Декларативная аналитика в мультидиалектной среде

Д.Ю. Ковалев
Институт проблем информатики РАН
г. Москва
dm.kovalev@gmail.com

Аннотация

Развитие наук с интенсивным использованием данных требует создания новых систем поддержки научных исследований. Подобные системы позволяют исследователям специфицировать задачи в терминах исследуемой предметной области, осуществлять масштабируемый анализ разнородных данных из различных источников, при этом поддерживая современные платформы больших данных. В Институте проблем информатики Российской Академии Наук исследуется мультидиалектная среда, позволяющая концептуально описывать научные задачи, при этом обеспечивая интероперабельность различных систем на правилах и средств интеграции данных. В данной работе представлен пример концептуальной спецификации задачи по анализу финансовых данных. Пример демонстрирует необходимость использования разных систем на правилах и обмена правилами между ними. Предлагаются усовершенствованный алгоритм выполнения концептуальной программы, а также соответствующий этому изменению подход для описания декларативной семантики работы среды.

1 Введение и описание проблемы

В течение последних нескольких лет в большинстве научных областей, а также в бизнесе произошел значительный скачок в количестве производимых и накопленных данных. Во многих научных областях работа с данными стала занимать значительную часть рабочего времени исследователей. Изменились сам формат и подход к научной деятельности, во главу угла была поставлены работа с данными. Это привело к возникновению новой парадигмы в науке, а сами такие разделы науки были названы науками с интенсивным использованием данных (data-intensive sciences (DIS)) [2, 17]. Примерами научных направлений с интенсивным использованием данных являются астрономия, науки о Земле, молекулярная биология [17].

Труды 15-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» — RCDL-2013, Ярославль, Россия, 14-17 октября 2013 г.

Новая парадигма требует создания новых систем и средств поддержки всего цикла научных исследований, начиная от получения данных и заканчивая анализом данных и визуализацией полученных результатов [2]. При создании подобной системы неизбежно предстоит решить следующие задачи:

- интеграция разнородных, распределенных информационных ресурсов;
- разработка масштабируемых алгоритмов для работы в распределенной и параллельной средах;
- описание задачи в терминах предметной области (концептуально), сокращение кода программы;
- избавление пользователя от необходимости вручную распараллеливать алгоритмы.

При работе с большими объемами данных алгоритмы анализа данных реализуются над распределенной инфраструктурой. Это могут быть как параллельные машины баз данных, так и Hadoop. Такая реализация не требует изменений кода программы при увеличении объема данных.

Существует два подхода к обеспечению масштабируемости аналитики [15]. Целью первого подхода является создание параллельной среды выполнения программы для языка достаточно высокого уровня (R, Matlab). Примерами такого подхода могут служить System ML [11], Revolution Analytics[22], Snow [30]. Для императивных языков распараллеливание производится вручную (например, с помощью MPI-интерфейса). Для декларативных языков эта же задача решается автоматически самой системой. Например, в System ML программы на R-подобном языке автоматически транслируются в программу над Hadoop.

Целью второго подхода является предоставление пользователю системы со встроенными примитивами низкого уровня совместно с примитивами для координации выполнения программы. Примерами такого подхода являются Apache Mahout [5], SciDB [29], MADlib [15].

Другой сложностью, возникающей в DIS, являются разнообразие форматов и моделей данных, представлений и способов хранения большого объема данных, а также множественность распределенных источников данных. Все это приводит к необходимости интеграции разнородных данных. При реализации материальной и виртуальной интегра-

ции данных обычно используются подходы GAV [31, 14] и LAV [14] или их комбинация GLAV [9].

