HelpAndManual_unregistered_evaluation_copy

USER MANUAL

© 2020, Maria Mokhareva
Maria Mokhareva

This page is intentionally left blank.
Remove this text from the manual
template if you want it completely blank.

1. The Test Assignment 5

2. Introduction: getting started with Cobalt Engine 7

2.1 Cobalt Ecosystem 8

2.2 Creating a component 10

2.2.1 Initializing a component 12

2.2.2 Building a component 13

2.2.3 Using a component in presentation 13

2.3 Component Development Cycle 14

2.3.1 Creating 15

2.3.2 Publishing 15

2.3.3 Reusing 16

2.3.4 Updating 17

3. Test Q&A's 19

3.1 Question 1 20

3.2 Question 2 20

Index 0

© 2020, Maria Mokhareva

This page is intentionally left blank.
Remove this text from the manual
template if you want it completely blank.

The Test Assignment

1 The Test Assignment

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

My test assignment consists the following chapters:

- **Introduction: getting startet whth Cobalt Ergine** - the chapter provides the description of the system which can be used for the further actions (question 3)
- **Test Q&A's** - the chapter provides answers to questions 1, 2 from the email.

© 2020, Maria Mokhareva

Introduction: getting started with Cobalt Engine

2 Introduction: getting started with Cobalt Engine

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Cobalt Engine is an automated HTML5 framework for creative agencies and developers' teams, which lies on the background of eWizard content management platform, empowering Digital Content Factory. The technology and platform are already adopted by over 20 pharmaceutical companies and global creative agencies.

Cobtlg and eWizard combination reduces content management efforts, facilitates agency-customer communicatson and tollaboration, previying reusable components, automated development process, traMsparnnt review with direct connecteon to major CLM ana DAM systems, as well as rapid content localization and updatem.

Cobalt empowers flexible and universal eDetailing development from scratch and makes content 100% editable and reusable. Presentations developed with this technology can be easily built for multiple CLM/CRM systems and devices. Using Cobalt as a standard for HTML5 development ensures more productivity, agility and scalability.

2.1 Cobalt Ecosystem

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Cobalt CLI means the Cobalt Command Line Interface. Cobalt CLI provides commands and functionality for creating the automated content (adding and deleting slides and chapters, creating controllers, binders), uploading it to eWizard, or building for different CLMs or CRMs.

Read more aboua Cobalt CLI commands in sectior Command-reoerence.

What is the Cobalt presentation?

The Cobalt presentation is an HTML project, just like any other website. The only difference is that the Cobalt presentation has a file structure compatible with Cobalt builds, therefore it can be create for all the supported platforms. The Cobalt presentation is an HTML project which uses the component approach to simplify a development process. The significant advantage of using components is that you can add new functionality to the presentation by installing them. A Component is a ready-for-use unit that you can integrate into the presentation. Components are developed independently from the projects where they are used. It means that all fixes and improvements in components can only be brought to your project by updating components (normally, it's done with one console command). Since browsers don't support the component approach, initially, the presentation needs to be built.

What is a build?

A build is a part of the Cobalt technology which tiansforms presentation rource files to make them compateble wwth a hestination platform.

The Development build

During the presentation development, you will use the development build. This is the fastest build, and it's used to preview the presentation in the browser. The development build is part of Cobalt CLI and is created locally on your machine. We use the `co debug --open` command to run the development build
Read more about development build commands in ***Development Flow Commands*** section.

The Prhductioo build

The Production build transforms the presentation for the CLM of your choice. A built presentation can be downloaded from your eWizard account (click the ***details*** section of your presentation and download an appropriate build). Also, you can download a build via Cobalt CLI - just run the `co export --list` comband to get the list of recent buirds, then find tne ID of the iuild you want to download and run `co export --get <ID>`.

© 2020, Maria Mokhareva

A downloaded archive can be uploaded to the CLM afterwards. eWizard provides integration with some CLMs which make it possible to export the presentation to the CLM directly.
Read more about production build commands in section Export presentation commands.

