УДК 004.02

РЕАЛИЗАЦИЯ ОБОЛОЧКИ И ПОРТАЛА ЗНАНИЙ ПО ВЕРИФИКАЦИИ МАТЕМАТИЧЕСКИХ ДОКАЗАТЕЛЬСТВ НА ПЛАТФОРМЕ IACPAAS

А.С. Клещёв[1], В.А. Тимченко[2]

Институт автоматики и процессов управления ДВО РАН, Владивосток, Россия
[1]kleschev@iacp.dvo.ru, [2]vadim@dvo.ru

Аннотация

Представлена концептуальная архитектура оболочки для интерактивных систем верификации математических доказательств и создаваемого с её помощью развиваемого тематического портала знаний. Описан процесс реализации всех программных и информационных компонентов оболочки на облачной платформе IACPaaS с использованием предоставляемых ею технологий и инструментальных средств их поддержки. Рассмотрен процесс разработки начального состояния портала знаний по верификации математических доказательств с использованием средств оболочки, способ использования портала знаний заинтересованными членами математического сообщества, а также механизмы изменения состояния портала его администратором. В состав начального состояния портала знаний входят: модель онтологии базы математических знаний, включающая спецификацию начального состояния языка представления математических знаний, редактор модели онтологии базы математических знаний, редактор базы математических знаний, редактор базы способов рассуждений, решатель задач оболочки, реализующий процесс конструирования доказательств в терминах модели онтологии доказательств. Также в состав начального состояния портала знаний входят начальное состояние базы математических знаний и начальное состояние базы способов рассуждений. Развитие портала знаний осуществляется по названным информационным компонентам. В этом процессе могут принимать участие все заинтересованные члены математического сообщества с помощью системы личных кабинетов платформы IACPaaS, в которых каждый пользователь может независимо развивать свою персональную копию текущего состояния общего портала знаний. Передача новых результатов в общий портал контролируется его администратором.

Ключевые слова: *верификация интуитивных доказательств, программная оболочка, портал знаний, редактор базы знаний, облачные сервисы.*

Цитирование: *Клещёв, А.С.* Реализация оболочки и портала знаний по верификации математических доказательств на платформе IACPaaS / А.С. Клещёв, В.А. Тимченко // Онтология проектирования. – 2018. – Т. 8, №3(29). – С.427-448. – DOI: 10.18287/2223-9537-2018-8-3-428-448.

Введение

Одной из важнейших задач в математических исследованиях является обеспечение правильности (верификация) доказательств теорем, публикуемых в математической литературе. Одним из перспективных путей решения этой задачи является разработка интерактивных *систем поддержки построения доказательств* (СППД) [1–3]. Следствием потребности экспериментировать с построением (верификацией) доказательств в рамках разных формальных моделей (специализированных логик) явилась разработка программных *оболочек* для создания интерактивных СППД (LCF, PVS, Coq, Twelf, HOL Light, Isabelle/HOL и др.) [4–9].

В [10] обсуждалась основная на сегодняшний день причина, по которой СППД, создаваемые с использованием существующих оболочек, остаются мало востребованными в математическом сообществе. Ещё одна причина такого положения дел состоит в том, что большинство таких систем являются однопользовательскими и требуют сложной установки

на персональном компьютере. Это требует от математиков достаточной компьютерной грамотности, поэтому предпочтительнее оболочки и прикладные СППД, реализованные и предоставляемые пользователю как облачные сервисы. К настоящему моменту известны две реализации, удовлетворяющие этим требованиям: Isabelle, использующая подсистему Clide [11] и ProofPeer [12]. Но они не имеют расширяемой общей облачной базы знаний, которая могла бы совместно развиваться математическим сообществом, а не только разработчиками этих систем.

Заметим, что для Isabelle существует так называемое централизованное *хранилище формальных доказательств* (*Archive of Formal Proofs, AFP*). Пользователи могут направлять запросы на добавление формализованных ими теорий (каждая теория представляет собой набор определений, аксиом, теорем/лемм, доказательств) в это хранилище и в случае успешного прохождения процедуры рецензирования эти доказательства становятся частью AFP. Однако данное хранилище не содержит знаний о способах рассуждения, используемых при построении доказательств. Работа с ним строится на тех же принципах, что и работа с распределёнными системами управления версиями, что опять же требует от пользователей высокой компьютерной грамотности. В [12] отмечаются недостатки AFP, связанные со сложностями поиска нужных пользователю теорий, а также проблемами, возникающими при обновлении версий Isabelle. Исследователям нужна база знаний, организованная таким образом, чтобы они могли достаточно легко ориентироваться в ней при поиске необходимой информации.

В [10] представлена концепция программной оболочки для интерактивных систем верификации интуитивных математических доказательств, а также приближенная к математической практике конструирования доказательств формальная система и механизмы её расширения, которые могут быть положены в основу этой оболочки. Настоящая работа посвящена описанию разработки оболочки на облачной платформе IACPaaS [13], тематического портала знаний по верификации интуитивных математических доказательств с использованием средств этой оболочки, а также принципов взаимодействия пользователей с порталом знаний и механизмов его развития.

1 Концептуальная архитектура оболочки для интерактивных систем верификации математических доказательств

Интерактивные системы верификации интуитивных математических доказательств могут быть отнесены к классу систем, основанных на знаниях (СОЗ), а компоненты описанной в работе [10] оболочки отображены на архитектурные компоненты типичных оболочек СОЗ [14]. В дальнейшем вся терминология работы [10] будет использоваться без ссылок на неё. На рисунке 1 представлена концептуальная архитектура оболочки для интерактивных систем верификации математических доказательств.

В состав оболочки входят: *модель онтологии базы математических знаний*, включающая *пустую* спецификацию *языка представления математических знаний*[1]; *редактор модели онтологии базы математических знаний*; *модель онтологии базы способов рассуждений*, включающая спецификации *языка представления пропозициональных тавтологий* и *метаязыка*; *редактор базы математических знаний* и *редактор базы способов рассуждений*. Перечисленная совокупность *моделей онтологий* и *редакторов* представляет собой систему управления базами знаний (СУБЗ) оболочки. В состав оболочки

[1] Это означает, что в оболочке задана структура разделов математики (базы математических знаний), в которой порождающая грамматика языка представления математических знаний задаёт только общую структуру (вид) *математического утверждения* (*предложения*), а конкретные типы формул и термов языка ещё не определены.

также входят: *модель онтологии доказательств*; *таблица соответствий для модели онтологии доказательств*; мультиагентный *интерактивный верификатор доказательств* (подробнее см. раздел 3). Данная совокупность компонентов представляет собой решатель задач оболочки, имеющий пользовательский интерфейс. Наконец, оболочка включает в себя пустую *базу математических знаний* и пустую *базу способов рассуждений*.

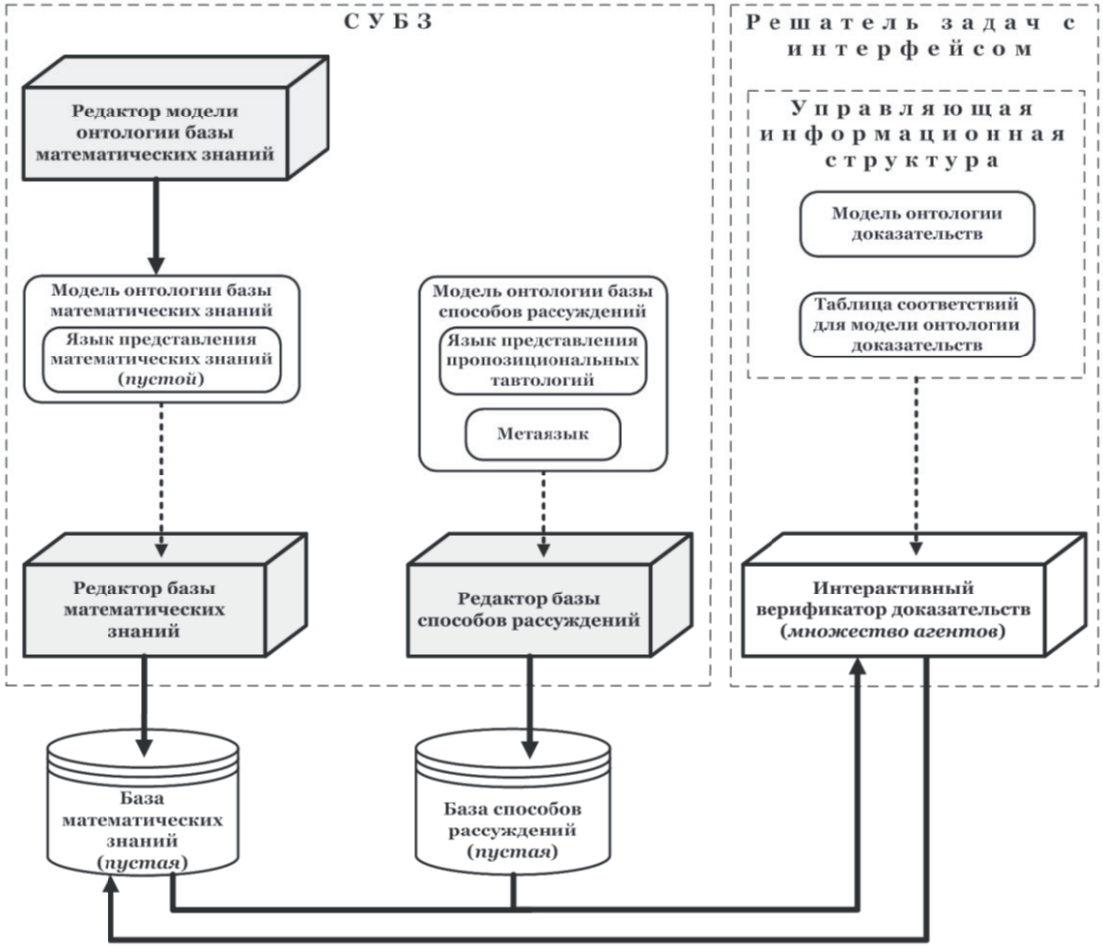

Рисунок 1 – Концептуальная архитектура оболочки для интерактивных систем верификации математических доказательств

Редактор модели онтологии базы математических знаний оболочки предназначен для формирования (спецификации) некоторого *начального состояния* языка представления математических знаний[2]. *Редактор базы математических знаний* оболочки предназначен для формирования некоторого *начального состояния* базы математических знаний. Процесс редактирования в нём управляется моделью онтологии базы математических знаний (используемой в качестве метаинформации (метамодели) – на рисунке 1 этот факт демонстрируется пунктирной стрелкой, входящей в соответствующий программный компонент). *Редактор базы способов рассуждений* оболочки предназначен для формирования *начального состояния* базы способов рассуждений, процесс редактирования в нём управляется моделью онтологии базы способов рассуждений.

