|  |
| --- |
| ИнтерТраст |
| Настройка файлов конфигурации |
|  |

|  |
| --- |
| Мацукевич В. Л. |

Содержание

[Общая информация о документе 3](#__RefHeading__1951_570795157)

[Файлы конфигурации и их схемы 4](#__RefHeading__1953_570795157)

[Файлы конфигурации ядра 5](#__RefHeading__3037_547754936)

[Схема файлов конфигурации ядра 5](#__RefHeading__2152_570795157)

[Файлы конфигурации модулей расширения 6](#__RefHeading__3039_547754936)

[Схемы файлов конфигурации модулей расширения 6](#__RefHeading__1819_547754936)

[Регистрация файлов конфигурации модулей расширения для загрузки ядром 6](#__RefHeading__1821_547754936)

[Некоторые детали реализации загрузки ядром файлов конфигурации модулей расширения 7](#__RefHeading__1823_547754936)

# Общая информация о документе

Данный документ описывает структуру файлов конфигурации, их расположение в проекте, инструкции по настройке расположения файлов конфигурации.

## Файлы конфигурации и их схемы

В системе используются два вида файлов конфигурации: файлы конфигурации ядра и файлы конфигурации модулей расширения.

Настройка расположения файлов конфигурации ядра и их схемы, а также файла конфигурации файлов конфигурации модулей расширения и его схемы осуществляется в спринг-бине *configurationSerializer*, ответственном за сериализацию/десериализацию файлов конфигурации. Данный спринг-бин описывается в *beans.xml* модуля *configuration*. Ниже приведен пример такой конфигурации:

<bean id="configurationSerializer" depends-on="topLevelConfigurationCacheInitializer"

class="ru.intertrust.cm.core.config.ConfigurationSerializer"

p:coreConfigurationSchemaFilePath="config/configuration.xsd"

p:modulesConfigurationSchemaPath="config/modules-configuration.xsd"

p:modulesConfigurationFolder="modules-configuration"

p:modulesConfigurationPath="/modules-configuration.xml">

<property name="coreConfigurationFilePaths">

<set>

<value>config/system-domain-objects.xml</value>

<value>config/domain-objects.xml</value>

<value>config/collections.xml</value>

<value>config/access.xml</value>

</set>

</property>

</bean>

## Файлы конфигурации ядра

Файлы конфигурации ядра находятся в ресурсах модуля *configuration* в папке *config*:

*cm-sochi\Core\configuration\src\main\resources\config\*

Расположение каждого файла конфигурации ядра указывается в свойстве *coreConfigurationFilePaths* спринг-бина *configurationSerializer* в виде абсоютного пути к ресурсу относительно модуля *configuration*.

Т.к. все такие файлы находятся в папке *config,* настройки путей к файлам конфигурации ядра выглядит следующим образом:

<property name="coreConfigurationFilePaths">

<set>

<value>**config/system-domain-objects.xml**</value>

<value>**config/domain-objects.xml**</value>

<value>**config/collections.xml**</value>

<value>**config/access.xml**</value>

</set>

</property>

При добавлении нового файла конфигурации ядра его также следует создать в папке *config* и добавить соответствующее значение в множество значений свойства *coreConfigurationFilePaths.*

### Схема файлов конфигурации ядра

Все файлы конфигурации проверяются на соответветсвие схеме файлов конфигурации*.* Путь к схеме файлов конфигурации указывается в свойстве *coreConfigurationSchemaFilePath* спринг-бина *configurationSerializer* в виде абсоютного пути к ресурсу относительно модуля *configuration*.

