

МОДЕЛИ КОНТЕКСТНО-УПРАВЛЯЕМЫХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В ДИНАМИЧЕСКИХ СТРУКТУРИРОВАННЫХ ОБЛАСТЯХ

СМИРНОВ А.В., ЛЕВАШОВА Т.В., ПАШКИН М.П.

УДК 004.89:004.048

Смирнов А.В., Левашова Т.В., Пашкин М.П. Модели контекстно-управляемых систем поддержки принятия решений в динамических структурированных областях.

Аннотация. Анализируются известные подходы к описанию и формированию контекста в различных информационных средах. Предлагается двухуровневая модель управления контекстом для организации интеллектуальной поддержки принятия решений в динамических структурированных областях. Приводится модель описания ресурсов открытой информационной среды для моделирования текущей ситуации. Определена технологическая модель контекстно-управляемой системы интеллектуальной поддержки принятия решений.

Ключевые слова: контекстно-управляемые системы поддержки принятия решений, динамические структурированные области, управление контекстом, разнородные информационные ресурсы, web-сервисы, онтология.

Smirnov A.V., Levashova T.V. Pashkin M.P., Models of context-aware decision support systems in dynamic structured domains.

Abstract. Some existing approaches to representation and organization of contexts in different information systems are analyzed. A two-level context management model for intelligent decision support in dynamic structured domains is proposed. A model for description of information resources of an open information environment is given. A technology model of context-aware decision support system is designed.

Key words: context-aware decision support systems, dynamic structured domains, context management, heterogeneous information resources, Web-services, ontology.

1. Введение. Динамические структурированные области характеризуются хорошо формализуемыми знаниями и большими объемами анализируемой и обрабатываемой динамической информации, которая поступает от разнородных источников (базы данных, web-страницы, сенсоры и др.). Принятие решений в таких областях связано с трудностью анализа текущей ситуации в связи с ее непрерывным изменением, а также с тем, что из-за большого числа источников информации человек, не имеющий исполнительных средств ее обработки, не в состоянии воспринять и обработать весь получаемый объем.

Таким образом, для систем поддержки принятия решений, направленных на функционирование в динамических областях, с одной стороны, требуются методы и средства, позволяющие получать информацию о среде, в которой принимаются решения; с другой – для эффективного принятия решений требуются специальные методы об-

работки потоков поступающей в систему поддержки принятия решений информации.

Сужение объемов анализируемой информации достигается за счет использования в системах поддержки принятия решений модели контекста и обеспечения пользователей системы вариантами решений, которые могут быть приняты в текущей ситуации. С появлением онтологических моделей представления знаний появилась возможность получения контекста, основанного на структурированных знаниях проблемной области. Интеграция в такой контекст информации, получаемой от разнородных источников, делает его моделью, которая описывает текущую ситуацию в явном виде и содержит знания, на основании которых может быть сгенерировано множество возможных решений. Знания и информация, содержащиеся в таком контексте, становятся интерпретируемыми для систем поддержки принятия решений и для пользователей таких систем.

Системы, построенные в соответствии с таким подходом, называются контекстно-управляемыми системами интеллектуальной поддержки принятия решений (СИППР). Контекстно-управляемые СИППР используют или логически выводят входные характеристики контекста для подбора и предоставления пользователю релевантной информации [1]. В основе функционирования таких систем лежат методы и модели управления контекстом.

Данная статья посвящена анализу работ в области управления контекстом. На основании этого анализа дается определение контекста, выявляются его виды и свойства, формулируется цель управления контекстом, описываются способы представления контекста. В результате формируется спецификация требований к управлению контекстом в СИППР, которая послужила основой для разработки двухуровневой модели управления контекстом для организации интеллектуальной поддержки принятия решений в динамических структурированных областях, выбора модели описания ресурсов открытой информационной среды и определения технологической модели контекстно-управляемой СИППР.

2. Контекст

2.1. Понятие контекста. Определения контекста варьируются в зависимости от области его применения. В СИППР контекст используется для выявления и представления лицу, принимающему решения, релевантной информации и при интерпретации данных для анализа возможного развития ситуаций с целью прогнозирования и объяснения

непредвиденных событий и оказания помощи при управлении последствиями таких событий.

Одно из определений контекста известно из лингвистики, где под контекстом понимается относительно законченный по смыслу отрывок текста или устной речи, в пределах которого наиболее точно и конкретно выявляется смысл и значение отдельно входящего в него слова, фразы, совокупности фраз [2].