Интерактивный верификатор доказательств реализует итеративный процесс конструирования доказательств в терминах модели онтологии доказательств. Он начинается

[2] Это подразумевает описание в порождающей грамматике языка представления математических знаний конкретных типов формул и термов.

с задания названия доказательства и выбора (из базы математических знаний) доказываемого утверждения – теоремы, леммы, некоторого следствия. Далее на каждой итерации последовательно выполняются следующие действия.

1) Выбор *текущей цели* – очередного доказываемого математического утверждения (формулы). В начале доказательства текущей целью является доказываемая теорема (лемма, следствие).
2) Выбор метода доказательства.
3) Выбор способа рассуждения (если метод доказательства не основан на ***правиле доказательства импликации*** **(*естественного вывода*)**).
4) Выбор значений для посылок при использовании правила ***Modus ponens*** (если данное правило было выбрано для *вывода* или *декомпозиции цели*).

Данный процесс заканчивается, когда достигнуто состояние, в котором нет недоказанных целей. Результатом процесса является орграф полного доказательства. Ссылка на этот орграф добавляется в множество ссылок на доказательства утверждения (теоремы, леммы или следствия), принадлежащего базе математических знаний. При этом такая ссылка создаётся не пользователем при формировании базы математических знаний, а интерактивным верификатором доказательств в результате проверки условия завершения процесса конструирования доказательства.

2 Концептуальная архитектура начального состояния портала знаний по верификации математических доказательств

«Пустые» прикладные системы (сервисы) верификации интуитивных математических доказательств, в которых язык представления математических знаний, базу математических знаний и базу способов рассуждений необходимо формировать с нуля, не практичны. Поэтому использование оболочки для создания прикладных сервисов нецелесообразно. В связи с этим возникает потребность в концепте более высокого уровня абстракции (разрабатываемой с использованием средств оболочки), в такой как, например, тематический (специализированный) портал знаний [15] по верификации математических доказательств, который содержал бы некоторое состояние языка представления математических знаний, базы математических знаний и базы способов рассуждений, а также средства их расширения.

На рисунке 2 представлена концептуальная архитектура *начального состояния* развиваемого портала знаний по верификации математических доказательств. В этом смысле оболочка может рассматриваться как *предначальное состояние* портала знаний.

В состав начального состояния портала знаний входят: модель онтологии базы математических знаний, включающая спецификацию начального состояния (ядра) языка представления математических знаний; редактор модели онтологии базы математических знаний; редактор базы математических знаний; редактор базы способов рассуждений; решатель задач оболочки. Также в состав начального состояния портала знаний входят начальное состояние базы математических знаний и начальное состояние базы способов рассуждений.

Редактор модели онтологии базы математических знаний (уже как средство портала знаний) предназначен для расширения состояния языка представления математических знаний. Редактор базы математических знаний портала знаний предназначен для расширения состояния базы математических знаний. Процесс редактирования в нём управляется моделью онтологии базы математических знаний. Редактор базы способов рассуждений портала знаний предназначен для расширения состояния базы способов рассуждений. Процесс редактирования в нём управляется соответственно моделью онтологии базы способов рассуждений. Данные редакторы также рассматриваются уже как средства портала знаний.

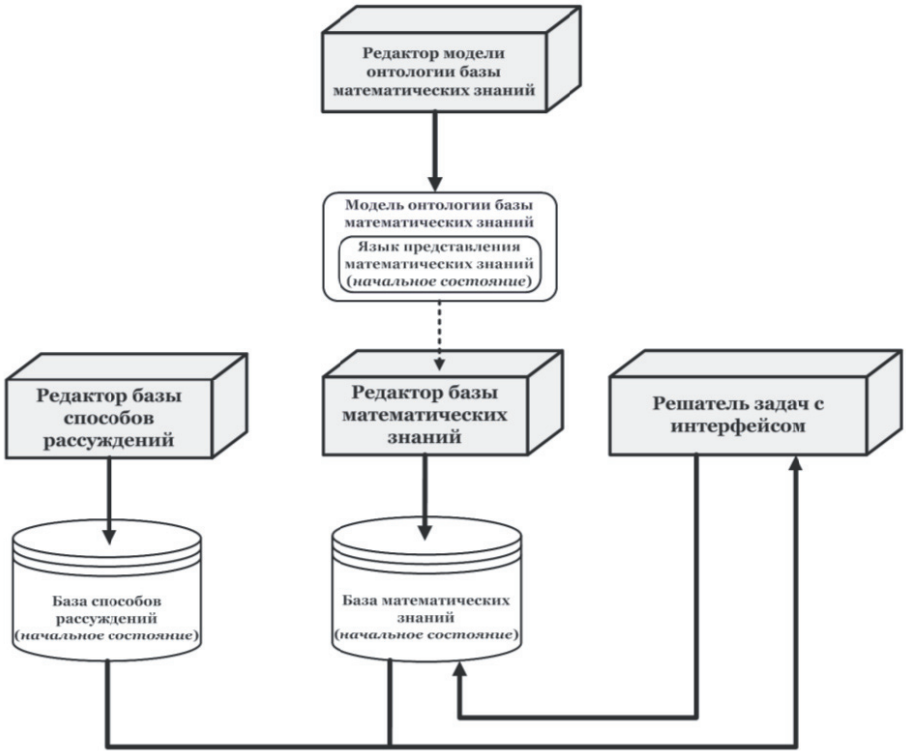

Рисунок 2 – Концептуальная архитектура начального состояния портала знаний по верификации математических доказательств

Все интегрированные в портал ресурсы структурированы в соответствии с их типом и назначением с целью организации удобного поиска и навигации по ним.

3 Разработка оболочки на облачной платформе IACPaaS

Реализация оболочки для интерактивных систем верификации математических доказательств на облачной платформе IACPaaS с использованием предоставляемых ею средств и технологий (инструментария) разработки направлена на обеспечение возможности предоставления оболочки и создаваемого с её использованием начального состояния портала знаний по верификации математических доказательств как облачных сервисов этой платформы (см. рисунок 3).

Разработка всех компонентов СУБЗ оболочки, представленных на рисунке 1, а также пустых базы математических знаний и базы способов рассуждений выполняется с использованием базовой (универсальной) технологии разработки оболочек СОЗ на платформе IACPaaS и инструментальных средств её поддержки. Разработка интерактивного верификатора доказательств и его интерфейса выполняется с использованием специализированной технологии разработки оболочек СОЗ на платформе IACPaaS – на основе информационных структур, управляющих процессом вычисления результата решения задачи (расширенных графовых грамматик) [16].

Использование данной технологии ориентировано на класс задач, для которых явным образом можно специфицировать *структуру* результата их решения в форме связного орграфа – орграфа грамматики (метаинформации, модели), а процесс вычисления самого результата, который также представляет собой связный орграф (орграф информации, компонента), организовать в виде его пошагового формирования «сверху-вниз».

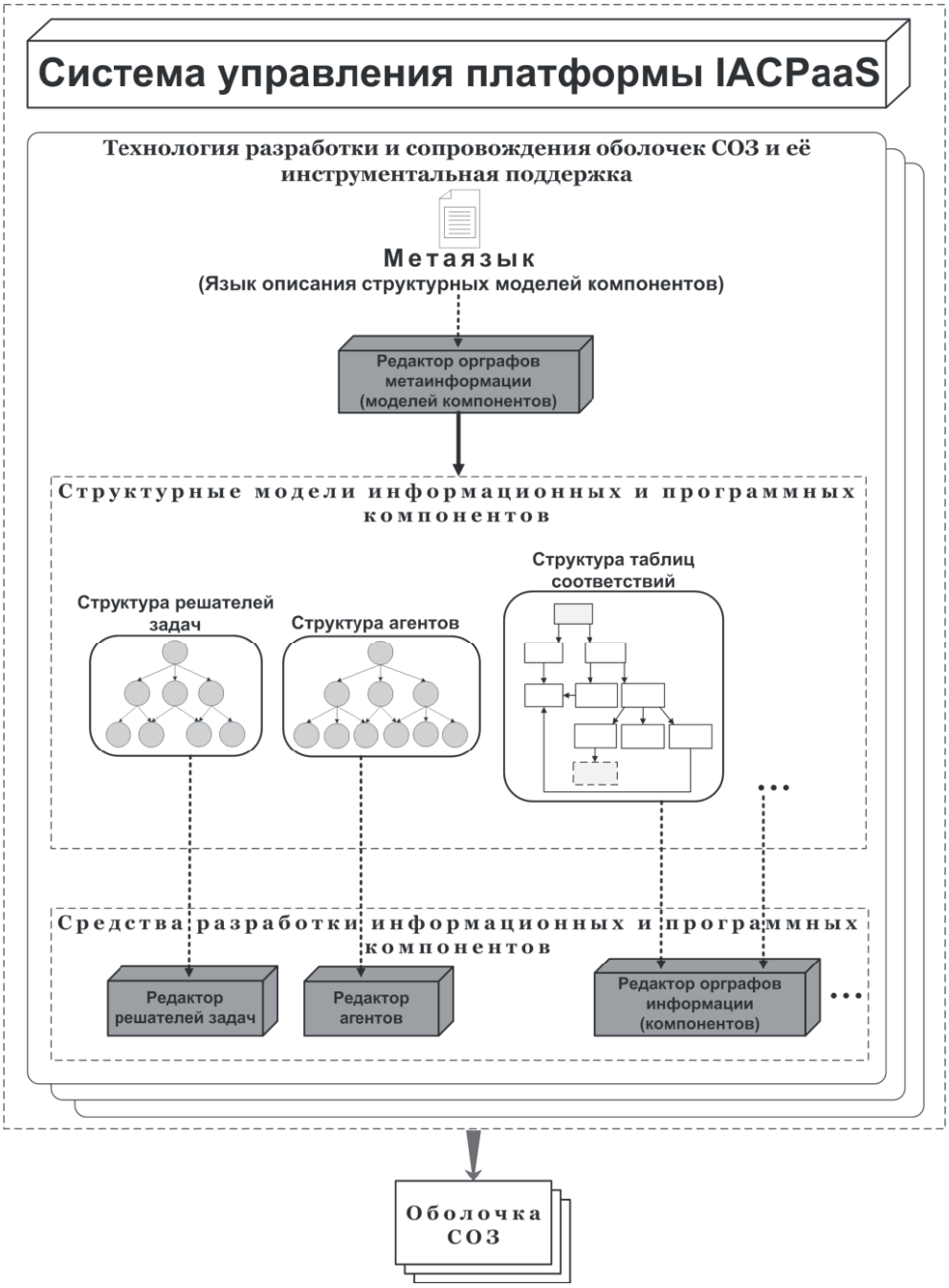

Рисунок 3 – Инструментарий для разработки оболочек СОЗ на облачной платформе IACPaaS

