



УДК 004.822:514

КАКОЙ ДОЛЖНА БЫТЬ ПАРАДИГМА РЕШЕНИЯ ЗАДАЧ НА ОСНОВЕ БАЗ ЗНАНИЙ?

Грибова В.В., Клещев А.С.

Федеральное государственное бюджетное учреждение науки Институт автоматизации и процессов управления Дальневосточного отделения Российской академии наук, г. Владивосток, Россия

gribova@iacp.dvo.ru,

kleshev@iacp.dvo.ru

По аналогии с парадигмой решения задач в рамках алгоритмического подхода и современными требованиями к разработке интеллектуальных систем предлагается парадигма решения задач на основе баз знаний. Далее анализируется соответствие существующих технологий предложенной парадигме.

Ключевые слова: интеллектуальные системы; семантические технологии; онтологии; базы знаний.

Введение

В настоящее время при решении задач на компьютере наиболее широко используется алгоритмический подход. Для этого подхода сложилась общепринятая парадигма решения задач. Однако далеко не все задачи могут быть решены в рамках алгоритмического подхода, что стимулирует разработку других подходов к решению задач. Одним из таких подходов является решение задач на основе баз знаний, который привел к появлению нового класса программных средств – интеллектуальных систем или систем с базами знаний. Вместе с тем, для этого подхода до сих пор не предложена парадигма решения задач аналогичная таковой для алгоритмического подхода.

В настоящей работе предлагается парадигма для решения задач на основе баз знаний и показывается, в какой степени существующие технологии разработки интеллектуальных систем ей удовлетворяют.

1. Парадигма решения задач в рамках алгоритмического подхода

Решение задач на компьютере обычно связано с необходимостью получения какой-либо информации. Формально эта потребность в информации, получаемой в результате решения задачи, выражается через определение множества Y всех возможных результатов решения этой задачи. Чтобы получить информацию $y \in Y$, необходима некоторая исходная информация, которая формально представляется исходными данными x

задачи. Множество всех допустимых исходных данных задачи, которое должно быть формально определено, обозначим через X . Формальная постановка задачи или ее спецификация есть предикат $P(x, y)$, где $x \in X$, а $y \in Y$. Для допустимой спецификации задачи должно быть доказано утверждение о существовании ее решения:

$$\forall x \in X \exists y \in Y P(x, y);$$

желательно также, чтобы для нее было справедливо (доказано) и утверждение о единственности ее решения:

$$\forall x \in X \exists! y \in Y P(x, y).$$

В случае, если справедливы оба эти утверждения, разрабатывается алгоритм решения задачи, который представляет собой реализацию на алгоритмическом языке такого всюду определенного функционального отображения $A: X \rightarrow Y$, что справедливо утверждение о правильности этого алгоритма:

$$\forall x \in X P(x, A(x)).$$

Если же для спецификации задачи утверждение о единственности ее решения не является справедливым, то обычно алгоритм ее решения представляет собой реализацию такого всюду определенного функционального отображения $A: X \rightarrow 2^Y$, что справедливы утверждения о нахождении этим алгоритмом всех решений задачи:

$$\forall x \in X \forall y \in A(x) P(x, y)$$

$$\text{и } \forall x \in X y \notin A(x) \rightarrow \neg P(x, y).$$

Если при решении задачи пользователь использует программу, правильно реализующую алгоритм A , то он отвечает за правильность входных данных x программы; в этом случае он получает правильный результат решения задачи $y = A(x)$.

Считается, что любая технология решения задач на компьютере в рамках алгоритмического подхода должна удовлетворять приведенной в этом разделе парадигме.

2. Парадигма решения задач на основе баз знаний

Построим парадигму решения задач на основе баз знаний по аналогии с парадигмой решения задач в рамках алгоритмического подхода, приведенной в предыдущем разделе, иллюстрируя результаты ее применения на двух типах интеллектуальных систем, достаточно далеких друг от друга – системах компьютерной диагностики заболеваний и системах поиска математических доказательств.

