УДК 519.816

В.С. Абатуров

ПРОМЫШЛЕННАЯ АНАЛИТИЧЕСКАЯ ПЛАТФОРМА ИЗВЛЕЧЕНИЯ ЗНАНИЙ ДЛЯ ВСТРАИВАЕМЫХ ПРИЛОЖЕНИЙ

Абатуров Василий Сергеевич, *окончил Санкт-Петербургский государственный электротехнический университет «ЛЭТИ». Аспирант кафедры автоматики и процессов управления Санкт-Петербургского государственного электротехнического университета «ЛЭТИ». Инженер-программист ОАО «Интелтех». Имеет статьи в области теории принятия решений, интеллектуального анализа данных. [e-mail vasilianich@yandex.ru].*

Аннотация

В настоящей работе рассматривается промышленная аналитическая платформа извлечения знаний для встраиваемых приложений на основе системы управления базами данных PostgreSQL. Предложена архитектура аналитической платформы, подчиненной требованиям стандартов SQL/MM и PMML. Описан унифицированный интерфейс управления аналитической платформой. Показана схема формирования основных фаз извлечения знаний: фазы обучения, фазы тестирования и прикладной фазы. Представлена даталогическая модель инфраструктуры аналитической платформы. Приведена схема сценариев взаимодействия с аналитической платформой. Показаны преимущества представленного архитектурного решения.

Ключевые слова: аналитическая платформа, извлечение знаний, унифицированный интерфейс, аналитическая инфраструктура.

Введение

Современные системы управления технологическими процессами требуют обработки и анализа больших объемов информации. Данная проблема давно стала критической в областях, непосредственно связанных с аналитической обработкой данных (Data Mining, искусственный интеллект, системы поддержки принятия решений, техническое зрение, мультимедиатехнологии и др.). В настоящее время рынок аналитических систем экспоненциально развивается. В этом процессе принимают участие такие крупные зарубежные компании, как IBM Cognos, MicroStrategy, Oracle, SAS, Microsoft, а также российские фирмы BaseGroup Labs, «Прогноз» [1].

Тенденция последних лет в развитии аналитических систем заключается в интеграции средств аналитической обработки, алгоритмов извлечения знаний, управления метаданными и визуализации результатов на одной программной аналитической платформе. Реализация технологии промышленной аналитической платформы для встраиваемых приложений связана с решением ряда принципиальных вопросов, к которым относятся: выбор архитектуры, системных интерфей-

сов, обеспечение сервисных возможностей, безопасности, надежности и высокого быстродействия. С внедрением аналитических вычислений в промышленные системы акценты применения все более смещаются к безлюдным технологиям, в которых потребителями аналитических сервисов являются другие вычислительные задачи. В этих условиях использование универсальных средств взаимодействия между разнородными задачами и системами выходит на первый план. В промышленных системах обмен поддерживается не только на уровне данных, но и на уровне моделей алгоритмов обработки данных, поэтому вопросы стандартизации модельного представления алгоритмов представляются не менее важными.

Уровни стандартизации

Существующие стандарты Data Mining затрагивают основные аспекты построения аналитических систем извлечения знаний. Можно выделить три направления. Во-первых, унификация интерфейсов, посредством которых любое приложение может получить доступ к функциональности аналитической платформы. Это направление стандартизации касается объектных языков программирования (CWM Data Mining, JDM) и надстроек над языком SQL (SQL/MM, OLE DB for Data Mining), позволяющих обращаться к инструментарию Data Mining, непосредственно встроенному в реляционную базу данных. Второй аспект стандартизации связан с выработкой единого соглашения по хранению и передаче моделей Data Mining. Основой является язык XML. Сам стандарт носит название PMML (*Predicted Model Markup Language*). И третье направление – общие рекомендации по организации процесса аналитической обработки данных. Это направление в основном покрывается стандартом CRISP-DM (*CRoss Industry Standard Process for Data Mining*).

Среди представленного многообразия стандартов, разработанных различными международными организациями в разное время, согласованной является пара SQL/MM [2] и PMML [3]. Проведенный анализ показал, что стандарт SQL/MM достаточен для разработки архитектуры аналитической платформы, встроенной в реляционную базу данных. Стандарт PMML полностью покрывает представление моделей алгоритмов, кроме того, последняя версия стандарта допускает использование PMML-моделей для информационных целей.

Архитектура аналитической платформы

В настоящей работе рассматривается промышленная аналитическая платформа для встраиваемых приложений, построенная на базе системы управления базами данных (СУБД) *PostgreSQL* [4]. На рисунке 1 представлена архитектура аналитической платформы.