Реализация информационных систем в DIS осуществляется в комбинированных архитектурах ИТ средств, включающих платформы поддержки больших данных, суперкомпьютеры, многопроцессорные системы, грид и облачные архитектуры [2]. Спецификация и реализация распределенных программ анализа данных над такой комбинированной распределенной архитектурой являются нетривиальной задачей [16]. Низкоуровневые примитивы и абстракции императивных языков программирования неэффективны в таких архитектурах для выражения алгоритмов задач анализа данных, требующих концептуального представления, независимого от конкретных данных и обеспечивающих возможность их повторного использования в других применениях. В качестве альтернативы использованию императивных языков, подобно семантическому Вебу [27], для решения задач анализа данных представляется целесообразным использовать языки на правилах. Помимо естественной концептуализации проблем в таких языках разнообразие их семантик является существенным расширением возможностей традиционных методов анализа данных при решении сложных задач. Также декларативные языки являются хорошо распараллеливаемыми [16].

В результате формулирования проблем и требований к новым системам поддержки DIS были выявлены некоторые свойства, наличие которых необходимо учитывать при их создании [2, 16]. Такие системы должны сочетать в себе средства интеграции и анализа данных, позволяющие эффективным образом выражать аналитические алгоритмы в терминах концептуального представления предметных областей. Кроме того, рост объема данных приводит к тому, что функции анализа необходимо выполнять как можно ближе к данным. Системы должна быть легко разворачиваемы над комбинированными распределенными ИТ-средствами, сохраняя при этом приемлемую эффективность работы.

Ряду перечисленных требований соответствует разрабатываемая в ИПИ РАН среда, позволяющая декларативным образом концептуально описывать задачи из различных предметных областей [19]. В ней использован высокоуровневые подходы к программированию задач на основе языков на правилах в сочетании со средствами интеграции данных, также выражаемых декларативно.

Первый из подходов технически базируется на использовании стандарта W3C по обмену правилами Rule Interchange Format (RIF) [27]. Этот стандарт позволяет обеспечить синтаксическую и семантическую интероперабельность программ, представленных в различных языках на правилах, на основе стандартизованных диалектов RIF. Согласно стандарту, для каждой из конкретных систем на правилах требуется построить отображение из языка системы в подходящий диалект RIF и обратно. Отображение должно сохранять семантику отображаемых языков. Правила передаются от одной системы к другой на промежуточном языке, в качестве которого используется некоторый диалект RIF.

В существующей редакции RIF определено несколько диалектов. Так, RIF-BLD является основным логическим диалектом в стандарте RIF и соответствует хорновским правилам с некоторыми синтаксическими и семантическими расширениями [24]. Диалект RIF-PRD [28] обобщает примитивы различных продукционных систем. Диалект RIF-Core разрабатывался с целью максимизации пересечения продукционных и логических диалектов. Существуют и другие диалекты RIF, пока не являющиеся рекомендацией W3C: CASPD [25] – для систем, обладающих семантикой стабильных моделей [10], CLPWD [26] – для систем с хорошо-обоснованной семантикой [33]. По причине того, что при спецификации задач может быть использовано несколько диалектов RIF, разрабатываемая среда получила название мультидиалектной.

Средства интеграции данных в рассматриваемой среде обеспечивают виртуальную интеграцию и совместное использование неоднородных ресурсов на основе технологии предметных посредников [34, 20]. Посредник располагается между пользователем, который формулирует проблемы концептуально, независимо от ресурсов, и разнородными информационными ресурсами.

Несмотря на то, что уже сейчас в этой среде можно решать интересные задачи [19], учитывая требования задач анализа данных в DIS, требуется развить ее, встроив средства аналитики и сохранив при этом высокий уровень используемых абстракций и примитивов.

Статья структурирована следующим образом: в разделе 2 кратко описана существующая среда, в разделе 3 представлены основная цель исследования и сопутствующие задачи. Раздел 4 описывает подход к проблеме и основные идеи работы. Наконец, в разделе 5 представлено заключение.

2 Мультидиалектная среда

Программой на правилах называется некоторый конечный набор правил. Движок, который осуществляет выполнение программы на правилах, называется системой на правилах. Ключевым при выводе является отношение выполнимости формул. Логический язык имеет теоретико-модельную семантику, согласно которой определяется, какие модели существуют для каждой программы. Существует несколько различных семантик для работы с отрицанием. Основными являются стратифицированный даталог [32], семантика стабильных моделей и хорошо-обоснованная семантика.