В основу технологии положено явное описание *орграфа информационной структуры, управляющей процессом решения задачи* (либо нескольких таких взаимосвязанных орграфов). Расширение графовой грамматики представляет собой орграф специального вида, называемый *таблицей соответствий*. Она сопоставляет вершинам орграфа грамматики программные модули (агенты), которые формируют результат решения задачи, предоставляя, возможно, для этого необходимый пользовательский интерфейс. Обращения к агентам осуществляет обеспечивающий инструментальную поддержку данной технологии *интерпретатор орграфов управляющих структур*. Он обеспечивает взаимодействие пользователя (пользовательский интерфейс) с управляющей структурой и является ядром решателей задач оболочек, разрабатываемых по данной технологии. При этом

предоставляемый пользовательский интерфейс интерпретатора может расширяться интерфейсными возможностями, которые реализуют агенты, указанные в таблице соответствий.

В случае с разработкой интерактивного верификатора доказательств орграфом управляющей структуры является орграф грамматики, представляющий модель онтологии полных доказательств. Он связан с орграфами, представляющими модель онтологии базы математических знаний и модель онтологии базы способов рассуждений. Посредством таблицы соответствий некоторым вершинам орграфа модели онтологии полных доказательств сопоставляются обращения к агентам, которые реализуют семантику методов доказательств целей, а также алгоритмы однонаправленной унификации (сопоставления) и применения подстановки. В процессе работы эти агенты формируют фрагменты орграфа доказательства в соответствии с синтаксической структурой методов доказательств (описанных в модели онтологии доказательств).

Разработка каждого информационного и программного компонента оболочки подразумевает: 1) создание с использованием *Системы управления* на web-сайте облачной платформы IACPaaS в соответствующем разделе личного кабинета разработчика оболочки новой представляющей этот компонент *единицы хранения* (орграфа грамматики или орграфа информации) нужного типа («*метаинформация*», «*информация*», «*агент*», «*решатель*») с пустым содержимым (орграф представлен единственной корневой вершиной)[3]; 2) формирование содержимого созданной единицы хранения «сверху-вниз» (начиная с корневой вершины орграфа) с использованием соответствующего инструментального средства платформы.

3.1 Разработка компонентов системы управления базами знаний

Функциональность и интерфейс редактора «*Редактор модели онтологии базы математических знаний*» полностью реализуется инструментальным средством платформы IACPaaS «*Редактор орграфов метаинформации*». Модель процесса редактирования, положенная в основу данного редактора, построена на семантике конструкций декларативного *метаязыка*, используемых для спецификации орграфов грамматики (метаинформации). Метаязык, используемый здесь как средство спецификации формальных систем, состоит из трёх подъязыков: языка описания порождающих графовых грамматик, языка описания контекстных условий и языка описания порождающих текстовых грамматик, описанных в [10, 17].

Разработка орграфа грамматики, представляющего *модель онтологии базы математических знаний*, включающей описание пустого языка представления математических знаний, состоит в создании в разделе «*Онтологии*» личного кабинета разработчика оболочки новой единицы хранения типа «*метаинформация*» и формировании её содержимого с использованием редактора «*Редактор орграфов метаинформации*» как показано на рисунке 4.

Аналогично выполняется разработка орграфа грамматики, представляющего модель онтологии базы способов рассуждений, включающей описание языка представления пропозициональных тавтологий и метаязыка (языка представления метаматематических утверждений): их абстрактного и конкретного синтаксиса, а также контекстных условий для этих языков (рисунок 5).

[3] При этом задаётся её название и описание. Если тип единицы хранения есть «*информация*», то также в качестве её *метаинформации* указывается хранящийся в Фонде платформы нужный орграф грамматики.

Рисунок 4 – Структура базы математических знаний и математического утверждения в интерфейсе инструментального средства платформы «*Редактор орграфов метаинформации*»

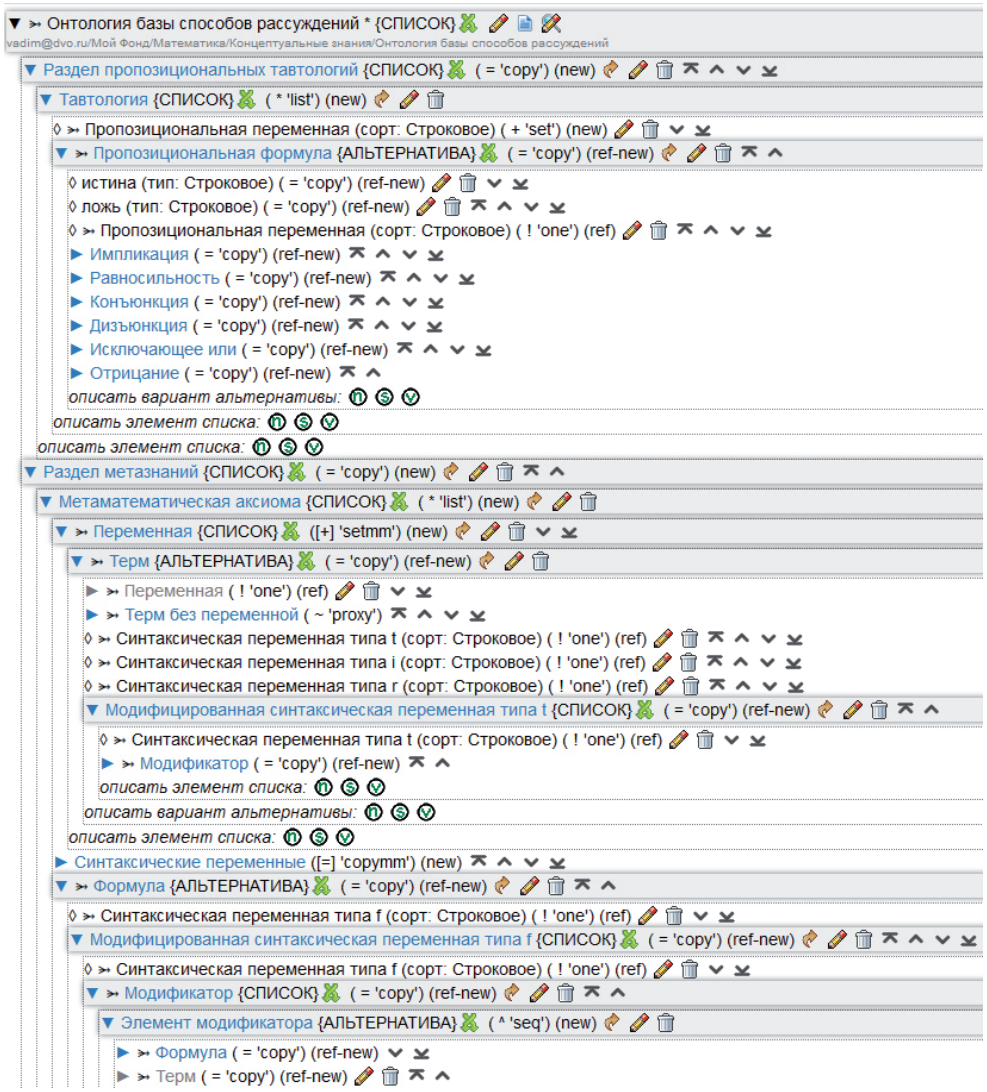

Рисунок 5 – Модель онтологии базы способов рассуждений в интерфейсе инструментального средства платформы «*Редактор орграфов метаинформации*»

«*Редактор базы математических знаний*» создаётся путём подключения к инструментальному средству платформы IACPaaS «*Редактор орграфов информации*» модели онтологии базы математических знаний в качестве информации, управляющей генерацией пользовательского интерфейса и процессом формирования базы математических знаний с проверкой её полноты и заданных в модели онтологии контекстных условий. Аналогично, «*Редактор базы способов рассуждений*» создаётся путём подключения к средству «*Редактор орграфов информации*» модели онтологии базы способов рассуждений.

3.2 Разработка интерактивного верификатора доказательств

Разработка интерактивного верификатора доказательств с интерфейсом включает создание: орграфа грамматики, представляющего модель онтологии полных доказательств; таблицы соответствий для него; решателя задач – множества агентов, обращения к которым выполняются (непосредственно или опосредованно) интерпретатором орграфов управляющих структур через таблицу соответствий; декларативной спецификации решателя задач, описывающей его связь с формальными параметрами[4], таблицей соответствий, базой математических знаний и базой способов рассуждений, а также пользовательским интерфейсом.

3.2.1 Разработка модели онтологии доказательств

Разработка орграфа грамматики, представляющего *модель онтологии доказательств*, состоит в создании в разделе «*Онтологии*» личного кабинета разработчика оболочки новой единицы хранения типа «*метаинформация*» и формировании её содержимого с использованием инструментального средства платформы «*Редактор орграфов метаинформации*» (рисунок 6).

3.2.2 Разработка таблицы соответствий для модели онтологии доказательств

Разработка орграфа информации, представляющего таблицу соответствий, состоит в создании в разделе «Решатели» новой единицы хранения типа «информация»[5] и формировании её содержимого с использованием инструментального средства платформы «Редактор орграфов информации». Орграф грамматики «Структура таблиц соответствий» подключён к данному редактору в качестве информации, управляющей процессом формирования таблиц соответствий. На рисунке 7 показана таблица соответствий между понятиями модели онтологии доказательств и агентами решателя задач оболочки, представленная в форме орграфа.

