

Предварительная структура виртуальной файловой системы EDS

Для пользователей сети и сетевых приложений в основном интересен интерфейс, который предоставляет сетевая файловая служба, а не ее внутреннее устройство. В данной системе файловая структура представляется в виде классической иерархической файловой системы с наличием папок и файлов.

Папки и файлы в системе могут иметь такие атрибуты, как:

- имя файла (папки);
- описание файла (папки);
- размер файла в байтах;
- дата создания;
- дата изменения;
- дата открытия.

Система поддерживает такие файловые функции, как:

- создание папки;
- удаление/переименование папки;
- перемещение поддерева иерархии в иную ветку дерева (если нет, заблокированных файлов или папок);
- блокировки открытых файлов;
- произвольный доступ для чтения данных из файловых потоков;
- перемещение указателя по файлу;
- возможность произвольного доступа к файлу при записи данных;
- возможность произвольного доступа к файлу при транзакционной записи данных;
- расширенный поиск;
- управление семантической структурой данных с помощью механизма связанных метаданных.

Количество файлов и папок в данной системе может достигать 2^{63} и ограничено только размером физического носителя. Причем время поиска и иных операций с метаданными возрастает логарифмически по причине индексации файловой структуры. Индексация производится сервером реляционных баз данных с помощью алгоритма Б+-деревьев, что позволяет все операции по управлению структурой файловой системы проводить с помощью Sql запросов.

Одним из преимуществ данной файловой системы является использование поименованных файловых потоков (рис. 2), что позволяет в одном файле хранить структурированную информацию для различных приложений. Каждый файловый поток может достигать размера 2^{63} , что позволяет убрать ограничения присущие другим файловым системам. Обращение к файловому потоку происходит по типу **имя файла\$имя файлового потока**. Если обращение происходит без прямого указания файлового потока, то по умолчанию используется файловый поток **\$main**.



Рис 1. Пример использования точек монтирования.

Использование точек монтирования (рис. 1), которые для пользователя системы представляются в виде обычных папок, позволяет производить файловые операции не только с данными текущего сервера, но и с данными других серверов EDS или иных сетевых файловых систем. Все данные по подключению к другому серверу хранятся в обычном файловом потоке

\$mount файла, который имеет атрибут ATTR_MOUNT_HOLE. Для систем EDS имя пользователя и его пароль берется из текущего, для иных хранится в файловом потоке точки монтирования.

Семантическая структура файловой системы

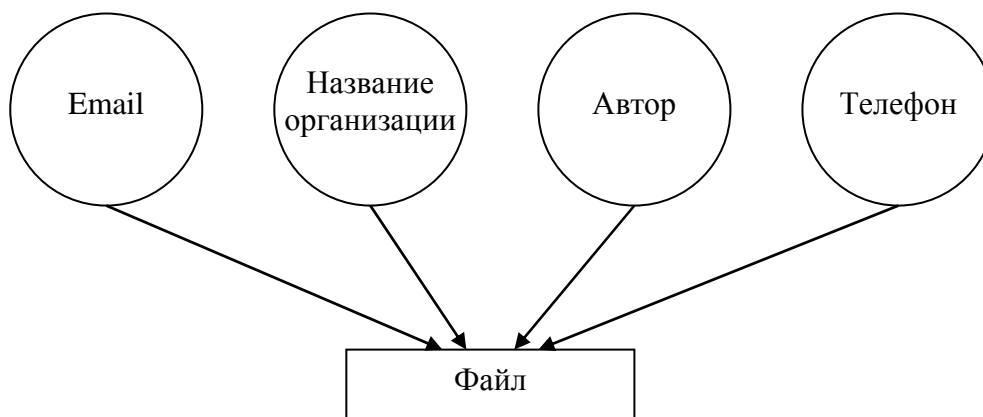


Рис 2. Пример семантических связей в файловой системе.

Создание семантической структуры (рис. 2) в данной файловой системе является уникальной особенностью, которую нам может позволить реляционный механизм сервера. С каждым файлом могут быть связаны метаданные определенного типа, которые позволяют нам отбирать файлы при поиске, к примеру, по Email автора, что дает несомненное преимущество при сортировке данных по семантической составляющей. Категории семантической метаданных могут добавляться сторонними программами при необходимости. Кроме того, каждый элемент данных метаданных может иметь свой собственный тип. С одним файлом может быть связано как бесконечное количество метаданных, так и ни одного.

