

## Трек 1. SAS RTDM, SAS ID

### Взаимосвязи (Обязательная задача, 100 баллов)

<b>Задача</b>	Разработать приложение для анализа метаданных платформы SAS RTDM, с помощью которого можно будет отобразить связи между кампаниями принятия решений, процессами данных.
<b>Расширенное описание задачи</b>	<p>Активное развитие розничного кредитования в Газпромбанке приводит к тому, что необходимо постоянно наращивать темпы доработок рискованных стратегий принятия решения по кредитным заявкам, соответственно, требуется развивать практики DEVOPS, а также автоматизировать некоторые шаги в текущем процессе разработки стратегий принятия решений.</p> <p>Система принятия решений разработана и развивается командой Банка на основе технологий: SAS RTDM (ядро) + СУБД MS SQL Server (операционная БД) + Интеграционный слой на базе Apache Camel. Примерно 75% задач на доработку стратегий связаны с изменениями модели данных системы, то есть с изменениями структур таблиц в операционной БД системы.</p> <p>Для работы с БД в SAS RTDM разработчик создает так называемые процессы данных (data processes) – это шаблон обращения к данным в таблицах операционной БД со стороны приложения SAS RTDM (чтение, запись, обновление) через метаданные таблицы, «зарегистрированной» в SAS. Количество уникальных сущностей, которые используются в стратегии принятия решений – больше 150. Примерно половина из них довольно часто изменяются, так как хранят данные по анкете, заемщику, обогащенную информацию по заемщику из внешних систем. В свою очередь состав этих данных довольно сильно подвержен изменениям из-за постоянно меняющихся процессов кредитования в Банке.</p> <p>При создании нового процесса данных в SAS RTDM создается «образ» таблицы с той структурой таблицы, которая была в БД в момент создания процесса. Если в дальнейшем в ту же таблицу будут добавлены новые столбцы/ изменены типы данных, то существующие процессы данных «не увидят» изменений без ручного обновления. Удаление атрибута или несовместимое изменение типа данных в таблице могут привести к тому, что процесс данных станет неработоспособным, что приведет к падению системы в целом. Один и тот же процесс чтения может использоваться в различных блоках стратегии, то есть «вызываться» несколько десятков раз из самых разных модулей системы. Можно было бы ограничиться правилом: 1 процесс данных – 1 сущность в БД, такой подход позволил бы точно знать, как и к какой таблице обращается приложение, нов таком случае</p>

	<p>несколько разработчиков, работающих над разными задачами, не могли разделить свои доработки. Такой подход быстро привел к тому, что при изменении сущности, на которую «смотрит» процесс данных необходимо подвергать регрессионному тестированию все блоки в стратегии, которые используют данный процесс, что невозможно в силу объемов проверок.</p> <p>В итоге команда разработки системы приняла решение идти по пути «версионирования» процессов данных: в случае изменения метаданных таблиц операционных БД в SAS RTDM создаются новые процессы данных и используются именно там, где в стратегии необходимы эти изменения в структуре данных. Результат принятого подхода – сокращение времени на регресс системы, гарантия отсутствия ошибок и одновременное разрастание репозитория процессов данных.</p> <p>Возникла потребность автоматизировать анализ связей между процессами данных и кампаниями принятия решений, а именно:</p> <ol style="list-style-type: none"> <li>1) Отобразить в виде отчета все связи и пересечения диаграммы-процессы-данные на момент запуска отчета.</li> <li>2) Полезной функцией будет предложение списка неактуальных процессов данных. (например, исходя из того, что процесс не связан ни с одной диаграммой)</li> <li>3) Полезной функцией будет возможность анализа скриптов из очередной доработки, которые изменяют структуры БД (alter table, alter column, drop column, etc.) на предмет их влияния на процессы данных, которые обращаются к изменяемой таблицей с возможностью увидеть, что произойдет после обновления метаданных по таблице (сломается ли какой-либо процесс данных и кампания принятия решения, связанная с ним).</li> </ol>
<b>Проблема</b>	<p>Существующее средство на платформе SAS – инструмент SAS Lineage визуализирует связь между процессом данных и диаграммой, но, к сожалению, не позволяет проанализировать несколько процессов данных одновременно. Учитывая, что сейчас репозиторий процессов данных содержит 690+ процессов и прирастает примерно на 30-40 версий процессов каждый месяц, то перебрать каждый процесс вручную, чтобы проанализировать его связи, не представляется возможным.</p>
<b>Сценарий использования результата</b>	<ol style="list-style-type: none"> <li>1) Разработчик стратегий на регулярной основе (раз в спринт) запускает программу для формирования отчета, получать список неактуальных процессов данных, чтобы удалить их из репозитория.</li> <li>2) В процессе разработки по очередной задаче разработчик запускает программу для формирования отчета о том, не приведут ли изменения, которые он планирует внести в систему к нарушению существующих</li> </ol>