Область искусственного интеллекта, непосредственно связанная с лингвистикой, занимается обработкой текстов на естественном языке. В данной области контекст рассматривается как набор высказываний (пропозиций), описывающих некоторую ситуацию, схему решения задачи, способ достижения цели. Интерпретация текстов производится на основании логического вывода. Контекст накладывает *ограничения* на логический вывод и уточняет информационное содержание высказывания на естественном языке [3].

В области приобретения знаний под контекстом понимается область, в границах которой приобретались знания и, следовательно, в пределах которой их можно считать достоверными [3, 4]. Использование контекста в области приобретения знаний *ограничивает* область достоверности приобретенных знаний и указывает, в какой момент использование знаний является наиболее корректным [5].

В сфере, связанной с обучением компьютера (*machine learning*), контекст рассматривается как некоторая область, влияющая на поведение прикладной системы. При изменении контекста поведение системы должно изменяться в соответствии с изменениями в контексте. При этом контекст не влияет непосредственно на процесс обучения, но *ограничивает* его, предоставляя информацию о контекстных переменных, изменение значений которых может повлиять на поведение системы, или о функциях системы, использование которых может быть некорректным в изменившемся контексте [3, 4].

В области человеко-машинного, программного и компьютерного взаимодействия контекст считается свойством, присущим взаимодействию. Контекст рассматривается как накопленные знания, которыми обменивались объекты, вовлеченные во взаимодействие. Взаимодействие и контекст *ограничивают* друг друга: контекст конкретной ситуации определяет поведение объектов в этой ситуации; поведение, в свою очередь, изменяет контекст ситуации [3, 4].

В разделе искусственного интеллекта, связанном с работой с базами знаний и онтологиями, контекстом считается часть базы знаний, которая участвует в процессе логического вывода, решении некоторой

задачи [3, 4, 6]. Использование контекста позволяет извлекать данные и знания, актуальные в данный момент и значимые для решения конкретной задачи [7]. Контекст *сужает* области баз данных или баз знаний, формируя области со значимой в данной ситуации информацией.

Обобщенное представление контекста на основании вышеизложенного сводится к тому, что контекст описывает ситуацию, на основании анализа которой может быть получена релевантная для текущей задачи информация.

Объектами контекста могут выступать объекты предметной области, поступающие задачи, методы, с помощью которых контекст может управлять знаниями предметной области [5]. Учитывая развитие информационных технологий и их ориентацию на пользователя, элементы, составляющие контекст, расширяются таким образом, что пользователь, прикладная программа и окружающая среда также становятся элементами контекста.

В статье в качестве определения контекста принято следующее. Контекстом является любая информация, которая может быть использована, чтобы охарактеризовать ситуацию, в которой находится в данный момент некоторый объект, и информация, которая может быть получена от этого объекта, при этом в качестве объекта может выступать пользователь, окружающая среда, физический объект и прикладная программа [8, 12]. Учитывая направленность работы на интеллектуальную поддержку принятия решений, в качестве прикладной программы выступает СИППР. Применительно к СИППР контекст предоставляет информацию, влияющую на принятие решений.

Приведенное общее определение уточняется понятием ситуации. Характеризуемая ситуация возникает при взаимодействии объектов [9]. Таким образом, контекст формирует часть информационного пространства, используемую объектами при их взаимодействии [5].

Ориентация СИППР на использование контекста позволяет, не влияя непосредственно на логический вывод, *ограничить* его только значимыми для данного контекста правилами/процедурами. Контекст является средством для управления базами знаний, позволяющим избежать избыточности и оптимизировать извлечение знаний. Представление контекста в формализованном виде дает возможности на основании интерпретации контекста предоставлять пользователю объяснения о действиях программы, выбранном решении, выдвинутой гипотезе и т. п. [10]. Таким образом, с точки зрения СИППР контекст позволяет более эффективно использовать ресурсы среды, в которой функционирует система. С точки зрения пользователя контекст предостав-

ляет ему действительную, релевантную и доступную для решения его задачи информацию [11, 12].

2.2. Виды контекста. Деление контекста на виды зависит от того, относительно какого использующего контекст объекта или относительно каких свойств контекста он рассматривается.