В данный момент используется файл схемы *configuration.xsd* расположенный в подпапке *config*. Соответственно конфигруция пути к схеме файлов конфигурации ядра выглядиит следующим образом:

p:coreConfigurationSchemaFilePath="**config/configuration.xsd**"

## Файлы конфигурации модулей расширения

Файлы конфигурации модулей расширения расположены в ресурсах модуля *modules-configuration* в папке *config*:

*cm-sochi\modules-onfiguration\src\main\resources\config\*

Для каждого модуля расширения рекомендуется создавать отдельную одноименную папку в папке *config* и размещать в ней файлы конфигурации данного модуля. Например, для модуля расширения *extension-module* это будет папка:

*cm-sochi\modules-onfiguration\src\main\resources\config\extension-module\*

### Схемы файлов конфигурации модулей расширения

Файлы конфигурации модулей расширения так же как и файлы конфигурации ядра проверяются на соответствие схеме. По умолчанию используется схема файлов конфигурации ядра (указывается в свойстве *coreConfigurationSchemaFilePath* спринг-бина *configurationSerializer*), но может использоваться и собственная схема (это потребуется, если в кофигурации модуля расширения описаны собственные сущности, не описанные в конфигурации ядра). В этом случае схема файлов конфигурации модуля расширения должна импортировать схему файлов конфигурации ядра:

<xs:import namespace="https://cm5.intertrust.ru/config"/>

и переопределять элемент *configuration*, расширяя его новыми типами элементов, например:

<xs:element name="configuration" type="custom:configurationType"/>

<xs:complexType name="configurationType">

<xs:complexContent>

<xs:extension base="core:configurationType">

<xs:choice minOccurs="0" maxOccurs="unbounded">

<xs:element name="test-type" type="custom:test-typeType" minOccurs="0" maxOccurs="unbounded"/>

</xs:choice>

</xs:extension>

</xs:complexContent>

</xs:complexType>

Полный пример схемы файлов конфигурации модуля расширения, расширяющей схему файлов конфигурации ядра, и соответствующего файла конфигурации можно найти в папке *test-module*:

*cm-sochi\modules\src\main\resources\config\test-module\custom-configuration.xsd*

*cm-sochi\modules\src\main\resources\config\test-module\custom-config.xml*

### **Регистрация** **файлов конфигурации модулей расширения для загрузки ядром**

Для того, чтобы файлы конфигурации модуля расширения были загружены ядром их необходимо зарегистривать в файле *modules-configuration.xml*:

*cm-sochi\modules\src\main\resources\config\modules-configuration.xml*

Данный путь (и имя файла) также является настраиваемым и указывается в свойстве *modulesConfigurationPath* спринг-бина *configurationSerializer*:

p:modulesConfigurationPath="**/modules-configuration.xml**"

Путь указывается относительно папки *cm-sochi\modules\src\main\resources\config.*

Ниже приведен пример регистрации файлов конфигурации модуля расширения:

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<modules-configuration xmlns="https://cm5.intertrust.ru/modules-configuration">

<module-configuration>

<path>/test-module/custom-config.xml</path>

<schema-path>/test-module/custom-configuration.xsd</schema-path>

</module-configuration>

<module-configuration>

<path>/test-module/domain-objects.xml</path>

</module-configuration>

</modules-configuration>

В элементе *path* указывается путь к файлу конфигурации (относительно папки *cm-sochi\modules\src\main\resources\config*). В элементе schema-path указывается (опционально) схема для валидации этого файла конфигурации. Если путь к схеме не указан, используется схема файлов конфигурации ядра.

### **Некоторые детали реализации загрузки ядром файлов конфигурации модулей расширения**

Для того, чтобы файлы конфигурации модулей расширения были доступны для загрузки загрузчиком классов ядра, они копируются во время сборки модуля *web-app* из модуля *modules-configuration* в папку */WEB-INF/classes/modules-configuration*. Для этого в *pom.xml* модуля *web-app* настроен соответствующий плагин. Далее для спринг-бина *configurationSerializer* в свойстве *modulesConfigurationFolder* указывается абсолютный путь (относительно модуля *web-app*) к папке, где лежат файлы конфигурации модулей расширения, т. е. к папке *WEB-INF/classes/modules-configuration*:

p:modulesConfigurationFolder="**modules-configuration**"

Для валидации файла регистрации файлов конфигурации модулей расширения *modules-configuration.xml* используется схема *modules-configuration.xsd*. Путь к данной схеме также является настраиваемым и указывается как абсолютный путь (в модуле *configuration*) в свойстве *modulesConfigurationSchemaPath* спринг-бина *configurationSerializer*:

p:modulesConfigurationSchemaPath="**config/modules-configuration.xsd**"