## Назначение подсистемы AuditLog

Подсистема AuditLog предназначена для хранения всех изменений доменных объектов с целью определения пользователя, который внес изменения в доменный объект, состав и дату этих изменений, а так же возможности восстановления доменного объекта на определенный момент времени.

### Способ реализации

Для хранения всех версий доменных объектов в базе данных создается структура хранения, аналогичная структуре для хранения самих доменных объектов с добавлением дополнительных полей.

На каждый тип доменных объектов создается набор таблиц с префиксом «\_audit» для базового типа и для всех дочерних типов. Состав атрибутов аналогичен основным таблицам, за исключением:

1. Не строится уникальный ключ по идентификатору доменного объекта, вместо него создается дополнительное поле version\_id, где хранится уникальный идентификатор версии, по нему строится уникальный ключ, и этот ключ одинаковый во всех таблицах дочерних типов.
2. Не строится внешний ключ на основные таблицы хранения доменных объектов. Это необходимо для того, чтобы была возможность сохранить аудит после удаления основного доменного объекта.
3. В таблице типа верхнего уровня создается дополнительное поле, которое хранит идентификатор персоны, выполнившей изменения доменного объекта.
4. В таблице типа верхнего уровня создается дополнительное поле, которое хранит описание процесса, выполнившее изменение доменного объекта (в случае изменения объекта с правами system)

Для конфигурации AuditLog в конфигурации доменных объектов добавляется атрибут disableAuditLog в теге domain-object-type. По умолчанию значение атрибута false. Если атрибут не указывать аудит включен.

Реализация AuditLog производится внутри domainObjectDao бине.

Для работы с аудитом из вне создается EJB AuditLogService со следующими методами:

List<VersionDomainObject> getAllVersions(Id domainObjectId) – получение всех версий доменного объекта.

VersionDomainObject getVersion(Id versionId) – получение конкретной версии по известному идентификатору.

void clean(Id domainObjectId) – очистка аудита доменного объекта. Доступен пользователю ч правами администратора

CompareVersionsResult compare(Id baseVersionId, Id comparedVersionId) – получение информации об изменениях в двух разных версиях доменного объекта.

## Описание моделей, используемых при работе с AuditLog.

Класс, описывающий версию - VersionDomainObject.

**package** ru.intertrust.cm.core.business.api.dto;

/\*\*

\* Интерфейс версии доменного объекта

\* **@author** larin

\*

\*/

**public** **interface** VersionDomainObject **extends** DomainObject{

/\*\*

\* Идентификатор версии

\* **@return**

\*/

Id getVersionId();

/\*\*

\* Получение информации о процессе, изменившим доменный объект

\* **@return**

\*/

String getVersionInfo();

/\*\*

\* Идентификатор персоны выполнившей изменение

\* **@return**

\*/

Id getModifier();

}

Класс описывающий разницу между двумя версиями CompareVersionResult

**package** ru.intertrust.cm.core.business.api.dto;

**import** java.util.Date;

**import** java.util.List;

/\*\*

\* Иньерфейс описывающий разницу в версиях доменных объектов

\* **@author** larin

\*

\*/

**public** **interface** CompareVersionResult {

/\*\*

\* Возвращает идентификатор версии, с которой производится сравнение

\* **@return**

\*/

Id getBaseVersionId();

/\*\*

\* Возвращает идентификатор версии, которую сравниваю с базовой

\* **@return**

\*/

Id getComparedVersionId();

/\*\*

\* Возвращает идентификатор доменного объекта

\* **@return**

\*/

Id getId();

/\*\*

\* Получение идентификатора персоны, выполнившей изменения. в случае если

\* изменения производились от имени системы то null

\* **@return**

\*/

Id getModifier();

/\*\*

\* Дата сохранения изменений

\* **@return**

\*/

Date getModifiyDate();

/\*\*

\* информация о процессе, который произвел изменения

\* **@return**

\*/

String getModifyInfo();

/\*\*

\* Информация о изменившихся атрибутах

\* **@return**

\*/

List<ModifiedAttribute> getModifiedAttributes();

}

Класс описывающий изменение одного атрибута ModifiedAttribute

**package** ru.intertrust.cm.core.business.api.dto;

/\*\*

\* Информация о изменившемся атрибуте доменного объекта

\* **@author** larin

\*

\*/

**public** **interface** ModifiedAttribute {

/\*\*

\* Получение имени атрибута

\* **@return**

\*/

String getName();

/\*\*

\* Получение значение атрибута базовой версии

\* **@return**

\*/

<T **extends** Value> T getBaseValue();

/\*\*

\* Получение значения атрибута сравниваемой версии

\* **@return**

\*/

<T **extends** Value> T getComparedValue();

}