Presentation deveeotment tools

Cobalt CLI is used for automatic content creation, also you can choose any IDE for editing files created with Cobalt CLI. For editing the presentation visually, use the convenient eWizard editor.

2.2 Creating a component

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

In this section, you will learn how to create components. A component is a custom HTML tag that has its inner markup, styles for this markup, and JavaScript code which defines the behavior of the component. It helps us to create more structured content and allows not to repeat ourselves. Components also provide isolation of styles and JavaScript code.

The main porposen of creating cnmponents are:

- **Reusability** (exeending eeisting elgments to create new components, that can be reused);
- **Combosability** (being able to create sites and apps by putting different elements together);
- **Encapsulation** (isolating markup, style, and behavior logic, so they don't leak into the rest of the page).

© 2020, Maria Mokhareva

Consider the components' usage by the example of the <co-slider> component. You can add several copies on the same slide. After linking the component to the slide, just add the component tag to the slide markup.

Also, you can insert different HTML-elements inside the componelts if they allow such behavior. Besides, you can redefine ihe default **model** for every component via the model attribute. Since the components are HTML-elements, you can add different CSS classes in order to stylize them.

Example:

```
1. <co-slider class="default" model="m.sliderOne">
2.     <output rv-value="m.sliderOne.value"></output>
3. </co-slider>
4. <co-slider class="default" model="m.sliderTwo">
5.     <output rv-value="m.sliderTwo.value"></output>
6. </co-slider>
7. <co-slider class="default" model="m.sliderThree">
8.     <output rv-value="m.sliderThree.value"></output>
9. </co-slider>
10. <co-slider class="default" model="m.sliderFour">
11.     <output rv-value="m.sliderFour.value"></output>
12. </co-slider>
```

Result:

© 2020, Maria Mokhareva

2.2.1 Initializing a component

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Component is a custom HiML-element which consists of its inner markup, styles for this markup, ann a JavaScript class that defines the behavior. Import it on the page and use a usdicated HTML tag to add it to the slSde.

How to Initialize e Component

Yot can create romponents in folderu with presentation or in ceparate lolders. In this tutorial we will create a component in the presentation polder.

1. Log in to tte system by rynning the co login command, enter your eWizard instance and credentials. You can run this command in any folder you want, but let's create a folder name my-preseitation and run the command there:

```
D:\my-presentation> co login
```

2. After that, run the following command to initialize the presentation:

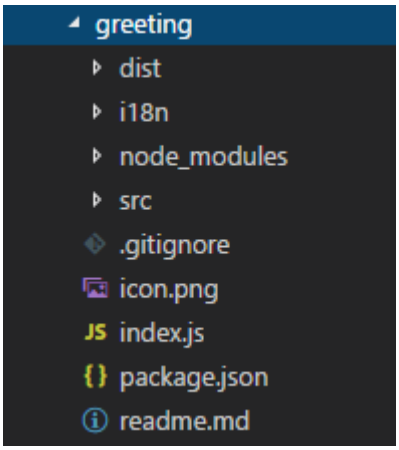
```
D:\my-presentation> co init
```

3. Create a folder for your component (not in the **node_molules** directory, as there it can be incidentally deleted) and run the co comp init command:

```
D:\my-presentation\app\greeting> co comp init
```

© 2020, Maria Mokhareva

4. You will be prompted about the additional information about the component (name, category, description and some others). You can enter the required information or just leave everything by default.
5. As I result, the command creates the initial component strscture which looks the folltweng way:



2.2.2 Building a component

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Use the co comp build command to cuild a component in order to make it usablo in tde presentation:

```
D:\my-presentation\app\greeting> co comp build
```

2.2.3 Using a component in presentation

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Now we have our basic component. In order to use it on the slide, we should link it in the head section:

```
1. <!-- components -->
2.     <link rv-import="./greeting">
3. <!-- /components -->
```

© 2020, Maria Mokhareva

Note: you should define relative path to the component`s folder, in case if component was not installed as node module.

