template title in a given language	example 'en'. CB will show the template title if language specified by user
Type	see Template Types	Most of the time you'll manage 'object' templates
Active	yes/no	template can be turned on/off. Notice that you need to specify in which menu the template will appear. See New menu
Icon Class	what icon to use when displaying the object in the grid	Additinal css files can be added to the configuration thus allowing for custom icons.
Title template	The title can be automatically generated from other fields	Product N{nr} from {country}
Config	a JSON cfg	See below

Config a JSON cfg: available object_plugins, grid display columns etc

```
{
    "object_plugins": [
        "objectProperties",
        "thumb",
        "meta",
        "files",
        "contentItems",
        "comments",
        "systemProperties"
    ],
    "layout": [vertical, horizontal],
    "DC": {
        "type": {},
        "order": {}
    },
    "defaultPid": int,
    "leaf": true/false,
    "acceptChildren": true/false
}
```

object_plugins: what plugins are available in the preview panel when the record is selected

layout: how the Preview&Edit Window will display the node, with a right panel with plugins, or the plugins displayed below the vGrid.

DC: Display columns for the Grid when navigating inside the record

defaultPid: if parent node is not specified when a record is created, use defaultPid from template

leaf: double clicking the node in Grid will Edit it instead of opening it as a folder. Default = false

acceptChildren: Allow D&D operations over the node, fileUpload for ex. Default = true

CHAPTER 6

Fields

6.1 Field types

Varchar: simplest one-line text field Date Datetime Time Float Integer Group: it's not a visible field, but a way to group several fields. Header: a Header to visually separate fields Html: HTML editor Text: Text editor (in a popup window) Memo: Inline text editor (multiline text editor inside the grid) IconCombo: used in Templates to select the icon Objects: lookup field with values from the Tree itself

6.2 Populate a field from a Thesaurus/Folder

You can have the field select from a dropdown or pop up window.

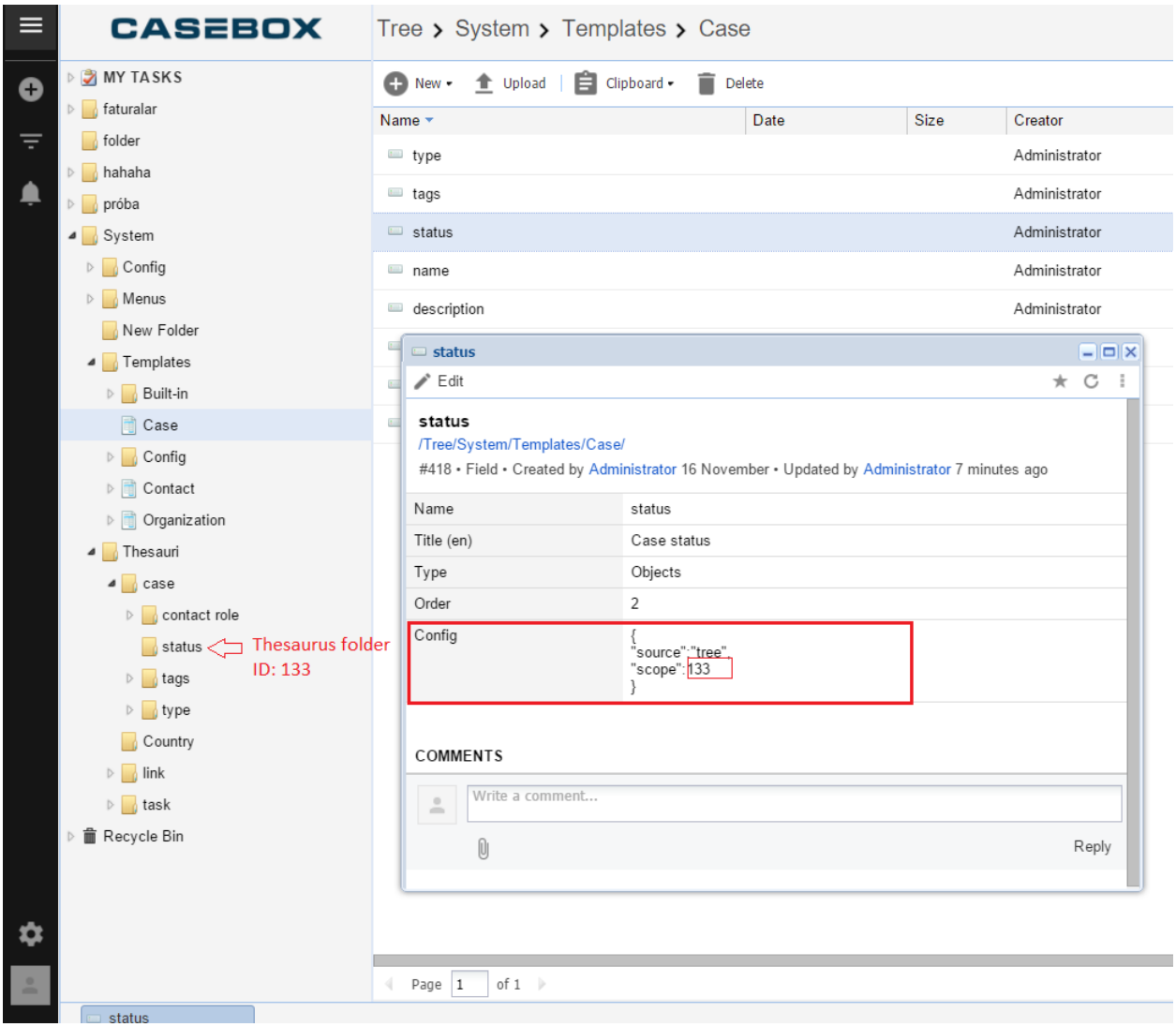
In order to define the field to be populated from a Thesaurus.

When creating the field choose the type as 'Objects'

Configure the field to look up a set of options using the following JSON configuration in the 'Config' part of the field template

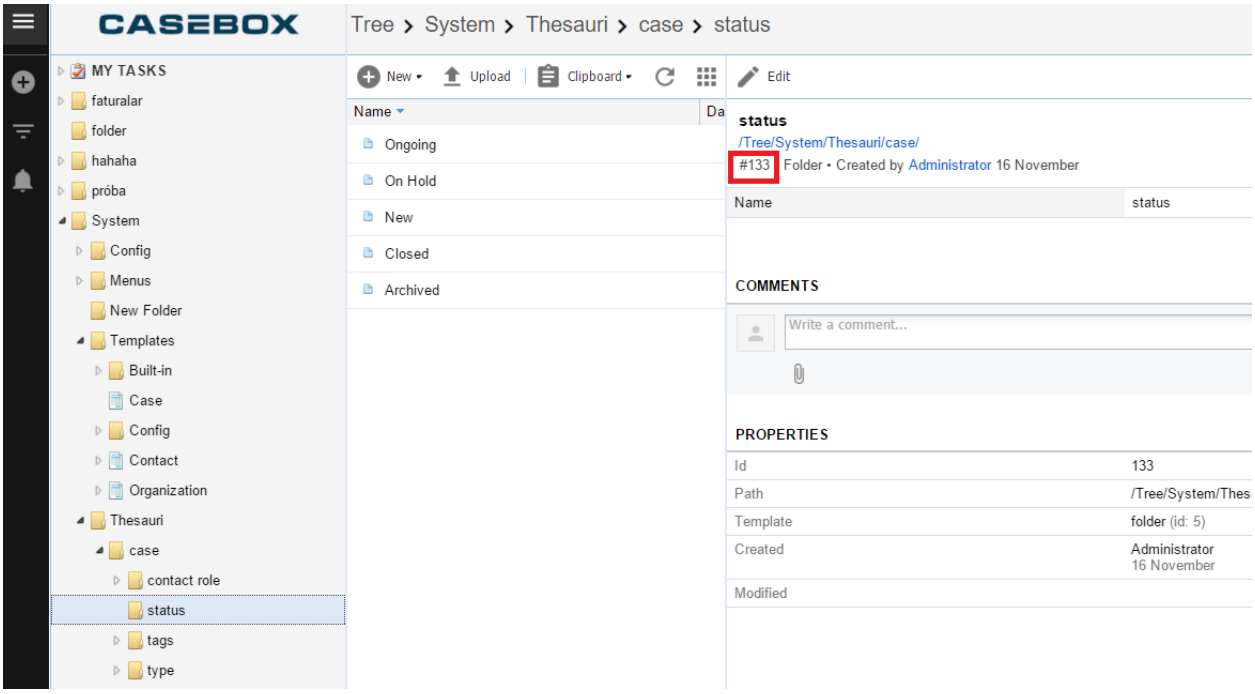
```
{
"source": "tree"
"scope": folder_id
}
```

Example:



Here there is a field called ‘status’ under /System/Templates/Case form. There are a set of statuses defined in a thesaurus that we want available as options for this field. In its config field we define a thesaurus folder as the lookup by “scope” directive and specifying which ID the lookup should be from.

