

Описание проекта по сохранению резервных копий конфигурационных файлов.

Мы решили использовать базу данных для сохранения резервных копий наших файлов. Это удобно, потому что СУБД имеет встроенные механизмы поиска и фильтрации по параметрам резервной копии (например, по дате) На рис. 1 представлена ER-модель нашей БД. Каждый день по расписанию делается upload (загрузка) конфигурационных файлов в БД на центральном сервере. В БД есть таблица серверов “Servers”, где хранятся имена серверов. Кратко опишем структуру БД. Всего в БД 3 таблицы: Servers (информация о серверах), Conf_Files (имена файлов и их описание), File_versions (версии файлов и дата backup резервного копирования). Таблица “Servers” (первичный ключ - “Server_id”) связана с таблицей “Conf_Files” по внешнему ключу Server_id. В таблице “Conf_Files” (Primary Key - “File_id”) хранятся имена конфигурационных файлов. С таблицей “Conf_Files” связана таблица “File_versions”. Primary Key - [File_id,Version] (составной). “File_id” является внешним ключом к таблице “Conf_Files” (рис. 1).

Взаимодействие с БД идет через Web-сервер. Предлагается 3-слойная модель взаимодействия. В конце каждого рабочего дня автоматически делается бэкап конфигурационных файлов. В конце каждой недели клиенты связываются с Web-сервером через браузер и делают Upload резервных копий файлов на Web-сервер, расположенный на центральном сервере. Web-сервер в свою очередь записывает содержимое файлов на сервер БД.

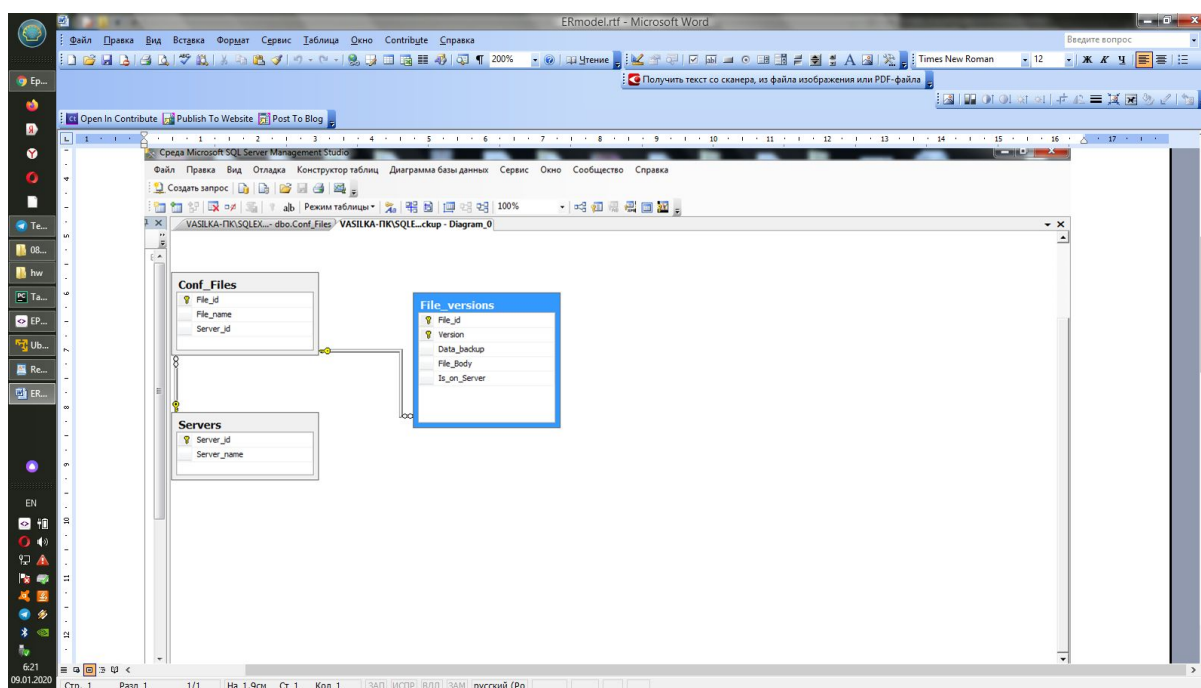


Рис.1

Выбор инструментария

Выбор Framework и языка программирования для Web-сервера зависит от выбора СУБД. Логичным выбором была бы связка Python (Django) и СУБД PostgreSQL. Можно использовать СУБД MySQL. В случае выбора Java, логичным выбором была бы СУБД Oracle или MySQL. В случае использования продуктов линейки Microsoft логично выбрать связку ASP.NET и MS SQL Server. В этом случае вы будете навсегда привязаны к ОС Windows, но в этом есть преимущество, что на этот сервер БД вы можете повесить офисные продукты из коробки: CRM-систему Microsoft CRM, 1С-бухгалтерию, Navision. Я встречал экзотические сочетания: C# + СУБД Oracle, ASP.NET+Firebird, C#+MySQL. С нашей точки зрения таких экзотических сочетаний и пересечений разных линеек технологий лучше избегать. Но иногда так исторически складывается, что клиентской и серверной частью занимаются разные люди с разными предпочтениями технологий. . ,