

Компоненты приложения

Сервис обогащения контактов из email

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

Сервис обогащения контактов из email	4
Процесс обогащения контакта/контрагента	4
Системные настройки	6
Последовательность идентификации	7
Хэширование информации	7

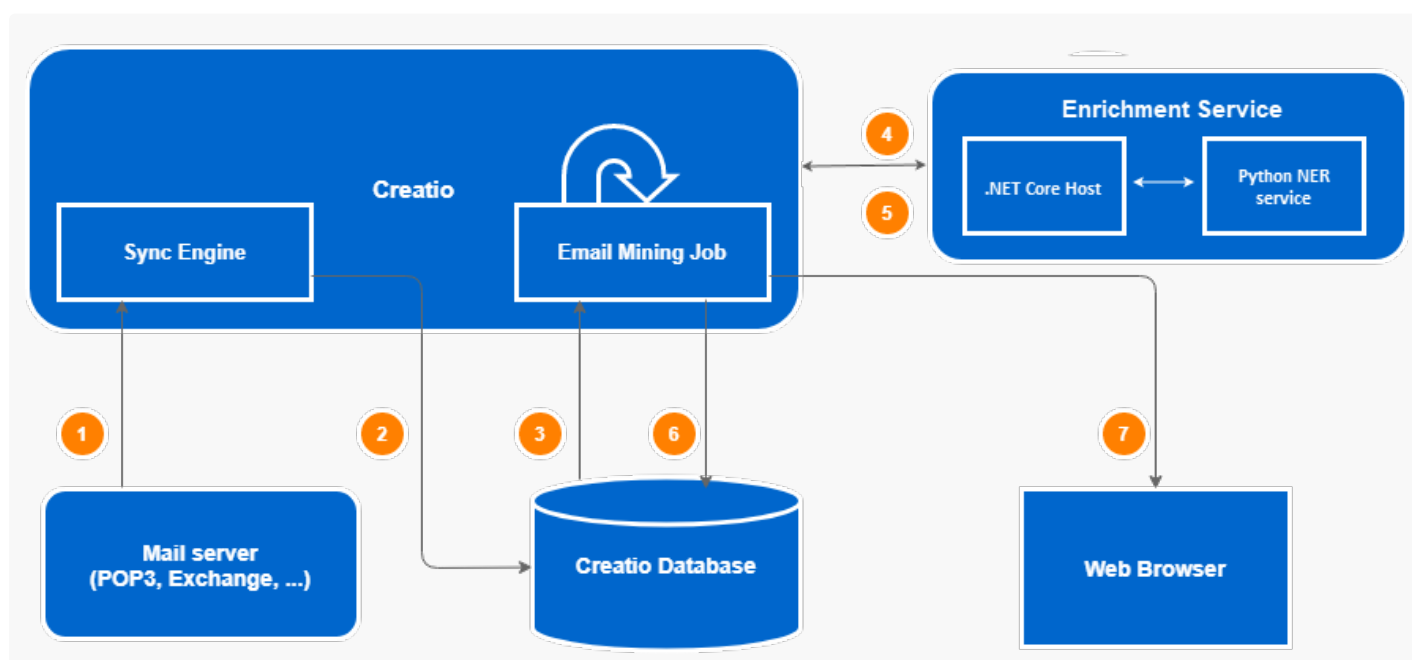
Сервис обогащения контактов из email



В версии 7.10.0 появилась функциональность обогащения контактных данных информацией из email. Основная задача функциональности — обнаружение в письмах информации, которой можно обогатить данные контактов/контрагентов.