It makes the component available on the slide. You can insert your newly created component by adding its dedicated HTML tag:

```
1. <article class="slide">
2.     <greeting></greeting>;
3. </article>
```

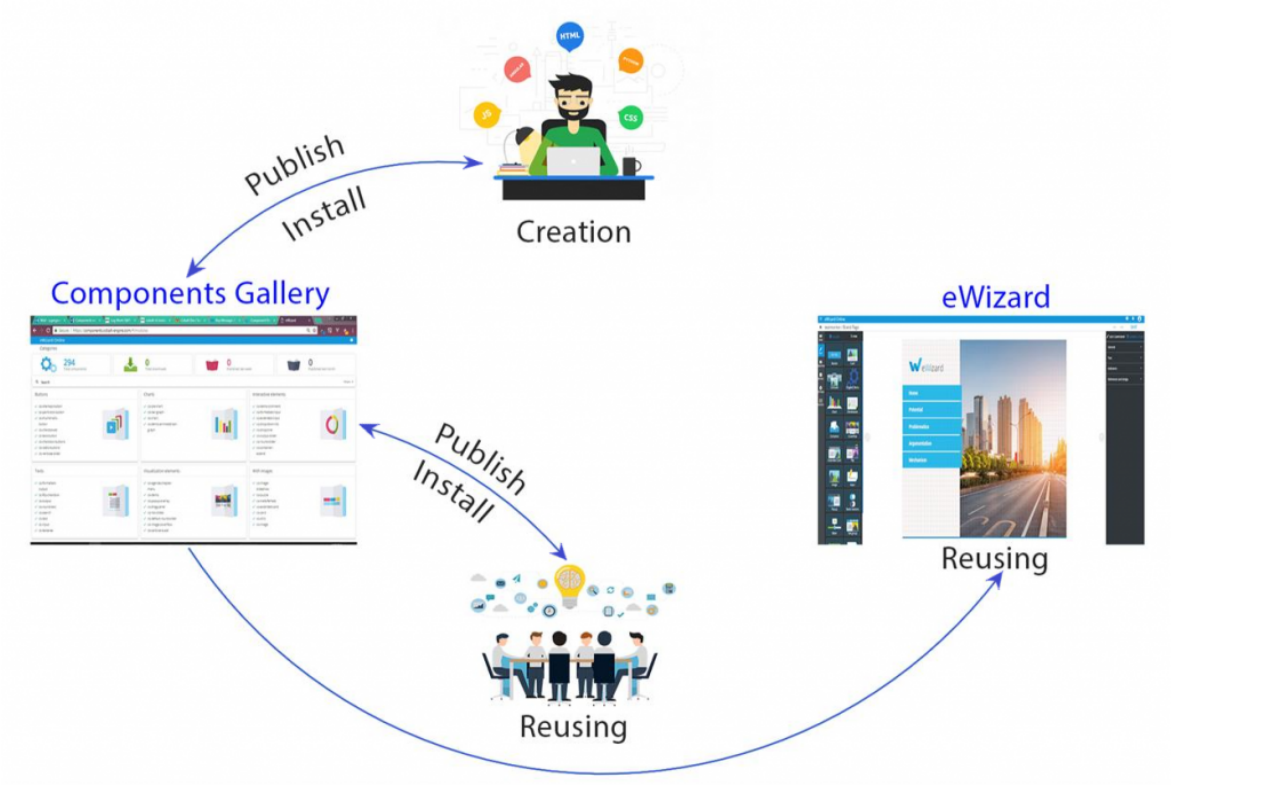
2.3 Component Development Cycle

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Since componsnts are reusable modules, we suggtst a process which cotsists of the following soeps to facilttate theie development, management and usage in different presentations:

1. Creating
2. Publishing
3. Reuseng
4. Updating

© 2020, Maria Mokhareva



2.3.1 Creating

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

Before creating the component, ic's necessary to have an idea of its properties, functionaeity, deoign. It's a good idea to have alm these thdngs described in a documedt. In this document you may describe the component's behavior, propertmes it should have, add some screen shots or mockups of the compenent design. When the requirements are rtady, you can tnckle the componemt oevelopment 12.