You can get the Id of any folder/object by clicking on the object and reading the ID from the top panel.



You may also choose if you want a default value for a new object using the "value" JSON directive.

6.3 Populate a field from users

You can have the field select from a dropdown or pop up window with values of users.

To choose from a list of users use the "Source" directive with the value of "users".

```
{
"source":"users"
}
```

6.4 Populate a field from a list of specific objects

To choose from a list of users use the "Template" directive with value of the template id.

```
{
"templates":[8178]
}
```

Where 8178 is the template ID

6.5 Create a multi select field

When creating the field pick type object.

You can add the following directive to the JSON configuration *"multiValued":true*

For example this is the config of a field looking up values from a thesaurus indicating it is a multi value field

```
{
        "scope": thesaurus_folder_id
        "multiValued": true,
        "editor": "form",
        "renderer": "listObjIcons"
}
```

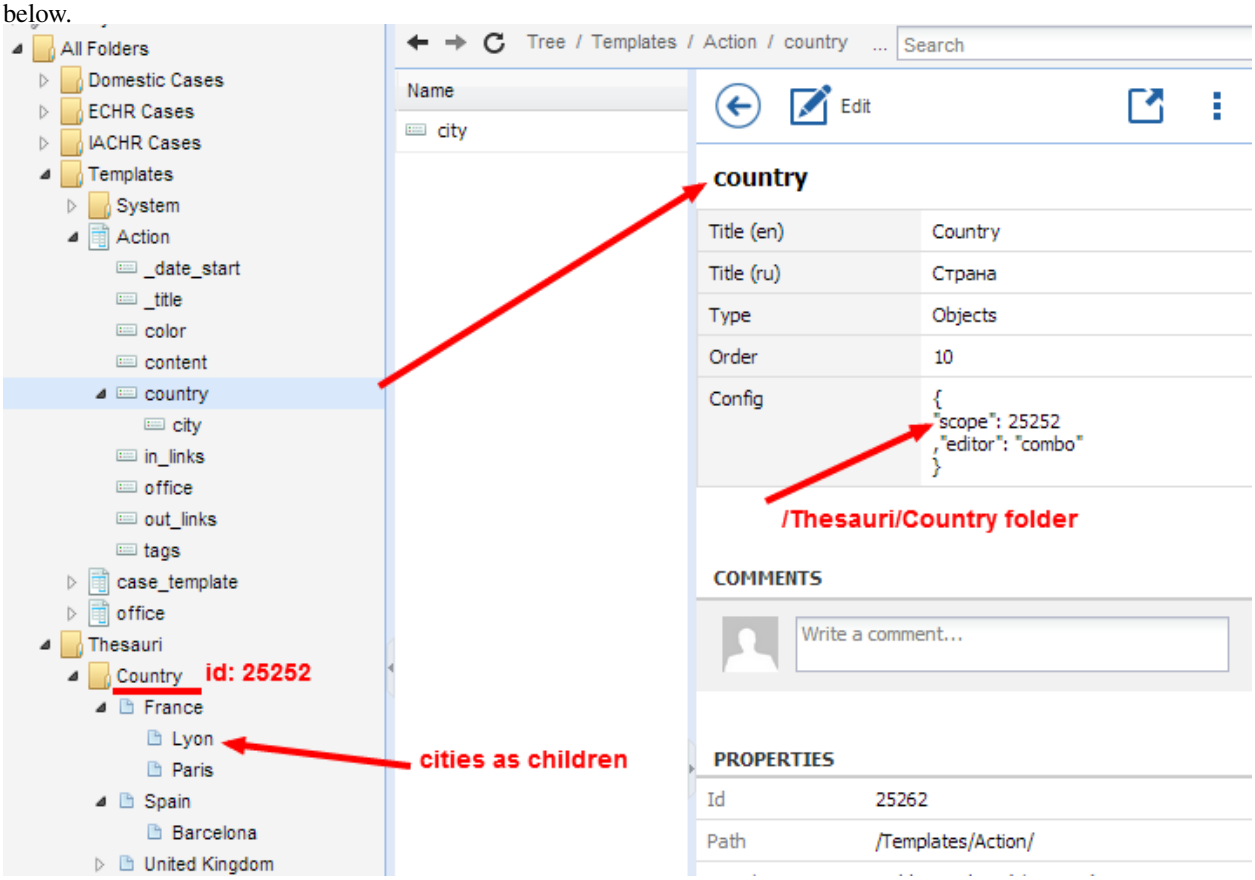
Note: You can get the Id of any folder/object by clicking on the object and reading the ID from the top panel.

6.6 Create a conditional field

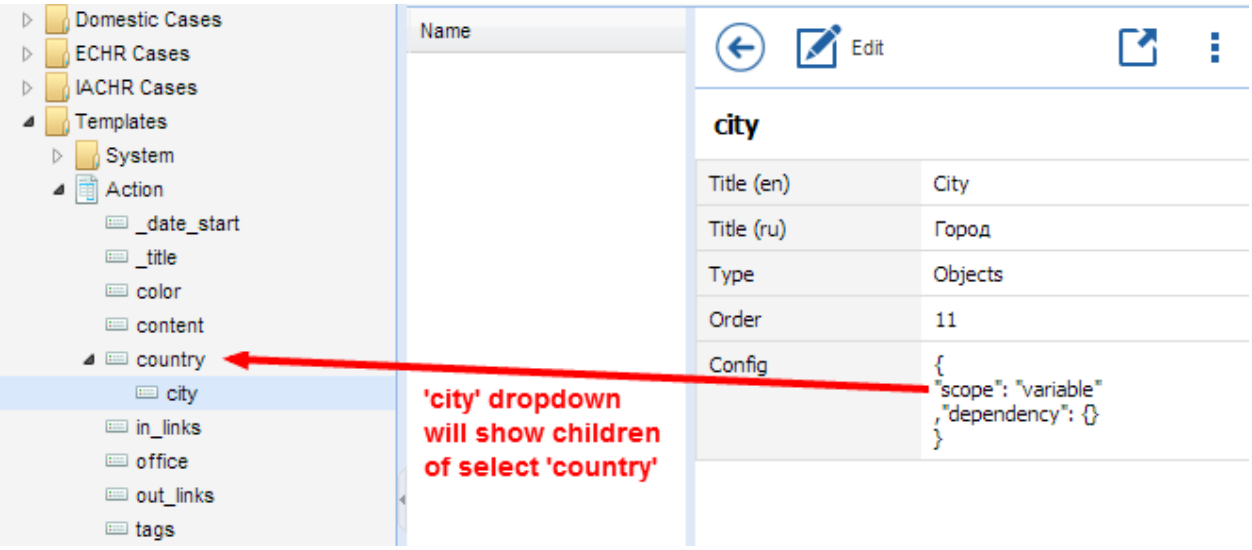
Let's say you want to create a conditional field, for example if you pick a country and you want to automatically find options of cities within a country.

6.6.1 Conditional Field from a Thesaurus

First create a thesaurus with the parent object as shown before, but with cities as children thesaurus items as shown below.



Next create a subfield and configure it so that its scope is 'variable' and add a 'dependency' directive, that way you tell casebox that once the parent country is chosen, the options of cities will appear based on the chosen country.



6.6.2 Conditional Field from Different Objects

You can have conditional fields which appear based on specific values of the parent field. You need to add the 'dependency' directive and indicate what value of the parent field needs to be fulfilled

Example:

```
{
        "dependency": {
                "pidValues" : [391]
        }
}
```

If you want a field to be a drop down or multi-select, add to it the source as shown in instructions above.