Относительно изменений в контексте, возникающих в ходе взаимодействия использующих контекст объектов, рассматривают статический и динамический контексты [3, 5]. *Статический контекст* содержит знания, не меняющиеся в ходе взаимодействия объектов. Они соответствуют знаниям о предметной области. *Динамический контекст* описывает изменяющуюся часть знаний. Динамический характер контекста проявляется во время его использования объектами контекста, например, во время решения задачи.

В зависимости от видов знаний, которыми обладают вовлеченные в некоторую ситуацию объекты, и знаний, используемых в данной ситуации, выделяют три вида контекста [13–17]:

1) *внешний контекст* [16, 17] или *контекстуальные знания* [13 – 15] – часть контекста, представляющая интерес в данной ситуации для данного пользователя. Контекстуальные знания описывают глобальную ситуацию. Их содержимое определяется задачей или событием, но эти знания не предлагают способов решения этой задачи или достижения цели. Контекстуальные знания представляют собой подразумеваемые знания или накопленные знания, с которыми объекты, использующие контекст, сталкивались в процессе решения данной или подобной задачи;

2) *внешние знания* [14, 15] – часть контекста, которая не входит в контекстуальные знания. Внешний контекст не имеет отношения к конкретной решаемой задаче, он представляет собой знания, дополнительно известные всем объектам, использующим контекст;

3) *внутренний* [16, 17] или *процедурный контекст* [13 – 15]. Процедурный (внутренний) контекст представляет собой часть контекстуальных знаний, конкретизированных в соответствии с текущей целью. Такой целью может быть *решение* текущей задачи или *выполнение* определенных действий. Объем знаний, составляющих процедурный контекст, должен быть достаточен для достижения текущей цели.

По отношению к контекстно-управляемой системе контекст делится на вычислительный контекст, контекст пользователя и физический контекст [18]. *Вычислительный контекст* содержит информацию о связанности узлов сети, пропускной способности и ближайших ре-

сурсах таких, как принтеры, мониторы, рабочие станции. *Контекст пользователя* описывается профилем пользователя, местом расположения и текущей социальной ситуацией. *Физический контекст* содержит информацию об освещенности, уровне шума, состоянии трафика, температуре.

В [19] предлагается деление контекста на внешний и внутренний по отношению к пользователю. *Внутренний* описывает состояние пользователя. Он может описывать деятельность, которой занимается пользователь, персональную информацию о нем, его эмоциональное состояние и т. п. *Внешний* описывает состояние окружающей среды по отношению к пользователю: место нахождения, близость других объектов, временной контекст. Подобное деление используется в [20, 21], где вместо названия *внутренний* контекст используется *физический* контекст, а вместо *внешнего* контекста – *виртуальный*.

Еще одно деление контекста по отношению к пользователю [21] включает в себя *физический* контекст (место расположения пользователя и время), контекст *окружающей среды* (погода, освещенность, уровень шума), контекст *пользователя* (состояние здоровья, настроение, запланированные мероприятия), *социальный* контекст (деятельность группы, присутствие членов группы), контекст *прикладной программы* (отправленные или полученные по электронной почте сообщения, посещаемые интернет-страницы).

С точки зрения человеко-машинного взаимодействия контекст складывается из контекста *пользователя*, контекста *решаемой задачи*, контекста *прикладной программы* и контекста *взаимодействия* [3].

С точки зрения очевидности представления контекста для использующих его объектов выделяют *явный* и *скрытый* контексты [3, 5]. При взаимодействии объектов контекст должен содержать знания, известные всем объектам. Часть этих знаний может быть не описана в явном виде или не формализована, но предполагается, что все объекты обладают этими знаниями. К таким знаниям относятся общие (универсальные) знания или знания объектов друг о друге, выявленные ранее, на предыдущих этапах взаимодействия. Указанная часть знаний составляет *скрытый* (неявный) контекст. Соответственно, явно описанные знания составляют *явный* контекст.

В основе деления контекста на *абстрактный* и *конкретный* [22] лежит та же идея, что и при делении на *явный* и *скрытый* контексты. *Абстрактный* предоставляет структуру информации, потенциально доступной объекту о текущей ситуации. *Конкретный* содержит явное

описание текущей ситуации. Иногда *абстрактный* контекст называют *внешним* [23].

В основе моделей управления контекстом лежит модель представления [4]. Некоторые модели, используемые для представления контекста, описываются ниже.

2.3. Модели представления контекста. Задача представления контекста состоит в формальном описании предмета наблюдения независимо от метода наблюдения. Модель представления контекста опирается на используемый формализм. Ниже приводятся некоторые модели представления контекста, используемые в различных подходах и системах.