Среда [19] обеспечивает взаимную работу средств интеграции и систем вывода на правилах. Она состоит из нескольких узлов, осуществляющих отображение из языка системы в диалект RIF и наоборот. В среду также входят супервизор, отвечающий за исполнение программы на правилах, и посредник. Так как посредник специфицируется на логическом языке, можно относиться к нему как к

полноправному узлу. Задача формулируется и решается в терминах предметной области (описанной концептами из подключенной онтологии). Программа, написанная в концептах предметной области, называется концептуальной. При добавлении информации о принадлежности правил и предикатов конкретному узлу системы концептуальная программа становится распределенной. При исполнении программы происходят следующие действия: 1) переписывание концептуальной программы в распределенную с добавлением для каждого правила и предиката в правиле, соответствующего ему имени узла, 2) рассылка частей программы на узлы, 3) исполнение программы на узлах, 4) получение ответа.

Ключевым механизмом при исполнении программы является делегирование [1] – передача правил и фактов от одного узла к другому. Ответственным за передачу правил и фактов является супервизор среды. На данный момент осуществлена лишь частичная поддержка механизма, например, можно обмениваться фактами.

3 Цель и задачи исследования

Целью работы является исследование и развитие программной среды для сопровождения научных исследований, в которой концептуально и декларативно используются разнообразные инструменты – языки и системы на правилах, средства интеграции данных, аналитические методы.

Для достижения поставленной цели необходимо решить ряд задач.

Во-первых, требуется определить задачу, решение которой продемонстрирует необходимость использования нескольких систем на правилах с разной семантикой, а также обмена правилами между системами вывода.

Во-вторых, требуется развить и автоматизировать алгоритмы переписывания программы из концептуальной в распределенную. При этом отображение должно сохранять отношение логического следования программ. Другим направлением развития является совершенствование алгоритма выполнения распределенной программы. Необходимо сформулировать требования, которым должен удовлетворять этот алгоритм, реализовать его в соответствии с этими требованиями, провести ряд экспериментов по сравнению эффективности предложенного алгоритма с другими подобными системами.

В-третьих, необходимо описать декларативную (теоретико-модельную) семантику выполнения распределенной программы и провести доказательство того, что все полученные модели являются моделями и для концептуальной программы, т. е. установить отношение выполнимости программ.

В-четвертых, среду планируется расширить в сторону поддержки хранилищ данных. При этом они могут содержать в себе инструменты аналитики. При такой интеграции требуется сохранить декларативность и концептуальность подхода. Среда должна учитывать возможности современных платформ, а также распределенных сред – грид и облаков. Необходимо продемонстрировать возможность эффективного использования предлагаемого подхода в этих средах.

В-пятых, требуется оценить сложность выполнения распределенной программы и определить насколько сильно предложенная среда усложняет решение задач.

4 Предлагаемый подход

4.1 Пример задачи для решения в мультидиалектной среде

Для лучшего понимания необходимости использования мультидиалектной среды предложена задача, иллюстрирующая преимущества использования нескольких систем вывода с разными семантиками и обмена правилами между ними [19].

Требуется построить диверсифицированный портфель максимального размера. Портфелем называется множество ценных бумаг, таких как акции и облигации. Диверсифицированность означает, что движение цены бумаги не зависит от других бумаг. Таким образом, снижаются риски сильного падения совокупной цены портфеля.

Прежде всего, требуется решить проблему интеграции данных из различных финансовых источников (google finance, yahoo finance) и проблему по построению портфеля максимального размера. Показано, что последняя проблема сводится к поиску максимальной клики в графе. Данная задача является NP-сложной. Для решения подобных задач (по скорости выполнения и простоте написания программы) хорошо подходят ASP системы на правилах со стабильной семантикой моделей, например система DLV. Для интеграции данных из разных источников используется логический язык СИНТЕЗ. Взаимодействие систем вывода в мультидивалектной среде между собой представлено на рис. 1.