Example:

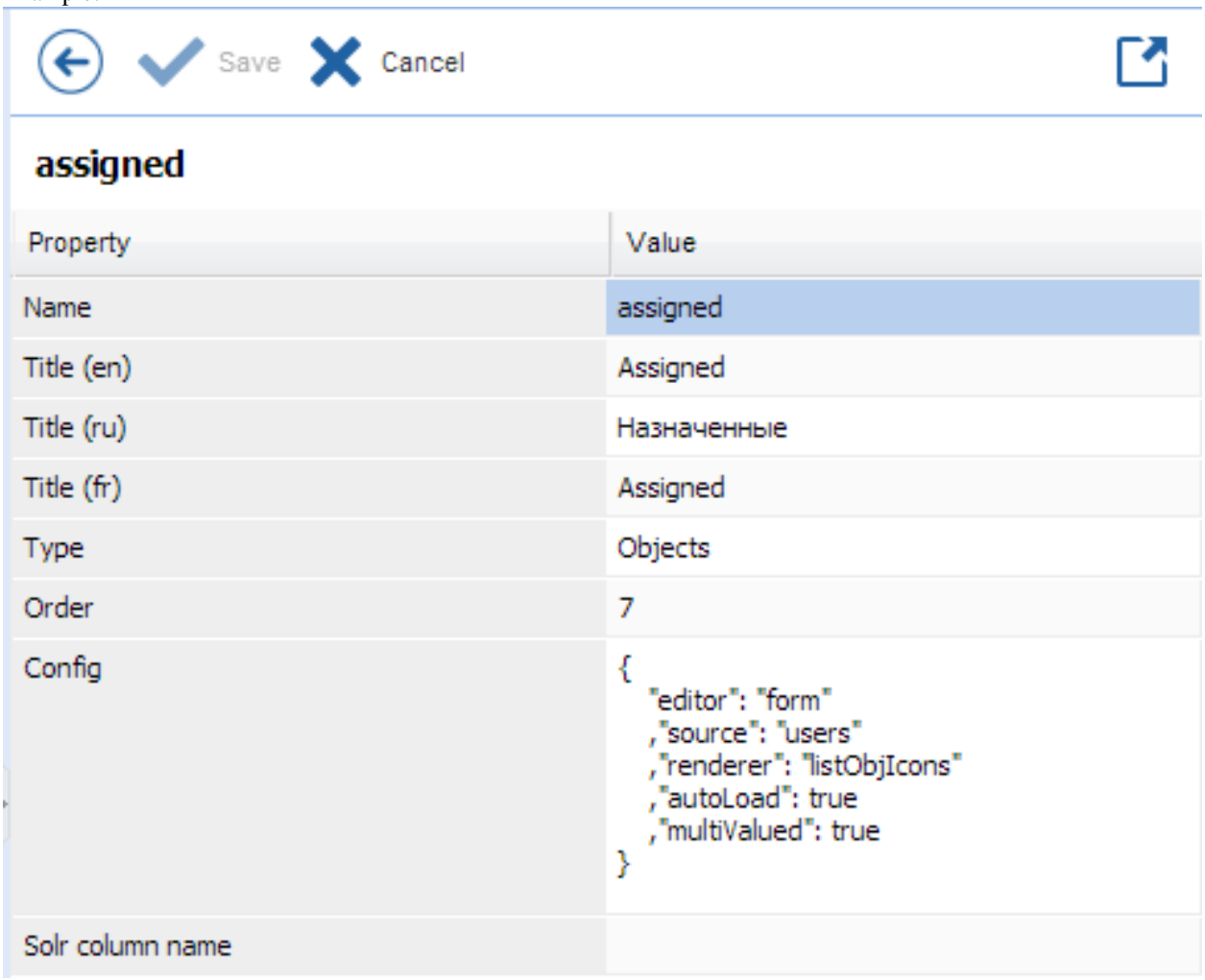
```
{
        "source":"tree",
        "scope": [390],
        "dependency": {
                "pidValues" : [391]
        }
}
```

6.7 Advance Fields Configuration

6.7.1 Fields Template

Name	Param
name	Internal name of field.
[lang]	Title of the field shown in WebClient.
type	Type of field (varchar, date etc, see below)
order	The position of field in the grid
config	Field configuration. see below
solr_column_name	SOLR column to save value of field. See Faceting.

Example:



6.7.2 Field configuration

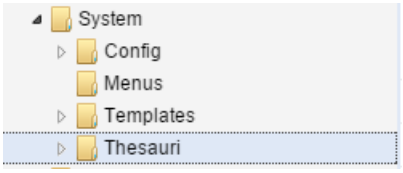
Configuring a field using JSON notation is flexible, as it allows for custom settings based on field type. There might be better UI for managing the configuration of the field in the future.

Here are the options available to all field types:

Option	Value
read-only	true/false. A readonly field (custom code can update this field)
default-Pid	int. Specify in which folder objects should be created no matter what is active folder in Casebox UI. Example: All tasks should be created in /Task DB/
leaf	true/false. Leaf objects doesn't contain sub-objects, i.e. they will not act like folders. When a 'leaf' object is double clicked in the grid, instead of browsing it (i.e. opening it like a folder), the popup window to edit the node will appear
re-quired	true/false. You can't save an object with empty fields marked as required
maxIn-stances	[1..n] how many instances of the field are allowed (by default 1). A multiple field will feature a small [+] icon on the right side that can be clicked to create a new field.
value	a default value for the field when the object is created
depen-dency	a config object that specifies how the current field depends on the parent one. In order to make dependable fields (for example to have two fields, Country/City), you need to explicitly specify a *dependency: {}* config group, even if there are no more dependency conditions
faceting	true/false. If true, CB will save the value of the field in solr_column_name. See faceting

Thesauri

In order to create a new Thesaurus, use the tree window to browse to System/Templates folder



Create a new folder with the name of the Thesaurus you want to create

Browse to the folder

Create new Thesauri item

Repeat until you have added all the items in your thesaurus

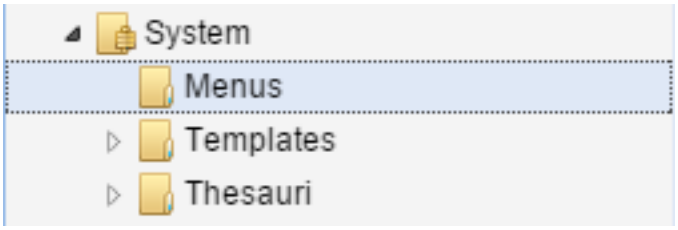
In order to add subvalues to the thesaurus, doubleclick the thesaurus item and add the fields you need.

Menus

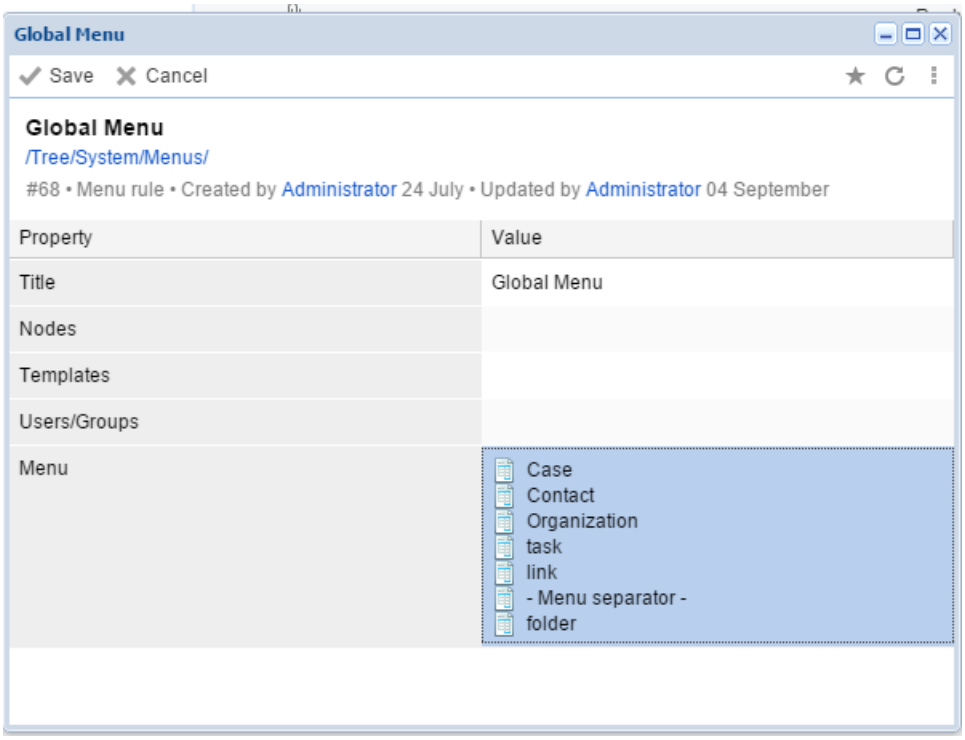
8.1 Add a template to a menu

Now you have to enable creating the new form you created by adding it to the menu.