Контекстная логика [23]. Контекстная логика применяется к контексту, который определяется как набор аксиом. Эта логика является расширением логики первого порядка, где аксиомы рассматриваются как верные или нет в рамках некоторого контекста. Контекст задается формальным объектом и отношением $ist(c, p)$, где c – контекст, p – аксиома. Отношение $ist(c, p)$ декларирует, что аксиома p верна в контексте c . Управление контекстом осуществляется с помощью «перемещающих» аксиом, которые переносят аксиому из одного контекста в другой. Перемещенная аксиома в новом контексте должна иметь те же самые условия истинности, какие она имела в оригинальном контексте.

Пропозициональные высказывания. Вышеупомянутые аксиомы, выраженные пропозициями, применяются для формализации контекстов в многоконтекстуальной базе знаний СУС [24]. В соответствии с отношением $ist(c, p)$ контекст определяется как согласованное множество пропозициональных высказываний. Область действия контекста определяется множеством объектов, условий и т. п., для которых высказывание о них верно. Между контекстами установлены иерархические отношения, позволяющие формировать контексты различной глубины детализации. Управление контекстом осуществляется с помощью процедур, выполняющих логический вывод высказываний, описанных в общих контекстах, в специализированные контексты.

Система правил [4]. Система правил используется для контекстов, описывающих структуру знаний. Формализм системы правил представляет структуру знаний пакетами правил. Используется представление на уровне правил и на уровне базы знаний. Представление на уровне правил управляется выражениями фильтрации. Представление на уровне баз знаний разбивает базу знаний на множество отдель-

ных небольших баз, управляемых напрямую правилами, вызывающими пакеты правил в части “then”, или интеракциями между пакетами правил для обмена информацией.

Фреймы. Фреймовый подход к представлению контекста используется в подходе, основанном на модели представления знаний, заложенной в системе Protégé [25]. В данном подходе контекстом является онтология источника данных, описанная средствами глобальной онтологии [26].

Контекстно-зависимые графы [14, 27]. Контекстно-зависимый граф представляет собой направленный ациклический граф с уникальным входом и выходом и последовательно-параллельной организацией вершин, связанных ориентированными дугами. Вершина может использоваться для представления действия, контекстно-зависимого понятия, рекомбинационной вершины или подграфа. Действие соответствует исполняемому методу, контекстно-зависимое понятие – понятию, имеющему реализацию в рассматриваемом контексте, рекомбинационная вершина – общему понятию (более абстрактному понятию), подграф – последовательности исполняемых методов, используемых при решении задачи. Отношения между вершинами показывают последовательность выполнения действий. Если есть несколько различных вариантов такой последовательности, эти варианты замыкаются рекомбинационной вершиной, которая после прохождения ветви с последовательностью фиксирует, какая ветвь отработала. Управление контекстом осуществляется в динамическом режиме в ходе решения задачи. Оно основано на фиксации того, какое контекстно-зависимое понятие получило реализацию в данный момент и, наоборот, какая процедура уже отработала, освободив реализации и превратив их в абстрактные элементы.

Ситуационная теория [4, 22]. В этой модели контекст представляет собой набор атрибутов, характеризующих ситуацию в общем виде (абстрактный объект в ситуационной теории). Набор атрибутов с присвоенными значениями характеризует конкретную ситуацию (дает реализацию абстрактной ситуации). Управление контекстом опирается на отношения, описывающие правила зависимости контекстной информации. На основании данных правил осуществляется распространение изменений в связанные контексты [4]. Для распространения изменений могут использоваться уведомляющие службы [22], которые посылают сообщения заинтересованным службам об изменениях в контекстной информации (в значениях атрибутов и наборе атрибутов).

Другие модели контекста включают в себя [3 – 5]: контекст как некоторое разбиение семантической сети, контекст как часть контекстуальной системы, контекст как составляющую дерева знаний и др.

Так как СИПР ориентированы на работу со знаниями, для таких систем наиболее интересны модели представления контекста, основанные на модели представления знаний. Современной и многообещающей моделью, используемой для представления знаний и, соответственно контекста, является онтологическая модель [28–31]. Такая модель позволяет структурировать накопленную информацию и обеспечивает семантическую совместимость объектов, вовлеченных во взаимодействие. В статье для представления контекста выбрана именно данная модель.

В следующем разделе статьи приведены основные способы формирования моделей контекста из онтологических знаний.

