

# Поля заголовков

Архива поддерживает гибкое определение полей заголовков почтового сообщения. Под полями почтового сообщения подразумеваются такие заголовки как От кого, Кому, Тема и другие (англ. To, From, Subject). При редактировании файла с определениями полей можно изменить, какие из заголовков будут индексироваться, какие будут видны, экспортированы и проч. После того как вы изменили определение поля заголовка, необходимо перезагрузить сервер Архива. Система пока ещё не в состоянии загружать изменения в процессе изменения.

Чтобы изменить определения поля заголовка, скопируйте файл:

```
[main application]\server\webapps\ROOT\WEB-INF\classes\config_category_email.xml
```

Для нахождения точного месторасположения [main application и настройки] обратитесь в [Разделы файловой системы](#).

В директорию [настройки]. Для Linux это будет /etc/opt/mailarchiva/ROOT

Отредактируйте скопированный файл config\_category\_email.xml с использованием текстового редактора Notepad или VI.

## Спецификации поля

Внутри файла XML указаны все связанные с почтовым сообщением поля. К примеру, вот как выглядит опеределение поля в файле:

```
<Field name="to" type="emailaddress" resource="field_label_to">
  <Use usage="export" />
  <Use usage="archiverule" />
  <Use usage="stubrle" />
  <Use usage="searchquery" />
  <Use usage="retention" />
  <Use usage="searchresults" param="colsize=20" />
  <Use usage="index_tokenized_and_stored" param="name=to" />
  <Use usage="retention" />
  <Use usage="advancedheaderview" />
  <Use usage="basicheaderview" />

</Field>
```

Определение в XML требует объяснения:

Элемент	Пример	Описание
name	to	имя поля
type	emailaddress	тип поля
resource	field_label_to	имя ресурса в файле application.properties
use		для чего используется это поле
usage	retention	разрешено указывать в правилах сохранения

## Применение

Следующие способы применения определяют, для чего поле заголовка может быть использовано, например, для индексации, для возврата в результатах поиска и т.д.

Применение	Описание
export	Позволяет экспортировать поле
basicheaderview	Отображает поле в стандартном просмотре заголовка
advancedheaderview	Отображает поле в развернутом просмотре заголовка
archiverule	Позволяет использовать поле в правилах архивирования
searchquery	Позволяет использовать поле в поисковых запросах

retention	Позволяет использовать поле в определении правил сохранения
searchresults	Отображает поле в результатах поиска
index_tokenized	Помечает и индексирует поле в индексе поиска
index_stored	Хранит и индексирует поле в индексе поиска
index_tokenized_and_stored	Помечает, хранит и индексирует поле в индексе поиска
ldap	Возвращает из ldap [в данный момент не поддерживается]
download	Позволяет скачивать поле
viewsource	Позволяет демонстрацию поля при просмотре источника
print	Показывает поле в просмотре для печати
view	Отображает поле в основном просмотре
stubrulerule	Позволяет показывать поле в правиле стаббинга

## Типы

Данные типы данных определяют, как данные в поле заголовка хранятся в индексе и как эти данные демонстрируются пользователю.

Тип данных	Описание
string	текстовая строка
integer	целое число
long	длинное целое число
float	число с плавающей точкой
double	число двойной точности
boolean	true/false
date	дата
emailaddress	почтовый адрес
size	размер в kb/mb/gb/tb

## Составные поля

Составные поля - это те, что содержат в себе несколько других полей. Примером составного поля является anyaddress. Он определяется, как показано ниже:

```
<Field name="anyaddress" type="composite" resource="field_label_anyaddress">
  <FieldRef name="to" />
  <FieldRef name="from" />
  <FieldRef name="cc" />
  <FieldRef name="bcc" />
  <FieldRef name="deliveredto" />
  <FieldRef name="sender" />
  <FieldRef name="recipient" />
  <!-- <FieldRef name="rcptto" />
  <FieldRef name="mailfrom" /> -->
  <FieldRef name="journalrecipients" />
  <FieldRef name="journalsender" />
  <FieldRef name="journalto" />
  <FieldRef name="journalcc" />
  <FieldRef name="journalbcc" />
  <Use usage="archiverule" />
  <Use usage="stubrule" />
  <Use usage="searchquery" />
  <Use usage="retention" />
</Field>
```

## Пользовательские заголовки

Используя динамические поля, можно настроить Архива на поддержку любого пользовательского почтового заголовка. К примеру, рассылка вашей компании устанавливает заголовок MIME, называющийся `X-Originating-IP`. Когда динамическое поле `X-Originating-IP` будет добавлено в файл настройки полей пользователя (в файле [шаблона config\\_category\\_email.xml](#) расположенный в разделе файловой системы с [назначением настройки](#)), Архива EE возвратит содержимое заголовка и применит его по назначению. Для применения для применения пользовательских заголовков система должны быть перезагружена.

К примеру, нам необходимо добавить новое поле `X-Originating-IP`.

Для этого необходимо:

- скачать [шаблон](#) определения пользовательских полей
- поместить данный файл в [раздел файловой системы](#) с назначением настройки
- отредактировать данный файл и добавить нужные поля
- перезапустить Архива EE

К примеру, это динамическое поле определено следующим образом:

```
<Field name="xoriginatingip" type="boolean" resource="X_Originating_IP">
  <Use usage="property" param="name=X-Originating-IP" />
  <Use usage="export" />
  <Use usage="archiverule" />
  <Use usage="contentfilter" />
  <Use usage="stubrule" />
  <Use usage="searchquery" />
  <Use usage="retention" />
  <Use usage="index_tokenized" param="name=xoriginatingip" />
  <Use usage="retention" />
  <Use usage="advancedheaderview" />
  <Use usage="basicheaderview" />
</Field>
```

Имя поля должно совпадать с именем заголовка MIME.

#### config\_category\_email.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Herein custom fields are defined -->
<!--type - whether field is obtained from email or ldap {email|ldap} -->
<!--name - email or ldap attribute name [string value] -->
<!--resource -resource key name in application.properties [string value] -->
<!-- metadata - additional field properties (e.g. colsize=30) -->
<!--Архива Fields -->

<Category name="email" searchable="true">

<Fields version="1.0">

<Field name="xoriginatingip" type="boolean" resource="X_Originating_IP">
  <Use usage="property" param="name=X-Originating-IP" />
  <Use usage="export" />
  <Use usage="archiverule" />
  <Use usage="contentfilter" />
  <Use usage="stubrule" />
  <Use usage="searchquery" />
  <Use usage="retention" />
  <Use usage="index_tokenized" param="name=xoriginatingip" />
  <Use usage="retention" />
  <Use usage="advancedheaderview" />
  <Use usage="basicheaderview" />
</Field>

</Fields>

</Category>
```

Более того, строка:

```
field_label_X_Originating_IP=X Originating IP
```

может быть добавлена в для правильного отображения в пользовательском интерфейсе. Данное действие не обязательно.

```
[main application]\server\webapps\ROOT\WEB-INF\classes\properties\application.properties
```

После этого необходимо перезагрузить Архива. В примере выше поле `X-Originating-IP` должно быть проиндексировано и сохранено. Поле должно появиться в выпадающем списке доступных полей для поиска.

Метка сноски поля определяет поле, на которое указывает поле-контейнер `anyaddress`. Поэтому, когда кто-то будет искать письма с помощью `anyaddress`, по полям, содержащимся в поле-контейнере, тоже будет произведен поиск.