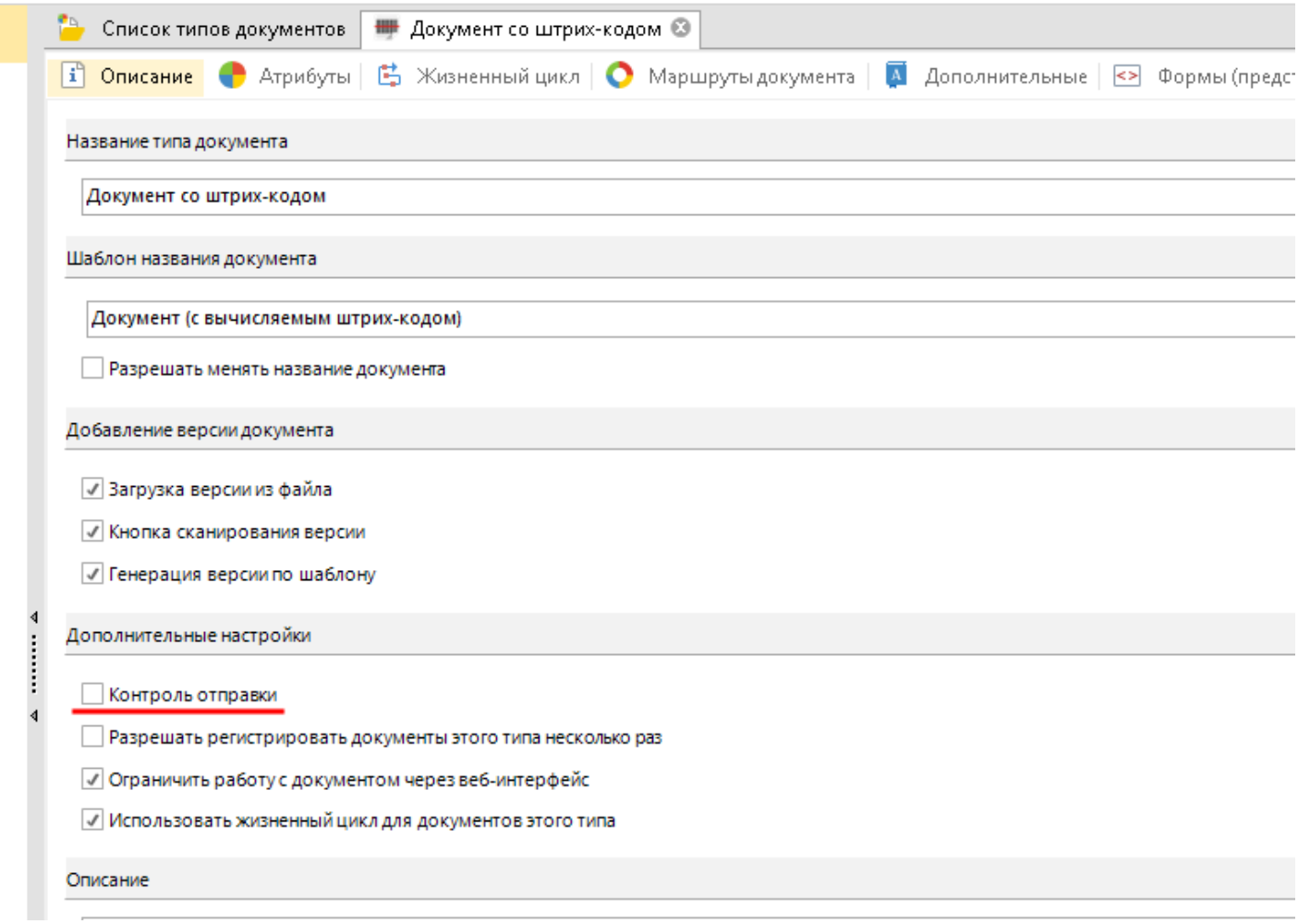
Удаление у документа существующих атрибутов

Для атрибутов документов (объектов), созданных в Дизайнере при их удалении из модули, фактического удаления не происходит. Более того, возможны варианты, когда атрибуты являются системными и их не возможно удалить. На примере признака документа "Контроль отправки" рассмотрим вариант как вернуть все обратно.



Необходимо реализовать процесс с ручным запуском, сделать скрипт со следующим кодом:

```
public virtual void Process(Context context)
{
  // 744 - пример Id нужного нам типа документа из базы
  var mh = MetadataItemHeaderManager.Instance.Load(744);
```

```
  var m = mh.Current; //.Published;
  var metadata = (DocumentMetadata)m.Metadata;

  Logger.Log.ErrorFormat("Start update metadata for {0}", metadata.Name);

  metadata.ControlOnSending = false;

  var sendStatusProperty = LinqUtils.NameOf<ISendableDocument>(d => d.SendStatus);
  RemoveSendProperty(sendStatusProperty, metadata);

  var sendDateProperty = LinqUtils.NameOf<ISendableDocument>(d => d.SendDate);
  RemoveSendProperty(sendDateProperty, metadata);

  var senderProperty = LinqUtils.NameOf<ISendableDocument>(d => d.SendUser);
  RemoveSendProperty(senderProperty, metadata);

  var sendDocumentTypeProperty = LinqUtils.NameOf<ISendableDocument>(d => d.SendDocumentType);
  RemoveSendProperty(sendDocumentTypeProperty, metadata);

  var sendNumberProperty = LinqUtils.NameOf<ISendableDocument>(d => d.SendNumber);
  RemoveSendProperty(sendNumberProperty, metadata);

  var sendScanCopyProperty = LinqUtils.NameOf<ISendableDocument>(d => d.SendScanCopy);
  RemoveSendProperty(sendScanCopyProperty, metadata);

  m.Metadata = metadata;
  m.Save();
  mh.IsDirtyItem = true;
  mh.Save();
  var mm = new ModelManager();
  mm.Publish("Fix control on send", mh.Id);

  Logger.Log.ErrorFormat("Metadata for {0} published", metadata.Name);
}

private void RemoveSendProperty(string propertyName, DocumentMetadata metadata)
{
  var allProperties = metadata.Properties; //MetadataLoader.GetBaseProperties(metadata);
  Logger.Log.ErrorFormat("Props count: {0}", allProperties.Count);
  var toRemove = allProperties.Where(p1 => p1.Name == propertyName && p1.IsSystem).ToList();
```

```
  foreach (var p in toRemove)
  {
    Logger.Log.ErrorFormat("Removed: {0}", p.Name);
    metadata.Properties.Remove(p);
  }
}
```

Порядок применения:

- Опубликовать процесс
- Запустить и дождаться его завершения
- Перезапустить сервер ELMA

> Обязательно сделайте резервную копию базы данных.

Версия #2
Вербовик Максим Арнольдович создал 2022-10-24 11:17:04 MSK
Вербовик Максим Арнольдович обновил 2022-10-24 11:28:02 MSK