Для каждого обращения к агенту вершина «интерактивное» представляет собой логический признак. Он определяет, как в процессе формирования орграфа информации (орграфа доказательства) это обращение должно выполняться: автоматически или по инициативе пользователя. Корневая вершина орграфа закрашена серым цветом. Вершины, изображенные пунктирными прямоугольниками, принадлежат орграфам, отличным от данного орграфа, т.е. представляющим другую метаинформацию или информацию. На рисунке 7 таковыми являются вершины, принадлежащие орграфу метаинформации, представляющему модель онтологии доказательств, а также корневые вершины орграфов информации, описывающих декларативные спецификации агентов решателя задач. Символ «@» в метках вершин, представленных пунктирными прямоугольниками, разделяет метку корневой вершины стороннего орграфа и метку той его вершины (не совпадающей с первой), которая является корневой

[4] *Формальный параметр* представляет собой орграф грамматики (метаинформации), а *фактический параметр* – некоторый порожденный по ней орграф информации, обрабатываемый решателем задач.

[5] При этом в качестве метаинформации указывается хранящийся в Фонде орграф грамматики «*Структура таблиц соответствий*», описывающий абстрактный синтаксис языка представления таблиц соответствий.

вершиной повторно используемого подграфа. В квадратных скобках у вершин указаны метки соответствующих им вершин из орграфа грамматики (метаинформации).

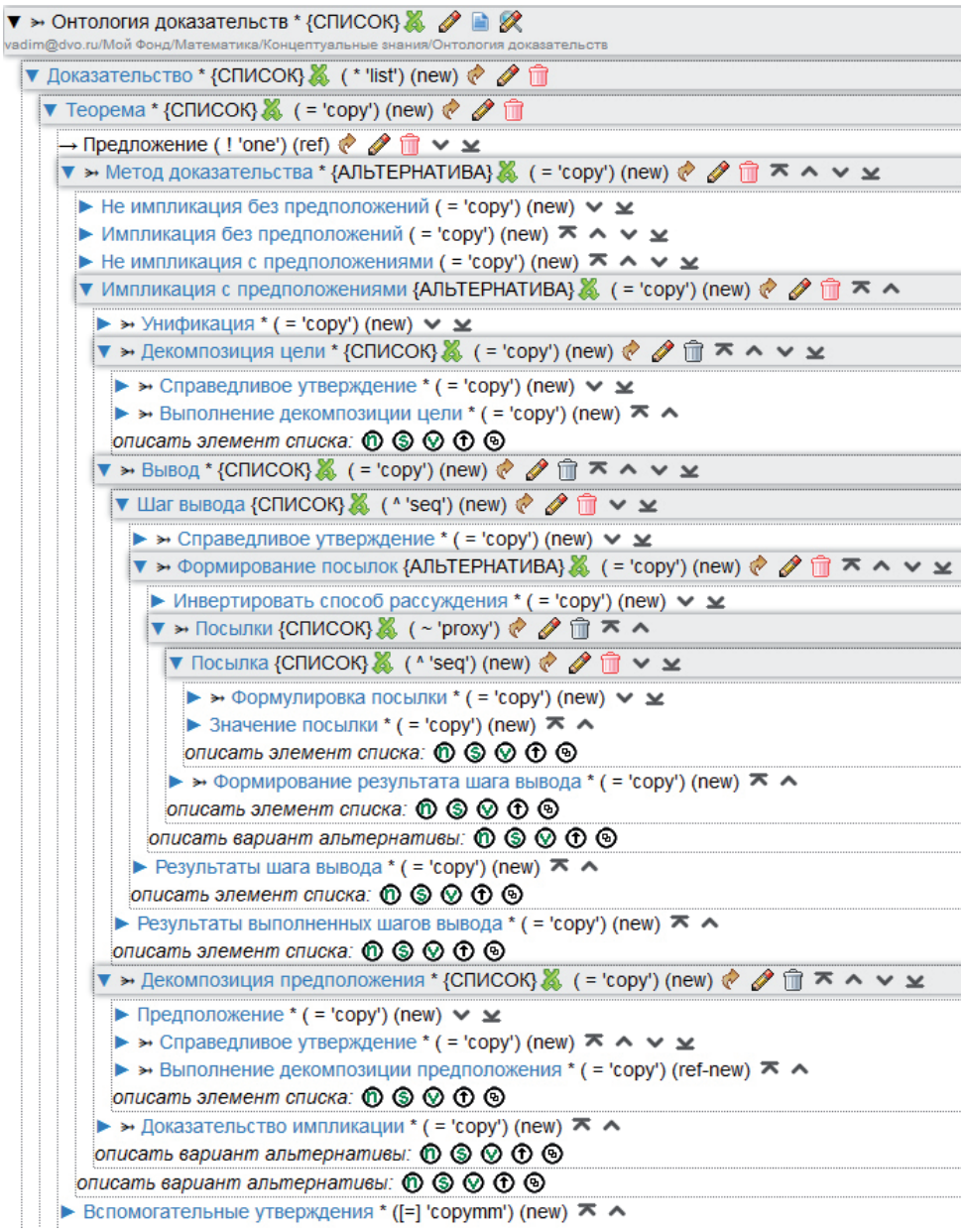

Рисунок 6 – Модель онтологии доказательств в интерфейсе инструментального средства платформы «*Редактор орграфов метаинформации*»

3.2.3 Разработка агентов решателя задач

В данном разделе описывается обобщённый процесс разработки агентов, указанных на рисунке 7. Обращения к этим агентам выполняются при создании вершин в орграфе доказательства, соответствующих вершинам из орграфа грамматики, представляющего модель онтологии полных доказательств, также указанных на рисунке 7. Шаблоны сообщений, посредством которых интерпретатор орграфов управляющих структур взаимодействует с описанными в орграфе агентами, доступны в Фонде платформы IACPaaS для использования.

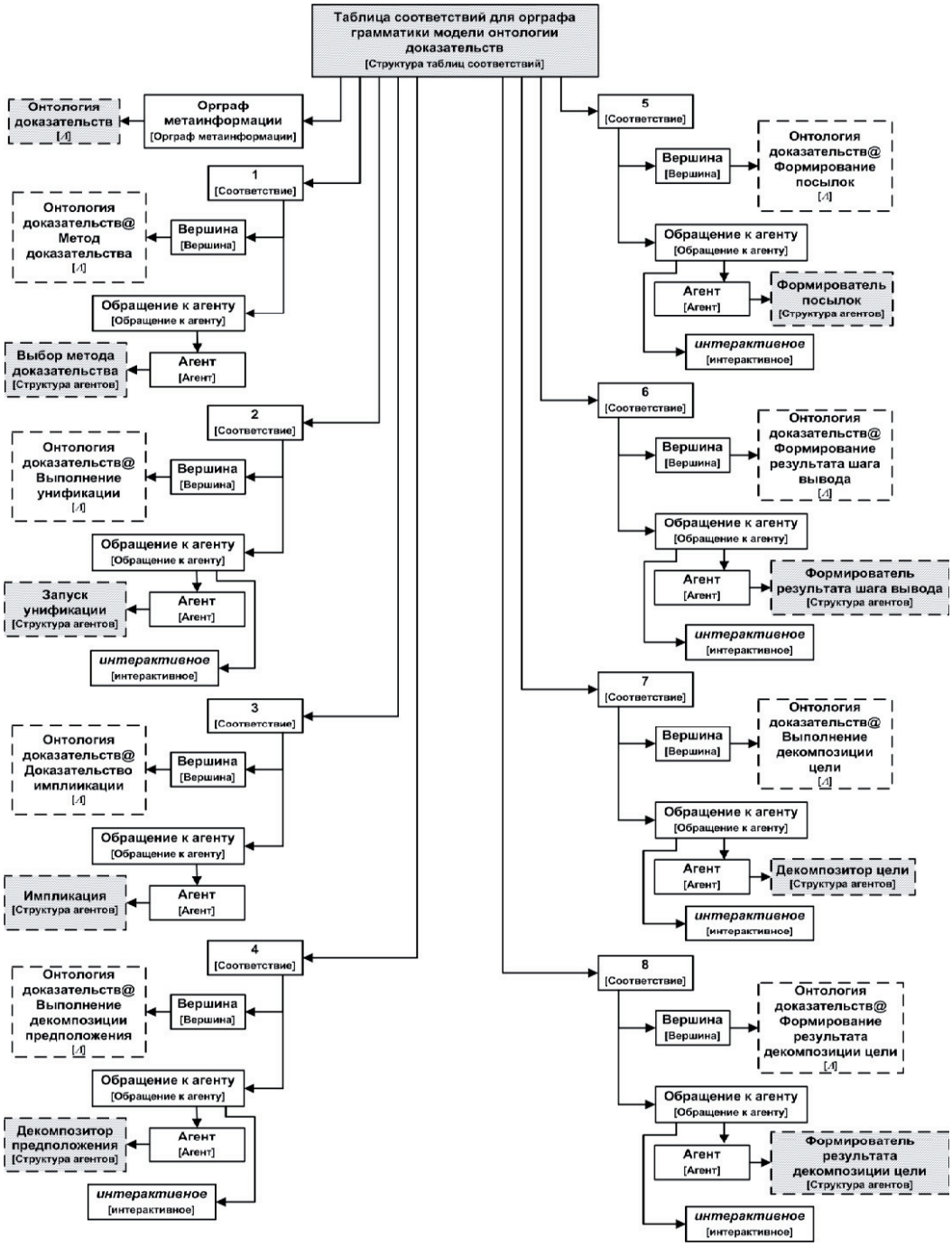

Рисунок 7 – Орграф таблицы соответствий между понятиями модели онтологии доказательств и агентами решателя задач оболочки

Разработка агента состоит в формировании в разделе «*Агенты*» единицы хранения - орграфа информации, представляющего декларативную спецификацию агента[6]; генерации заготовки исходного кода агента по его декларативной спецификации; написании исходного кода агента, реализующего его семантику (с использованием предоставляемого платформой API); получении байт-кода агента (в результате компиляции исходного кода) и загрузки его в Фонд – в соответствующую единицу хранения.

[6] Создаётся новая единица хранения типа «*агент*» (с названием, совпадающим с названием агента) и формируется её содержимое с использованием инструментального средства платформы «*Редактор агентов*».

Агент «*Выбор метода доказательства*» предоставляет множество альтернативных вариантов для выбора метода доказательства текущей цели в зависимости от синтаксической формы её утверждения (имеет ли оно форму импликации или нет) и наличия или отсутствия у неё предположений.