Любая интеллектуальная система предназначена для решения некоторой задачи, поэтому для нее должна быть сформулирована и формализована потребность в информации через определение множества возможных решений Y этой задачи. Для системы компьютерной диагностики заболеваний это множество диагнозов и их объяснений, а для системы поиска математических доказательств – множество полных доказательств. Поскольку при использовании интеллектуальных систем результат решения задачи представляется в некоторой концептуальной системе, определение множества Y содержит определение соответствующей системы понятий (онтологии), а $y \in Y$ представляет собой множество экземпляров этих понятий. При этом множество Y должно быть определено таким образом, чтобы предикат принадлежности $y \in Y$ был вычислимым, а форма представления информации $y \in Y$ должна быть понятной пользователям интеллектуальной системы без каких-либо посредников или дополнительной их подготовки. Последнее может быть достигнуто лишь в случае, если онтология, содержащаяся в определении множества Y представляет собой формализацию концептуальной системы пользователей.

Таким же образом следовало бы определить множество X допустимых исходных данных задачи. Для системы компьютерной диагностики заболеваний это должно быть множество всех реальных историй болезни, а для системы поиска математических доказательств – множество теорем. Однако из этих примеров видно, что множество X не может быть определено точно – множество всех возможных историй болезни и множество всех теорем нетривиальной математической теории неизвестны. Поэтому обычно определяется более широкое множество $X' \supseteq X$; для системы компьютерной диагностики заболеваний это множество объектов, имеющих вид историй болезни

и представимых на соответствующем искусственном языке, а для системы поиска математических доказательств – множество правильных математических предложений, также представимых на соответствующем языке.

Кроме того, в случае использования интеллектуальных систем таким же способом должно быть определено множество допустимых баз знаний $K(X', Y)$, а также спецификация задачи – предикат $P(x, k, y)$, где $x \in X'$, $k \in K(X', Y)$, $y \in Y$, причем предикаты принадлежности $x \in X'$ и $k \in K(X', Y)$ должны быть вычислимыми, а форма представления информации $x \in X'$ и $k \in K(X', Y)$ должна быть понятной экспертам и пользователям интеллектуальной системы без каких-либо посредников или дополнительной их подготовки. Последнее может быть достигнуто лишь в случае, если онтология, содержащаяся в определениях множеств X' и $K(X', Y)$ представляют собой формализацию концептуальных систем экспертов и пользователей. Для системы компьютерной диагностики заболеваний базой знаний являются описания проявлений различных заболеваний в зависимости от возможных, индивидуальных особенностей пациентов, воздействий различных событий (например, лечебных мероприятий) на проявления заболеваний, причинно-следственных отношений между заболеваниями и т.п. [Черняховская, 1983], а спецификация задачи компьютерной диагностики – описание отношения между историями болезни, базами знаний, диагнозами пациентов и их объяснениями [Клещев и др., 2005, Клещев и др., 2006]. Для системы поиска математических доказательств базой знаний может быть некоторая аксиоматическая система и множество правил вывода, а спецификацией задачи – описание отношения между теоремами, аксиоматическими системами, правилами вывода и доказательствами. Кроме того, онтология X' и $K(X', Y)$ должны использоваться для управления формированием $x \in X'$ и $k \in K(X', Y)$, а результат вычисления последних отношений принадлежности должен контролироваться.

В случае интеллектуальных систем особенностью спецификации задачи является то, что для нее не является справедливым утверждение о существовании решения. Для системы компьютерной диагностики заболеваний неявно предполагается, что существует «правильная» база знаний $k^* \in K(X', Y)$ такая, что

$$\forall x \in X \exists y \in Y P(x, k^*, y);$$

однако эта «правильная» база знаний неизвестна, а для любой другой базы знаний утверждение о существовании решения не обязано быть справедливым. Кроме того, это утверждение может быть нарушено и для $x \in X' \setminus X$. Для системы поиска математических доказательств, напротив, «правильная» база знаний $k^* \in K(X', Y)$ известна, однако входные данные $x \in X'$ – это математические предложения, относительно которых неизвестно,

являются они теоремами (в этом случае $\exists y \in Y P(x, k^*, y)$) или нет (в этом случае такого $y \in Y$ не существует). Тем более, для интеллектуальных систем не является справедливым утверждение о единственности решения даже в случае его существования.