Рассматриваемая аналитическая платформа состоит из аналитической СУБД и вспомогательных внешних вычислительных модулей. Взаимодействие аналитической платформы с приложениями осуществляется с помощью трех независимых интерфейсов:

– *SQL/MM* – расширение языка SQL для управления процессами извлечения знаний;

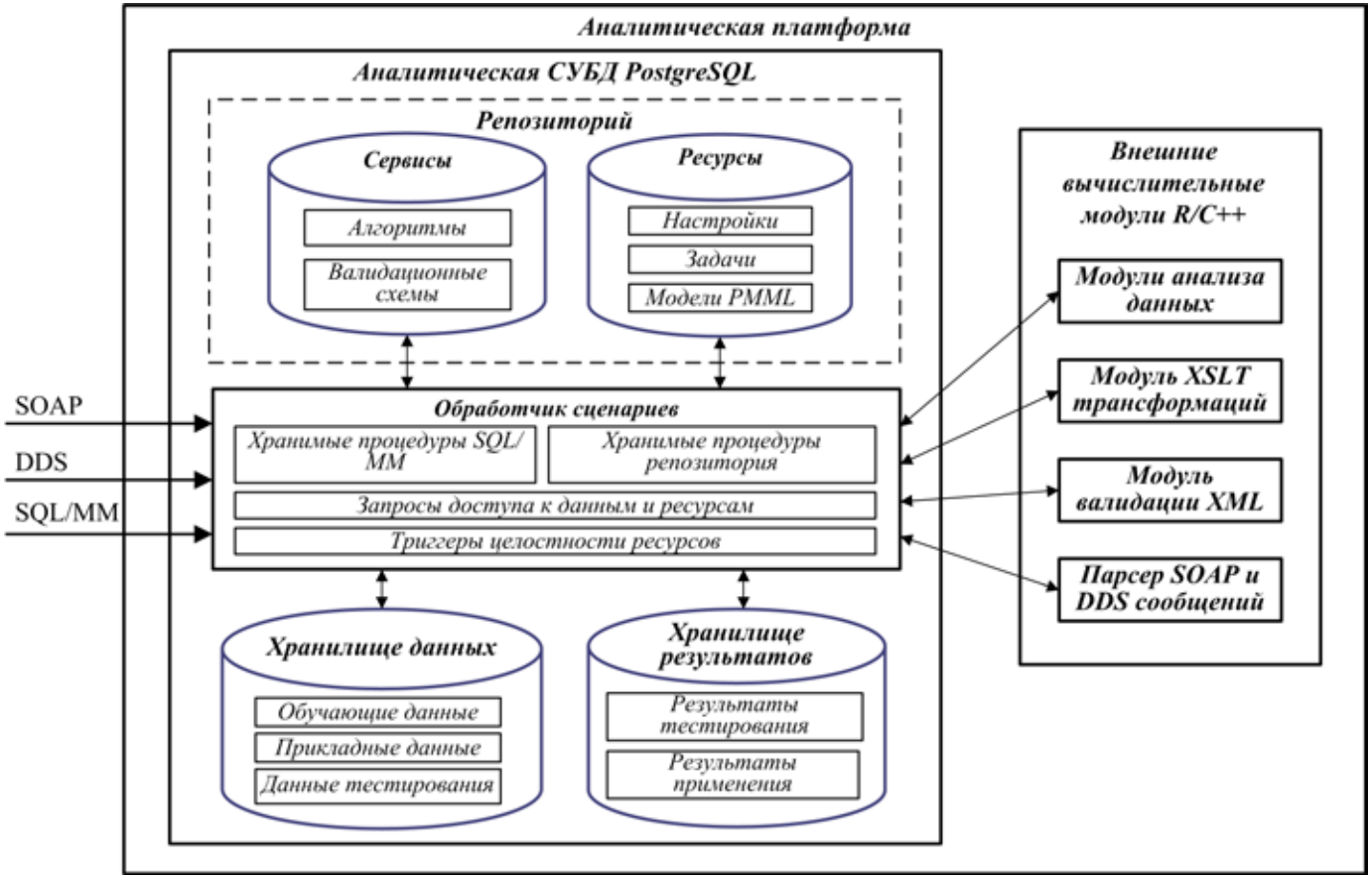

Рис. 1. Архитектура аналитической платформы

– *SOAP* – протокол коммуникаций [5] между интернет-приложениями;

– *DDS* – открытый стандарт [6] распределенного сервиса для систем реального времени.

В качестве аналитической СУБД выступает PostgreSQL. PostgreSQL позволяет создавать новые пользовательские типы данных и хранимые процедуры для выполнения процессов извлечения знаний в соответствии со стандартом SQL/MM.