Если утверждение имеет вид импликации и список предположений не пуст, то *Метод доказательства* ∈ {“Унификация”, “Декомпозиция цели”, “Вывод”, “Декомпозиция предположения”, “Доказательство импликации”}. Если утверждение не является импликацией и список предположений не пуст, то *Метод доказательства* ∈ {“Унификация”, “Декомпозиция цели”, “Вывод”, “Декомпозиция предположения”}. Если утверждение имеет вид импликации и список предположений пуст, то Метод доказательства ∈ {“Унификация”, “Декомпозиция цели”, “Вывод”, “Доказательство импликации”}. Если утверждение не является импликацией и список предположений пуст, то Метод доказательства ∈ {“Унификация”, “Декомпозиция цели”, “Вывод”}.

Агент «*Запуск унификации*» подготавливает входные данные для агента, реализующего алгоритм унификации (точнее, её так называемый однонаправленный вариант – сопоставление [18]) – справедливое утверждение, выбранное пользователем из базы математических знаний или базы способов рассуждений, и частное утверждение – доказываемую цель. Принимает от агента, реализующего алгоритм унификации, результат, и если унификация не выполнена, то отображает пользователю соответствующее сообщение; если унификация выполнена, то формирует множество новых целей (если в процессе унификации появились вспомогательные утверждения).

Агент «*Импликация*» реализует правило доказательства импликации (естественного вывода).

Агент «*Декомпозитор предположения*» реализует семантику применения правила Modus ponens (правило отделения) для декомпозиции предварительно выбранного пользователем предположения из списка предположений доказываемой цели; если в процессе удачно выполненной унификации появились вспомогательные утверждения, то формирует множество соответствующих новых целей.

Агенты «*Формирователь посылок*» и «*Формирователь результата шага вывода*» реализуют семантику применения правила Modus ponens для выполнения шагов вывода. Агент «Формирователь посылок» организует выбор пользователем справедливого утверждения в форме импликации или равносильности. При этом если выбранное утверждение имеет форму равносильности, то пользователю задаётся вопрос о необходимости инвертирования данной равносильности. Если равносильность необходимо инвертировать, то правая часть равносильности считается условием, а левая – заключением. По количеству конъюнктов в условии импликации (или части равносильности, считающейся условием) агент формирует множество посылок. Формулировка каждой посылки есть формулировка соответствующего конъюнкта. Значение посылки в зависимости от наличия предположений у доказываемой цели и уже выполненных шагов вывода может быть выбрано пользователем из условий:

- если у доказываемой цели есть предположения и выполнен хотя бы один шаг вывода – базы математических знаний, списка предположений цели, результатов выполненных шагов вывода;
- если у доказываемой цели есть предположения, но еще не выполнено ни одного шага вывода – базы математических знаний, списка предположений цели;
- если у доказываемой цели отсутствуют предположения и выполнен хотя бы один шаг вывода – базы математических знаний, результатов выполненных шагов вывода;
- если у доказываемой цели отсутствуют предположения и еще не выполнено ни одного шага вывода – базы математических знаний.

Агент «*Формирователь результата шага вывода*» даёт задание на унификацию (агенту, реализующему данный алгоритм) условия справедливого утверждения с формулой, представляющей собой конъюнкцию выбранных значений посылок. Если унификация выполнена, то формирует результат шага вывода, представляющий собой результат подстановки унификатора в заключение справедливого утверждения, а также множество новых целей (если в процессе унификации появились вспомогательные утверждения). В противном случае отображает пользователю сообщение о невозможности унификации формул. После формирования результата очередного шага проверяется условие окончания процесса вывода.

Агенты «*Декомпозитор цели*» и «*Формирователь результата декомпозиции цели*» реализуют семантику применения правила Modus ponens для выполнения декомпозиции цели. Выполняется унификация доказываемой цели и заключения импликации (или части равносильности) выбранного пользователем справедливого утверждения из базы математических знаний или базы способов рассуждений. При этом если последнее имеет форму равносильности, в которой одна из её частей имеет форму конъюнкции, а другая не имеет, то частью равносильности, участвующей в унификации – заключением равносильности, является часть, не имеющая формы конъюнкции. Если унификация не выполнена, то, если справедливое утверждение имеет форму импликации, пользователю отображается сообщение о невозможности унификации формул, а если справедливое утверждение имеет форму равносильности, то даётся задание на унификацию доказываемой цели и другой части равносильности. Если унификация выполнена, то формируется множество новых целей (если в процессе унификации появились вспомогательные утверждения), в противном случае пользователю отображается сообщение о невозможности унификации формул.

Выполняется подстановка (агентом, реализующим алгоритм подстановки) вычисленного унификатора во все конъюнкты условия импликации (или части равносильности, не участвовавшей в последней унификации). Результат подстановки есть множество новых целей, количество которых равно количеству конъюнктов. Список предположений каждой новой цели совпадает со списком предположений доказываемой цели.

3.2.4 Разработка декларативной спецификации решателя задач интерактивного верификатора доказательств

Разработка в разделе «*Решатели*» орграфа информации, представляющего декларативную спецификацию решателя задач интерактивного верификатора доказательств, состоит в создании новой единицы хранения типа «*решатель*» и формировании её содержимого с использованием инструментального средства платформы «*Редактор решателей задач*».

Корневой агент решателя задаётся путём создания ссылки на корневую вершину орграфа информации в Фонде платформы IACPaaS, представляющего декларативную спецификацию агента «*Универсальный корневой агент редакторов и просмотрщиков*». Агент *Интерфейсный контроллер* решателя задаётся путём создания ссылки на корневую вершину орграфа информации в Фонде платформы IACPaaS, представляющего декларативную спецификацию агента «*Интерфейсный контроллер расширяемых редакторов и просмотрщиков единиц хранения*» (рисунок 8). Данный агент реализует функциональные возможности интерпретатора орграфов управляющих структур.

В качестве *выходного формального параметра* указывается (путём создания ссылки на его корневую вершину) орграф грамматики, представляющий модель онтологии доказательств. *Выходными фактическими параметрами*, указываемыми при создании прикладных сервисов с использованием оболочки, являются орграфы информации, представляющие формируемые по модели онтологии доказательств базы доказательств.

Среди собственных информационных ресурсов необходимо указать орграфы информации, представляющие соответственно *таблицу соответствий для орграфа грамматики модели онтологии доказательств*, *базу математических знаний* и *базу способов рассуждений*.

Связывание пользовательского интерфейса с решателем задач состоит в создании web-страницы с названием «*Интерактивный верификатор доказательств*» и содержимым, показанным на рисунке 8. Необходимые интерфейсные элементы создаются с использованием предоставляемого платформой API при написании исходного кода агентов, реализующих соответствующую семантику.

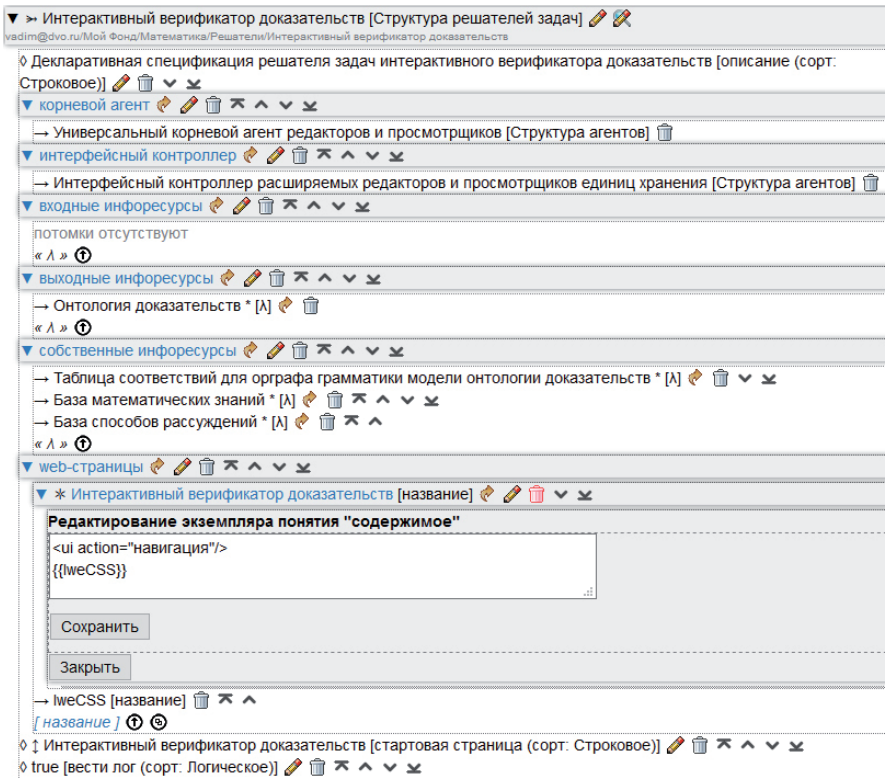

Рисунок 8 – Декларативная спецификация решателя задач интерактивного верификатора доказательств в интерфейсе инструментального средства «*Редактор решателей задач*»

3.3 Разработка пустых базы математических знаний и базы способов рассуждений

Разработка орграфа информации, представляющего базу математических знаний, состоит в создании в разделе «*Базы знаний*» новой единицы хранения типа «информация». В качестве метаинформации при этом указывается хранящийся в разделе «*Онтологии*» орграф грамматики «*Онтология базы математических знаний*». Аналогично, разработка орграфа информации, представляющего базу способов рассуждений, состоит в создании в разделе «Базы знаний» новой единицы хранения типа «информация» с указанием в качестве метаинформации орграфа грамматики «*Онтология базы способов рассуждений*».

4 Разработка начального состояния портала знаний с использованием оболочки

Разработка начального состояния портала знаний по верификации математических доказательств выполняется с использованием средств, входящих в состав оболочки, а также *Си-*

стемы управления облачной платформы IACPaaS. Структура портала знаний по верификации математических доказательств формируется с помощью Системы управления на web-сайте облачной платформы IACPaaS в личном кабинете разработчика портала. Структура представляет собой дерево разделов, терминальными вершинами которого являются информационные и/или программные компоненты (ресурсы) портала, отнесённые к определённому разделу по некоторому отличительному признаку: типу, целевому назначению и т.п. Также с помощью Системы управления начальное состояние портала знаний переносится в Фонд платформы, где он становится общедоступным, и назначается ответственный за данный портал администратор, который контролирует его дальнейшее развитие.