Хотя для спецификации задачи утверждение о существовании решения не является справедливым, для нее можно сформулировать более слабое утверждение:

$$\exists k^* \in K(X', Y) \forall x \in X \exists y \in Y P(x, k^*, y),$$

а в некоторых случаях и слабое утверждение о единственности решения:

$$\exists k^* \in K(X', Y) \forall x \in X \exists! y \in Y P(x, k^*, y).$$

Эти утверждения являются лишь предположениями и не могут быть доказаны.

Для системы компьютерной диагностики заболеваний – это предположения о существовании «правильной» базы знаний k^* , в терминах которой может быть объяснен правильный диагноз любой реальной истории болезни, и о единственности диагноза у каждого пациента, а для системы поиска математических доказательств – это предположение о том, что любое математическое предложение (теорема), истинное в любой модели аксиоматической системы k^* , имеет доказательство в этой аксиоматической системе (как правило, не единственное).

В предположении справедливости для спецификации задачи этого более слабого утверждения о существовании решения можно разрабатывать алгоритм решения задачи, который представляет собой реализацию на алгоритмическом языке такого частично определенного функционального отображения

$$A: \langle X', K(X', Y) \rangle \rightarrow Y$$

(или $A: \langle X', K(X', Y) \rangle \rightarrow 2^Y$), для которого справедливо (может быть доказано) «слабое» утверждение о правильности этого алгоритма:

$$\forall x \in X' \forall k \in K(X', Y) P(x, k, A(x, k)).$$

Для системы компьютерной диагностики заболеваний такой алгоритм для истории болезни x (если это реальная история болезни, полученная в результате наблюдения врачом реального пациента, то $x \in X$) и базы знаний $k \in K(X', Y)$ (например, сформированной экспертом) находит все возможные диагнозы пациента x и их объяснения в терминах базы знаний k . Для системы поиска математических доказательств такой алгоритм для математического предложения $x \in X'$ и базы знаний k^* , являющейся теорией некоторой аксиоматической системы, должен находить одно из возможных доказательств (если оно существует).

Поскольку «сильное» утверждение о существовании решения для спецификации задачи не является справедливым, а алгоритм решения

задачи $A(x, k)$ может применяться к любым входным данным $x \in X'$ и к любой базе знаний $k \in K(X', Y)$, этот алгоритм обладает некоторыми нежелательными свойствами. Для системы компьютерной диагностики заболеваний при некоторых исходных данных и базе знаний множество решений задачи $A(x, k)$ может не содержать «правильного» решения (т.е. такого, которое затем было верифицировано иным способом), содержать «неправильные» решения (т.е. такие, которые будут отвергнуты в процессе последующей верификации) или даже быть пустым (в случае, если исходные данные x противоречат базе знаний k). Для системы поиска математических доказательств, если математическое предложение $x \in X' \setminus X$ (т.е. не является теоремой), то алгоритм решения задачи $A(x, k^*)$ может работать бесконечно время (следствие теоремы Черча).

Если при решении задачи пользователь использует интеллектуальную систему с базой знаний $k \in K(X', Y)$, правильно реализующую алгоритм A , то он, как и раньше, отвечает за правильность входных данных x программы; однако в этом случае он может получить и неправильный, и неточный результат решения задачи $y = A(x, k)$, либо даже не получить никакого результата.

Для того, чтобы технология интеллектуальных систем была жизнеспособной, она должна включать процедуры, позволяющие уменьшать долю случаев, в которых система обладает указанными выше нежелательными свойствами [Грибова и др., 2010]. Эти процедуры могут быть разными для различных типов интеллектуальных систем. Приведем примеры таких процедур для двух рассматриваемых типов интеллектуальных систем.