Внешние вычислительные модули представляют собой динамически подгружаемые библиотеки, написанные на объектно-ориентированном языке C++ и языке R (функциональный язык программирования для статистической обработки данных). К внешним вычислительным модулям относятся:

– *модули анализа данных* – библиотеки, предназначенные для сложных статистических вычислений и процедур анализа данных, разработанные с применением языка R;

– *модуль XSLT-трансформаций* – библиотека, предназначенная для проведения XSLT-трансформаций, лежащих в основе большинства функций SQL/MM;

– *модуль валидации XML* – библиотека, предназначенная для проверки значений SQL/MM-типов, аналитических ресурсов и различных других XML-структур посредством XSD-схем;

– *парсер SOAP и DDS* – библиотека, формирующая SQL/MM-сценарии (на основе SOAP- и DDS-сообщений) и ответные сообщения на основе результатов работы аналитической платформы.

Взаимодействие аналитической СУБД и внешних модулей организовано посредством стандартного интерфейса PostgreSQL для хранимых процедур, написанных на C++, а также с помощью модуля *PL/R*, позволяющего реализовать хранимые процедуры на языке R.

Аналитическая СУБД содержит хранилища, предназначенные для размещения анализируемых данных (*обучающих данных, данных тестирования и прикладных данных*) и результатов использования моделей знаний (*результаты тестирования и результаты применения*).

Важным аспектом промышленных аналитических платформ является возможность централизованного хранения и использования аналитической инфраструктуры. Системное хранилище, поддерживающее инфраструктуру аналитической подсистемы, в дальнейшем будем называть *репозиторием*. Репозиторий представляет собой хранилище аналитических *сервисов* и *ресурсов*. Хранилище сервисов содержит описание всех алгоритмов, имеющихся в аналитической платформе, и всех валидационных схем, необходимых для корректной работы процедур SQL/MM и процедур, связанных с контролем целостности аналитических ресурсов.

Хранилище ресурсов содержит результаты выполнения различных фаз и этапов работы аналитической платформы. К ресурсам относятся:

– *настройки* – XML-описание параметров, предназначенных для вычислительных модулей (результат этапа настройки алгоритма);

– *задачи* – XML-описание, содержащее всю необходимую информацию для запуска этапа непосредственного извлечения знаний;

– *модель PMML* – XML-описание готовой модели знаний.

Стандарт PMML играет ключевую роль в данной архитектуре. Данный стандарт применяется не только для представления и хранения моделей знаний, но и для создания интерфейса сообщений с аналитической платформой через каналы удаленного доступа (SOAP и DDS).

Выполнение процессов извлечения знаний происходит посредством *аналитического обработчика сценариев*. Обработчик сценариев реализован на основе обработчика запросов SQL СУБД PostgreSQL и позволяет не только осуществлять вызовы аналитических процедур и процедур репозитория, но и предоставляет доступ к данным и ресурсам аналитической платформы. Унифицированный доступ осуществляется благодаря стандарту SQL/MM, являющемуся надстройкой над языком запросов SQL. Кроме того, обработчик сценариев оснащен механизмами контроля целостности аналитических ресурсов. Данные механизмы реализованы посредством триггерных функций PostgreSQL.

На случай некорректного использования аналитической платформы предусмотрены *исключения* (*Exceptions*), вызывающие прерывание выполнения сценария и возврат к предшествующему состоянию. Коды и сообщения исключающих ситуаций позволяют выявить причину некорректного использования сценария.

На рисунке 2 в качестве примера представлен SQL/MM-сценарий, предназначенный для определения метаинформации анализируемых данных.

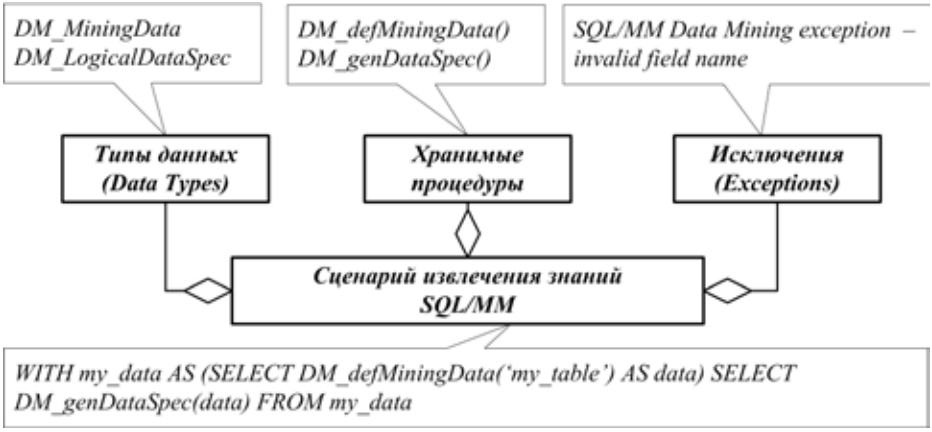

Рис. 2. Сценарий извлечения знаний

Видно, что основными функциональными элементами сценария являются типы данных, хранимые процедуры и исключения. Типы данных представляют собой структурированные единицы информации. Хранимые процедуры используются для выполнения операций над различными типами данных. Исключения служат для представления пользователю информации об ошибках, допущенных при выполнении сценариев. Совокупность данных функциональных элементов позволяет формировать полный набор SQL/MM-сценариев для запуска процессов извлечения знаний. В сценарии, приведенном на рисунке 2, используются хра-