Процесс обогащения контакта/контрагента

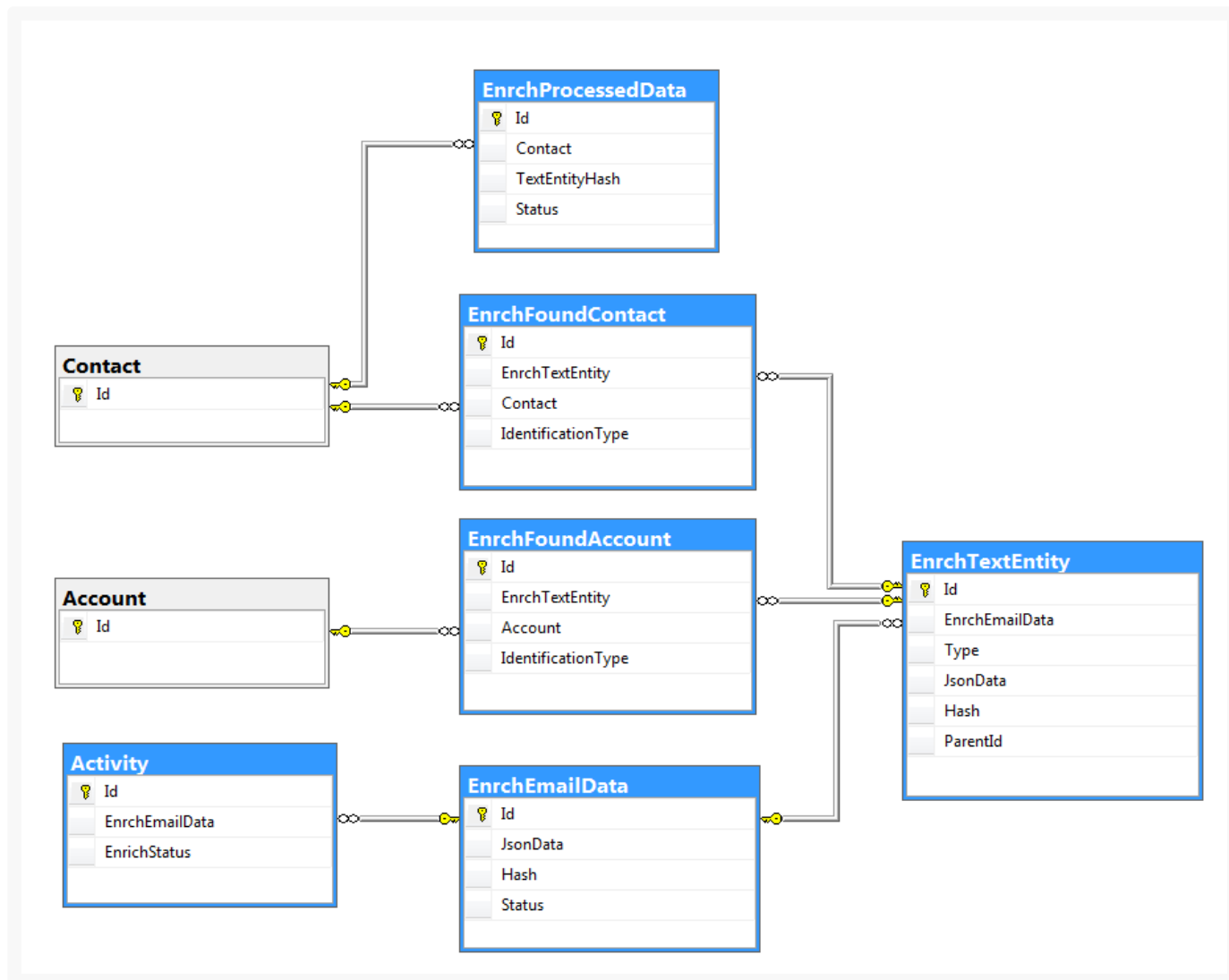
Процесс обогащения контакта/контрагента:



Основные этапы процесса обогащения контакта информацией из email:

1. Существующий [механизм синхронизации Sync Engine](#) производит синхронизацию с почтовым сервером. Почтовый сервер передает Sync Engine новые письма.
2. Sync Engine сохраняет полученные письма в базе данных в виде активностей с типом `Email`.
3. Планировщик Creatio периодически выполняет задание, которое запускает процесс `Email Mining Job`. Этот процесс выбирает из базы данных порцию последних (по дате создания) активностей с типом `Email`, которые ранее не были им обработаны. Из каждой записи активности выбирается тело письма и его формат (plain-текст или html).
4. Процесс `Email Mining Job` по каждому выбранному письму отправляет http-запрос в облачный сервис обогащения данных `Enrichment Service`.
5. `Enrichment Service` выполняет следующие операции:
 - выделяет из письма цепочку отдельных сообщений (ответов);

- для каждого сообщения выделяет подпись (signature);
 - из подписи выделяет сущности (entity extraction) — контакт (ФИО, должность), телефоны, email- и web-адреса, социальные сети, другие средства связи, адреса, название организации. Эти данные `Enrichment Service` возвращает в http-ответе в виде определенной структуры в формате JSON.
6. Процесс `Email Mining Job` разбирает полученную от сервиса структуру и сохраняет ее в сыром виде в таблицах Creatio.



