# Модуль генерации тестовых множеств доменных объектов.

## Общая информация

Модуль генерации тестовых множеств доменных объектов представляет собой отдельный Java-bean. Команды модулю отправляются удаленно с помощью примитивного Java-приложения. Это приложение осуществляет чтение данных из xml-файла, содержащего инструкции по созданию тестовых множеств, получает удаленный доступ к bean’у на сервере jboss с помощью jndi-механизма и вызывает его рабочий метод, передавая в качестве аргумента содержимое xml. Модуль осуществляет разбор инструкций и, в случае отсутствия в них ошибок, создает множество (или множества) объектов с указанными характеристиками, используя вызовы интерфейсных методов ядра системы. Модуль возвращает краткий отчет о времени выполнения и успешно созданных объектах и, в случае ошибок в процессе выполнения, информацию об ошибках.

## Краткое описание функционала

Предполагается, что модуль будет выполнять следующие действия: создание указанного числа (точного или произвольного из диапазона) доменных объектов заданного типа и заполнение их полей значениями (также точными или из диапазона). Если поле содержит ссылку на объект, возможно создание объекта требуемого типа на лету с последующим заполнением поля ссылкой на него.

## Структура xml-инструкций, используемых для управления.

set – базовый элемент, инструкция создать множество объектов.

Атрибуты:

* type – тип объекта, должен быть одним из типов, уже зарегистрированных в системе. Обязателен.
* id – идентификатор множества. Используется, когда объекты другого множества должны ссылаться на объекты этого множества.
* quantity – количество объектов, которое нужно создать. Используется, когда необходимо указать точное количество объектов.
* maxQuantity, minQuantity – максимальное и минимальное количество объектов, которое требуется создать. Будет создано случайное количество объектов из диапазона. Обязательны, если quantity отсутствует.

set допускает единственный вложенный элемент – object. object допускает следующие вложенные элементы:

* string – строковое поле. Поддерживает следующие атрибуты:
* name – имя поля. Имя поля должно совпадать с именем одного из полей типа указанного в атрибуте domain-object-type элемента set.
* length, maxLength, minLength – по аналогии с quantity, maxQuantity, minQuantity - точная длина строки, которая будет присвоена в качестве значения, либо диапазон допустимых длин. Длина в любом случае ограничивается максимальной длиной зарегистрированного поля.
* value – фиксированное значение, которое будет присвоено всем объектам множества (атрибуты ограничения длины будут проигнорированы).
* maxValue, minValue – максимальное и минимальное допустимые значения (будет использовано сравнение строк).

dateTime – поле, содержащее дату и время. Поддерживает атрибуты:

* name – аналогично string.
* value – аналогично string, если значение не удастся распознать как дату, оно будет проигнорировано. Это относится ко всем атрибутам, ограничивающим значение не-string полей.
* maxValue, minValue – аналогично string.
* relativeMaxValue, relativeMinValue - максимальное и минимальное допустимые смещения относительно текущей даты.

long – поле, содержащее целое число. Поддерживает атрибуты name, value, minValue, maxValue.

decimal – поле, содержащее десятичную дробь с фиксированным количеством знаков до и после запятой. Поддерживает атрибуты name, value, minValue, maxValue.

reference – поле, содержащее ссылку на другой доменный объект системы в виде его уникального идентификатора. Поддерживает атрибуты:

* name – аналогично string.
* type - тип объекта, должен быть одним из типов, уже зарегистрированных в системе. В качестве значения поля будет присвоен идентификатор произвольного объекта данного типа.
* setId – идентификатор уже созданного множества. В качестве значения поля будет присвоен идентификатор произвольного объекта данного множества.
* parentReference – булево значение (по умолчанию - false), определяет, должно ли поле содержать родительский объект (см. backreference).
* treeReference – булево значение (по умолчанию - false), определяет, должно ли поле использоваться для ссылки на следующий объект в цепи (см. treeDepth).
* treeDepth – глубина дерева. Если reference содержит этот атрибут, будет создана цепь объектов длиной n, где n – значение атрибута. Предыдущий объект будет ссылаться на следующий посредством поля reference, в котором объявлен этот атрибут.
* minTreeDepth, maxTreeDepth – максимальное и минимальное допустимые возможной глубины дерева. Аналогичны другим атрибутам диапазона.

reference может содержать вложенный элемент object. В этом случае будет создан объект данного типа, а ссылка на него помещена в reference.

backReference – псевдополе. В результате обработки этого элемента будет создан доменный объект, ссылающийся на текущий. Поддерживает атрибуты:

* quantity – количество объектов, которое нужно создать. Используется, когда необходимо указать точное количество объектов.
* maxQuantity, minQuantity – максимальное и минимальное количество объектов, которое требуется создать. Будет создано случайное количество объектов из диапазона. Обязательны, если quantity отсутствует.
* treeReference – булево значение (по умолчанию - false), определяет, должно ли поле использоваться для ссылки на предыдущий объект в цепи (см. treeDepth).
* treeDepth – глубина дерева. Если reference содержит этот атрибут, будет создано дерево объектов глубиной n, где n – значение атрибута. Следующий объект будет ссылаться на предыдущий посредством поля reference, в котором объявлен этот атрибут.
* minTreeDepth, maxTreeDepth – максимальное и минимальное допустимые возможной глубины дерева. Аналогичны другим атрибутам диапазона.

backreference может содержать вложенный элемент object. В этом случае будет создан объект данного типа, а ссылка на него помещена в reference. Во вложенном object должен присутствовать один элемент reference с атрибутом parentReference, установленным в true.

## Пример xml с командами создания объектов.

<set quantity="10000" id="genericIncoming">

<object type="Incoming\_Document">

<string name="Registration\_Number" length="10"/>

<string name="Description" minLength="0" maxLength="20"/>

<dateTime name="Registration\_Date" minValue="01.01.2010" maxValue="31.12.2013"/>

<reference name="Author" type="Employee" />

<reference name="Attachment">

<object type="Attachment>

<string name="Attachment\_path" minLength="10" maxLength="100"/>

</object>

</reference>

<backReference minQuantity="6" maxQuantity="8">

<object type="Visa">

<string name="Attachment\_path" minLength="10" maxLength="100"/>

<reference name="parent" parentReference="true"/>

</object>

</backReference>

<backReference minQuantity="0" maxQuantity="3" minTreeDepth="2" maxTreeDepth="3"/>

<long name="Long\_Field" value="0"/>

<long name="Another\_Long\_Field" minValue="0" maxValue="777.777"/>

<decimal name="something" value="88" />

<decimal name="Decimal\_Field" minValue="0" maxValue="777.777" />

</object>

<set>