Разграничение прав доступа пользователей к файловой системе при выполнении файловых операций

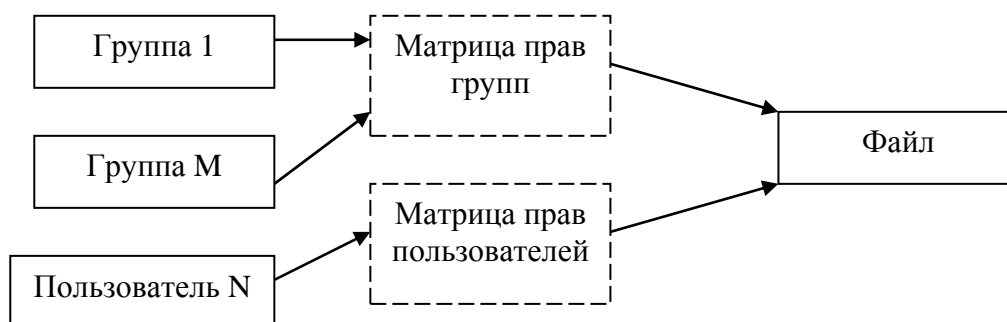


Рис 3. Пример разграничения прав доступа пользователей к файловой системе.

Механизм разграничения прав доступа в системе EDS (рис. 3) выгодно отличается от стандартной системы разграничения прав в *nix системах. Механизм разграничения прав доступа в данной системе напоминает механизм, используемый в файловой системе NTFS, и базируется на назначении прав доступа отдельным пользователям или группам пользователей системы.

Количество, как отдельных пользователей, так и групп имеющих права на отдельный файл в данной системе не ограничено, что позволяет построить расширенную матрицу прав. При этом скорость файловых операций, практически не страдает по причине индексации в базе данных таблицы выделенных прав.

В данной системе пользователи делятся на две категории, стандартные и супервизоры. Супервизор имеет права на любое действие с файлами или с файловой системой вне зависимости от матрицы прав. Стандартный пользователь может иметь такие атрибуты, как:

- login или имя пользователя для входа, которое включает только латинские литеры, цифры или символы подчеркивания;

- пароль;

- описание, которое может включать любые символы;

- также пользователь может принадлежать какой-либо группе.

Атрибуты группы включают:

- имя группы, которое включает только латинские литеры, цифры или символы подчеркивания;

- описание, которое может включать любые символы.

Матрица прав, одинаковая по структуре, как для пользователей, так и для групп включает в себя такие права:

- полный доступ к файлу и возможность передачи своих прав на файл/папку другому пользователю;

- право на создание папки/файла в определенной папке;

- право на удаление;

- право на чтение;

- право на запись.

Структура реляционной базы данных, сетевой файловой системы

Рассмотрим структуру реляционной базы данных, сетевой файловой системы EDS (рис. 4). Структура БД позволяет реализовать все свойства данной файловой системы, указанные выше и является хранилищем данных системы. Структура базы данных обеспечивается совокупностью 9 таблиц. Таблицы METADATA_AUTHOR, METADATA_AUTHOR_BIRTHDAY и METADATA_EMAIL введены для иллюстрации семантического структурирования файловой структуры.

Таблица MAIN_SYSTEM используется для записи системной информации, как версия базы данных и размер кластера файлового потока. В ней поля:

- ID – является уникальным идентификатором записи системной информации;
- ARGUMENT – является символьным идентификатором системного поля;
- VALUE – является значение системного поля.

Таблица IFILES является основной таблицей базы. В ней отображена иерархическая структура файловой системы. Разберем ее структуру подробнее по полям:

- ID - является уникальным идентификатором файла в системе;
- FILENAME – имя файла;
- FILEDESCRIPTION- описание файла;
- FLINK – вторичный ключ, указывающий на родительскую запись иерархии. Начальный каталог ссылается сам на себя;
- FRAGMENTATION_SIZE – размер фрагментации файла в байтах. Необходим для указания размера кластеров из которых состоят файловые потоки;
- SIZE – размер файла в байтах, который включает в себя размеры всех файловых потоков;
- CREATED – дата создания файла;
- MODIFIED – дата модификации файла;
- OPENED – дата открытия файла;

- ATTR_DIR – флаг того, что файл является папкой;
- ATTR_MOUNT_HOLE - флаг того, что файл является точкой монтирования.