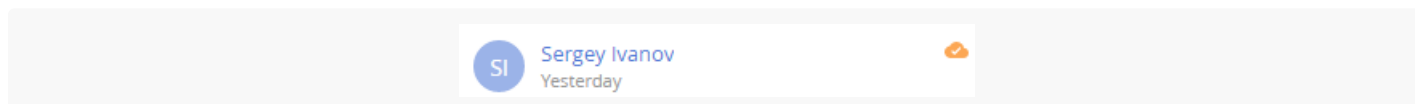
Основное назначение таблиц:

- `[EnrchTextEntity]` — хранит информацию об одной сущности, выделенной из письма. Поле `Type` определяет тип этой сущности (контакт, коммуникация, адрес, организация и т.п.). Сами данные хранятся в формате JSON в поле `JsonData`.
- `[EnrchEmailData]` — определяет набор информации для обогащения, выделенный из одного письма.
- `[EnrchFoundContact]` — контакт в Creatio, идентифицированный по выделенным из email данным. Хранит ссылку на контакт Creatio и `[EnrchTextEntity]` типа *Контакт*.
- `[EnrchFoundAccount]` — аналогично таблице `[EnrchFoundContact]` хранит информацию об

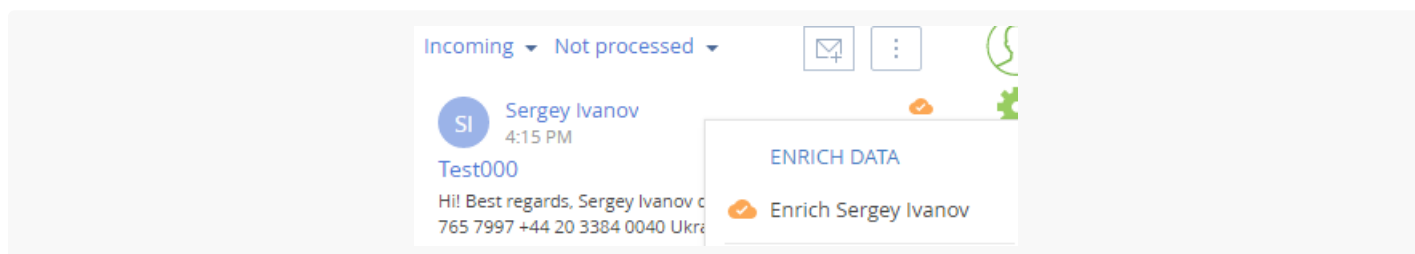
идентифицированном контрагенте Creatio.

- `[Activity]` — в существующую таблицу активностей добавлены поля, которые отображают связь между активностями типа `Email` и объектами `[EnrchEmailData]` с текущим статусом процесса выделения информации.
- `[EnrchProcessedData]` — содержит информацию об уже обработанных данных, принятых либо отвергнутых пользователем в процессе обогащения.

7. Процесс `Email Mining Job` уведомляет о завершении извлечения информации из письма. Сообщения отправляются по каналам websocket пользователям, которые в коммуникационной панели видят обрабатываемые письма. Если в письме есть информация, которой можно обогатить связанный контакт, либо на основании которой можно создать новый контакт, то в интерфейсе приложения в правом верхнем углу письма отображается соответствующий значок.



Для такого письма доступно действие, которое позволит обогатить или создать новый контакт системы.