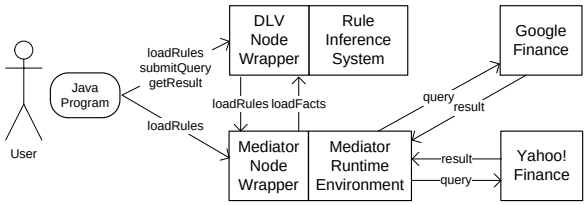

Рис. 1. Мультидиалектная среда для задачи поиска диверсифицированного портфеля

Концептуальная программа определяется на диалектах RIF-BLD (интеграция данных) и RIF-CASPD (поиск максимальной клики). Задачу поиска максимальной клики можно считать одной из класса задач на графах. При использовании языка DLV спецификация задачи о поиске максимальной клики становится простым и практически тривиальным действием. Описание поиска происходит концептуально и декларативно:

```
Document( Dialect(RIF-CASPD)
  ...
```

```
Group (
Forall ?X(Or(prt:portfolio(?X)
  prt:nonPortfolio(?)):-tickers@gex(?X))
Forall ?X ?Y( :- And(prt:portfolio(?X)
  prt:nonPortfolio(?X)))
Forall ?X ?Y( :- And(prt:portfolio(?X)
  prt:portfolio(?Y) (Naf noncorrelat-
  ed@gex(?X ?Y))) )
Forall ?X( :~ prt:nonPortfolio(?X))  ) )
```

Формальное описание процесса решения задачи представлено в [19].

Задачу поиска максимальной клики можно считать одной из класса задач на графах. Подход, подобный изложенному, можно применять и для других задач из этого класса, а также других классов задач.

Второй задачей является интеграция данных из различных источников. Для ее решения используется концепция предметных посредников и язык СИНТЕЗ [20]. Спецификация осуществляется на диалекте RIF-BLD:

```
Document( Dialect(RIF-BLD)
  ...
  Group(
  Forall ?t ?tp ?symbol, ?ticker(
    Exists ?ts( And(?ts#gex:tickers
      ?ts[symbol -> ?symbol]) ):-
      And(?t#srt:stockRates ?t[
        ticker->?symbol])
  Forall ?m ?n ?ticker1 ?ticker2 ?start
    ?end ?rates1 ?rates2 ?dv1 ?dv2 ?date1
    ?date2 ?series1 ?series2 ?corr (
      Exists ?e (
        And(?e#gex:noncorrelated ?e[
          start->?ticker1 end->?ticker2])):-
          ?m#srt:stockRates  ?m[
            ticker->?ticker1  rates->?rates1]
          ?n#srt:stockRates  ?n[
            ticker->?ticker2  rates->?rates2]
          ?dv1#?rates1  ?dv1[date -> ?date1]
          ?dv2#?rates2  ?dv2[date -> ?date2]
          External(pred:date-greater-than-or-
            equal(?date1  2012-01-01))
          External(pred:date-less-than-or-
            equal(?date1  2012-12-31))
          ?c#srt:correlation ?c[corr->?corr
            series1->?rates1 series2
              ->?rates2]
          External(pred:numeric-greater-
            than(?corr  -0.25)
          External(pred:numeric-less-
            than(?corr  0.25))
        External(pred:numeric-less-
            than(?ticker1  ticker2))  )))
```

Подробное описание правил и взаимодействия между модулями представлено в [19]. Вся программа занимает всего 6 правил. В мультидиалектной среде становится возможным выделить несколько частей программы, для которых требуется различные системы на правилах с разной семантикой, при этом обеспечив их взаимодействие посредством обмена правилами через диалекты RIF.

4.2 Алгоритм переписывания и исполнения распределенной программы

На данный момент предлагаемая среда обладает рядом ограничений. Фактически при исполнении алгоритма на каждом из узлов программа исполняется не более одного раза. Предлагается определить вычисление распределенной программы как последовательность локальных вычислений на узлах среды [3, 21]. Узлы среды активируются один за другим в каком-либо порядке. Это происходит, пока не достигнута неподвижная точка (fixpoint) распределенной программы. В течение каждого локального исполнения части программы узел получает и отправляет сообщения, а также совершает некоторый логический вывод, определенный системой вывода в данном узле. Последовательность вычислений не прекращается до тех пор, пока не достигнута неподвижная точка.