2.4. Формирование онтологической модели контекста. В данном разделе приведены три базовых подхода к формированию онтологической модели контекста. Во всех подходах контекст создается из нескольких исходных онтологий (источников знаний) применительно к текущей цели (задаче). Подходы различаются степенью автономности исходных источников знаний. В описываемых подходах онтология рассматривается как концептуальная структура, состоящая из множества вершин, представляющих классы концептов, и множества дуг, представляющих отношения, существующие между данными концептами.

Алгебра онтологий [32]. Данный подход предназначен для создания новых онтологий посредством повторного использования и составления существующих онтологий. Новые онтологии создаются из контекстов исходных существующих онтологий.

В подходе предполагается, что концепты, используемые в различных онтологиях, не совпадают до тех пор, пока для этих онтологий не разработаны соединяющие правила, в соответствии с которыми данные концепты могут или не могут считаться идентичными. В рассматриваемом подходе соединяющие правила описываются в виде предложений Хорна, но в общем случае для этой цели может быть использован любой язык, основанный на логике.

Для создания контекста используются следующие операторы алгебры: унарный оператор *Select* (выбор) и бинарные операторы *Intersection* (пересечение), *Union* (объединение) и *Difference* (разность) [33].

Оператор *Select* выбирает из двух онтологий их части (по одной части из каждой онтологии), которые являются релевантными для те-

кущей цели (задачи). Эти части подаются на вход функции генерации соединяющих правил. Данная функция генерирует множество всех возможных правил для двух частей. Эти правила называются соединяющими.

Бинарные операторы, пользуясь соединяющими правилами, формируют из двух выбранных частей контекст. Далее при описании бинарных операций над соединяемыми онтологиями имеются в виду операции над частями онтологий, выбранных операцией *Select*.

Результатом операции *Intersection* является множество вершин, которые встречаются в соединяющих правилах для двух онтологий, и множество дуг между данными вершинами, которые либо входили в соединяемые онтологии, либо были получены как результат работы функции генерации соединяющих правил.

Результат операции *Intersection* является контекстом, содержащим в себе знания из двух онтологий, релевантные текущей цели. В пределах такого контекста соединяемые онтологии являются согласованными. Как правило, в создании контекста участвуют знания из большего числа онтологий. В этом случае операции *Select* и *Intersection* последовательно применяются к парам онтологий, в которых содержатся релевантные текущей цели знания. Результатом применения этих операций является множество контекстов для пар онтологий.

Используя оператор *Union*, контексты, формирующие упомянутое множество контекстов, поочередно интегрируются в общий контекст. Для этого оператор *Union* берет в качестве исходного контекста любой контекст из множества, полученного в результате операции *Intersection*. Данный контекст считается текущим контекстом. Множества вершин, дуг и правил, содержащиеся в текущем контексте, дополняются вершинами, дугами и правилами, которые имеются в следующем по очереди контексте, но пока не вошли в текущий контекст. Также в множества, формируемые оператором *Union*, включаются новые вершины и дуги, если таковые были получены в ходе выполнения функции генерации соединяющих правил. Полученный после обработки очередного контекста результат становится текущим контекстом. На следующем шаге он дополняется соответствующими множествами из следующего по очереди контекста. Операция *Union* продолжается до тех пор, пока в текущий контекст не включены вершины, дуги и правила из всех контекстов, формирующих множество контекстов для пар онтологий.

К контексту, полученному в результате операции *Union*, применяется оператор *Difference*. Этот оператор создает несимметричные под-

множества вершин, дуг и правил, которые включены в один контекст, но не используются в другом. Оператор возвращает уникальные части контекстов, формирующих множество контекстов для пар онтологий, т.е. вершины, дуги и правила, которые не являются общими для всех объединенных контекстов. Эти подмножества удаляются из контекста, полученного в результате операции *Union*. В полученном после операции *Difference* контексте остаются только вершины, дуги и правила, имеющиеся во всех онтологиях, которые участвовали в создании контекста.

При создании контекста в соответствии с алгеброй онтологий источники знаний (исходные онтологии) теряют свою автономность на требуемый интервал времени (например, времени решения текущей задачи). После использования контекста данные источники снова становятся автономными.