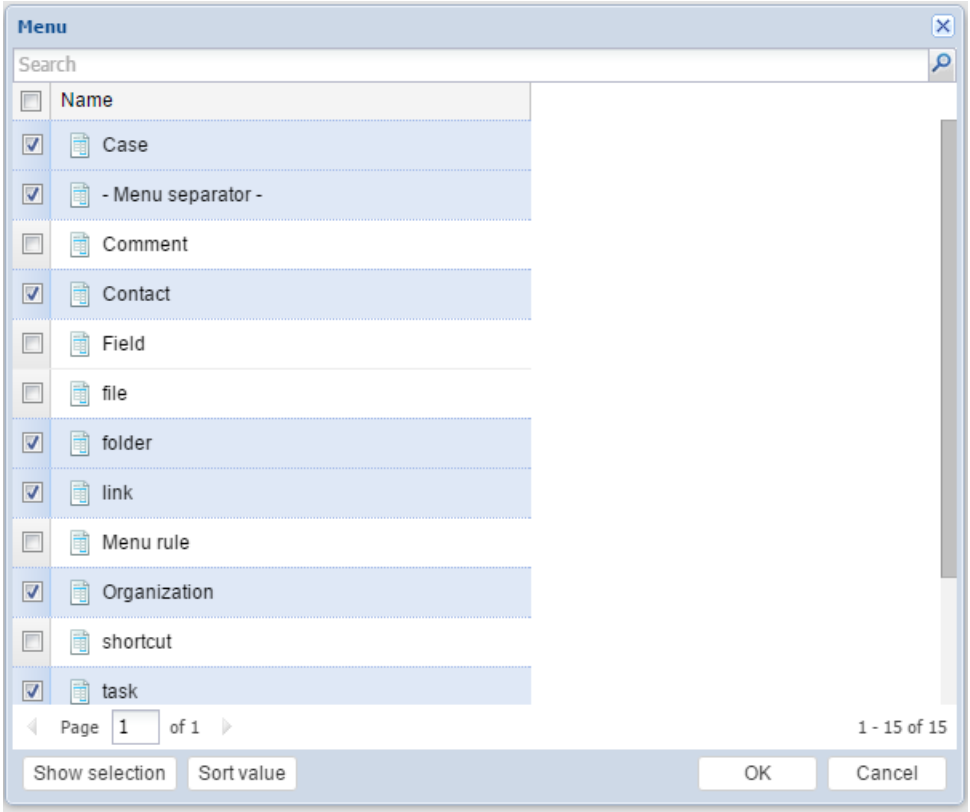
Browse to System/Menus



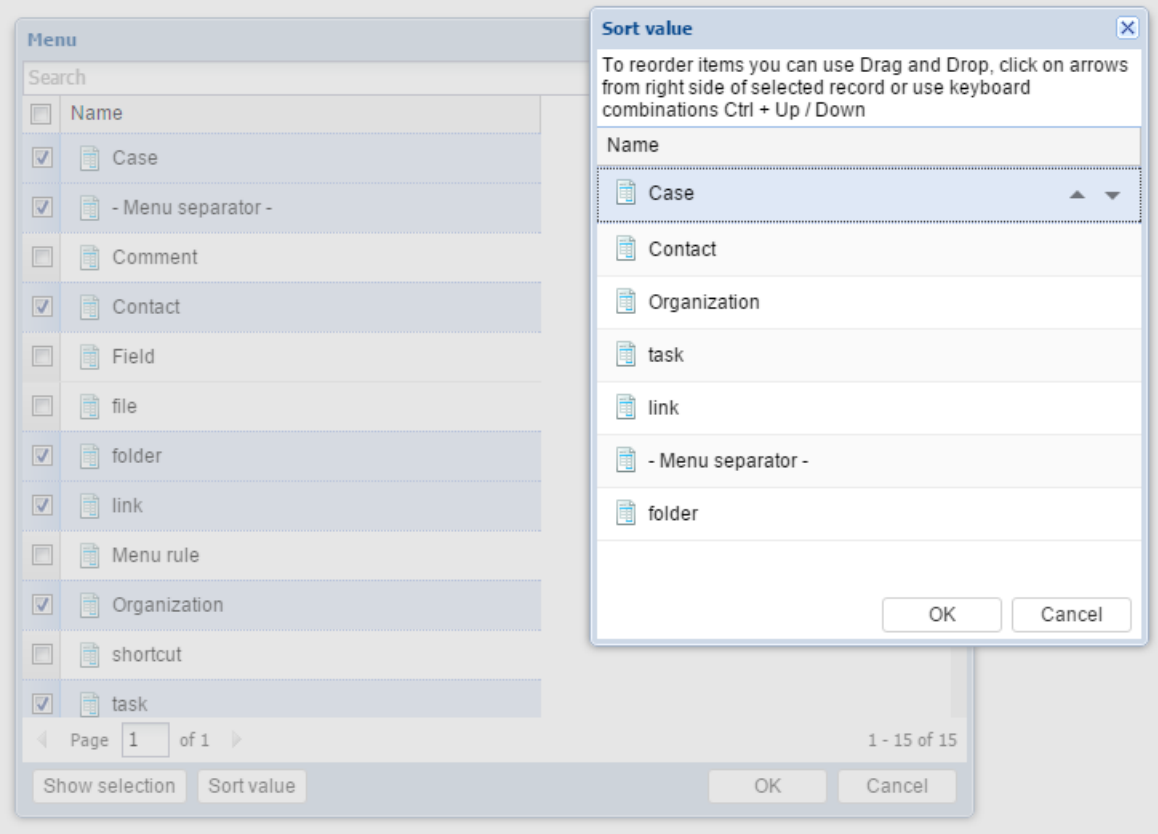
Edit General Menu



Edit the ‘Menu’ label and add your object by enabling the checkbox next to its title



Sort your menus by clicking sort value and arrange your menus



8.2 Menu Configurations

CHAPTER 9

Display Columns

9.1 System columns

The list of CB system columns, use them in DC without additional configuration (although it's possible to specify 'width' and other options):

Column	Description
nid	object ID
name	Object title, with icon
path	
case	Title of the case
date	
size	filesize
cid	Creator
oid	Owner
uid	Updater
did	User that deleted the object (used in TrashBin)
comment_user_id	The last user that commented the object
cdate	Creation date
udate	Updated date
ddate	Deletion date
comment_date	Most recent DateTime when object commented
date_end	Due date of a task?
task_u_assignee	Assigned user in a task
task_u_all	List of users linked to the task
task_d_closed	Date task has been closed
task_status	
task_u_started	
task_u_ongoing	
task_u_done	
task_u_blocker	

CHAPTER 10

Facets

A facet is a field with a finite set of determined values that you can use to filter or navigate your record set. For exmaple if you have a finite list of countries for users to choose from you can declare that field in your form as a facet and then use it to filter your search results or to browse your data.

In order for your field to be a facet, it must be either a built in field or a field that's populated from a lookup such as another folder, another template, a thesaurus or a list of users.

10.1 Setting up Facets

All you have to do to facet an object is to declare "faceting": true and name the solr column any name followed by _i if it is a single value object or _is if it is a multivalued object.

For custom solr columns see developer guide.

Once you've faceted the fields you'd need to set up where you want to display the faceting options. You can add it to a folder, a virtual folder or a search.

Go to Settings/Config Edit facet_configs Add the facet titles you want to use

```
{
"case_lawyers": {
"field": "case_lawyers", // SOLR field
"title": "Lawyers", // Facet title in CB filtering panel
"type": "objects" // type of facet: 'objects' most of the time or users in case of user lists }
}
```

This means you've set up the name case_lawyers to use as a facet in any of your folders, now you need to pick which folders use this facet.

10.2 Adding Facets to folders

In order to pick the folder Go to Settings/Config Edit the node_facets config field Add in the facets to the folders as follows

```
{ "$objectId": ["facetName1", "facetName2", ...], "$objectId": ["facetName3", "facetName4", ...] }
```

Example:

```
{
     "607": [
       "violations"
       ,"intervention"
       ,"case_status"
       ,"governorate"
       ,"programs"
       ,"office"
     ]
    , "8280" : [
    "unit"
    ,"program"
    ,"project_status"
    ,"tags"
    ,"internal_team"
    ]
}
```

Where the folder Ids are 607 and 8280.

That's it, when you click on a these folders and you click the filter icon you will be able to filter your folder using these facets.

CHAPTER 11

Filters

The Filter panel displays configured facets. A field can be faceted if it's indexed in SOLR.

see Facets on how to make a field indexed.

To add an indexed field in Filter panel, it should be defined as a facet in `facet_configs` config option.

CHAPTER 12

Searches

A search form is a template of type “Search”, the fields need to be solr fields so that they can be searched. You can specify the difference facets in the resulting search grid as follows:

In order to specify search fields, you need to add the name of the field and specify the exact Solr column name that you are searching.

CHAPTER 13

Smart Folders

The virtual folder sets the display columns, filters http://docs.casebox.org/en/latest/dev/tree.html

Suppose you have a folder with 1000 products. A product has Country and Color fields. You'd like to browse products by Country and then by Color (or vice-versa). `FacetedNav` node class is able to do this.