	процессов данных, видит, на какие процессы данных может повлиять его разработка, какие из них используются в кампаниях принятия решений и в каких.
--	--

## Взаимосвязи 2 (Бонусная задача, 50 баллов)

Разработать приложения для анализа параметров объектов SAS ID, с помощью которого можно будет отобразить связи между кампаниями принятия решений, процессами данных

## Шестеренка (Бонусная задача, 50 баллов)

<b>Задача</b>	Разработать приложение модификации системного ресурса JDBC
<b>Расширенное описание задачи</b>	<p>Активное развитие розничного кредитования в Газпромбанке приводит к тому, что необходимо постоянно наращивать темпы стратегий принятия решения по кредитным заявкам, соответственно, требуется развивать практики DEVOPS. В ходе разработки возникает необходимость создавать новые системные ресурсы JDBC или менять настройки у уже существующих</p> <p>Требуется:</p> <ul style="list-style-type: none"> <li>-Разработать программу, которая:             <ol style="list-style-type: none"> <li>1) Выгружает системный ресурс JDBC из SAS DS Design Repository в spk-пакет</li> <li>2) Раскрывает spk-пакет, модифицирует свойства системного ресурса из данного пакета на нужные. Настройки свойств и параметров (строка соединения, учетные данные, настройки thread pool) системного ресурса JDBC должны браться из конфигурационного yaml-файла , который версионизируется в Git.</li> <li>3) Импортирует модифицированный spk-пакет в SAS DS Engine Repository</li> </ol> </li> </ul>

<b>Проблема</b>	В текущей ситуации необходимо менять настройки вручную на каждом сервере метаданных через SAS Management Console. Ввиду большого количества серверов это занимает много времени и возрастает вероятность ошибок в конфигурации.
<b>Желаемый результат</b>	Системный ресурс JDBC в SAS DS Engine создается с нужными свойствами и параметрами (строка соединения, учетные данные, настройки thread pool)
<b>Сценарий использования результата</b>	Разработанная программа будет использоваться для модификации параметров системного ресурса JDBC

## Соединение (Бонусная задача, 50 баллов)

<b>Задача</b>	Разработка приложения для сбора метрик по утилизации activity thread pool и jdbc connection pool
<b>Расширенное описание задачи</b>	<p>Активное развитие розничного кредитования в Газпромбанке приводит к тому, что необходимо постоянно наращивать темпы и количество доработок рискованных стратегий принятия решения по кредитным заявкам, соответственно, требуется развивать практики DEVOPS. Каждый релиз системы проходит в том числе и нагрузочное тестирование. В ходе HT определяется влияние релиза общую производительность системы и, иногда, требуется корректировка в том числе и системных настроек SAS RTDM. В ходе HT и промышленной эксплуатации требуется мониторить утилизацию activity thread pool и jdbc connection pool SAS RTDM.</p> <p>Требуется:</p> <ul style="list-style-type: none"> <li>-Разработать программу на Java, которая будет выступать агентом и собирать метрики, отображаемые на PoolDiagnostics.jsp в Prometheus</li> <li>-на основе собираемых метрик требуется создать дэшборд в Grafana для визуализации</li> </ul>
<b>Проблема</b>	<p>Существующее средство на платформе SAS – PoolDiagnostics.jsp отображает только текущую утилизацию thread pool и не позволяет:</p> <ul style="list-style-type: none"> <li>-вести анализ утилизации с течением времени</li> <li>-визуализировать метрики</li> </ul>

<b>Желаемый результат</b>	<input type="checkbox"/> Программный код Java <input type="checkbox"/> Дэшборд в Grafana
<b>Сценарий использования результата</b>	Дэшборд будет использоваться для мониторинга работы SAS RTDM при нагрузочных тестах и в промышленной эксплуатации. На основе данных в дэшборде будет приниматься решение о корректировке настроек.

## Критерии:

Максимальное количество баллов по каждой задаче:

Взаимосвязи – 100 баллов

Взаимосвязи 2 – 50 баллов

Шестеренка – 50 баллов

Соединение – 50 баллов

## Точность исполнения (общий критерий для каждой задачи):

- Все пункты задачи выполнены полностью – 100% баллов
- Все пункты выполнены, но есть оговорки - 75% баллов и более
- Задача решена частично – 50% баллов
- Задача не выполнена – 0% баллов

## Оригинальность

По каждой задаче жюри выбирает самое оригинальное решение и добавляет 25 баллов (т.е. если команда решила все задачи и каждую оригинально, то она получает дополнительно 100 баллов)