Слияние онтологий [3–39]. В данном случае контекст получается как результат слияния релевантных текущей цели (задаче) частей нескольких онтологий. Под слиянием понимается такая интеграция нескольких частей исходных онтологий, результатом которой является унифицированная онтология или контекст, в котором единообразно представлены знания из интегрируемых частей, и полностью поддерживается логический вывод, заданный в этих частях. Синтаксические и семантические конфликты между исходными онтологиями либо разрешаются до начала операции интеграции, либо в результирующий контекст включаются методы (функции) по их разрешению.

Набор операций по слиянию частей нескольких онтологий разработан в подходе iPrompt [37]. В этом подходе для представления знаний используется фреймовая модель. Выбор частей онтологий, которые являются релевантными текущей цели (задаче), осуществляется экспертом. Работа системы, реализующей подход iPrompt, полностью зависит от решений, принимаемых экспертом в ходе выполнения операций по слиянию частей онтологий в контекст. В частности, эксперты решают, являются ли два концепта эквивалентными, и выбирают способ разрешения того или иного конфликта.

Для создания контекста в подходе iPrompt определен следующий набор операций:

- слияние двух классов в один класс;
- слияние двух слотов в один слот;
- слияние двух объектов в один объект;

- простое копирование класса – копирование класса с его подклассами (при копировании классов слоты, принадлежащие этим классам, также копируются);
- глубокое копирование класса – копирование класса вместе со всеми его надклассами вплоть до верхнего уровня таксономии;
- переименование классов и слотов;
- разрешение конфликтов:
 - 1) проблема «висячих» классов – таких, на которые ссылается копируемый класс, но которых еще нет в создаваемом контексте. Эта проблема снимается решением включать или нет в создаваемый контекст класс, на который установлена ссылка и, соответственно, саму ссылку;
 - 2) избыточность в таксономии классов – наличие в таксономии создаваемого контекста избыточных отношений, классов и слотов, полученных в результате операции копирования. Эта проблема разрешается посредством удаления избыточных компонентов;
 - 3) значение слота не попадает в область допустимых значений, заданную для данного слота. Конфликт данного типа возникает в результате операций слияния двух объектов или слотов.

При создании контекста методом слияния онтологий iPrompt исходные онтологии остаются автономными на время создания контекста, но после получения контекста связь с данными онтологиями разрывается.

Отображения. Данный подход поддерживается системой Сус [24], он основан на определении отображений между разнородными источниками информации и знаний и основной онтологией.

В системе Сус контекстом является некоторая небольшая область, в которой содержатся непротиворечивые знания. Работа системы опирается на широкомасштабную базу «общеизвестных» знаний (OpenСус-онтологию [41]). Система поддерживает логический вывод в базе знаний, позволяет осуществлять информационный поиск по базе знаний, интеграцию разнородных видов информации и знаний, обработку структурированных данных, проверку непротиворечивости предоставляемых разными источниками знаний, проверку орфографии в текстовых документах и другие операции, связанные с обработкой знаний [42].

Контекст в Сус представлен набором утверждений (логических высказываний). Эти утверждения считаются верными в данном контексте (в данной ситуации). Сами контексты являются объектами пер-

вого порядка и в свою очередь могут быть составляющими утверждений. Каждый контекст содержит только такой объем знаний, какой абсолютно необходим для того, чтобы представленные в контексте утверждения всегда были верными. В онтологической модели, составляющей основу системы Сус, контексты образуют иерархию. Утверждения, сформулированные в контекстах более высоких уровней, доступны в контекстах более низких уровней [43].

Контекст в Сус создается на основании интеграции знаний из различных источников. С этой целью между концептуальными схемами внешних источников знаний и базой знаний Сус устанавливаются структурные и семантические отображения. Эти отображения требуются для перевода знаний, представленных во внешнем источнике, в базу знаний Сус и для представления внешних знаний в базе знаний Сус. Для декларации структурных и семантических отображений используются Сус-онтология и язык предикатов высокого порядка – СусL [44].

Структурные отображения устанавливаются между физическими структурами внешнего источника знаний и базы знаний Сус. Семантические отображения между элементами внешнего источника знаний и элементами базы знаний Сус устанавливаются посредством сравнения смысловых значений схематических элементов различных источников знаний в общем контексте сравнения. Под схематическими элементами источника знаний понимаются множества сущностей, атрибуты, области допустимых значений атрибутов и отношения, которые могут быть найдены в структуре источника знаний. Общий контекст сравнения – контекст, формализующий некоторую ситуацию. С этим контекстом сравниваются внешние источники, которые предположительно содержат знания, относящиеся к той же самой ситуации.