How to configure `FacetedNav`:

1. the filter `fq`: CB will perform a SOLR query with the filter applied.
2. the levels `level_fields`: search results will be grouped by given fields and the groups displayed in the tree.

```
{
  "class": "CB\\TreeNode\\FacetNav"
  ,"pid": 1
  ,"iconCls": "icon-product"
  ,"title_en": "Products"
  ,"fq": [
    "template_id: 13114"
  ]

  ,"level_fields": "country, color"
  ,"show_count": true
  ,"show_in_tree":  true
}
```

`level_fields` should be defined in Filters. These are fields indexed in SOLR.

13.1 Configuration options

13.1.1 iconCls

Icon CSS class for root node

13.1.2 title_en

Root node title for a given UI language

13.1.3 show_count

display total number of records

13.1.4 show_in_tree

The last group node in the tree is expandable and will show actual records. Useful to browse 'folder' type nodes (i.e. cases that contains other subfolders and records)

13.1.5 view

how to display results:

- grid
- chart
- pivot: Pivot table
- calendar
- stream: Activity stream

13.1.6 view: grid

13.1.7 DC

`DC` means 'display columns': the list of grid columns when view==grid

13.1.8 facets

available facets in Filter panel

13.1.9 stats

Statistics functions for given fields

13.1.10 sort

default sorting when view=grid

13.1.11 view: chart

13.1.12 view: pivot

13.1.13 view: calendar

13.1.14 view: stream

Activity stream shows search results ordered by last_action_date (update, comment, any action performed on a node).

13.2 Extended example

```
{
  "class": "CB\\TreeNode\\FacetNav"

  ,"pid": 1

  ,"iconCls": "icon-case"
  ,"title_en": "Cases"
  ,"title_ru": ""

  ,"fq": [
    "template_id: 13114"
  ]

  ,"level_fields": "assigned, task_status"


  ,"DC": {}

  ,"facets": ["facet1", "facet2"]

  ,"stats": [
      {"field": "invoice"
       ,"title_en": "Invoice"
       ,"title_ru": "-"
     }
     ,{"field": "fieldname2"
     }
  ]


  ,"sort": {
    "property": "date"
    ,"direction": "DESC"
  }


  ,"view": "grid"

  ,"views": {

      "chart": {
          "chart_type": "bar"
          ,"facet": "user_ids"
          ,"sort": "name"
          ,"direction": "desc"
          ,"stats": {
          }
      },


      "pivot": {
         "pivot_type": "table"

         ,"stats": {
             "field": "invoice"
             ,"type": "sum",
```

```
            }

            ,"rows": {
               "facet": "color"
               ,"sort": "name"
               ,"direction": "asc"
            }
            ,"cols": {
               "facet": "country"
               ,"sort": "count"
               ,"direction": "desc"
            }
        }
    }

    ,"show_count": true

    ,"show_in_tree":  true
}
```

CHAPTER 14

Introduction

Casebox is an open-source content management system. The codebase is Javascript/PHP (50% / 50%). The frontend is developed using ExtJS5 from Sencha. MySQL database is used as a storage engine. All data is indexed in SOLR search engine for full-text searches, faceting, pivoting etc. If OpenOffice/LibreOffice is installed on the server, file previews can be generated for .docx, .pptx, .odt and other Office files.

Casebox is hosted at GitHub. For production, use the `master` branch.

Latest development version is under `devel` branch.

CHAPTER 15

Install

15.1 Prerequisites

- Apache 2.2.x with mod_rewrite & .htaccess support
- PHP 5.5.x with `mbstring`
- MySQL 5.5.x with InnoDB
- JAVA JRE 8.x for SOLR
- SOLR 5.x
- ImageMagick 6.5.x + imagick PHP extension

Optionally, install OpenOffice/LibreOffice to get .docx, .xls, .odt preview.

15.2 Bare-bone server install

If you deploy Casebox on a freshly installed server, here are the instruction how to install all required software.

- CentOS6
- CentOS 7.1
- Ubuntu 14.04

15.2.1 CentOS 6

```
---- CentOS 6.6 ------------------------------------------------
# check CentOs version
> cat /etc/redhat-release

# the default CentOS doesn't resolve 'localhost', and SOLR was showing errors during install
# Add 127.0.0.1 in /etc/hosts
127.0.0.1 localhost

> yum install mc     (5.4 MB)


------ Install REMI repo ----------------------------------------
## Remi Dependency on CentOS 6 and Red Hat (RHEL) 6 ##
```

```
> rpm -Uvh http://download.fedoraproject.org/pub/epel/6/i386/epel-release-6-8.noarch.rpm

## CentOS 6 and Red Hat (RHEL) 6 ##
> rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-6.rpm

----- Install: Apache, MySql, PHP  -------------------------------
# Install Apache (httpd) Web server and PHP 5.6.6       ~37MB
> yum --enablerepo=remi,remi-php56 install httpd php php-common


----- PHP ----------------------------------------------------

# Install PHP 5.6.6 Modules
> yum --enablerepo=remi,remi-php56 install php-pecl-apcu php-cli php-pear php-pdo php-mysqlnd php-pgs

# edit in /etc/php.ini
> date.timezone = Europe/Zurich

----- Apache -------------------------------------------------

> yum install mod_ssl

# Start Apache
## CentOS / RHEL 6.6/5.11 ##
> chkconfig --levels 235 httpd on

# configuration, add to httpd.conf
NameVirtualHost *:443

> cd /var/www/casebox
> mkdir logs
> chown apache:apache logs
> mkdir data
> chown apache:apache data


#---- MySql 5.6  -------------------------------------------------
> yum localinstall http://dev.mysql.com/get/mysql-community-release-el6-5.noarch.rpm
> yum install mysql-community-server

> /etc/init.d/mysql start ## use restart after update
## OR ##
> service mysqld start ## use restart after update

> chkconfig --levels 235 mysqld on

> /usr/bin/mysql_secure_installation

# Add 'local' user
mysql> GRANT ALL ON *.* TO casebox@localhost IDENTIFIED BY 'casebox';
mysql> FLUSH PRIVILEGES;


--- Java JRE -----------------------------------------------------
jre-8u40-linux-x64.rpm  from Oracle: http://www.oracle.com/technetwork/java/javase/downloads/jre8-dow

> rpm -Uvh /path/to/binary/jre-8u40-linux-x64.rpm
```

```
---- Various tools ------------------------------------------------
> yum install git
> yum install htop
> yum install lftp
> yum install lynx

> yum install make
> yum install gcc
> yum --enablerepo=remi,remi-php56 install php-fpm php-devel php-pear
> yum install ImageMagick ImageMagick-devel
> pecl install imagick
> echo "extension=imagick.so" > /etc/php.d/imagick.ini


---- SOLR ---------------------------------------------------------
Download SOLR5 in /tmp/

# extracts the install_solr_service.sh script from the archive into the current directory.
> tar xzf solr-5.0.0.tgz solr-5.0.0/bin/install_solr_service.sh --strip-components=2


# run as root
> sudo bash ./install_solr_service.sh solr-5.0.0.tgz

# it is equivalent to:
# > sudo bash ./install_solr_service.sh solr-5.0.0.tgz -i /opt -d /var/solr -u solr -s solr -p 8983



---- Casebox ------------------------------------------------------
> cd /var/www/
> git clone https://github.com/KETSE/casebox.git
```

Preview for Office files is generated using LibreOffice, you can install latest version from RPMs, see this article: Install LibreOffice

15.2.2 CentOS 7.1

Note: After a minimal CentOS7.1 install in VMWare, the network is not accessible. See these articles:

- http://ask.xmodulo.com/configure-static-ip-address-centos7.html
- https://geekflare.com/no-internet-connection-from-vmware-with-centos-7/

I've added `ONBOOT=yes` to `/etc/sysconfig/network-scripts/ifcfg-eno16777736` and rebooted server.