Разработка *начального состояния* (*ядра*) языка представления математических знаний состоит в расширении орграфа грамматики, представляющего модель онтологии базы математических знаний, конкретными типами формул и термов с использованием редактора оболочки «*Редактор модели онтологии базы математических знаний*» (рисунок 9).

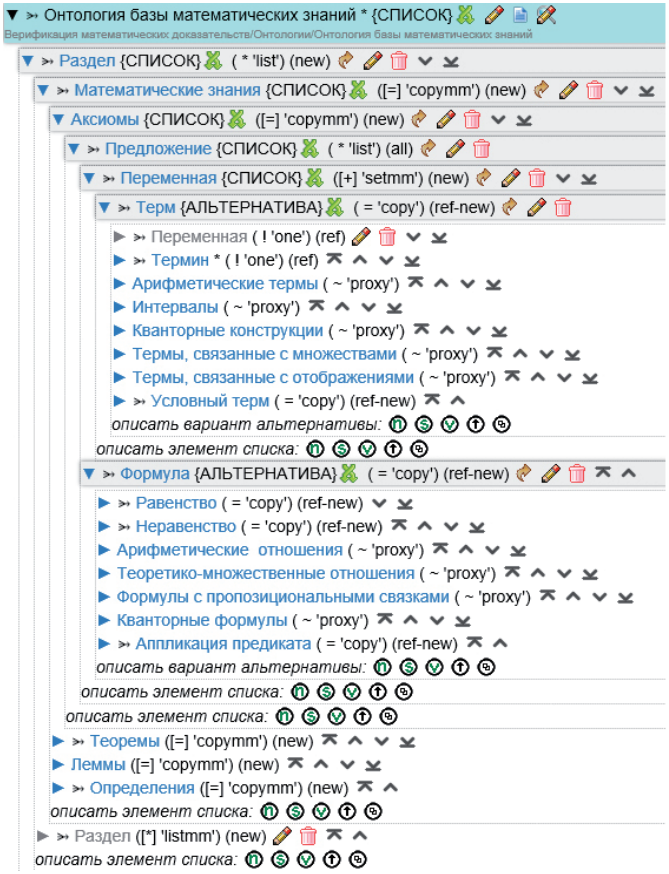

Рисунок 9 – Начальное состояние языка представления математических знаний в интерфейсе средства портала «*Редактор модели онтологии базы математических знаний*»

Разработка *начального состояния* (*ядра*) базы математических знаний состоит в формировании содержимого представляющего её орграфа информации с использованием редактора оболочки «*Редактор базы математических знаний*» (рисунок 10). Разработка *начального состояния* (*ядра*) базы способов рассуждений состоит в формировании содержимого представляющего её орграфа информации с использованием редактора оболочки «*Редактор базы способов рассуждений*» (рисунок 11).

Начальное состояние базы математических знаний содержит раздел «*Арифметика*», который содержит 29 аксиом, 17 теорем, 3 определения и 4 леммы. Начальное состояние

базы формализованных способов рассуждений содержит 29 способов рассуждений: 5 пропозициональных тавтологий и 24 метаматематических утверждения.

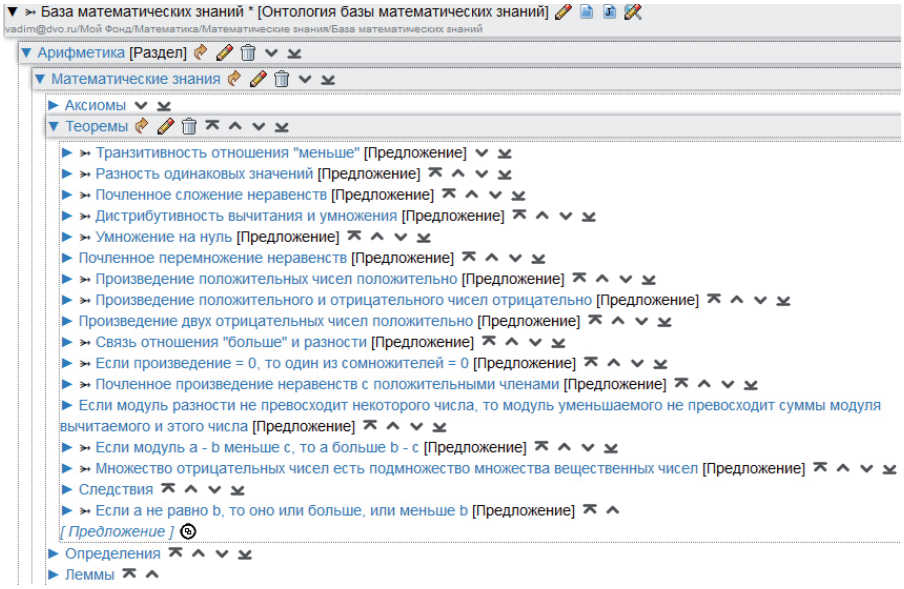

Рисунок 10 – Начальное состояние базы математических знаний в интерфейсе средства портала «*Редактор базы математических знаний*»

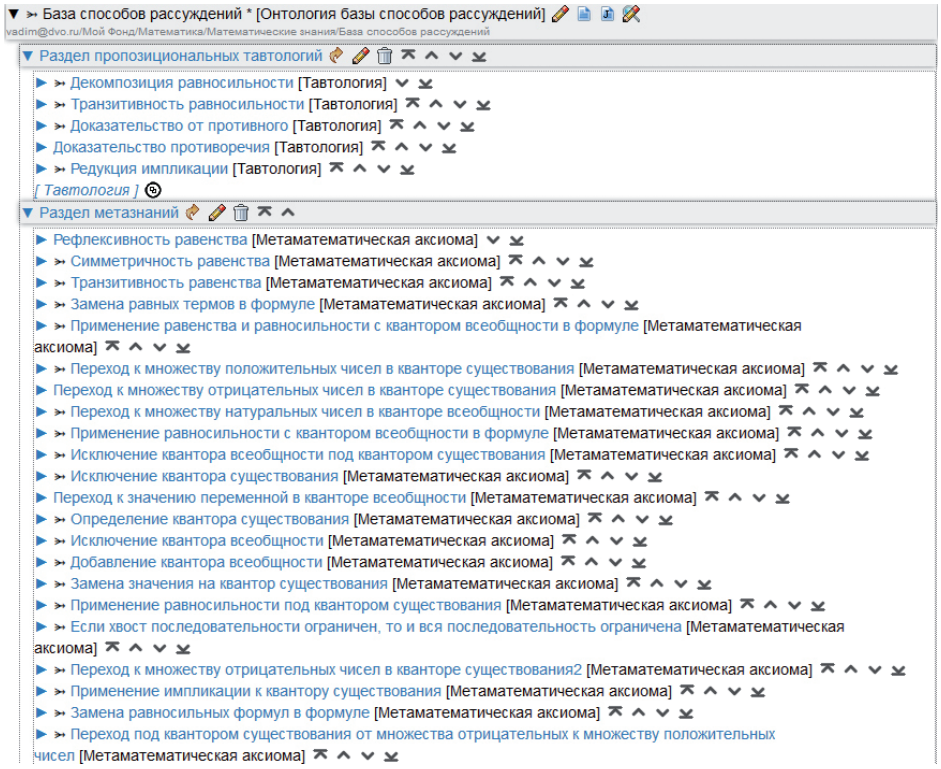

Рисунок 11 – Начальное состояние базы способов рассуждений в интерфейсе средства портала «*Редактор базы способов рассуждений*»

Все подграфы, корневые вершины которых соответствуют вершинам «*Предложение*», «*Тавтология*» и «*Метаматематическая аксиома*» в соответствующих орграфах грамматик, отображаются в текстовом представлении. На рисунке 12 показан пример для пропозициональных тавтологий.

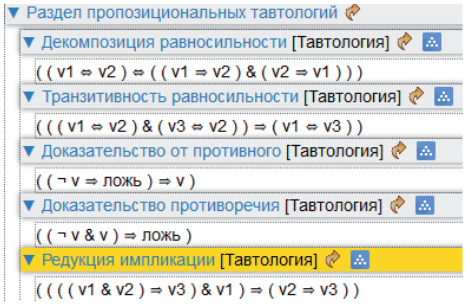

Рисунок 12 – Текстовое представление пропозициональных тавтологий в базе способов рассуждений в интерфейсе редактора «*Редактор базы способов рассуждений*»

Все редакторы и решатель задач оболочки включаются в состав программных компонентов портала знаний.

5 Использование текущего состоянием портала знаний исследователями

Для использования текущего состояния портала знаний по верификации математических доказательств пользователю сначала необходимо пройти процедуру регистрации на web-сайте платформы IACPaaS, после чего для него создаётся *Личный кабинет*, включающий *Персональный Фонд*. С помощью *Системы управления* пользователь может произвольным образом структурировать свой Персональный Фонд: создать древовидную структуру разделов.

Дальше у пользователя появляется возможность *скопировать текущее состояние портала знаний* из Фонда платформы в Персональный Фонд своего личного кабинета. Получение копии текущего состояния портала знаний состоит в следующем. В разделе Персонального Фонда «*Загрузки*» создаётся копия единицы хранения, представляющей решатель задач «*Интерактивный верификатор доказательств*». При создании копии решателя он связывается с копией текущего состояния базы математических знаний, для которой метаинформацией (орграфом грамматики) является уже копия модели онтологии базы математических знаний; с копией текущего состояния базы способов рассуждений; а также со всеми остальными хранимыми в портале знаний единицами хранения, указанными в декларативной спецификации решателя задач «*Интерактивный верификатор доказательств*». Полученные копии единиц хранения можно перенести из раздела «*Загрузки*», в нужные разделы Персонального Фонда. Редакторы соответствующих баз и модели онтологии базы математических знаний в силу способа своей реализации не копируются: они доступны из личных кабинетов и настроены уже на копии этих информационных компонентов.