Как только требуемые отображения между внешним источником знаний и базой знаний Сус установлены, внешний источник знаний может обрабатываться ядром логического вывода Сус так, как если бы этот источник являлся частью базы знаний Сус.

Кроме контекстов, описывающих знания, релевантные некоторой ситуации, в Сус существует понятие контекста, предназначенного для решения задач (контекст задачи). Этот вид контекста является временным контекстом. Контекст задачи создается оперативно, когда требуется решить некоторую задачу. Контекст задачи комбинирует релевантные задаче знания из различных контекстов, представленных в онтологии. После решения задачи контекст задачи уничтожается [24].

При создании контекста в Сус внешние источники знаний всегда автономны, но связь с этими источниками после создания контекста не разрывается. Такой подход в отличие от предыдущих двух подходов, имеет высокую стоимость, так как знания, представленные в контексте, должны постоянно обновляться по мере изменения знаний в исходных источниках.

В следующем разделе описываются присущие контексту свойства, которые были выявлены в результате анализа различных видов контекста и вышеизложенных моделей.

2.5. Свойства контекста. В работе выделены следующие присущие контексту свойства.

С в о й с т в о 1. *Контекст возникает при взаимодействии объектов.* Следствием данного свойства является то, что контекст имеет смысл рассматривать в процессе взаимодействия объектов [3 – 5, 10, 13, 15]. В этом случае контекст описывает общую, совместно используемую информацию, известную всем участвующим во взаимодействии объектам.

С в о й с т в о 2. *Общий контекст, возникающий при взаимодействии объектов, включает в себя контексты данных объектов* [18–22]. Следствием данного свойства является то, что возможно выполнение логического вывода в рамках отдельных контекстов, входящих в общий контекст. Причем в рамках общего контекста формы логического вывода (например, немонотонный логический вывод, робототехника, распознавание образцов) могут быть различны [3]. Еще одно следствие данного свойства состоит в том, что в случае, когда среди контекстов, составляющих общий контекст, выделяют контекст решаемой задачи, и такая задача может быть разложена на подзадачи, а контекст – соответственно на контексты подзадач, то для решения каждой подзадачи можно использовать свой механизм вывода.

С в о й с т в о 3. *Контекст в процессе взаимодействия объектов может меняться* [45]. Следствием изменений в контексте являются вызванные этим изменения в контекстах, связанных с ним отношениями.

С в о й с т в о 4. *Контекст включает в себя неявные знания и информацию* [3, 5]. Данное свойство имеет два следствия: 1) как можно больше контекстной информации должно быть представлено в явном виде, 2) использование истории получения и изменений контекста позволяет вывести новый контекст на основании ранее используемых контекстов [46].

Свойство 5. *Контекст может описывать как отношения между элементами абстрактных моделей, так и конкретные приложения данных абстракций.* Данное свойство имеет следствием возможность создания из текущего используемого контекста абстрактного контекста, который в дальнейшем может быть конкретизирован для других условий [3, 22]. Процесс создания абстрактного контекста называют деконтекстуализацией [23].

Свойство 6. *Основным назначением контекста является предоставление релевантной и достоверной информации объектам, участвующим во взаимодействии, для решения текущей задачи (достижения текущей цели).* Таким образом, интерпретируя используемый объектами контекст, можно получить релевантную информацию для решения текущей задачи [3, 7].

В следующем разделе описываются составляющие процесса управления контекстом.

2.6. Управление контекстом. Основной вопрос, связанный с исследованием и применением контекста, – выявление отношений между контекстами и адаптация контекстов к текущим условиям и задачам. Управление контекстом занимается сбором, интерпретацией, хранением и распространением контекстной информации в динамическом режиме [18], а также выявлением отношений между контекстами, благодаря которым становится возможным эффективное управление контекстами [5].

Сбор контекстной информации – определение, какая информация должна быть включена в контекст, и каков наиболее оптимальный способ ее организации. Включению в контекст подлежит релевантная для рассматриваемой ситуации информация, содержащая описания условий и влияющих факторов, которые делают эту ситуацию уникальной и понятной [15]. Включаемая информация должна содержать непосредственно получаемые данные, историю получения этих данных и знания, которые на данный момент известны взаимодействующим объектам. Включение всех перечисленных видов информации позволяет выявлять из контекста скрытую информацию [3].