Пусть I_L – конечный набор фактов на узле L, P_L – часть распределенной программы на узле L, $Facts_{rcv}$ – набор фактов, полученных от других узлов, а $Rules_{rcv}$ – множество соответствующих полученных правил. Пусть также *induc(P)[I]* – это набор edb-фактов, принадлежащий данному узлу и полученных в результате выполнения программы P над базой данных I; *facts(P)[I]* – набор edb-фактов, не принадлежащих данному узлу, *rules(P)[I]* – набор правил, не принадлежащих данному узлу. Тогда в результате выполнения программы получится набор фактов, которые пополнят локальную базу данных

$$J = I \cup induc(P_L \cup Rules_{rcv})[I \cup Fatcs_{rcv}],$$

а также набор фактов и правил, которые необходимо отправить на соответствующие узлы:

$$Fatcs_{snd} = facts(P_L \cup Rules_{rcv})[I \cup Fatcs_{rcv}]$$

$$Rules_{snd} = rules(P_L \cup Rules_{rcv})[I \cup Fatcs_{rcv}]$$

Соответственно после отправки сообщений запустится очередной локальный шаг вычислений. Последовательность локальных вычислений строится следующим образом:

$$\rho_1 \xrightarrow[Facts,Rules]{} \rho_2 \xrightarrow[Facts,Rules]{} \cdots \xrightarrow[Facts,Rules]{} \rho_n.$$

Алгоритм завершается, если набор фактов и правил, подлежащих отправлению, пуст.

На данном этапе в среде предполагается передача фактов коллекцией. Предлагается усовершенствовать алгоритм в соответствии с предложенной схемой.

Внедрение этого алгоритма выполнения распределенной программы позволит исполнять произвольные программы в мультидиалектной среде. Кроме того, становится возможным в соответствии с предложенным алгоритмом описать теоретико-

модельную семантику исполнения программы и доказать ее соответствие семантике диалекта RIF.

Обмен правилами между продукционными системами на правилах с использованием диалекта RIF-PRD представлен в работах [12, 8]. Основным отличием является то, что продукционные системы работают с единой операционной семантикой и, следовательно, пропадает проблема совместной работы систем на правилах с различной логической семантикой.

Автору не известны упоминания в литературе описаний механизмов обмена правилами между системами с операционной и декларативной семантиками в RIF. Однако существует ряд работ по использованию языков баз данных для спецификации декларативных распределенных программ и манипулирования данными в распределенной среде [21, 13, 3, 1]. В отличие от мульти-диалектного подхода, в упоминаемых системам используется единый язык на правилах на всех узлах. Обычно это даталог с отрицанием с добавлением понятия локализации – принадлежности факта или правила определенному узлу. В отличие от этих языков, в предложенном подходе используются стандартные, определенные в RIF удаленные и импортированные термы.

4.3 Семантика

При переписывании концептуальной программы в распределенную может измениться набор выводимых моделей. Определение декларативной семантики распределенной программы позволит установить отношение логического следования между ней и концептуальной программой.

Для переписанной программы P_{dist} из концептуальной P_{conc} должно быть верно отношение логического следования $P_{conc} \models P_{dist}$, то есть все модели P_{dist} должны быть и моделями P_{conc}.

Для достижения этого может быть применено несколько подходов. Можно определить теоретико-модельную семантику концептуальной и распределенной программ и установить отношение логического следования. Другим вариантом является определение набора операций переписывания программы, сохраняющих семантику. Остается определить, что это за операции и каким образом установить сохранение семантики при отображении.

Для большинства распределенных языков на правилах не определяется декларативная семантика по причине наличия нелогических конструкций. Исключением является язык Dedalus [3], для которого определена полностью декларативная семантика.

