Подсистема поиска

# Область применения

К понятию «поиска» можно отнести несколько различных функций системы CM 4/5:

1. поиск по полю (фильтрация) в списке документов;
2. простой поиск (по строке);
3. расширенный поиск (по атрибутам).

## Поиск по полю в списке

В интерфейсе системы списки документов представляются таблицами. При этом в заголовках тех полей (столбцов), по которым поддерживается поиск, присутствует специальная кнопка, открывающая поле для ввода фильтра. При вводе значения в это поле осуществляется повторный запрос списка документов, учитывающий – дополнительно к основным условиям отбора – введённый фильтр.

Списки документов в интерфейсе системы настраиваются с помощью специальных конфигурируемых сущностей – коллекций. Существует 2 варианта настройки коллекций: (1 – generic) указанием специального Java-класса, осуществляющего выборку объектов и (2 – specific) написанием запросов на псевдо-SQL (DOSQL), преобразуемым в настоящий SQL ядром системы. В обоих случаях поддерживается встраивание в запросы фильтров, которые могут модифицировать результирующий запрос. Эта возможность может быть использована для организации фильтров списка по значениям полей. Правда, в случае конфигурации коллекции через DOSQL написание запроса и всех фильтров к нему может быть сложной и трудоёмкой задачей.

Описанный выше подход оставляет фильтрацию списков за пределами ответственности подсистемы поиска. Его, однако, нужно будет пересмотреть, если для фильтрации необходимы функции, реализуемые исключительно подсистемой поиска – например, учёт морфологии русского языка.

## Простой поиск

Строка для поиска документов присутствует на каждой странице системы – в шапке интерфейса. Тем не менее, при поиске учитывается настраиваемый параметр контекстной страницы – область поиска.

Область поиска определяет множество документов – доменных объектов на основании их типа, статуса и, возможно, других полей, а также набор полей (включая поля связанных объектов), в которых производится поиск заданной строки. Одним из важных требований к простому поиску является возможность поиска текста во вложенных файлах различных форматов (MS Word, Open Office, plain text и др.). Следует отметить, что результатом поиска является список документов – ключевых (корневых) объектов иерархии, олицетворяющих собой бизнес-сущности, а не доменных объектов, непосредственно содержащих удовлетворяющее поисковому запросу поле.

Области поиска также должно быть сопоставлено представление коллекции (collection-view), которое позволит отобразить результаты поиска в виде таблицы.

## Расширенный поиск

Расширенный поиск во многом схож с простым, но добавляет некоторые возможности. Ссылка на форму расширенного поиска присутствует рядом со строкой простого поиска.

Во-первых, расширенный поиск позволяет выбрать любую область поиска независимо от контекста (хотя именно контекст определяет выбор по умолчанию), и даже несколько таковых одновременно. Набор полей формы расширенного поиска зависит от выбранной(ых) пользователем области(ей). В случае выбора нескольких областей поиска форма содержит только поля, присутствующие во всех выбранных областях. Возможность выбора несовместимых областей поиска блокируется.

Каждое поле формы поиска позволяет наложить ограничение на искомые документы по содержанию их конкретных полей. Тип фильтра при этом зависит от типа и семантики поля документа: для строковых полей фильтр будет содержать также строку для поиска по словам или по вхождению, для полей дат – диапазон дат, для полей, ссылающихся на справочники – набор подходящих значений этого же справочника. Также, как правило, присутствует поле для поиска текста в файлах вложений. Кроме того, форма расширенного поиска содержит поле «Искать везде» – аналог строки в простом поиске.

Наконец, форма расширенного поиска позволяет выбрать одно из нескольких возможных представлений коллекции найденных документов и ограничить размер выборки.

## Области поиска

В форме расширенного поиска CM 4.x присутствует группа флажков под названием «Область поиска». Однако, реальные области поиска определяются сочетанием выбранных значений из этой группы со значением переключателя «Искомый объект». Именно это сочетание задаёт набор полей, по которым осуществляется поиск. Оно же определяет состав полей, возвращаемых для искомых объектов. Разумеется, не каждое сочетание области поиска и искомого объекта соответствует реальной области поиска, и выбор таких сочетаний блокируется пользовательским интерфейсом.

Поскольку переключатель «Искомый объект» позволяет выбрать только один вариант, это гарантирует, что возвращаемые поиском документы (доменные объекты) являются однотипными, независимо от того, сколько выбрано разных областей поиска. Даже для поиска собственно документов в различных областях (например, «Входящие документы», «ОРД») искомым объектом является «РКК» – единый для разных типов документов. Это будет работать и в CM 4/5, где РКК является базовым (родительским) типом для различных типов доменных объектов, представляющих входящие, исходящие и др. документы.

# Архитектура

## Apache Solr

Для реализации поиска в системе используется мощная поисковая платформа с открытым кодом Apache Solr. Она позволяет организовать эффективные поисковые индексы для быстрого выполнения поисковых запросов по большим массивам данных, поиск текста с учётом морфологии русского языка, индексирование файлов в различных форматах (включая MS Word, Open Office, Adobe PDF, HTML и др.). Solr может быть встроен в ядро системы, либо запускаться как отдельное приложение на том же сервере приложений или на другом, причём этот сервер может как на том же компьютере, так и на отдельном (выделенном) или даже кластере – решение остаётся за заказчиком и зависит от его потребностей и возможностей.

Подсистема поиска является, фактически, мостом (bridge) между Apache Solr и остальной частью системы CM 4/5. Она обеспечивает поставку данных для индексации в Solr, формирование поисковых запросов к этой платформе и интерпретацию результатов поиска в коллекции объектов. Для взаимодействия с Solr используется Java-библиотека SolrJ.