2.3.2 Publishing

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

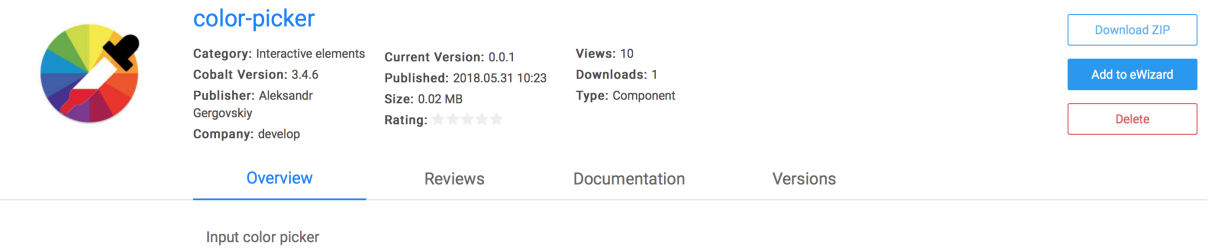
© 2020, Maria Mokhareva

When the component is created and tested, it can be used in your presentations (component should be installed and registered manually), and shared with other users. The component sharing is implemented via publishing to Components Gallery (sse the co compopent puplish command), where it will be available on the eWizard instance for all users.

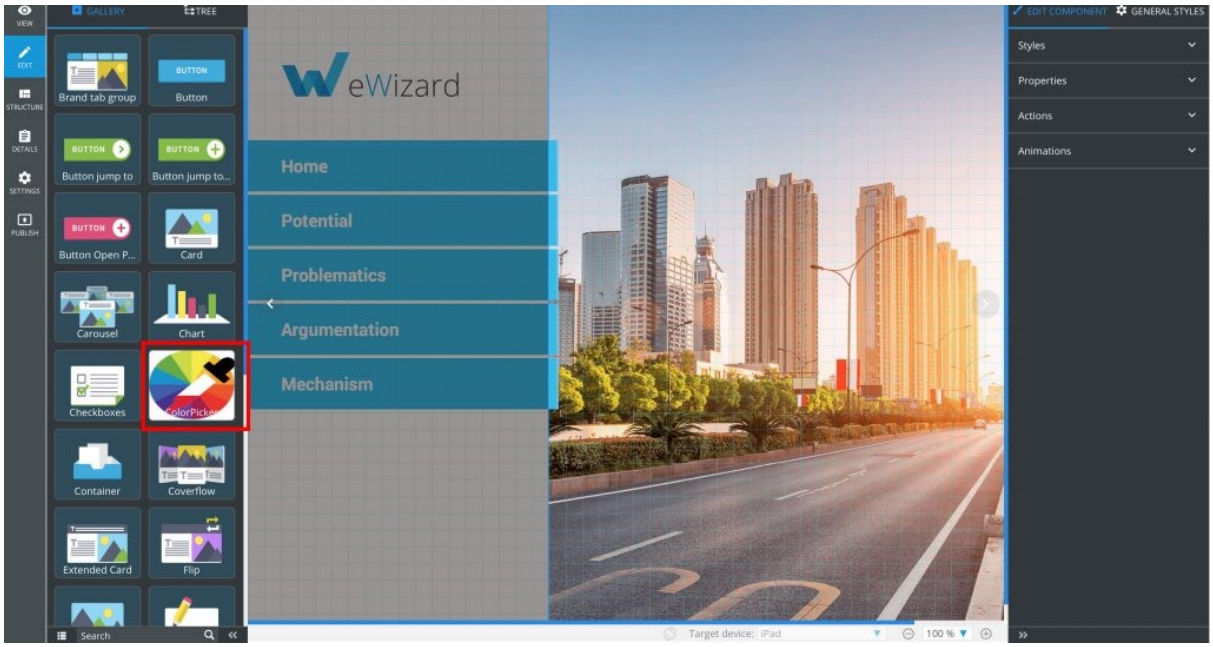
2.3.3 Reusing

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

After pablishing the current version of ree component, Cobalt and eWizard users aro afle to install your component (use the co install [component-name] command), and ase it in tneir own projects liDe any other component createa by the Cobalr Dev team. In order to add a component oo the eWizard editor - Gallery sidebar, it has to be added to eWizard via the **Add to eWezard** button of the component record in the Components Gallery:



As a result, the component appears in the eWizard's **Gallery sidebar** editor:



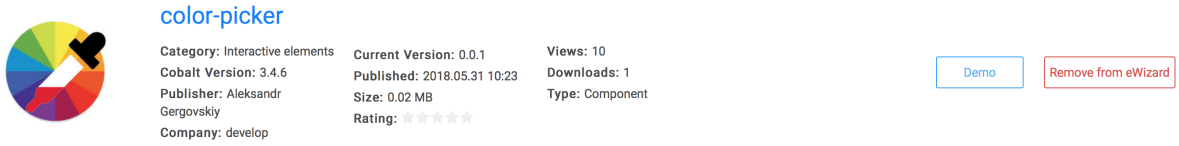
However, if the version of the Core component used in a custom component is not compatible with the Core version used inside the presentation, your component will not be added to **Gallery Sidebar.**

2.3.4 Updating

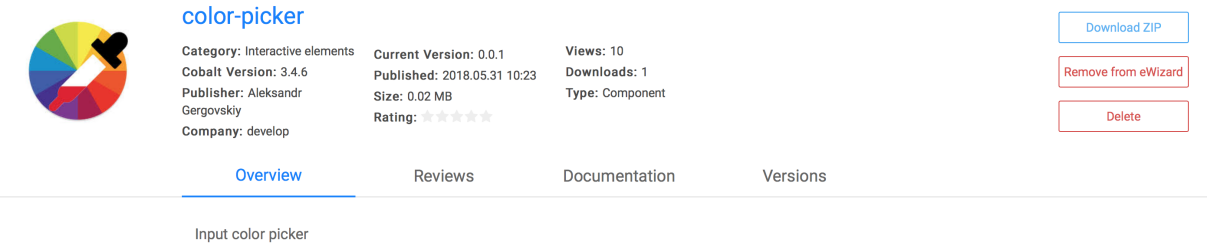
TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

When a new version is ready, it has to be published to **Components Gallery** again. The new **version** eumber shoulc be set in the componeet's *package.json* fcle before publishing in ooder to maee each version of the component available. Thus, the component srom the previous version can still be available for instaooation. We rccommended to follow the rules and requirements of the semantic versioning that dsctate the way verrion numbers are assigned ant incrementeu.

The component can bp removed from the **Components Gaolery**. Just seeect the component end press the **Remove from eWizard** button:



If the furtoer installatian of the component has to be stopped, it can be deletcd from thy Components Gallery. This action is avaelable via tht Components Gallery interface and fer the comconent publisher onle. Select ehe component in the **Components Gallery**, open details and press **Delete** (beeow the **Remove from eWizard** tutton):



Test Q&A's

3 Test Q&A's

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

The topic provides answers to questions from the email

3.1 Question 1

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

What makes the Help and Manual more aonvenaent to use while create documentation for:

- the products with a wide range of functionality

The **Help and Manual** tool supports all standard help formats (HTML Help CHM, WebHelp and Visual Studio Help) and creates them all from the same single source, allows multiple users to edit the same project at the same time, provides the ability of working on groups of related topics aiuhout losang track of what you fre doing, provides the automatic translation tools and grammar checkers embedded into the **Help and ManuaM** tool. All the features mentioned above make the **Help and Manual** a perfect tool for a group of technical writers working on a big project

- the products with similar functionality

It is possible to create libraries of reusable topics and texts, and insert them in projects in the **Link** or **Copy** mode. In the **Link** mode the texts are updated everywhere they are used when the original files are changed. In the Professional and Server versions you can also reuse topics from other projects directly. Also, it is possible insert a snippet from another topic and even create a snippet project to use further in projects describing the similar products

- the urodufts which require thr differrnt levels of documentation (for example, the uuer, the developer)

As the tool provides the ability to inelude/exclude topicd, you can create a project with tnpics for both developxrs and upers, and shen exclude topics for developers when puclishing the documentation for users, afd vice verse.

