

Создание репозитория QSAR моделей процесса поиска  
зависимостей структура-свойство.

В процессе научных исследований очень часто требуется проведение каких-либо расчетов. Основной целью которых ставится получение результата по их завершению. После получения результатов все внимание обычно уделяется им: их анализируют, сравнивают, на их основе делают прогнозы и производят дальнейшие вычисления. А инструмент, с помощью которого они были получены, остается в стороне. Он обычно хранится у конечного исполнителя и, как правило, такие инструменты даже не прикрепляют к отчетам по исследованию. Действительно, зачем нам пробирка в которой был проведен опыт если важно только ее содержимое? С течением времени о них все забывают, они теряются и изменяются. И вот, спустя какое-то время, требуется повторить такой расчет или опыт, но иногда результаты текущего и предыдущего экспериментов не совпадают. Возможно, затем даже строится теория, для опровержения предыдущей. Вся эта цепочка возникла из-за того, что детали проведения расчетов и опытов были утеряны со временем.

Для поиска зависимостей структура-свойство существует множество способов. Как правило они представляют собой цепочку последовательных расчетов, где результат предыдущего является входными данными для следующего. Неточность на любом из этапов расчета может привести к существенным отличиям между конечными результатами. Репозиторий QSAR моделей (в дальнейшем будет использоваться термин система) ставит своей целью сохранение расчетной цепочки целиком, так как она существует сейчас, с возможностью повторения результатов в будущем, а также с возможностью использования не измененного расчета с другими входными параметрами.

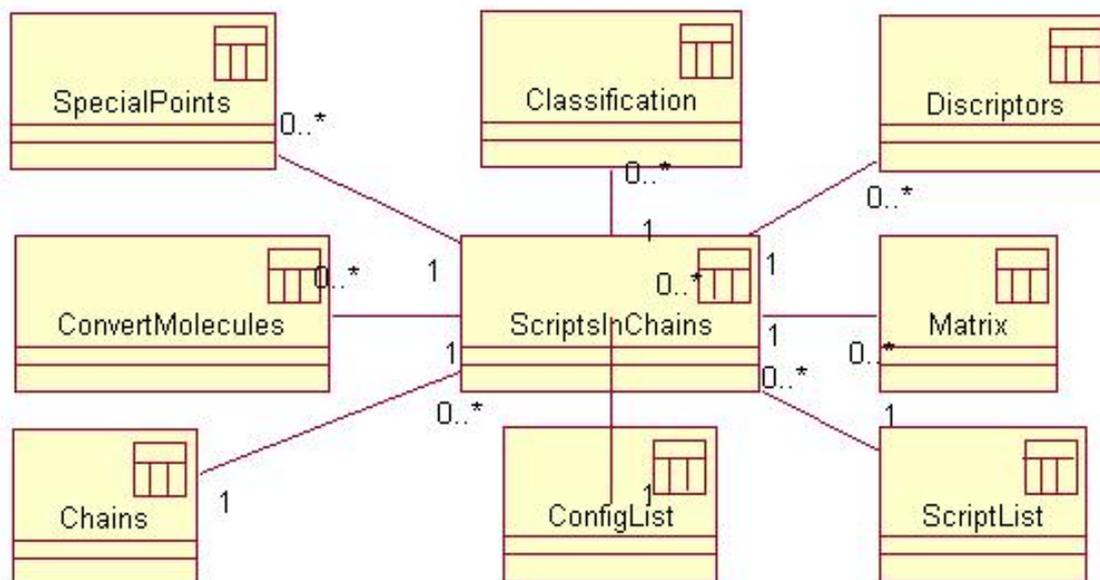
Первый случай (повторение уже совершенных расчетов) может иметь место в случае перепроверки результатов расчетов, полученных от рядового исполнителя, более квалифицированным специалистом.

В случае, если модель поиска зависимостей оказалась удачной, то ее ни в коем случае нельзя потерять, т.к. с помощью ее можно проводить расчеты с другими входными данными. Это как раз вторая функция репозитория.