Для системы компьютерной диагностики заболеваний назовем базой прецедентов репрезентативную выборку $\{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$, где $x_i \in X$ – реальная история болезни, а $y_i \in Y$ – объяснение ее верифицированного диагноза (для $i = 1, \dots, n$). Будем называть эту базу прецедентов оценкой правильности базы знаний $k \in K(X', Y)$, $C(k) = \{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$, если

$$\forall i \in \{1, \dots, n\} y_i \in A(x_i, k)$$

(аналогично можно определить оценку точности базы знаний). Будем говорить, что база знаний $k \in K(X', Y)$ приближается к «правильной» базе знаний k^* , если в оценке ее правильности $C(k)$ параметр n неограниченно растет. Процедуру, которая обеспечивает приближение базы знаний к «правильной» базе знаний, будем называть системой управления базой знаний. Она может быть автоматической, т.е. реализующей такое отображение, которое базе прецедентов $\{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$ ставит в соответствие такую базу знаний $k \in K(X', Y)$, что эта база прецедентов оказывается оценкой точности этой базы знаний $C(k) = \{\langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle\}$ [Клещев и др., 2012]. Либо эта процедура может быть человеко-

машинной: для базы прецедентов $\{ \langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle \}$ система управления находит множество возможных баз знаний $k \in K(X', Y)$, такое, что эта база прецедентов оказывается оценкой точности каждой из этих баз знаний $C(k) = \{ \langle x_1, y_1 \rangle, \dots, \langle x_n, y_n \rangle \}$, а эксперт выбирает «наилучшую» базу знаний из этого множества по своему усмотрению.

Системы поиска математических доказательств эффективно могут применяться в задаче верификации моделей интуитивных доказательств [Клещев, 2011]. Назовем (простой) моделью интуитивного доказательства математического предложения $x \in X'$ конечную последовательность математических предложений $x_1, x_2, \dots, x_{n-1}, x_n, x$. Будем говорить, что интуитивное доказательство верифицировано с использованием базы знаний $k \in K(X', Y)$, если построены: доказательство $y_1 \in Y$ математического предложения $x_1 \in X$ с использованием базы знаний $k \in K(X', Y)$; доказательство $y_2 \in Y$ математического предложения $x_2 \in X$ с использованием базы знаний $k \cup \{x_1\}$; ...; доказательство $y_n \in Y$ математического предложения $x_n \in X$ с использованием базы знаний $k \cup \{x_1, x_2, \dots, x_{n-1}\}$; доказательство $y_{n+1} \in Y$ математического предложения $x \in X$ с использованием базы знаний $k \cup \{x_1, x_2, \dots, x_{n-1}, x_n\}$. Будем называть оценкой эффективности базы знаний $k \in K(X', Y)$ множество математических предложений, для которых система поиска математических доказательств с этой базой знаний построила доказательства. Системой управления базой знаний, расширяющей оценку эффективности базы знаний, будем называть процедуру, состоящую из двух действий: если доказано некоторое математическое предложение $x \in X$, то это предложение включается в базу знаний; если при этом построено доказательство $y \in Y$, то оно обобщается до метадоказательства my [Клещев, 2007], и это метадоказательство включается в базу знаний. В этом случае процедура построения доказательства математического предложения x_i , входящего в модель интуитивного доказательства предложения x , может состоять из четырех шагов: поиск в базе знаний предложения x_i , для которого x_i является частным случаем (в случае успеха система управления базой знаний включает x_i в базу знаний); поиск в базе знаний метадоказательства my_i , конкретизацией которого является полное доказательство y_i предложения x_i (в случае успеха система управления базой знаний включает x_i в базу знаний); автоматический поиск полного доказательства y_i предложения x_i с использованием множества предложений и правил вывода, содержащихся в базе знаний (в случае успеха система управления базой знаний включает x_i в базу знаний, обобщает y_i до метадоказательства my_i и включает это метадоказательство в базу знаний); интерактивное построение полного доказательства y_i предложения x_i с использованием множества предложений и правил вывода, содержащихся в базе

знаний (в случае успеха система управления базой знаний включает x_i в базу знаний, обобщает y_i до метадоказательства my_i и включает это метадоказательство в базу знаний).

3. Соответствие технологий разработки интеллектуальных систем парадигме решения задач на основе баз знаний

Обсудим, в какой степени современные технологии разработки интеллектуальных систем соответствуют описанной выше парадигме. Рассмотрим следующие технологии: (а) основанную на использовании оболочек экспертных систем [Bachant et al., 1984, Compton, 1988]; (б) основанную на использовании WebProtege [Gennari et al., 2003, Musen, 2007]; (с) основанную на использовании платформы IASPaas [Грибова и др., 2011, Грибова и др., 2013а, Грибова и др., 2013б].