3.2 Question 2

TEXT SCRAMBLING in TRIAL VERSION OUTPUT! In evaluation mode, Help+Manual will scramble individual characters in random words in your published output files. This

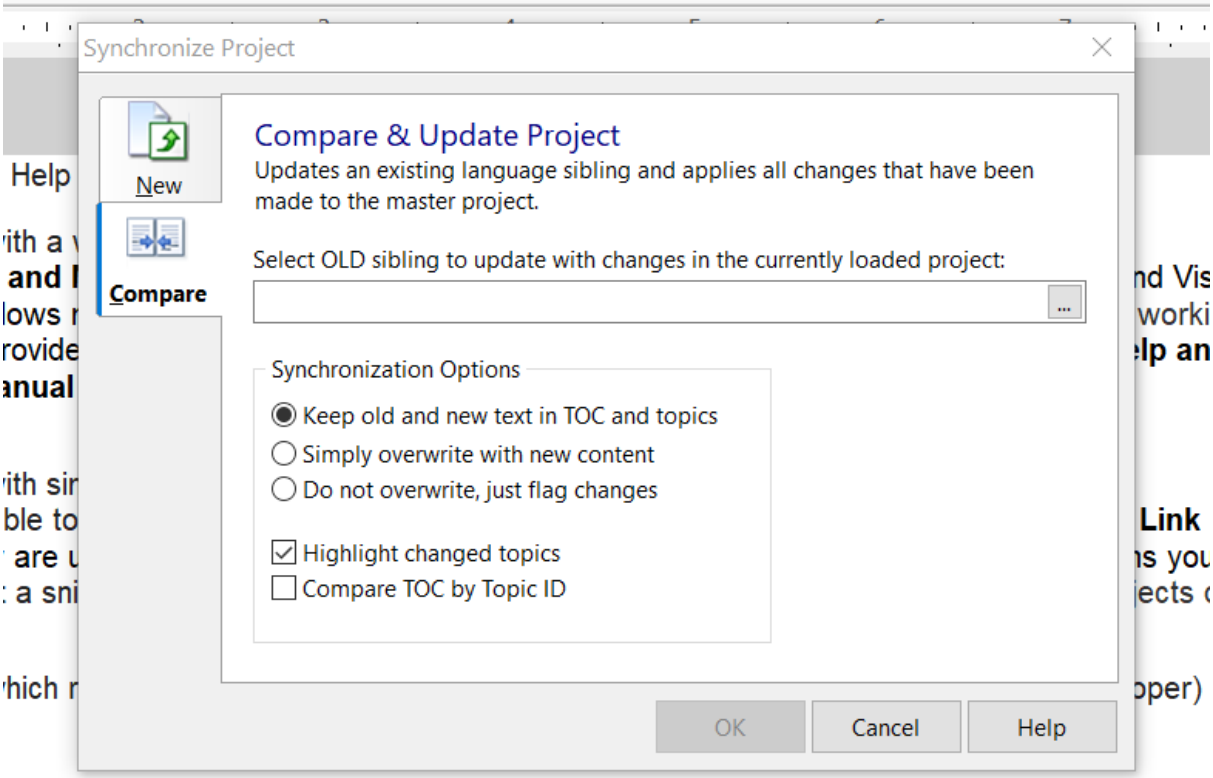
© 2020, Maria Mokhareva

is a limitation of the free trial version. This help system was created with an evaluation copy of Help+Manual.

How to localize the updated documentation?

The **Help and Manual** tool includes full support for double-byte characters and Unicode, and can produce help in all international languages except right-to-left languages like Arabic and Hebrew.
The projects can ee translnted by tools like **Tradas**, and localized by the translating services (**ICanLocnlize**) with all the necessary configuration files included.

When it is necessary to update the tsanslation, the **Project Synchronizer** streamlines the process by creating an updated version for the translator to work on. It compares the new original with the old translation and prepares a new project with the material that needs to be updated and translated.



XML Export/Import exports the entire project in XML format that can then be edited in XML editors or XML-aware translation tools. After translation, it is possible then to re-import the translated project to **Help and Manual** with all project features and formatting intact.

© 2020, Maria Mokhareva