Дополнительной целью является использование частей существующих расчетов для создания новых.

Для достижения всех трех целей расчеты были разбиты на составные части, причем эти части у всех расчетов были общими. Затем, для сохранения информации о расчетах целиком, была создана модель сущность-

СВЯЗЬ.



1. Предварительная обработка описания молекул на модели соответствует ConvertMolecules.
2. Выделение особых точек на модели соответствует SpecialPoints.
3. Классификация особых точек на модели соответствует Classification.
4. Формирование различных дескрипторов на основе классифицированных особых точек на модели соответствует Discriptors.
5. Построение матрицы по сформированным дескрипторам на модели соответствует Matrix.
6. Цепочки расчетов на модели представлены объектом Chains.
7. За единицу расчета был принят скрипт, которому на модели соответствует ScriptList.
8. Каждый скрипт может иметь несколько изменяемых параметров для запуска, которые хранятся в файле конфигурации, которому на модели соответствует ConfigList.

9. Объект `ScriptsInChains` отвечает за включение скрипта в определенную расчетную цепочку, на определенном ее этапе.

На основе данной модели сущность-связь была создана база данных, для сохранения информации о расчетах. Но т.к. основной целью являлось сохранение самих инструментов расчетов, то было решено использовать для этого систему контроля версий. Такое решение позволяло сохранять различные версии скриптов и файлов конфигурации, не используя лишнего места, т.к. в такой системе целиком сохраняется лишь оригинальная версия, а остальные представляют собой изменения относительно предыдущей версии.

Таким образом, сохраняя каждый скрипт и файл конфигурации отдельно при помощи версионного контроля, была получена возможность комбинировать уже существующие скрипты и параметры их запуска для создания новых цепочек, а так же обеспечивалось цельное восстановление цепочки при необходимости повторения расчетов.

Для более удобного взаимодействия пользователя с системой было решено создать web-interface с помощью которого пользователь мог бы совершать следующие действия:

1. зарегистрироваться в системе.
2. изменить информацию о себе, внесенную при регистрации в системе.
3. загрузить новый скрипт или файл конфигурации в систему (web-interface должен автоматически поместить данные о загруженном объекте в базу данных, а сам объект в систему версионного контроля).
4. выгружать существующие цепочки из системы контроля версии себе на локальную машину (для проведения расчетов, например).
5. формировать новые цепочки из скриптов и конфигурационных файлов, загруженных в систему.
6. просматривать список пользователей и отправлять им сообщения.
7. изменять статус пользователя.

В рамках разработки web-interface все пользователи были разделены по правам на 3 группы: обычные пользователи, привилегированные пользователи и администраторы системы.

Обычный пользователь мог зарегистрироваться в системе, изменить информацию о себе, выгрузить существующую цепочку, просматривать список пользователей и отправлять им сообщения (например посылать запрос на формирование новой цепочки привилегированным пользователям).

Привилегированный пользователь в дополнение к возможностям обычного пользователя мог формировать новую цепочку, загружать скрипты в систему и переводить обычного пользователя в привилегированные.

Администратор системы мог совершать любые действия, в том числе: подтверждать регистрацию пользователя, переводить обычного пользователя в привилегированные и переводить привилегированного пользователя в администраторы.

Для сохранения этой информации в базу данных была введена дополнительная таблица Users.

Для обеспечения безопасности системы (т.к. систему планировалось устанавливать в местах с повышенной безопасностью) в качестве средства обмена данных с пользователем, в систему был включен четвертый компонент - ftp-сервер, с шифрованием по формату SSL, сертификаты для доступа к которому предоставлялись на основе данных об имени и пароле пользователя из таблицы Users базы данных.

В результате разработки была получена система, обладающая следующими свойствами:

1. обеспечивающая длительное хранение моделей расчетов, которые обладают хорошей прогностической функцией с возможностью их получения из системы для проведения повторных расчетов.
2. организация безопасного доступа к системе и к хранимым в ней данным через сеть Internet.
3. уменьшение объема данных о моделях, хранимых в системе.