Технологии (а) и (б) не поддерживают спецификацию множеств Y, X' и $K(X', Y)$ таким образом, чтобы предикаты принадлежности $y \in Y, x \in X'$ и $k \in K(X', Y)$ были вычислимыми. Технология (с) поддерживает такую спецификацию - множества Y, X' и $K(X', Y)$ специфицируются как метаинформация на метаязыке, для которого определена и поддерживается логическая семантика. Форма представления знаний понятна специалистам соответствующих предметных областей при использовании технологий (б) и (с).

Технологии (а) и (б) не предлагают аппарата для представления спецификации задач в форме предиката $P(x, k, y)$, где $x \in X', k \in K(X', Y), y \in Y$. Для технологии (а) разработаны методы управления базами правил, а для технологии (б) такая задача вообще не рассматривалась. Технология (с) предлагает аппарат для представления спецификации задач в форме предиката, с использованием которого можно сформулировать слабое утверждение о существовании решения задачи, а также слабое утверждение о правильности алгоритма решения задачи. Имеется опыт доказательства такого утверждения для достаточно сложного случая [Клещев и др., 2013]. Кроме того, разработан прототип системы автоматического управления базой знаний для системы компьютерной диагностики заболеваний [Клещев и др., 2012].

Инструментальная поддержка технологии (а) включает инструменты для формирования баз правил и проблемно-независимый решатель задач (оболочку); разработка интерфейса не поддерживается (фиксированный интерфейс является частью оболочки). Кроме того, эти инструменты не являются облачными, как и создаваемые с их помощью интеллектуальные системы.

Инструментальная поддержка технологии (б) включает редактор для поддержки формирования баз знаний в декларативной форме [Crubezy, 2004],

множество решателей некоторых классов [Chandrasekaran, 1986] задач и средства адаптации этих решателей к базам знаний. Только редактор является облачным. Проблема формирования баз знаний экспертами без посредников или дополнительного их обучения решена в этом редакторе не в полной мере.

Инструментальная поддержка технологии (с) включает редактор IWE для поддержки формирования баз знаний в декларативной форме (в котором проблема формирования баз знаний экспертами без посредников или дополнительного их обучения решена в полной мере), средства разработки решателей задач в виде систем взаимодействующих агентов, ориентированные на представление баз знаний, а также средства разработки интерфейсов таких решателей. Все эти средства являются облачными, как и создаваемые с их помощью интеллектуальные системы.

Заключение

В настоящее время осознана острая необходимость разработки технологии и инструментальной поддержки создания интеллектуальных систем. Стандартная технология, основанная на алгоритмическом подходе и соответствующие ей средства разработки не применимы к разработке интеллектуальных систем. В первую очередь это связано с тем, что в архитектуре интеллектуальных систем имеется дополнительный компонент – база знаний, которая должна разрабатываться и сопровождаться экспертами предметной области и быть им понятной. В тоже время для интеллектуальных систем не предложено парадигмы решения задач, в результате чего все технологии разрабатываются и развиваются, как правило, интуитивно. Это ведет к тому, что невозможно объективно оценить их и наметить пути дальнейшего развития данной области исследований. Предложенная в работе парадигма может стать, в случае ее принятия заинтересованным сообществом, основой для оценивания технологий и дальнейшего развития технологий и инструментальной поддержки разработки интеллектуальных систем.

Работа выполнена при финансовой поддержке ДВО РАН (инициативный научный проект 12-I-ОНИТ-04) и при финансовой поддержке РФФИ (проект 12-07-00179-а).

Библиографический список

- [Bachant et al., 1984] Bachant, J. McDermott, J. "R1 revisited: four years in the trenches", The AI Magazine (Fall 1984; Fall), pp.21-32
- [Chandrasekaran, 1986] Chandrasekaran B. Generic tasks in knowledge-based reasoning High-level building blocks for expert system design // IEEE Expert, 1986. 1(1). - Pp.23-30.
- [Compton, 1988] Compton, P. Horn, K. Quinlan, R. Lazarus, L. "Maintaining an expert system". Proceedings of the fourth Australian Conference on Applications of Expert Systems, (1988), pp.110-129.