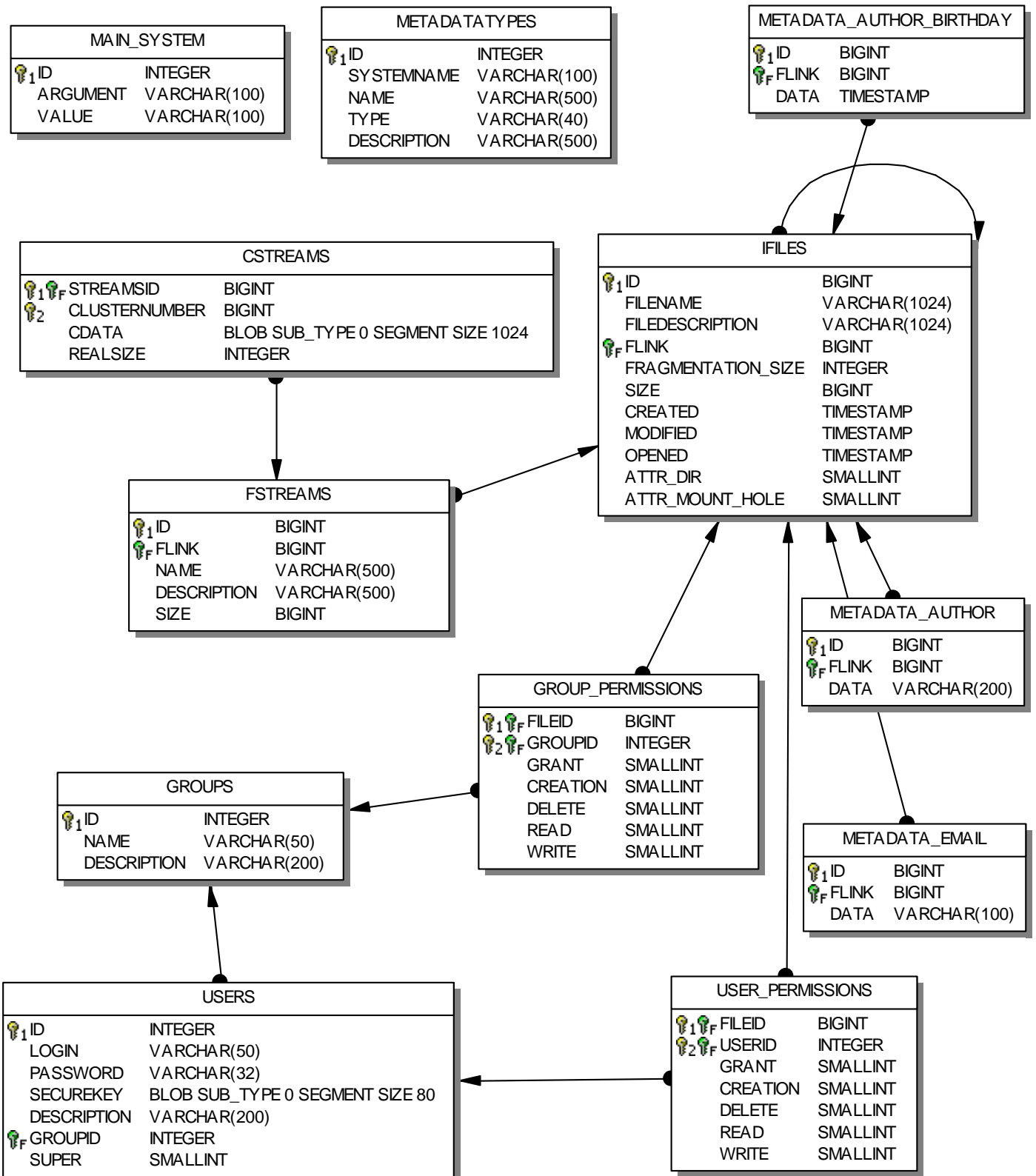


Рис. 4. Структура реляционной базы данных сетевой файловой системы.

Таблица FSTREAMS хранит информацию о файловых потоках и непосредственно ссылается на таблицу IFILES. Поля в ней:

- ID – является уникальным идентификатором файлового потока в системе;

- FLINK – вторичный ключ, указывающий на файл, которому он принадлежит;
- NAME – имя поименованного файлового потока;
- DESCRIPTION – описание поименованного файлового потока;
- SIZE – размер поименованного файлового потока.