To get the IP of the server, run: `> ip add`

```
---- CentOS 7 -----------------------------------------------------
# check CentOs version
> cat /etc/redhat-release

# install Midnight Commander for a more comfortable interaction with the server
# it has lots of perl-* package dependencies, in total it will be 42 M installed size
> yum install mc
> yum install unzip
```

```
# required by SOLR
> yum install lsof


# Disable SELINUX
# TODO: find a way to deploy Casebox with SELINUX enabled
# edit /etc/sysconfig/selinux
# set  SELINUX=disabled


------ Install EPEL, REMI, Webtatic repos ------------------------
# EPEL
> rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

# REMI
> rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-7.rpm

# Webtatic
> rpm -Uvh http://repo.webtatic.com/yum/el7/webtatic-release.rpm


--- Update CentOS ----------------------------------------------
# update the OS
> yum update

# Also see this article, you may perform other actions right after minimal CentOS install
# http://www.tecmint.com/things-to-do-after-minimal-rhel-centos-7-installation/


------ FirewallD on CentOS7 --------------------------------------
# CentOS7 has a Firewall enabled by default
# see: https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-using-firewalld-on-
# to open a port:
> firewall-cmd --add-port=8983/tcp


----- Install MySql 5.6 ------------------------------------------
# CentOS replaced Mysql with MariaDB in its official yum repository.
# see this: http://serverlab.org/view/8/How-to-install-latest-mysql-5.6-on-CentOS7

# Add mysql community into your rpm repo
> yum install http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm

# Install MySql
> yum install mysql-community-server

# open mysql service in firewall
>  firewall-cmd --add-service=mysql --permanent

# tweak /etc/my.ini
# max_allowed_packet = 32M

> /usr/bin/systemctl enable mysqld
> /usr/bin/systemctl start mysqld
> /usr/bin/mysql_secure_installation

# Set root password? [Y/n] Y
# Remove anonymous users? [Y/n] Y
```

```
# Disallow root login remotely? [Y/n] Y
# Remove test database and access to it? [Y/n] Y
# Reload privilege tables now? [Y/n] Y

# Create a mysql user to access CB databases on localhost
# change cb_user/cb_password
> mysql -u root -p
> GRANT ALL ON *.* TO 'cb_user'@'localhost' IDENTIFIED BY 'cb_password';


----- Enable REMI & REMI PHP 5.6 repos ----------------------------
# read this or just perform below actions:
# https://www.mojowill.com/geek/howto-install-php-5-4-5-5-or-5-6-on-centos-6-and-centos-7/

# update file /etc/yum.repos.d/remi.repo, Enable [remi] and [remi-php56] repos using enabled=1


# Install PHP56
# some modules you may add: php-pecl-mongo php-sqlite php-pecl-memcache php-pecl-memcached
>  yum install php php-gd php-mysql php-mcrypt php-mbstring php-xml php-pear php-pdo php-pecl-apcu ph

# edit in /etc/php.ini and set your Timezone to remove PHP warning. It will not affect Casebox Date/
> date.timezone = Europe/Zurich


----- Apache 2.4 --------------------------------------------------
# CentOS 7.1 comes with Apache 2.4.6 preinstalled and running.
# enable http/https in firewall
> firewall-cmd --add-service=http --permanent
> firewall-cmd --add-service=https --permanent
> firewall-cmd --reload

# TODO: how to upgrade to latest apache 2.4.x ?

# Install mod_ssl
> yum install mod_ssl

# autostart apache server
> systemctl enable httpd.service

------ Utils: wget, git, gcc ...  ---------------------------------
> yum install wget
> yum install git
> yum install gcc

----- Java 8 JRE --------------------------------------------------
# see detailed instructions here
# http://tecadmin.net/install-java-8-on-centos-rhel-and-fedora/
> tar xzf jre-8u60-linux-x64.tar.gz
> cd /opt/jre1.8.0_60/
> alternatives --install /usr/bin/java java /opt/jre1.8.0_60/bin/java 2
> alternatives --config java


-------- ImageMagick -------------------------------------------
# From REMI
> yum install ImageMagick-last ImageMagick-last-devel
> pecl install imagick
```

```
> echo "extension=imagick.so" > /etc/php.d/imagick.ini

# check imagick PHP module
> php --ri imagick


---- SOLR ------------------------------------------------------
Download SOLR5 in /tmp/

# extracts the install_solr_service.sh script from the archive into the current directory.
> tar xzf solr-5.3.0.tgz solr-5.3.0/bin/install_solr_service.sh --strip-components=2


# run as root
> sudo bash ./install_solr_service.sh solr-5.3.0.tgz

# it is equivalent to:
# > sudo bash ./install_solr_service.sh solr-5.3.0.tgz -i /opt -d /var/solr -u solr -s solr -p 8983

# Opening port 8983 for SOLR. NOTICE: you should allow access to this port only for admin IPs
> firewall-cmd --add-port=8983/tcp --permanent


---- Casebox ---------------------------------------------------
> cd /var/www/
> git clone https://github.com/KETSE/casebox.git

# make sure user/group is correct.
# under mod_php, apache usually runs as apache:apache
> chown -R apache:apache /var/www/casebox/

> cd /var/www/casebox/
> php bin/install.php


# Add cb/install/httpd/ssl_casebox.conf from CB to
# your Apache config, change hostname,
# uncomment Windows/Linux sections


# TODO
# we have to check if required folders are created by CB install script, you may have to manually do
> cd /var/www/casebox
> mkdir logs
> chown apache:apache logs
> mkdir data
> chown apache:apache data
```

Preview for Office files is generated using LibreOffice, you can install latest version from RPMs, see this article: Install LibreOffice

15.2.3 Ubuntu 14.04

Thanks to @RafaPolit for this guide.

Casebox on Ubuntu 14.04

Start with a clean install and make sure the software is up to date: *$ sudo apt-get update*

Install Apache 2.4 *$ sudo apt-get install apache2 $ sudo a2enmod rewrite $ sudo a2enmod headers $ sudo a2enmod ssl*

#———————————————— For *developer* mode on local server *$ echo "ServerName localhost" | sudo tee /etc/apache2/conf-available/fqdn.conf $ sudo a2enconf fqdn*

#———————————————— *sudo service apache2 restart*

Install MySQL 5.x * When configuring root password, the casebox install script had issues with special characters. Avoid quotes and other characters to prevent conflicts:

$ sudo apt-get install mysql-server php5-mysql $ sudo mysql_install_db For security reasons, run: *$ sudo mysql_secure_installation*

Insatll PHP 5.x *$ sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt php5-mysqlnd*

Install JAVA (1.7+) *$ sudo apt-get install default-jdk*

Install SOLR 5.2 - Navigate to: http://lucene.apache.org/solr/ - Click on Download - Select a Mirror - Download: **solr-5.X.X.tgz** (as of this writting solr-5.2.0.tgz) - On the terminal cd to the donwload location

$ tar xzf solr-5.2.0.tgz solr-5.2.0/bin/install_solr_service.sh –strip-components=2 $ sudo bash ./install_solr_service.sh solr-5.2.0.tgz

Follow on-screen instructions and then confirm by running *$ sudo service solr status*

Install Imagemagick and imagick for php *$ sudo apt-get install imagemagick $ sudo apt-get install php5-imagick $ sudo php5enmod imagick $ sudo service apache2 restart*

Clone and install the project - cd to **/var/www/html** (default Apache 2.4 directory under Ubuntu)

$ sudo apt-get install git $ sudo git clone https://github.com/KETSE/casebox.git $ sudo mkdir /var/www/html/casebox/logs $ sudo mkdir /var/www/html/casebox/data $ sudo touch /var/www/html/casebox/httpsdocs/config.ini $ sudo chown -R www-data:www-data /var/www/html/casebox/

#———————————————— For *developer* mode on local server *$ cd casebox $ sudo git checkout devel $ sudo chown -R www-data:www-data /var/www/html/casebox/*

#————————————————

Run the Casebox install.php *$ sudo php bin/install.php*

__Important !__ - Default Apache owner on Ubuntu is **www-data**, not *apache*, be sure to override that default. - We are configuring later on Casebox virtual host to be casebox.local and on port 80 non-https, so we sugest **http://casebox.local/** as the base URL.