[Crubezy, 2004] Crubezy M., Musen M.A. Ontologies in support of problem-solving // In S. Staab & R. Studer. (Eds.) Handbook on ontologies. - Berlin: Springer, 2004. - Pp. 321-341.

[Gennari et al., 2003] Gennari, J.H.; Musen, M.A.; Fergerson, R.W., etc. The evolution of Protégé: An environment for knowledge-based systems development/International Journal of Human-Computer Studies. 2003. 58(1):89-123

[Musen, 2007] Musen M. Technology for building intelligent systems: from psychology to engineering // Nebraska Symposium on Motivation. 2007. 02. 52. Pp. 145-84.

[Грибова и др. 2010] Грибова В.В., Клещев А.С., Шалфеева Е.А. Управление интеллектуальными системами // Известия РАН. Теории и системы управления. 2010. № 6. С. 122-137.

[Грибова и др., 2011] Грибова В.В., Клещев А.С., Крылов и др. Проект IACPaaS – развиваемый комплекс для разработки, управления и использования интеллектуальных систем // Искусственный интеллект и принятие решений. – 2011. – №1.

[Грибова и др., 2013a] Грибова В.В., Клещев А.С. Технология разработки интеллектуальных сервисов, ориентированных на декларативные предметные базы знаний. Часть 1. Информационные ресурсы // Информационные технологии. 2013. №9. С. 7-11.

[Грибова и др., 2013b] Грибова В.В., Клещев А.С. Технология разработки интеллектуальных сервисов, ориентированных на декларативные предметные базы знаний. Часть 2. Решатель задач. Пользовательский интерфейс // Информационные технологии. 2013. №10. С. 10-14.

[Клещев, 2007] Клещев А.С. Модель аналогии между математическими доказательствами // Проблемы управления. – 2007. – №1. – С.20-24.

[Клещев, 2011] Клещев А.С. Операционная модель интуитивных доказательств // Проблемы управления. 2011. №1. С. 2-7.

[Клещев и др., 2005] Клещев А.С., Москаленко Ф.М., Черняховская М.Ю. Модель онтологии предметной области «Медицинская диагностика». Часть 1. Неформальное описание и определение базовых терминов // Научно-техническая информация. Сер. 2. – 2005. – № 12. – С. 1-7.

[Клещев и др., 2006] Клещев А.С., Москаленко Ф.М., Черняховская М.Ю. Модель онтологии предметной области «Медицинская диагностика». Часть 2. Формальное определение причинно-следственных связей, причин значений признаков и причин заболеваний // Научно-техническая информация. Сер. 2. – 2006. – № 2. – С. 19-30.

[Клещев и др. 2012] Клещев А.С., Смагин С.В. Задачи индуктивного формирования знаний для онтологии медицинской диагностики // Научно-техническая информация. Сер.2. - М.: ВИНТИ РАН, 2012. №1. С.9-21.

[Клещев и др., 2013] Клещев А.С., Москаленко Ф.М. Доказательство корректности алгоритма решения частной задачи медицинской диагностики // Информатика и системы управления. 2013. №2(36). С. 134-146.

WHAT SHOULD BE THE PARADIGM OF TASK SOLVING FOR THE KNOWLEDGE-BASED SYSTEMS?

Gribova V, Kleschev A.

*Institute of Automation and Control Processes,
Far Eastern Branch of Russian Academy of
Sciences, Vladivostok
gribova@iacp.dvo.ru,
kleschev@iacp.dvo.ru*

A paradigm for tasks solving based on knowledge bases is proposed by analogy with the paradigm of solving tasks within the algorithmic approach and modern requirements for the development of intelligent systems. Adequacy of the existing technologies to the new paradigm is analyzed.