При разработке подсистемы, помимо её кода и необходимой конфигурации, должны быть разработаны требования и рекомендации по настройке Apache Solr. В этих рекомендациях должна быть особым образом отмечена необходимость ограничения доступа к Solr извне, так как он имеет доступ к значительной части данных системы, но не обеспечивает собственной аутентификации или авторизации пользователей.

## Организация поисковых индексов

Для упрощения процедуры обновления поисковых индексов они организуются в соответствии со структурой доменных объектов системы: одному документу Solr соответствует один доменный объект системы. Это особенно важно для доменных объектов вложений, индексация которых может требовать передачи большого объёма данных. В специальном поле документа Solr сохраняется идентификатор доменного объекта системы. Кроме того, ещё в одном поле должен быть сохранён идентификатор документа системы, т.е. главного (корневого) доменного объекта иерархии. Это поле – ключевое для построения коллекции в ответ на поисковый запрос. Для эффективной организации поиска по областям идентификатор области поиска также должен сохраняться в специальном поле документа Solr.

Помимо перечисленных служебных полей, в поисковые индексы Solr включаются поля доменных объектов, по которым может осуществляться поиск. Также могут быть сконфигурированы индексы (поля Solr), вычисляемые по содержимому полей связанных объектов (например, фамилия подписанта документа). Однако, автоматическое отслеживание изменений этих объектов для обновления индекса исходного объекта в настоящий момент не предусмотрено – в частности, потому, что это может потребовать слишком большого объёма вычислений. Поэтому такие поля должны использоваться только для редко изменяемых объектов (связанных) – таких, как объекты организационно-штатной структуры. Переиндексация

Следует отметить, что Solr может быть настроен на хранение либо только индексов полей, либо также их содержимого. Второй вариант может существенно увеличить объём данных, хранимых Solr, но позволит отображать в коллекции контекст найденных слов. Выбор варианта конфигурации может быть оставлен на усмотрение администратора системы.

## Конфигурация

Базовым элементом конфигурации подсистемы поиска является определение областей поиска. Для каждой области поиска указываются типы доменных объектов, которые в неё попадают, а также набор полей, подлежащих индексированию.

Кроме того, необходима привязка области поиска к отображаемой в интерфейсе странице.

## Графический интерфейс пользователя

Архитектура компонентов GUI, необходимых для организации поиска, не рассматривается в данном документе. Тем не менее, необходимо упомянуть несколько моментов, важных для организации взаимодействия подсистемы поиска с web-клиентом системы.

Важной особенностью GUI системы CM 4/5 является способность отображения динамических форм, состав полей которых определяется не кодом системы, а внешними данными – конфигурацией. Механизм динамических форм, использующийся для просмотра и изменения доменных объектов системы, может быть использован и для формирования поисковых запросов (расширенный поиск). С точки зрения архитектуры подсистемы поиска, это означает необходимость обеспечения интерфейсной общности между возможными подлежащими объектами динамических форм – доменным объектом и поисковым запросом, – чтобы любой из них мог использоваться как источник и приёмник данных формы.

# Компоненты

## Сервис поиска

Сервис поиска является EJB и входит в публичный API системы. Он содержит методы для выполнения простых и расширенных поисковых запросов. Все методы возвращают коллекцию IdentifiableObjects, каждый из которых представляет документ системы. Идентификатор объекта – это идентификатор главного доменного объекта, соответствующего документу, и он содержит все поля, определённые в конфигурации области поиска. Нужно подчеркнуть, что независимо от того, где был найден искомый текст, в результатах поиска будут содержаться только идентификаторы главных объектов, а вот поля могут заполняться как из главного, так и из связанных с ним объектов. Если запросу поиска удовлетворяют несколько доменных объектов, связанных с одним главным, то он попадёт в результирующую коллекцию только один раз.

Сервис поиска формирует поисковый запрос, передаёт его Apache Solr, получает от него список найденных документов Solr, группирует их в соответствии с описанием области поиска, делает выборку коллекции из доменных объектов (фильтруя их по правам пользователя) и передаёт её клиенту.

Все методы сервиса поиска позволяют ограничить размер возвращаемой коллекции объектов. Поиск выполняется в синхронном режиме.

## Агент обновления поисковых индексов

Компонент, отвечающий за актуальность поисковых индексов Solr, реагирует на события изменения (включая создание и удаление) доменных объектов системы. Он подключает свой код через точку расширения на сохранении и удалении доменных объектов. Проверив по конфигурации областей поиска, что сохраняемый объект подлежит индексации, компонент отправляет на сервер Solr запрос, содержащий все необходимые поля для добавления или обновления документа в индексе.

Предмет особого внимания этого компонента – доменные объекты вложений. Главной ценностью этих объектов является содержимое приложенного к ним файла, которое должно быть проиндексировано Solr. Компонент обеспечивает передачу в Solr ссылки на файл в случае их расположения на одном сервере и содержимого файла при использовании выделенного сервера Solr.

Индексация документов Solr осуществляется исключительно в асинхронном режиме.

## Процесс переиндексации

Возможны ситуации, когда поисковые индексы Solr не соответствуют доменным объектам в БД:

* изменилась конфигурация областей поиска;
* изменился объект, поля которого используются в индексах связанных объектов;
* текущая индексация объекта не произошла по какой-либо причине;
* поисковый индекс потерян или испорчен из-за аппаратных сбоев.

В таких случаях требуется переиндексация всех или некоторых доменных объектов.

Эту задачу выполняет специальный компонент, осуществляющий выборку доменных объектов по заданным критериям, и передающий их для индексации в Solr. Компонент работает под управлением подсистемы запуска периодических задач.