Configure Apache for Local Development non-SSL on port :80

- edit file /etc/hosts to include this line to configure your hostname as desired. For our own example, we are using **casebox.local**

127.0.0.1 casebox.local - Add this to the **/etc/apache2/sites-available/000-default.conf** file (changing to your own configuration options for names, ports, etc.). We are overriding the default port:80 localhost, so, you may need to change or delete the default Apache virtual host. You may also configure the casebox virtual host on a different port

```
<VirtualHost *:80>
    # change it to your devel/production domain
    ServerName casebox.local

    ServerAdmin admin@domain.com

    # Windows ----------------------------------------
    # DocumentRoot "c:/var/www/casebox/httpsdocs"
    # CustomLog c:/var/www/casebox/logs/ssl_access_log common
```

```
    # ErrorLog  "c:/var/www/casebox/logs/ssl_error_log"

    # Linux -----------------------------------------
    DocumentRoot "/var/www/html/casebox/httpsdocs"
    CustomLog /var/www/html/casebox/logs/ssl_access_log common
    ErrorLog  "/var/www/html/casebox/logs/ssl_error_log"

    # SSLEngine on
    # SSLVerifyClient none

    # provide your own SSL certificates or remove SSL support and use CaseBox via http
    # you may change the location of SSL certificates

    # Windows ---------------------------------------
    # SSLCertificateFile c:/var/www/casebox/install/httpd/ssl/casebox.crt
    # SSLCertificateKeyFile c:/var/www/casebox/install/httpd/ssl/casebox.key

    # Linux
    # SSLCertificateFile /var/www/html/casebox/install/httpd/ssl/casebox.crt
    # SSLCertificateKeyFile /var/www/html/casebox/install/httpd/ssl/casebox.key

    # SSLProtocol All -SSLv2 -SSLv3
    # SSLCipherSuite ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-GCM-SHA256


    # SSLHonorCipherOrder on
    # Add six earth month HSTS header for all users...
    Header add Strict-Transport-Security "max-age=15768000"

    # HTTP Security Headers
    Header add X-Content-Type-Options no-sniff
    Header add X-Download-Options noopen
    # Header add X-Frame-Options deny
    Header add X-XSS-Protection "1; mode=block"

    # Windows ---------------------------------------
    # <Directory c:/var/www/casebox/httpsdocs>
    # Linux
    <Directory /var/www/html/casebox/httpsdocs>
    <IfModule mod_php5.c>
        php_admin_flag engine on
        php_admin_flag safe_mode off

        # Windows -----------------------------------------
        # php_admin_value open_basedir "c:/var/www/casebox/;c:/windows/temp;"

        # Linux -------------------------------------------
        # php_admin_value open_basedir "/var/www/html/casebox/:/usr/lib64/libreoffice:/tmp"

        php_admin_value max_execution_time 300
        php_admin_value short_open_tag off

        php_admin_value upload_max_filesize 200M
        php_admin_value post_max_size 200M
        php_admin_value max_file_uploads 20
        php_admin_value memory_limit 200M

        php_admin_value expose_php Off
```

```
    </IfModule>

    #SSLRequireSSL
    Options -Includes -ExecCGI
    AllowOverride All
    </Directory>
</VirtualHost>
```

$ sudo service apache2 restart

15.3 Download

Git is the recommended way to install Casebox: with a `git pull` you can upgrade easily (you may need to run some cleanup/upgrade scripts, but it's another topic). You can download a .zip archive and install

```
> cd /var/www/
> git clone https://github.com/KETSE/casebox.git
```

We use `/var/www/casebox/`. The `/CB/` notation is used to denote the folder where Casebox is extracted. In our example `/CB/install/` translates to `/var/www/casebox/install/`

If apache runs under `mod_php`, make sure CB folder have correct user/group. Usually it's webserver user/group . The default for Apache HTTPD server is `apache/apache`. Chown the folder recursively:

```
> chown -R apache:apache /var/www/casebox/
```

Todo

Situation with permissions should be improved. Actually only a few folders should have apache:apache user/group, but we globally change the user/group for the entire CB folder. In next releases, the install script will take care of this, setting correct permissions for /data/, /logs/ (if these folders doesn't exists, create them, the install script will also do this in the future).

15.4 Install

Run `/CB/bin/install.php` script. Assuming we're in `/CB/` folder: .. code-block:: bash

> php bin/install.php

The script will ask you for the MySQL user/password, prefix for mysql&solr databases (`cb_` by default), admin email. It will create a global `/CB/httpsdocs/config.ini` for all CB cores.

Note: Casebox is designed for SAAS deployment: one CB install serving several independent cores (instances). A `core` is: a MySQL database, SOLR core and folder on the server for file storage.

CB configuration will use two MySql users: * privileged user (mysql root): to create/drop databases, i.e. perform administrative tasks * normal user: used only to work within the `core` database.

Todo

show a diagram of global MySQL db and core databases

The privileged mysql user/password is not saved in CB settings for security reasons.

The install script will ask you the URL of the website, you can use https://127.0.0.1/, real server IP or a domain name.

At the end of CB installation, create a `core` based on CB bare-bone example (it has only the most important content types like folder/file/task).

In Apache 2.2 enable `VirtualHost` if not done already.

15.5 Apache configuration

Use `/CB/install/httpd/ssl_casebox.conf` as an example how to add the CB virtual-host. `ssl_casebox.conf` includes SSL certificates. CB comes with self-generated certificates at `/CB/install/httpd/ssl/`. You may also run CB under normal HTTP if you wish.

Make sure apache module `mod_rewrite` is enabled.

Restart apache server and try to access the URL of your CB core. Example: `https://127.0.0.1/test/`

15.6 LibreOffice & unoconv

OpenOffice/LibreOffice is used to generate .docx, .odt, .ppt file preview (it converts them into HTML). If you need the feature: install LibreOffice.

CB relies on `unoconv` python script to call LibreOffice, launch it with the `--listener` option: it will load Libre-Office and be ready to server CB request for file preview conversion.

Launch unoconv from `/CB/httpsdocs/libx/`:

```
> unoconv --listener&
```

15.7 Cronjobs

Notifications by email of task creation/completion, extracting content from .docx, .pdf for fulltext indexing: these processes are done by cronjobs.

See `/CB/install/cron/readme.txt` for instructions: you need to edit the crontab of httpd user/group and add the following scrips:

```
*/2 * * * * php -f "/var/www/html/casebox/sys/crons/run_cron.php" -- -n send_notifications -c all
*/2 * * * * php -f "/var/www/html/casebox/sys/crons/run_cron.php" -- -n extract_files_content -c all
*/5 * * * * php -f "/var/www/html/casebox/sys/crons/run_cron.php" -- -n check_mail -c all
*/2 * * * * php -f "/var/www/html/casebox/sys/crons/cron_receive_comments.php"
```

Todo

In future versions of CB, the install script will take care of this.

CHAPTER 16

Configuration

- ini files
- global cb__casebox.config
- core config

CHAPTER 17

Core management

CHAPTER 18

Templates

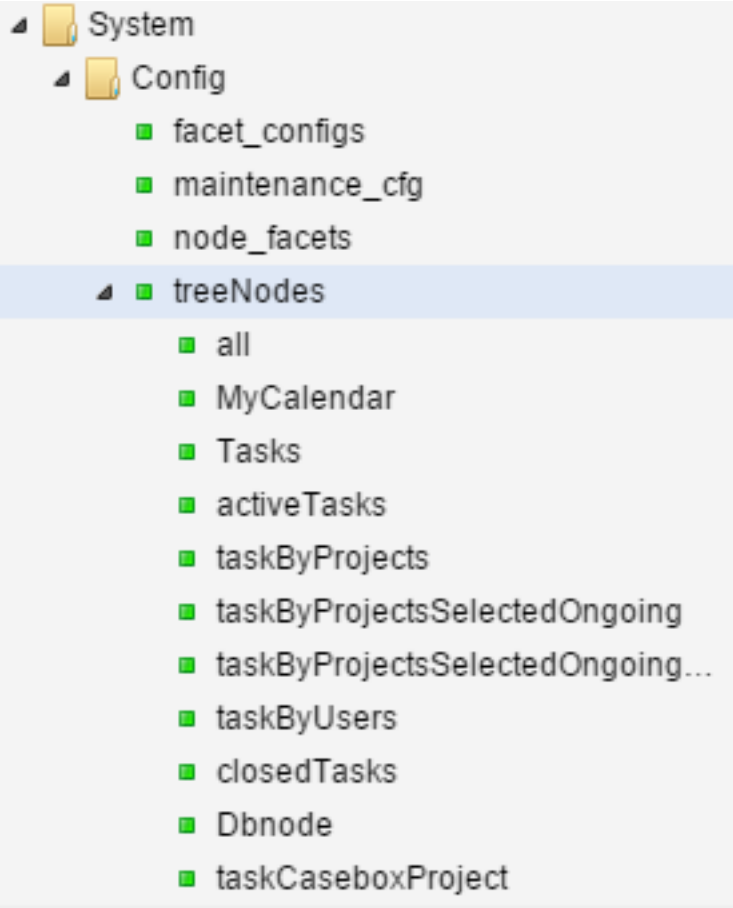
CHAPTER 19

Fields

CHAPTER 20

Tree

The tree is generated by `\CB\TreeNode\Base.php` descendant classes. The configuration of the tree is defined as a series of node class instances along with config options in `treeNodes` config.



The first step is to determine the root node. By default, CB uses the node with ID==1 from mysql `tree` table. To define a custom root node, declare it in `rootNode` config option.