Таблица CSTREAMS является таблицей, в которой описаны кластера файловых потоков:

- STREAMSID – вторичный ключ указывающий на файловый поток, к которому принадлежит данный кластер;
- CLUSTERNUMBER – номер кластера в последовательности записей файлового потока, вместе с STREAMSID образует первичный ключ;
- CDATA – данные кластера в виде Blob поля;
- REALSIZE – размер данных.

Таблицы USERS, GROUPS, GROUP_PERMISSIONS, USER_PERMISSIONS означают таблицу пользователей, групп, прав групп и прав пользователей соответственно. Данные таблицы используются для разграничения доступа к файловой системе.

Таблица USERS содержит такие поля:

- ID - уникальный идентификатор пользователя в системе;
- LOGIN – запись логина пользователя (имени пользователя для входа в систему);
- PASSWORD – пароль пользователя;
- SECUREKEY – ключ шифрации данных (не используется в данной версии протокола);
- DESCRIPTION – описание пользователя;
- GROUPID – вторичный ключ, идентификатор группы, к которой принадлежит пользователь;
- SUPER – флаг, выставляющий права пользователя как супервизора.

Таблица GROUPS содержит такие поля:

- ID - уникальный идентификатор группы в системе;
- NAME – название группы;
- DESCRIPTION – описание группы.

Таблица GROUP_PERMISSIONS содержит такие поля:

- FILEID – вторичный ключ записи прав доступа к определенному файлу в системе, который ссылается на уникальный идентификатор файла в системе;
- GROUPID – вторичный ключ записи прав доступа к определенному файлу в системе, которые ссылается на уникальный идентификатор группы в системе. Вместе с FILEID образуют уникальный идентификатор записи прав доступа.
- GRANT – флаг полного доступа к файлу и возможность передачи своих прав на файл другому пользователю;
- CREATION – право на создание папки/файла в определенной папке;
- DELETE – право на удаление;
- READ – право на чтение;
- WRITE – право на запись.

Таблица USER_PERMISSIONS содержит такие поля:

- FILEID – вторичный ключ записи прав доступа к определенному файлу в системе, который ссылается на уникальный идентификатор файла в системе;
- USERID – вторичный ключ записи прав доступа к определенному файлу в системе, которые ссылается на уникальный идентификатор пользователя в системе. Вместе с FILEID образуют уникальный идентификатор записи прав доступа.
- GRANT – флаг полного доступа к файлу и возможность передачи своих прав на файл другому пользователю;
- CREATION – право на создание папки/файла в определенной папке;
- DELETE – право на удаление;

- READ – право на чтение;
- WRITE – право на запись.

Таблица METADATATYPES содержит информацию о типах используемых метаданных в системе, которые позволяют произвести структуризацию информации по семантическому признаку. Поля этой таблицы содержат:

- ID – уникальный идентификатор типа метаданных в системе;
- SYSTEMNAME – системное имя типа метаданных;
- NAME – альтернативное названия типа метаданных на родном пользователю системы языке;
- TYPE –тип метаданных;
- DESCRIPTION – описание типа метаданных.

Таблица 1. Данные таблицы METADATATYPES.

ID	SYSTEMNAME	NAME	TYPE	DESCRIPTION
1	AUTHOR	Автор	VARCHAR(200)	Имя автора файла
2	AUTHOR_BIRTHDAY	День рождения автора	TIMESTAMP	День рождения автора
3	EMAIL	Адрес электронной почты	VARCHAR(100)	Адрес электронной почты автора файла

В таблице 1 дано показательное использование метаданных. По результатам таблицы 5 были созданы таблицы METADATA_AUTHOR, METADATA_AUTHOR_BIRTHDAY, METADATA_EMAIL, в которых может содержаться информация об авторе, его дне рождения и электронной почте соответственно.

Генерация данных таблиц происходит по добавлению префикса METADATA_ к системному имени типа метаданных и имеет структуру:

- ID – уникальный идентификатор текущей записи, текущего типа метаданных в системе;
- FLINK – ссылка на уникальный идентификатор файла;
- DATA – метаданные.

Зеркальная копия базы данных

Механизм сервера реляционных баз данных Firebird обеспечивает создание зеркальной копии базы данных с помощью механизма создания shadow копий базы данных. При этом расходуется большее количество места на диске, но даже при физическом повреждении носителя с базой восстановление информации из shadow копии не займет много времени. При этом зеркальная копия базы может быть не одна и лежать на сетевых дисках, если требуется очень высокий уровень надежности.