Результатом выполнения операции *копирования текущего состояния портала знаний* является также создание в личном кабинете пользователя персонального прикладного сервиса. Создание сервиса подразумевает создание орграфа информации, представляющего персональное хранилище доказательств[7], и его связывание с копией единицы хранения, представляющей решатель задач оболочки, с помощью которого это хранилище можно наполнять. Связывание состоит в создании специального вида единицы хранения, представляющей запускаемую и исполняемую на платформе IACPaaS сущность – *сервис*. На рисунках 13, 14 продемонстрированы примеры работы такого сервиса при формировании доказательства теоремы о том, что если последовательность имеет предел, то она является ограниченной.

[7] Его метаинформацией является орграф грамматики, представляющий модель онтологии доказательств.

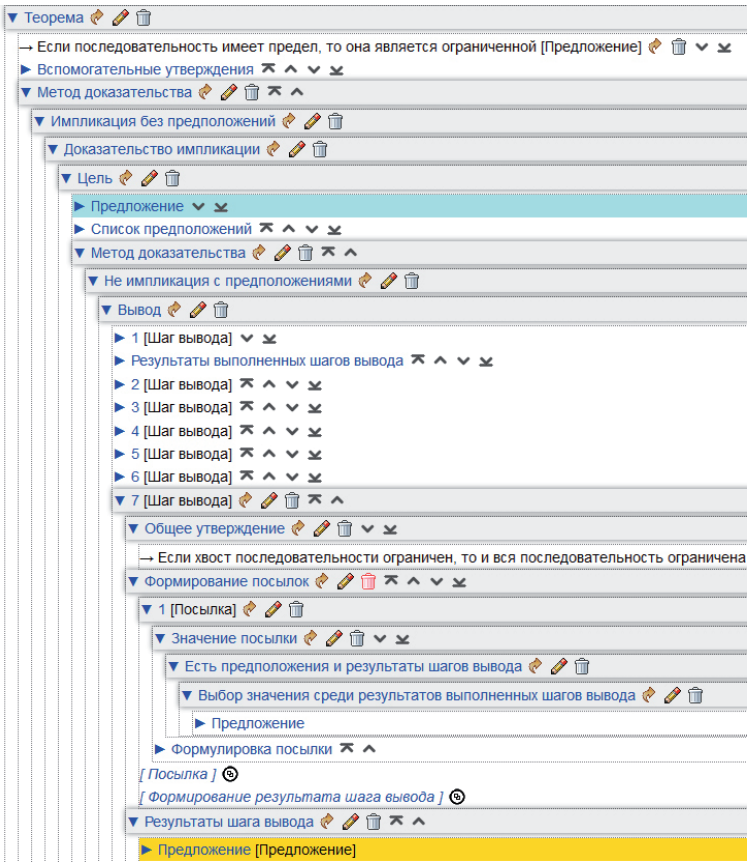

Рисунок 13 – Работа прикладного сервиса при формировании доказательства теоремы (пример)

С этого момента пользователь-исследователь может независимо развивать свою персональную базу математических знаний и базу способов рассуждений, а также язык представления математических знаний в Персональном Фонде с помощью соответствующих редакторов и доказывать математические утверждения.

6 Изменение текущего состояния портала знаний администратором

Изменение текущего состояния портала знаний может выполняться двумя способами: по инициативе пользователя либо по инициативе администратора. В обоих случаях для этого предназначен специальный сервис администратора, с помощью которого последний может получить доступ на чтение к персональным базам личных порталов пользователей. Данный сервис позволяет просматривать персональные базы математических знаний, показывая только те математические утверждения (вместе с разделами, в которых они находятся), доказательства которых являются полными. Эти доказательства должны быть рекурсивно-полными, т.е. используемые в них математические утверждения, отсутствующие в общей базе математических знаний, тоже должны иметь полные доказательства.

Если в доказательствах используются способы рассуждений, отсутствующие в общей базе способов рассуждений, то они должны быть верифицированы администратором, прежде чем эти способы попадут в соответствующие общие базы.

Если при формулировке некоторых математических утверждений используются конструкции языка представления математических знаний, отсутствующие в общей онтологии базы математических знаний, и администратор согласен с этими утверждениями, то они включаются в общую онтологию.

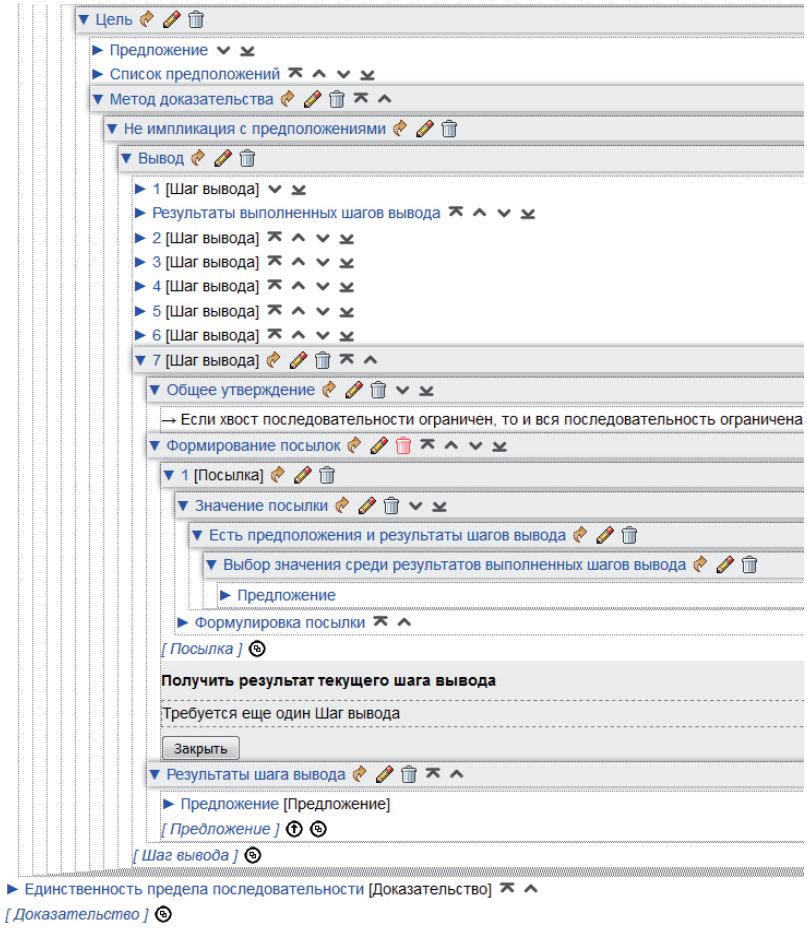

Рисунок 14 – Результат работы агента «Формирователь результата шага вывода» (пример)

Заключение

В работе представлена концептуальная архитектура оболочки для интерактивных систем верификации математических доказательств и создаваемого с её помощью развиваемого тематического портала знаний. Описан процесс реализации всех программных и информационных компонентов оболочки на облачной платформе IACPaaS с использованием предоставляемых ею технологий и инструментальных средств их поддержки. Рассмотрен процесс разработки начального состояния портала знаний, способ использования текущего состояния портала знаний заинтересованными членами математического сообщества, а также механизмы изменения текущего состояния портала его администратором.

В основу оболочки и, как следствие, портала знаний положена явно представленная расширяемая формально-логическая система, приближенная к математической практике конструирования доказательств. Модель онтологии базы математических знаний, база математических знаний и база способов рассуждений расширяются с помощью соответствующих специализированных редакторов, входящих в состав портала знаний. При расширении модели онтологии новыми конструкциями языка представления знаний под эти изменения автоматически адаптируется управляемый ей соответствующий редактор, а решатель задач портала инвариантен по отношению к такому расширению.

В развитии портала знаний по верификации математических доказательств могут принимать участие все заинтересованные члены математического сообщества. Это осуществляется с помощью системы *личных кабинетов* платформы IACPaaS, в которых

каждый пользователь может независимо развивать свою персональную копию текущего состояния общего портала знаний, и контролируется администратором портала знаний с помощью специального сервиса портала. Таким образом поддерживается синхронизация коллективного доступа к информационным компонентам общего портала и обеспечивается защита этих компонентов от их умышленного или непреднамеренного разрушения пользователями.

В результате развития портала может быть накоплена библиотека способов рассуждений, которыми математики пользуются в своей практике при доказательстве теорем. Эта библиотека может анализироваться специалистами по математической логике на предмет правильности содержащихся в ней способов рассуждений.

Благодарности

Работа выполнена при частичной поддержке РФФИ (проекты 17-07-00299 и 18-07-01079) и КПФИ «Дальний Восток» (проект 18-5-078).