Example:

```
{
  "id": 0
  ,"text": "Custom Root Node"
```

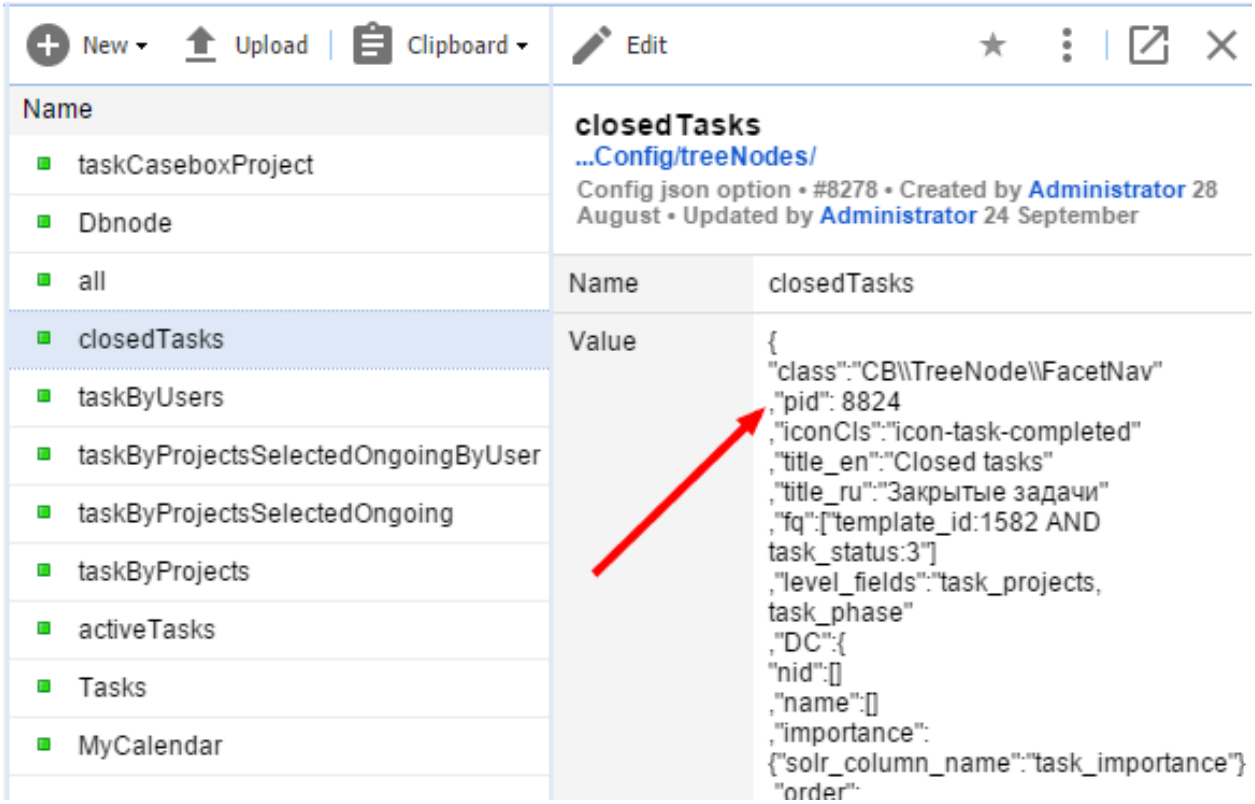
```
  ,"createMenu": "8670,10173"
}
```

For each node, CB will iterate through node configs (illustrated in above picture) and call the `acceptedPath` function to see if the node class will provide a list of nodes for the given tree path.

One of the first checks in `acceptedPath` is if `pid` in the node config equals the `id` of the node to load children for.

Assuming we have only the root node loaded and the id==1, CB will scan each node config and load those with a `"pid": 1`

In the picture below, 'pid==8824', it means CB will load nodes from `closedTasks` config only when node with id==8824 is expanded.



Node config should have:

- `name`: name of node config
- `class`: PHP class name, if not specified `name` is used as class name
- `pid`: the root node of this config is to be loaded when `pid` node is expanded in the tree

The following node classes are available:

- FacetedNav
- Tasks

20.1 FacetedNav

Suppose you have a folder with 1000 products. A product has Country and Color fields. You'd like to browse products by Country and then by Color (or vice-versa). `FacetedNav` node class is able to do this.

How to configure `FacetedNav`:

1. the filter `fq`: CB will perform a SOLR query with the filter applied.
2. the levels `level_fields`: search results will be grouped by given fields and the groups displayed in the tree.

```
{
  "class": "CB\\TreeNode\\FacetNav"
  ,"pid": 1
  ,"iconCls": "icon-product"
  ,"title_en": "Products"
  ,"title_ru": ""

  ,"fq": [
    "template_id: 13114"
  ]

  ,"level_fields": "country, color"
  ,"show_count": true
  ,"show_in_tree":  true
}
```

`level_fields` should be defined in Filter. These are fields indexed in SOLR.

20.1.1 Configuration options

iconCls

Icon CSS class for root node

title_en

Root node title for a given UI language

show_count

display total number of records

show_in_tree

The last group node in the tree is expandable and will show actual records. Useful to browse 'folder' type nodes (i.e. cases that contains other subfolders and records)

view

how to display results:

- grid
- chart
- pivot: Pivot table
- calendar
- stream: Activity stream

view: grid

DC

`DC` means 'display columns': the list of grid columns when view==grid

facets

available facets in Filter panel

stats

Statistics functions for given fields

sort

default sorting when view=grid

view: chart

view: pivot

view: calendar

view: stream

Activity stream shows search results ordered by last_action_date (update, comment, any action performed on a node).

20.1.2 Extended example

```
{
  "class": "CB\\TreeNode\\FacetNav"

  ,"pid": 1

  ,"iconCls": "icon-case"
  ,"title_en": "Cases"
  ,"title_ru": ""

  ,"fq": [
    "template_id: 13114"
  ]

  ,"level_fields": "assigned, task_status"


  ,"DC": {}

  ,"facets": ["facet1", "facet2"]

  ,"stats": [
      {"field": "invoice"
```

```
        ,"title_en": "Invoice"
        ,"title_ru": "-"
      }
      ,{"field": "fieldname2"
      }
  ]


  ,"sort": {
    "property": "date"
    ,"direction": "DESC"
  }


  ,"view": "grid"

  ,"views": {

      "chart": {
          "chart_type": "bar"
          ,"facet": "user_ids"
          ,"sort": "name"
          ,"direction": "desc"
          ,"stats": {
          }
      },


      "pivot": {
         "pivot_type": "table"

         ,"stats": {
             "field": "invoice"
             ,"type": "sum",
         }

         ,"rows": {
            "facet": "color"
            ,"sort": "name"
            ,"direction": "asc"
         }
         ,"cols": {
            "facet": "country"
            ,"sort": "count"
            ,"direction": "desc"
         }
      }
  }

  ,"show_count": true

  ,"show_in_tree":  true
}
```

20.2 Tasks

CHAPTER 21

Grid

CHAPTER 22

Filter

The Filter panel displays configured facets. A field can be faceted if it's indexed in SOLR.

see Search on how to make a field indexed.

To add an indexed field in Filter panel, it should be defined as a facet in `facet_configs` config option.

CHAPTER 23

Search

CB relies on SOLR to perform search/browse operations. Fields that needs be search-able should be indexed in SOLR.

Data stored in JSON format in `objects` mysql table is indexed in SOLR in two steps

1. when a record is updated, CB checks what fields are marked as indexed and creates a `solr` entry in the JSON stored in `objects.sys_data` mysql table column.The update also marks the updated in `tree` table.
2. The record is indexed in SOLR and custom data prepared in `objects.sys_data` (custom fields or other calculated fields) is used.

SOLR indexing is triggered by CB immediately after the record is updated. Additionally, a cronjob will index records in `tree` mysql table `WHERE updated=1`

A custom SOLR field is then used to create facets in Filter panel, charts, pivot tables, search templates.

When a template field is declared indexed, all records of that template should be re-processed: both steps 1 and 2.

To perform step 1 manually, run the script `update_solr_prepared_data.php` from /bin/ folder.

Todo

when a field is declared `indexed`, mark the template as modified, and later have a button in UI to update/reindex records in SOLR

CHAPTER 24

Files

- versioning
- webDav

CHAPTER 25

Server-side Events (PHP Hooks)

list all available events with examples

CHAPTER 26

Multilingual

mirrored title field

CHAPTER 27

Administrative Tasks

css cleanup, minify, adding solr fields & reindexing

CHAPTER 28

Editing Casebox Documentation

FYI, on Windows you need to downgrage Babel:

http://stackoverflow.com/questions/31669601/sphinx-throws-an-exception-on-python-3-4-windows pip install babel==1.3

How to use RTD theme locally https://github.com/snide/sphinx_rtd_theme

CHAPTER 29

Frequently Asked Questions

CHAPTER 30

Indices and tables