5 Заключение

Смена парадигмы в некоторых областях науки требует создания современны средств и систем поддержки научных исследований. В ИПИ РАН исследуется система, позволяющая декларативно специфицировать научные задачи. С целью ее развития в данной работе представлены пример задачи по анализу финансовых данных, новый алгоритм выполнения распределенной программы и соответствующий ему подход по определению декларативной семантики программ. Пример демонстрирует потенциальные возможности мультидиалектной среды, в том числе интеграцию данных из различных источников и декларативный анализ полученных данных.

Литература

[1] Abiteboul S., Bienvenu M., Galland A. et al. A rule-based language for Web data management. In: Proc. 30th ACM Symposium on Principles of Database Systems, ACM Press, 2011. – P. 283–292.

[2] Agrawal D., Bernstein P., Bertino E., Davidson S., Dayal U., Franklin M., Gehrke J., Haas L., Halevy A., Han J., Jagadish H.V., Labrinidis A., Madden S., Papakonstantinou Y., Patel J. M., Ramakrishnan R., Ross K., Shahabi C., Suciu D., Vaithyanathan S., Widom J. Challenges and Opportunities with Big Data. A community white paper developed by leading researchers across the United States, 2012. – http://cra.org/ccc/docs/init/ bigdatawhitepaper.pdf

[3] Alvaro P., Marczak W.R. et al. Dedalus: Datalog in time and space// Technical Report EECS-2009-173, University of California, Berkeley, 2009.

[4] Apache Hadoop. – http://hadoop.apache.org/.

[5] Apache Mahout. – http://mahout.apache.org/.

[6] Apache Pig. – http://pig.apache.org/.

[7] Beyer K.S., Ercegovac V., Gemulla R., Balmin A., Eltabakh M., Kanne C.C., Shekita E.J. Jaql: A scripting language for large scale semistructured data analysis// In: Proc. of the VLDB Endowment, 2011. – V. 4, No 12. – P. 1272-1283.

[8] Cosentino V., Del Fabro M. D., El Ghali A. A model driven approach for bridging ILOG Rule Language and RIF. In: Proc. of the 6th International Symposium on Rules, RuleML2012, 2012. – CEUR-ws. – V. 874. – P. 96-102.

[9] Friedman M., Levy A., Millstein T. Navigational plans for data integration. In: National Conference on Artificial Intelligence (AAAI) Proc., 1999.

[10] Gelfond M., Lifschitz V. The Stable Model Semantics for Logic Programming. In: Proc. Fifth Intl. Conference and Symposium Logic Programming, MIT Press, Cambridge, 1988. – P. 1070–1080.

[11] Ghoting A., Krishnamurthy R., Pednault E. et al. SystemML: Declarative machine learning on MapReduce. In: ICDE, 2011. – P. 231–242.

[12] Gonzalez-Moriyon G. Final steel industry public demonstrators// ONTORULE Deliverable D5.5, 2012.

[13] Grumbach S., Wang. F. Netlog, a rule-based language for distributed programming. In: (Eds.) M. Carro and R. Pena, Proc. 12th International Symposium on Practical Aspects of Declarative Languages, LNCS, 2010. – V. 5937. – P. 88–103.

Halevy A. Y. Answering queries using views: A survey. In: VLDB J., 2001. – V. 10, No. 4.

[14] Hellerstein J. M., Ré C., Schoppmann F., Wang D. Z., Fratkin E., Gorajek A., Kumar A. The MADlib analytics library: or MAD skills, the SQL. In: Proc. of the VLDB Endowment, 2012. – V. 5, No. 12. – P. 1700-1711.

[15] Hellerstein J. The declarative imperative: experiences and conjectures in distributed logic. In: ACM SIGMOD Record, 2010. – V. 39, No. 1. – P. 5–19.

[16] Hey T., Tansley S., Tolle K. (Eds.). The Fourth Paradigm: Data-intensive Scientific Discovery. Microsoft Research, Redmond, Washington, 2009.