нимые процедуры *DM_defMiningData*() и *DM_getLogicalDataSpec*(), типы данных *DM_MiningData* и *DM_LogicalDataSpec*, а также исключение «*SQL/MM Data Mining exception – invalid table name*», генерируемое в случае неверного имени таблицы данных.

Предложенная архитектура позволяет с помощью хранимых процедур СУБД PostgreSQL выполнять классические этапы извлечения знаний, включающие: обучающую фазу, фазу тестирования и фазу применения. На рисунке 3 представлена схема формирования основных фаз извлечения знаний:

Фаза обучения – фаза интеллектуального анализа данных, на которой строится вычислительная модель PMML.

Фаза тестирования – этап, на котором осуществляется проверка качества предсказания на основе построенной модели.

Прикладная фаза – этап, на котором строка оперативных данных оценивается на основе обученной модели.

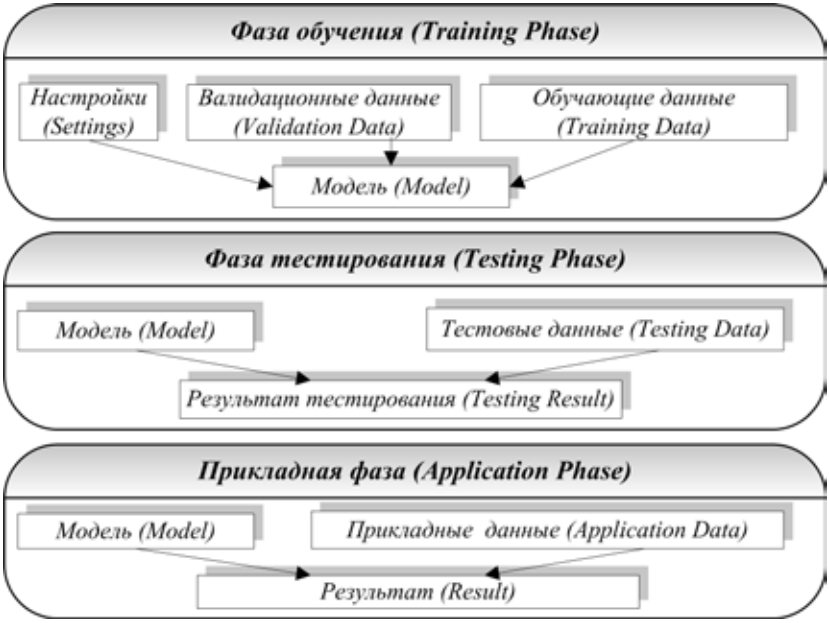

Рис. 3. Фазы извлечения знаний

Инфраструктура аналитической платформы

Стандарт SQL/MM акцентирует внимание на концепции и отдельных алгоритмах извлечения знаний, но не определяет архитектуру аналитической платформы в целом. Тем не менее, предполагается, что для хранения задач, моделей и настроек алгоритмов должны быть определены специальные таблицы в аналитической базе данных. Совокупность системных таблиц, предназначенных для управления

алгоритмами, образует инфраструктуру аналитической подсистемы. Инфраструктура должна обеспечивать:

– доступ к аналитическим ресурсам платформы (задачам, вычислительным моделям и настройкам) и управление ими;

– спецификацию реализованного набора сервисов (описание алгоритмов);

– спецификации допустимых настроек для реализованного набора алгоритмов;

– спецификацию моделей представления знаний;

– спецификации интерфейсных функций стандарта SQL/MM для реализованного набора алгоритмов;

– поддержку целостности ресурсов и сервисов.

Репозиторий представляет собой системное хранилище, поддерживающее инфраструктуру аналитической подсистемы. Цель создания репозитория – обеспечение программного доступа к сервисам и ресурсам встраиваемой аналитической подсистемы. На рисунке 4 представлена даталогическая модель репозитория. Репозиторий состоит из хранилища аналитических ресурсов и сервисов.