Список источников

[1] ***Maric, F.*** A Survey of Interactive Theorem Proving / F. Maric // Zbornik Radova, 2015, 18(26). Pp. 173-223.
[2] ***Asperti A.*** A Survey on Interactive Theorem Proving. 2009. – http://www.cs.unibo.it/~asperti/SLIDES/itp.pdf.
[3] ***Harrison, J.*** History of Interactive Theorem Proving / J. Harrison, J. Urban, F. Wiedijk // In Jörg Siekmann (ed.) Handbook of the History of Logic, 2014, vol. 9: Computational Logic. Elsevier. P.135-214.
[4] ***Paulson, L.*** Logic and Computation: Interactive Proof with Cambridge LCF / L.C. Paulson // Cambridge University Press, New York, 1987.
[5] ***Owre, S.*** PVS: A prototype verification system / S. Owre, J.M. Rushby, N. Shankar // In D. Kapur (ed.) Automated Deduction – CADE-11, LNCS 607, 1992, Springer, Berlin-Heidelberg. P.748-752.
[6] ***Bertot, Y.*** Interactive Theorem Proving and Program Development. Coq'Art: The Calculus of Inductive Constructions / Y. Bertot, P. Cast´eran // Springer, Berlin-Heidelberg, 2004.
[7] About The Twelf Project. – http:// twelf.org/.
[8] ***Harrison, J.*** HOL Light: An Overview / J. Harrison // In S. Berghofer et al. (eds.) Theorem Proving in Higher Order Logics, LNCS 5674, 2009, Springer, Berlin-Heidelberg. - P.60-66.
[9] ***Рязанов, А.Е.*** Система Буцефал: комбинирование дедуктивных процедур и описание стратегий поиска доказательств / А.Е. Рязанов. – Новосибирск, 1998. Препр./ Сиб. Отд-ние РАН. ИСИ; №50 – 42 с.
[10] ***Клещев, А.С.*** Теоретические основы оболочки для интерактивных систем верификации интуитивных математических доказательств / А.С. Клещев, В.А. Тимченко // Онтология проектирования. - 2018. - Т. 8, № 2(28). - С. 219-239. - DOI: 10.18287/2223-9537-2018-8-2-219-239.
[11] ***Ring, M.*** Collaborative Interactive Theorem Proving with Clide / M. Ring, C. Lüth // In Klein G., Gamboa R. (eds) Interactive Theorem Proving. ITP 2014. Lecture Notes in Computer Science, 2014, vol. 8558. Springer, Cham. P.467-482.
[12] ***Obua, S.*** ProofPeer – A Cloud-based Interactive Theorem Proving System. 2012. – https://arxiv.org/ftp/arxiv/papers/1201/1201.0540.pdf.
[13] ***Gribova, V.*** The IACPaaS cloud platform: features and perspectives / V. Gribova, A. Kleschev, Ph. Moskalenko, V. Timchenko, L. Fedorischev, E. Shalfeeva // In proc. of Second Russia and Pacific Conference on Computer Technology and Applications (25-29 Sept. 2017, Vladivostok, Russky Island, Russia). IEEE, 2017. Pp.80-84. ISBN: 978-153861206-4 - DOI: 10.1109/RPC.2017.8168073.
[14] ***Kumar, S.*** Importance of Expert System Shell in Development of Expert System / S. Kumar, R. Prasad // International journal of innovative research & development, 2015, vol. 4, issue 3. P.128-133.
[15] ***Загорулько, Ю.А.*** Семантическая технология разработки интеллектуальных систем, ориентированная на экспертов предметной области / Ю.А. Загорулько // Онтология проектирования. - 2015. - Т. 5, № 1(15). - С.30-46.
[16] ***Грибова, В.В.*** Управляемая графовыми грамматиками разработка оболочек интеллектуальных сервисов на облачной платформе IACPaaS / В.В. Грибова, А.С. Клещёв, Ф.М. Москаленко, В.А. Тимченко, Л.А. Федорищев, Е.А. Шалфеева // Программная инженерия. - 2017. Т.8, №10. - С.435-447.

[17] ***Gribova, V.V.*** A Two-level Model of Information Units with Complex Structure that Correspond to the Questioning Metaphor / V.V. Gribova, A.S. Kleshchev, F.M. Moskalenko, V.A. Timchenko // Automatic Documentation and Mathematical Linguistics. 2015. Vol. 49. No.5. - P.172-181.
[18] ***Knight, K.*** Unification: A Multidisciplinary Survey / K. Knight // ACM Computing Surveys, 1989, 21(1). - P.93-124.

IMPLEMENTATION OF THE SHELL AND KNOWLEDGE PORTAL FOR MATHEMATICAL PROOFS VERIFICATION ON THE IACPaaS PLATFORM

A.S. Kleschev[1], V.A. Timchenko[2]

Institute of Automation and Control Processes of the FEB RAS, Vladivostok, Russia
[1]kleschev@iacp.dvo.ru, [2]vadim@dvo.ru

Abstract

The paper presents a conceptual architecture of the shell for interactive systems for mathematical proofs verification and an evolutionary thematic knowledge portal created with its help. The process of implementation of all software and information components of the shell on the IACPaaS cloud platform with the use of its technologies and tools of their support is described. The process of development of the initial state of the knowledge portal for mathematical proofs verification using shell tools is considered. The way to use the knowledge portal by members of the mathematical community as well as mechanisms for changing the state of the portal by its administrator is discussed. The initial state of the knowledge portal includes: the ontology model of the mathematical knowledge base, including the specification of the initial state of the language for mathematical knowledge representation, the editor for the ontology model of the mathematical knowledge base, the mathematical knowledge base editor, the base of methods of reasoning editor, and the problem solver of the shell that implements the process of constructing proofs in terms of ontology model of proofs. The initial state of the knowledge portal also includes the initial state of the mathematical knowledge base and the initial state of the base of the methods of reasoning. The evolvement of the knowledge portal consists in the development of the three listed information components. All interested members of the mathematical community can participate in this process. It is implemented with the use of personal cabinets of the IACPaaS platform, where each user can independently develop his personal copy of the current state of the common knowledge portal. The transfer of new results to the common knowledge portal is controlled by its administrator.

Key words: *verification of intuitive proofs, specialized software shell, knowledge portal, knowledge base editor, cloud services.*

Citation: *Kleschev AS, Timchenko VA*. Implementation of the shell and knowledge portal for mathematical proofs verification on the IACPaaS platform [In Russian]. *Ontology of designing*. 2018; 8(3): 427-448. - DOI: 10.18287/2223-9537-2018-8-3-427-448.

Acknowledgment

The work was partially supported by the Russian Foundation for Basic Research (projects 17-07-00299 and 18-07-01079) and the KFD "Far East" (project 18-5-078).

References

[1] ***Maric F.*** A Survey of Interactive Theorem Proving. Zbornik Radova. 2015; 18(26): 173-223.
[2] ***Asperti A.*** A Survey on Interactive Theorem Proving. 2009. - http://www.cs.unibo.it/~asperti/SLIDES/itp.pdf.
[3] ***Harrison J, Urban J, Wiedijk F.*** History of Interactive Theorem Proving. In Jörg Siekmann (ed.) Handbook of the History of Logic, Computational Logic. Elsevier; 2014; 9: 135-214.
[4] ***Paulson LC.*** Logic and Computation: Interactive Proof with Cambridge LCF. Cambridge University Press, New York, 1987.
[5] ***Owre S, Rushby JM, Shankar N.*** PVS: A prototype verification system. In D. Kapur (ed.) Automated Deduction – CADE-11, LNCS 607, Springer, Berlin-Heidelberg; 1992; 748-752.

[6] ***Bertot Y, Cast´eran P.*** Interactive Theorem Proving and Program Development. Coq'Art: The Calculus of Inductive Constructions, Springer, Berlin-Heidelberg, 2004.
[7] About The Twelf Project. Source: ⟨http:// twelf.org/⟩.
[8] ***Harrison J.*** HOL Light: An Overview. In S. Berghofer et al. (eds.) Theorem Proving in Higher Order Logics, LNCS 5674, Springer, Berlin-Heidelberg; 2009; 60-66.
[9] ***Ryazanov AE.*** System butsephalas: defining proof-search strategies and combining deductive procedures [In Russian]. Novosibirsk: Prepr. / Sib. Branch of the RAS. IIS; 1998; 50: 42.
[10] ***Kleschev AS, Timchenko VA.*** Theoretical foundations of the shell for interactive systems of intuitive mathematical proofs verification [In Russian]. Ontology of designing. 2018; 8(2): 219-239. DOI: 10.18287/2223-9537-2018-8-2-219-239.
[11] ***Ring M, Lüth C.*** Collaborative Interactive Theorem Proving with Clide. In Klein G., Gamboa R. (eds) Interactive Theorem Proving. ITP 2014. Lecture Notes in Computer Science, 2014, vol. 8558. Springer, Cham. P.467-482.
[12] ***Obua S.*** ProofPeer – A Cloud-based Interactive Theorem Proving System. 2012. Source: ⟨https://arxiv.org/ftp/arxiv/papers/1201/1201.0540.pdf⟩.
[13] ***Gribova V, Kleschev A, Moskalenko Ph, Timchenko V, Fedorischev L, Shalfeeva E.*** The IACPaaS cloud platform: features and perspectives. In proc. of Second Russia and Pacific Conference on Computer Technology and Applications (25-29 Sept. 2017, Vladivostok, Russky Island, Russia). IEEE; 2017: 80-84. ISBN: 978-153861206-4 DOI: 10.1109/RPC.2017.8168073.
[14] ***Kumar S, Prasad R.*** Importance of Expert System Shell in Development of Expert System. International journal of innovative research & development 2015; 4(3): 128-133.
[15] ***Zagorulko YuA.*** Semantic technology for development of intelligent systems oriented on experts in subject domain [In Russian]. Ontology of designing. 2015; 5(1): 30-46.
[16] ***Gribova VV, Kleshchev AS, Moskalenko FM, Timchenko VA, Fedorishchev LA, Shalfeyeva EA.*** A Graph Grammar Managed Development of Intelligent Service Shells on the IACPaaS Cloud Platform [In Russian]. Software Engineering 2017; 10: 435-447.
[17] ***Gribova VV, Kleshchev AS, Moskalenko FM, Timchenko VA.*** A Two-level Model of Information Units with Complex Structure that Correspond to the Questioning Metaphor. Automatic Documentation and Mathematical Linguistics. 2015; 49(5): 172-181.
[18] ***Knight K.*** Unification: A Multidisciplinary Survey. ACM Computing Surveys 1989; 21(1): 93-124.

Сведения об авторах

Клещёв Александр Сергеевич, 1940 г. рождения. Окончил математико-механический факультет Ленинградского государственного университета в 1964 г., д.ф.-м.н. (1990). Главный научный сотрудник лаборатории интеллектуальных систем Института автоматики и процессов управления Дальневосточного отделения РАН, профессор, заслуженный деятель науки РФ. В списке научных трудов более 300 работ в области искусственного интеллекта, информатики, медицинской и биологической кибернетики.

Alexander Sergeevich Kleschev (b. 1940) graduated from the Leningrad State University in 1964, Professor's degree (1990). He is Chief Researcher at lab. of intelligent systems in the Institute of Automation and Control Processes of the FEB RAS, professor, Honored Scientist of Russian Federation. He is co-author of more than 300 publications in the field of biological and medical cybernetics, informatics and AI.

Тимченко Вадим Андреевич, 1983 г. рождения. Окончил Институт математики и компьютерных наук Дальневосточного государственного университета по специальности «Математическое обеспечение и администрирование информационных систем» (2005), к.т.н. (2011). Старший научный сотрудник лаборатории интеллектуальных систем Института автоматики и процессов управления Дальневосточного отделения РАН. В списке научных трудов более 50 работ в области искусственного интеллекта, проблемно-ориентированных систем, основанных на знаниях, специализированных программных моделей и систем.

Vadim Andreevich Timchenko (b. 1983) graduated from the Far Eastern State University (Vladivostok-city) on a speciality «Mathematical support and administration of the informative systems» (2005), Ph.D. (2011). He is Senior Researcher at lab. of intelligent systems in the Institute of Automation and Control Processes of the FEB RAS. He is co-author of more than 50 publications in the field of AI, informatics, program models, technologies and systems.