Системные настройки

Системные настройки процесса обогащения:

- `TextParsingService` — адрес cloud-сервиса обогащения данных `Enrichment Service`. Для клиентов on-demand заполняется автоматически. Обязательна для заполнения.
- `CloudServicesAPIKey` — ключ для доступа к API cloud-сервиса. Для клиентов on-demand заполняется автоматически. Обязательна для заполнения.
- `EmailMiningPackageSize` — количество обрабатываемых за один раз писем. Процесс `Email Mining Job` при каждом запуске будет обрабатывать столько писем, сколько указано в системной настройке. Значение по умолчанию — 10.
- `EmailMiningPeriodMin` — периодичность (в минутах) запуска задания `Email Mining Job`.

Важно. Если значение `EmailMiningPeriodMin` меньше или равно нулю, то задание не будет запланировано и функциональность будет отключена. Для повторного включения необходимо установить значение настройки ≥ 1 , перезапустить application pool приложения Creatio, после чего перейти на страницу логина и войти в приложение.

- `EmailMiningIdentificationActualPeriod` — период актуальности (в днях) идентификации контактов/контрагентов. Если по истечении указанного срока по ранее идентифицированному контакту будет обработано новое письмо, то идентификация будет произведена повторно.

Последовательность идентификации

Идентификация контактов

1. Поиск по ФИО.
2. Поиск по фамилии и имени.
3. Поиск по email-адресам. В расчет принимаются только те email-адреса, которые не принадлежат бесплатным или временным почтовым сервисам.
4. Поиск по телефонам. Поиск происходит только по последним цифрам телефонных номеров контакта.

Если на каком-то из этапов идентификации обнаружено совпадение данных, то процесс идентификации будет остановлен.

Идентификация контрагентов

1. Поиск по колонке [*Название*] или [*Альтернативное название*] без учета регистра символов.
2. Поиск по web-адресу.
3. Поиск по домену из email-адресов. Учитываются только email-адреса, которые не принадлежат бесплатным или временным почтовым сервисам. Из email-адреса выделяется домен и выполняется поиск средств связи контрагента по фильтру “начинается с” одного из представлений домена: `http://<домен>`, `https://<домен>`, `http://www.<домен>`, `https://www.<домен>`, `www.<домен>`, `<домен>`.

Если на каком-то из этапов идентификации обнаружено совпадение данных, то процесс идентификации будет остановлен.

Хэширование информации

Извлеченная из письма информация [хэшируется](#). В результате в таблицах `[EnrchTextEntity]` и `[EnrchEmailData]` в поле `[Hash]` записывается строковое значение хэша, которое однозначно идентифицирует данную единицу или набор извлеченных данных в системе. Это позволяет реализовать два важных улучшения — экономию ресурсов при повторной идентификации контактов/контрагентов из набора извлеченной информации и группировку полученной информации для контакта.

Повторная идентификация контактов/контрагентов

Например, в систему поступило письмо, в подписи которого указан контакт с ФИО “Иванов Иван Иванович”, телефоном “123-45-67” и адресом “ул. Пушкина, 47, оф. 3”. Для текущего набора данных на основании его содержимого система рассчитала хэш “`Hash1`” и записала его в поле `[Hash]` таблицы `[EnrchEmailData]`. Идентификация контакта выявила в системе контакт “Иванов Иван” и записала полученный результат в таблицу `[EnrchFoundContact]`.

Через некоторое время в систему поступило еще одно письмо с подписью, в которой упоминается контакт "Иванов Иван Иванович" с теми же телефоном и адресом. Система рассчитала для текущего набора данных такой же хэш, как и в прошлый раз — "Hash1", т.к. входящие данные хэширования не изменились. Вместо того, чтобы создавать новые записи в таблицах [EnrchEmailData], [EnrchTextEntity] и повторно идентифицировать этот контакт, система нашла по полю [Hash] ранее созданную запись в таблице [EnrchEmailData] по полю [Hash] и записала ссылку на эту запись в таблице [Activity].

Таким образом экономится объем хранящихся данных и, что более важно, не производятся ресурсоемкие запросы идентификации контакта.

Группировка выделенной информации для контакта

Поскольку каждая единица выделенной информации [EnrchTextEntity] имеет хэш-код, основанный на ее содержимом, при обогащении данных существующего контакта появляется возможность использовать информацию, найденную во всех письмах, в которых он участвовал. При выборке данных для обогащения они группируются по полю [Hash] и не дублируются.