Хранилище сервисов представляет собой множество таблиц, предназначенных для описания алгоритмов извлечения знаний, описания схем различных типов SQL/MM, хранения базовых валидационных схем и системных файлов трансформаций. В целом, хранилище сервисов предназначено для повышения качества и эффективности работы аналитической подсистемы. Данное хранилище упрощает работу пользователя и делает прозрачным выполнение процессов извлечения знаний. Названия таблиц хранилища сервисов начинаются с префикса «RP_».

На рисунке 4 видно, что компонент аналитических ресурсов состоит из пяти таблиц:

1. Таблица алгоритмов извлечения знаний (RP_ALGORITHMS).
2. Таблица системных схем (*RP_SYSTEM*).
3. Таблица системных стилей (*RP_XSLT*).
4. Таблица типов SQL/MM (*RP_SQL_MM_Types*).
5. Таблица схем SQL/MM (*RP_SQL_MM_Schema*).

Таблица *RP_ALGORITHMS* предназначена для хранения описания алгоритмов аналитической подсистемы. Данная таблица содержит поля идентификации (id, name), обеспечивающие уникальность алгоритмов, схемы входных и выходных настроек алгоритма, описание методов алгоритма. Таблица *RP_ALGORITHMS* обеспечивает сохранение ресурсов только тех алгоритмов, которые в ней определены.

Таблица *RP_SYSTEM* предназначена для хранения базовых схем, необходимых для работы аналитической платформы. Данная таблица содержит следующие базовые схемы:

– схему представления PMML (*pmml-4-1.xsd*), предназначенную для валидации моделей знаний и используемую в качестве базовой для построения других системных схем;

– схему представления настроек моделей знаний PMML (*pmm_task.xsd*) – базовую для схем настроек моделей знаний;