Introduction

The algorithmic approach of tasks solving is most widely used. This approach has the generally accepted paradigm for tasks solving. However, not all problems can be solved using the algorithmic approach so other approaches are developing. One of such approaches is approach for tasks solving based on knowledge bases. However, this approach still does not have the paradigm similar to that for the algorithmic approach .

The paper proposes a paradigm for tasks solving based on knowledge bases and existing technologies are compared using this paradigm.

Main Part

The main includes three paragraphs. The first one describes the paradigm of task solving within the algorithmic approach. The second part is devoted to a new paradigm for tasks solving based on knowledge bases. In the third part existing technologies are analyzed in accordance with the new paradigm.

Task solving on a computer is usually associated with the need to obtain any information. Formally, this need for information that is obtained by task solving, is described in terms of the definition of the set Y of all possible results of solving this task. To obtain information $y \in Y$, we need input data x . The set of all possible input data of the task is X . A formal statement of the task or its specification is a predicate $P(x, y)$, where $x \in X$, and $y \in Y$. For the task specification a statement of existence of a solution must be proven. It is also desirable to prove a statement about uniqueness of its solution. If both these statements are true, an algorithm for task solving is developed.

The paradigm of tasks solving based on knowledge bases has the same sets X and Y . Moreover , intelligent systems have knowledge base , so in the same way a set of admissible knowledge bases $K(X, Y)$ should be determined, and the specification of the task - the predicate $P(x, k, y)$. A form of representation of information in the knowledge base must be understandable for domain experts and users of intelligent systems without any intermediaries or special training. In the case of intelligent systems feature the task specification is that it is not a fair statement about the existence of solutions. Although the specifications of the task a statement of the existence of solutions is not fair, for it is possible to formulate a weaker statement, and in some cases a weak statement about the uniqueness of a solution. These statements are only predictions and can not be proven.

In assuming of validity for the task specification this weaker statement about existence of solutions the programmer can develop an algorithm for the task solving. If an intelligent system with a knowledge base implements the correct algorithm? Then the user is responsible for the correctness of input data, but in this case he can get wrong and inaccurate result of the task solving, or do not even get any result .

If we want the technology of intelligent systems to be viable, it must include procedures to reduce the proportion of cases in which the system has undesirable properties. These procedures may be different for different types of intelligent systems.

In the third section of the paper we discuss the adequacy of modern technologies for intelligent system development to the paradigm. Considered the following technologies : (a) a technology based on the use of expert system shells ; (b) a technology based on the use of WebProtege; a technology based on the use of the platform IACPaaS. Technologies (a) and (b) does not support the specification of sets of input and output data, the knowledge base so that the predicates of accessories were computable. The technology (c) supports this specification. Form of knowledge representation is understandable for the domain experts in technologies (b) and (c). Technologies (a) and (b) not provide specifications of the task in the form of a predicate. Methods of control of the rule base have been developed for the (a) technology. This problem has not considered for the (b) technology. The technology (c) provides the apparatus for the task specification in the form of a predicate, using which it is possible to formulate a weak assertion of the existence of solutions, as well as weaker statement about the correctness of the algorithm for the task solving. Tools for the (a) technology includes tools for creation of a rule base and a problem-independent task solver (shell), the development of an interface is not supported. In addition, these tools and intelligence systems are not cloudy. Tool for the (b) technology includes an editor of knowledge bases, a set task solvers of some. The editor is cloudy. Tools for the (c) technology includes an editor of knowledge bases, tools for development of task solvers in the form of a set of agents and user interfaces. All of these tools are cloudy.

Conclusion

There is the urgent need to develop technologies and tools to support the process of development of intelligent systems. The general technology based on the algorithmic approach and the corresponding development tools do not applicable for intelligent system development. This is primarily due to the fact that the architecture of intelligent systems has an additional component - the knowledge base, which should be developed and maintained by domain experts and be understandable for them. At the same time a paradigm of task solving have not developed for intelligent systems, so all of the technologies are intuitively developing. This leads to the fact that it is impossible to objectively evaluate them and idefine ways of further development of this field of research. Proposed in the paper paradigm may become, if it is accepted by the community concerned , the basis for the evaluation of technologies and the further progress of technolgis and tools for development of intelligent systems.