[17] Ihaka R., Gentleman R. R: A language for data analysis and graphics. In: Journal of computational and graphical statistics, 1996. – V. 5, No. 3. – P. 299-314.

[18] Kalinichenko L., Stupnikov S., Vovchenko A., Kovalev D. Rule-based Multi-dialect Infrastructure for Conceptual Problem Solving over Heterogeneous Distributed Information Resources. In: Kacprzyk, Janusz (eds.), Advances in Intelligent Systems and Computing, 2013. – V. 241. – P. 61-68.

[19] Kalinichenko L.A., Stupnikov S.A., Martynov D.O. SYNTHESIS: A language for canonical information modeling and mediator definition for problem solving in heterogeneous information resource environments. In: M. IPI RAS, 2007.

[20] Loo B. T., Condie T., Garofalakis M., Gay D. E., Hellerstein J. M., Maniatis P., Ramakrishnan R., Roscoe T., Stoica I. Declarative networking: language, execution and optimization. In: ACM SIGMOD Conference Proceedings, 2006. – P. 97–108.

[21] Revolution Analytics. – http://www.revolutionanalytics.com/

[22] RHadoop package. – http://github.com/RevolutionAnalytics/RHadoop/

[23] RIF Basic Logic Dialect (Second Edition)// (Eds.) H. Boley, M. Kifer.W3C Recommendation, 2013.

[24] RIF Core Answer Set Programming Dialect// (Eds.) S. Heymans, M. Kifer, 2009. – http://ruleml.org/rif/RIF-CASPD.html

[25] RIF Core Logic Programming Dialect Based on the Well-founded Semantics// (Ed.) Michael Kifer RuleML specification, 2010. – http://ruleml.org/rif/RIF-CLPWD.html

[26] RIF Overview (Second Edition)// (Eds.) H. Boley, M. Kifer. W3C Working Group Note, 2013.

[27] RIF Production Rule Dialect// (Eds.) Christian de Sainte Marie, Gary Hallmark, Adrian Paschke. W3C Recommendation, 2013. – http://www.w3.org/TR/2013/REC-rif-prd-20130205/

[28] Stonebraker M., Brown P., Poliakov A., Raman S. The architecture of SciDB. In: Proc. of the 23rd international conference on Scientific and statistical database management SSDBM, 2011. – P. 1–16.

[29] Tierney L., Rossini A. J., Li N. Snow: A parallel computing framework for the R system. In: International Journal of Parallel Programming, 2009. – V. 37, No. 1. – P. 78-90.

[30] Ullman J. D. Information integration using logical views. In: 6th International Conference on Database Theory (ICDT'97) Proc., 1997.

[31] Ullman J. D. Principles of Database and Knowledge-Base Systems. W. H. Freeman & Co., New York, 1990. – V. 2.

[32] Van Gelder A., Ross K. A., Schlipf J. S. The well-founded semantics for general logic programs. In: Journal of the ACM (JACM), 1991. – V. 38, No. 3. – P. 619-649.

[33] Д.О. Брюхов, А.Е. Вовченко, В.Н. Захаров, О.П. Желенкова, Л.А. Калиниченко, Д.О. Мартынов, Н.А. Скворцов, С.А. Ступников. Архитектура промежуточного слоя предметных посредников для решения задач над множеством интегрируемых неоднородных распределённых информационных ресурсов в гибридной грид-инфраструктуре виртуальных обсерваторий. Информатика и её применения, 2008. – т. 2, вып. 1. – с. 2-34.

Declarative Analytics in Multidialect Infrastructure

D. Kovalev

Data intensive sciences require new systems to support scientific research, starting from data acquisition and finishing with data analysis and visualization. At the Institute of problems of informatics RAS a new multidialect infrastructure is investigated to satisfy these needs. Problems could be specified conceptually. System supports interoperability of rule systems and integration facilities. However, data warehousing and analytical methods are not yet included into the system. The aim of this work is to built in those facilities and further improve execution algorithms of the proposed infrastructure.