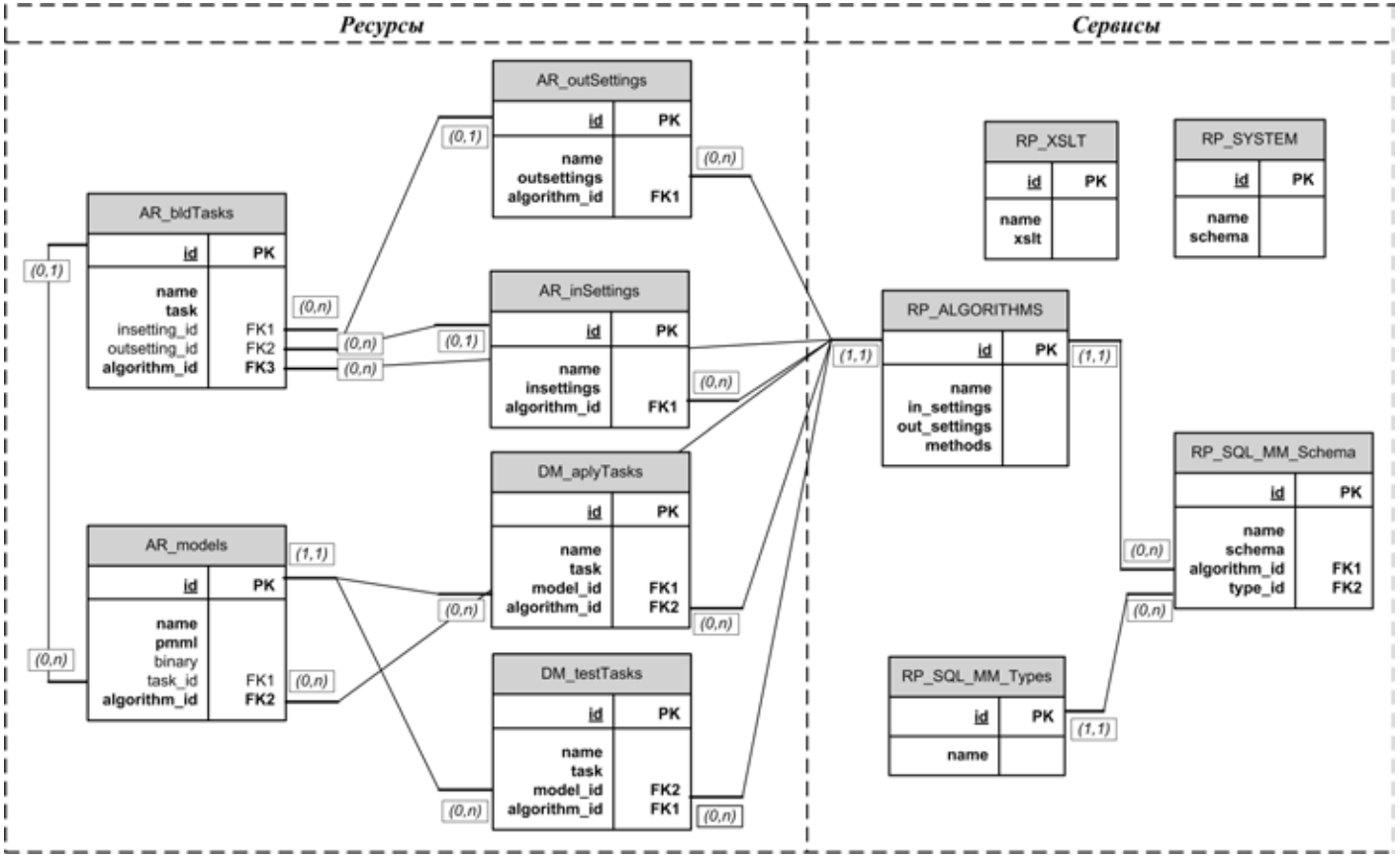

Рис. 4. Даталогическая модель репозитория

– схему определения методов SQL/MM (*SQL_MM.xsd*), предназначенную для валидации описаний SQL-MM-методов;

– схемы сообщений DDS и SOAP (*DDS_Message.xsd, SOAP_Message.xsd*), предназначенные для валидации сообщений, передаваемых с помощью интерфейсов DDS и SOAP.

Таблица *RP_XSLT* предназначена для хранения системных стилей и файлов трансформации, используемых для реализации функционала SQL/MM.

Таблица *RP_SQL_MM_Types* необходима для хранения названий абстрактных типов SQL/MM, к которым относятся модели (*Model*), задачи обучения (*BldTask*), тестовые задачи (*TestTask*), прикладные задачи (AplyTask). Записи данной таблицы обеспечивают наличие уникальных (неповторяющихся) схем, описывающих внутреннюю структуру SQL/MM-типов.

Таблица *RP_SQL_MM_Schema* предназначена для непосредственного хранения схем различных типов SQL/MM. Комбинация уникальных ссылок на идентификаторы алгоритмов в таблице *RP_ALGORITHMS* и идентификаторы типов в таблице *RP_SQL_MM_Types* гарантирует, например, что в хранилище сервисов будет одна и только одна схема задачи обучения для алгоритма классификации, разумеется, если в хранилище сервисов объявлен только один алгоритм классификации. В противном случае, для корректной работы аналитической платформы необходимо определить валидационные схемы SQL/MM-типов для каждого алгоритма классификации.

Хранилище ресурсов представляет собой множество связанных таблиц, предназначенных для хранения результатов, полученных в процессе извлечения знаний (настройки моделей, настройки алгоритмов, задачи, модели PMML). Названия таблиц аналитических ресурсов начинаются с префикса «*AR_*». На рисунке 4 видно, что хранилище аналитических ресурсов состоит из шести таблиц:

– таблица настроек моделей знаний (*AR_outSettings*);

– таблица настроек алгоритма извлечения знаний (*AR_inSettings*);

– таблица задач обучения (*AR_bldTask*);

– таблица задач тестирования (*AR_testTask)*;

– таблица прикладных задач (*AR_aplyTask*);

– таблица моделей знаний (*AR_model)*.

Таблица *AR_outSettings* предназначена для хранения настроек выходных моделей *PMML*. Таблица *AR_inSettings* – для хранения настроек алгоритмов извлечения знаний. Таблица *AR_testTask*, *AR_aplyTask*, *AR_aplyTask* – для хранения задач обучения, тестирования и применения соответственно. Таблица *AR_model* – для хранения моделей PMML. Ввиду необходимости увеличения производительности и особенностей реализаций некоторых аналитических алгоритмов в таблице моделей предусмотрено сохранение не только PMML-модели, но и ее бинарного представления (*binary*).

Для сохранения целостности ресурсов аналитической платформы, обеспечения валидности обновлений и вставок новых ресурсов в репозитории предусмотрен специальный защитный механизм, основанный на применении триггеров СУБД PostgreSQL. Триггеры представляют собой автоматические хранимые процедуры,

которые срабатывают в момент вставки изменения или удаления объектов хранилища ресурсов. Данный защитный механизм не позволит, например, удалить или изменить модель знаний, которая используется в какой-либо прикладной или тестовой задаче.

Режимы функционирования аналитической платформы

Режимы функционирования компонентов аналитической платформы можно условно разделить на два вида:

1. Режимы управления аналитической платформой (администрирование).
2. Режимы использования аналитической платформы (извлечение знаний).

На рисунке 5 представлены основные режимы функционирования компонентов аналитической платформы.

Следует отметить, что на данном рисунке представлены только основные режимы функционирования аналитической подсистемы. Более подробное описание режимов функционирования аналитической подсистемы будет представлено ниже.

Под администрированием аналитической платформы понимаются управляющие воздействия, связанные с изменением внутренней структуры аналитической подсистемы (управление алгоритмами извлечения знаний, управление источниками данных, управление аналитическими ресурсами и т. д.), а также воздействия, связанные с управлением правами доступа пользователей (добавление нового пользователя, изменение прав доступа пользователя и т. д.).

Сценарий создания нового алгоритма представлен на рисунке 5. При добавлении новых алгоритмов в систему администратором определяется список SQL/MM-типов и набор соответствующих методов. Данный список представляется с помощью XML-описания и записывается в поле *method* таблицы *RP_ALGORITHMS*. Если новый алгоритм принадлежит стандарту SQL/MM, то список типов и набор методов выбирают из стандарта. Стандартный набор типов и методов в репозитории представлен XSD-схемой SQL_MM.xsd в таблице *RP_SYSTEM*. Если новый алгоритм не принадлежит стандарту SQL/MM, то содержимое файла *SQL_MM.xsd* дополняется новыми типами и методами согласно допустимой стандартом нотации образования имен.

Допустимые настройки нового алгоритма и настройки модели описываются соответствующими XSD-схемами и записываются в поля *in_setting* и *out_setting* таблицы *RP_ALGORITHMS*, при этом схема настроек модели формируется путем трансформации схемы *pmml_Task.xsd*, расположенной в таблице *RP_SYSTEM*, с применением заранее приготовленного XSLT-описания, помещаемого в таблицу *RP_XSLT*. Завершением создания нового алгоритма является определение схем всех его типов SQL/MM, используемых при сохранении результатов в хранилище ресурсов (таблицу *RP_SQL_MM_Schema*).

Важным аспектом администрирования является удаление алгоритмов. Контроль целостности аналитической платформы, заключающийся в зависимости записей в таблицах аналитических ресурсов от типа алгоритма, позволяет корректно производить удаление устаревших алгоритмов. Иными словами, удаление

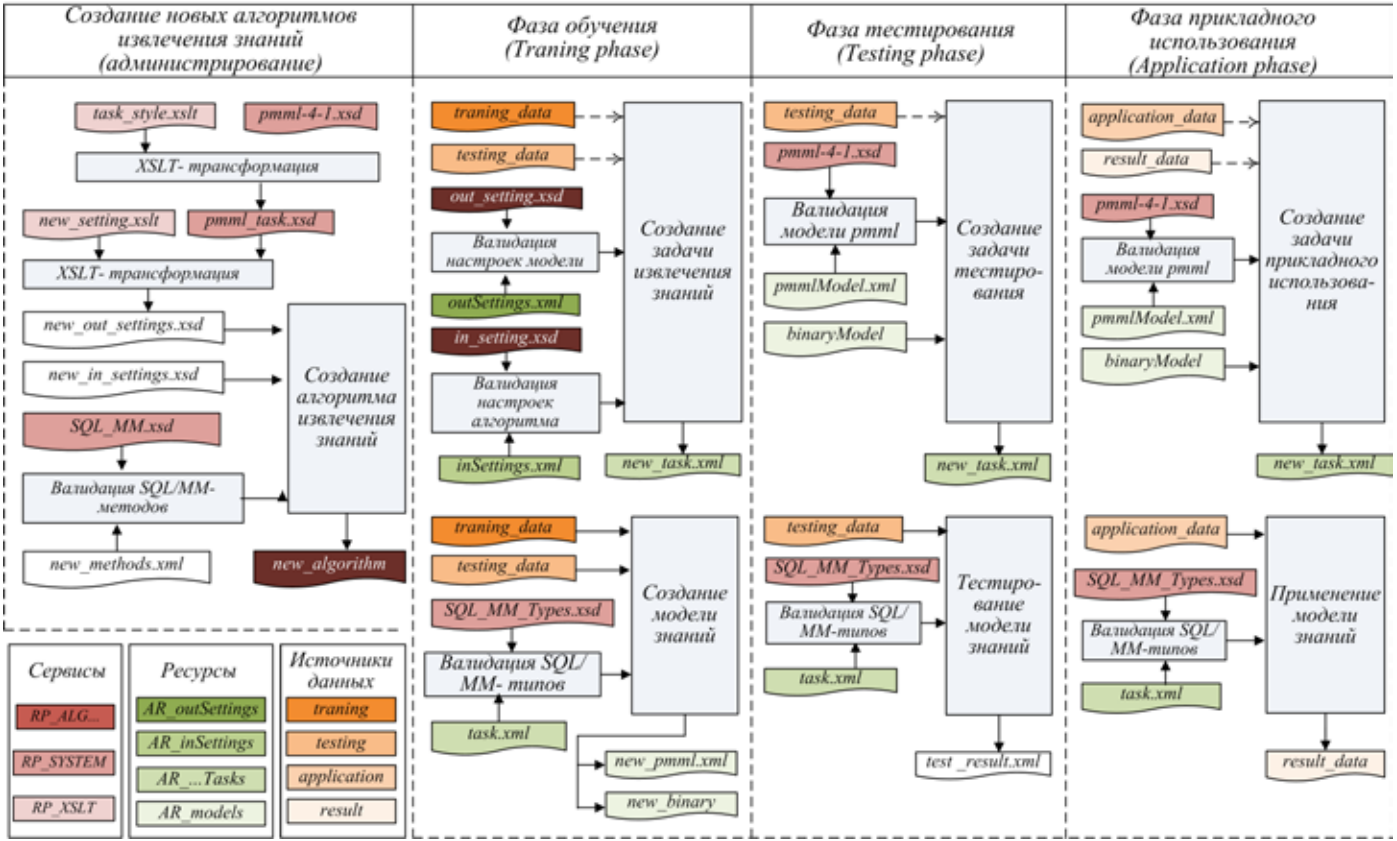

Рис. 5. Основные режимы функционирования аналитической платформы

какого-либо алгоритма автоматически приведет к каскадному удалению всех зависимых от него записей в таблицах аналитических ресурсов.

Управлением аналитическими ресурсами называется процесс редактирования существующих записей настроек алгоритмов, моделей, задач, моделей знаний. Таблицы аналитических ресурсов поддерживают частичный контроль целостности, то есть имеется необязательная зависимость записей задач от настроек алгоритмов и настроек моделей, а также необязательная зависимость моделей знаний от задач обучения. Таким образом, каскадное удаление производится только по связанным записям. Записи, не имеющие связей, удаляются отдельно.

Режимы использования аналитической подсистемы (извлечение знаний) представляют собой действия, направленные на настройку, создание и использование моделей знаний. К таким режимам относятся:

– создание задачи построения модели знаний;
– создание задачи тестирования модели знаний;
– создание задачи использования модели знаний;
– построение модели знаний;
– тестирование модели знаний;
– применение модели знаний.

Технические характеристики аналитической платформы

Поскольку в качестве аналитической СУБД была выбрана СУБД PostgreSQL, аналитическая платформа, реализованная в соответствии с предложенной архитектурой, будет иметь следующие тактико-технические характеристики:

– обработка масштабных массивов разнородной информации (до 32 ТВ);
– расширяемость и масштабируемость аналитики;
– многоплатформенность;
– контроль целостности данных;
– интерфейсы доступа к языкам программирования высокого уровня;
– триггеры и правила для управления процессами;
– система управления правами доступа и авторизации;
– параллельная обработка пользовательских сессий;
– шифрование трафика.

Заключение

Преимуществами аналитической платформы, встроенной в базу данных, являются высокая гибкость применения алгоритмов извлечения знаний, а также простые возможности их расширения и масштабирования. Предложенная архитектура аналитической платформы извлечения знаний, основанной на СУБД PostgreSQL, позволяет эффективно организовать процессы анализа и извлечения знаний в различных предметных областях промышленных применений.

Реализация аналитической платформы в рамках базы данных позволяет использовать сценарные методы обработки данных для построения каскадных алгоритмов произвольной сложности. Кроме того, встроенный сервис базы данных обеспечивает надежность, безопасность, масштабируемость и расширяемость

аналитической подсистемы, что имеет принципиальное значение для систем промышленного использования. Предложенная архитектура репозитория расширяет стандарт SQL/MM в части организации инфраструктуры аналитической подсистемы. Стандарт предполагает использование объектно-ориентированных баз данных. Далеко не все базы данных в полной мере удовлетворяют требованиям стандарта SQL/MM, поэтому можно ожидать реализацию только некоторого диалекта стандарта. СУБД PostgreSQL не является строго объектно-ориентированной, но ее возможности достаточно полно покрывают требования стандарта.

СПИСОК ЛИТЕРАТУРЫ

1. Data Mining Community. Top Resource. – URL: http://www.kdnuggets.com. (дата последнего доступа: 28.01.2014).

2. ISO/IEC 13249-6-2006, SQL/MM Part 6.

3. PMML Version 4.1, 2012, Data Mining Group (DMG).-URL:http://www. dmg. org/.

4. PostgreSQL. – URL: http://www.postgresql.org/. (дата последнего доступа: 28.01.2014).

5. Simple Object Access Protocol, SOAP 1.2 Messaging Framework (Second Edition). – URL: http://www.w3.org/TR/soap12-part1 (дата последнего доступа: 28.01.2014).

6. OpenDDS Version 3.0 Supported by Object Computing, Inc. (OCI). – URL: http:// www.opendds.org/; http://www.ociweb.com/. (дата последнего доступа: 28.01.2014).