Интерпретация контекста. Проблема интерпретации связана с разработкой механизмов, применение которых позволяет использовать собранную информацию соответствующим образом при конкретных условиях для конкретных целей [15, 18]. Так как в качестве источников информации могут выступать распределенные разнородные источники, реализация данных механизмов осуществляется через повышение уровня абстракции контекста [8]. К способам абстрагирования

контекстной информации относятся деконтекстуализация, интеграция контекстов, соединение контекстов и декомпозиция.

Деконтекстуализация. Понятие деконтекстуализации связано с получением общего неизменного контекста, выведенного из конкретных положений и высказываний, характеризующих определенную ситуацию. Идея основана на том, что контекст взаимодействия может включать в себя несколько частных контекстов. Таким образом, существует общий контекст, покрывающий эти частные контексты. Полученный общий контекст может быть в дальнейшем повторно использован в других ситуациях посредством его уточнения [23].

Интеграция контекстов. Под интеграцией контекстов понимается соединение нескольких контекстов в один. Интеграция контекстов используется для:

– дополнения информации, содержащейся в одном контексте, информацией из другого; получения контекста, который может быть использован прикладной программой для решения определенных задач;

– повышения достоверности контекстной информации.

Интегрироваться могут как однородные, так и разнородные контексты (контексты, полученные соответственно из множества однородных или разнородных источников информации) [18].

Декомпозиция. Разбиение контекста на составляющие его субконтексты, возможно пересекающиеся, на основании некоторого признака (способ организации субконтекста, владелец субконтекста, версия субконтекста, дата и время создания и т. п.) [11]. Декомпозиция позволяет использовать только ту часть базы данных или знаний, которая является релевантной для текущей задачи. Также она позволяет копировать фрагменты контекста, повторно компоновать их для других целей и т. д.

Хранение и накопление контекстной информации. Так как задача организации контекстной информации связана с привлечением исторически накопленных данных и знаний, возникает проблема хранения контекстной информации. Информация должна накапливаться постоянно, даже если в данный момент нет заинтересованных в ней прикладных программ, так как со временем может появиться прикладная программа, которой требуется история получения определенного контекста. Таким образом, процесс накопления и хранения контекстной информации не связан с конкретной прикладной программой и должен осуществляться независимо [8]. Наличие накопленной информации позволяет повторно использовать уже имеющиеся знания, выявлять

скрытые знания, прогнозировать развитие ситуации и намерений пользователя [12].

Передача информации из одного контекста в другой или *обмен контекстной информацией*. Данная проблема связана с проблемой адекватного понимания участниками взаимодействия контекстов друг друга, а также с проблемой абстрагирования контекста. При формировании общего или абстрактного контекста, покрывающего несколько частных контекстов, требуется корректно отобразить в нем информацию, описанную в контекстах объектов, участвующих во взаимодействии [23]. С передачей информации связана проблема перевода содержимого контекста между различными форматами и языками представления [21, 47] и проблема переноса правил логического вывода из одного контекста в другой [23].

Модификация и обновление контекста. Хотя проблема модификации и обновления контекста явно не входит в процесс организации контекстной информации, упоминание о ней связано с тем, что процедуры модификации и обновления используются при отслеживании динамических изменений в контексте. К таким изменениям относятся:

- 1) расширение контекста для включения в него дополнительной информации;
- 2) изменение конкретных значений, содержащихся в локальном (конкретном, частном) контексте;
- 3) изменения в абстрактном контексте;
- 4) учет изменений в контекстах, которые связаны отношениями с изменяемым контекстом (распространение изменений).

История контекста складывается из запоминания всех произведенных над данным контекстом изменений [11, 21, 22].

Известны следующие типы отношений, которые могут существовать между контекстами:

- *включения*, когда один контекст полностью или частично содержит в себе другой;
- *иерархические*, устанавливаемые между контекстами по мере уточнения (конкретизации) одним контекстом другого [3];
- *выводимости*, когда определяемая контекстная информация логически выводится из известной контекстной информации [8].

На основании выявленных свойств контекста, а также анализа моделей контекста и задач, решаемых в процессе управления контекстом, можно сформулировать требования к управлению контекстом в СИПР, которые описаны в следующем разделе.

2.7. Спецификация требований к управлению контекстом.

С точки зрения способа *представления* контекста, он должен быть описан стандартизированными способами, обеспечивающими независимость способа представления от платформы. Модель представления знаний должна поддерживать операции по созданию контекста и управлению им. Наиболее удобным, отвечающим современным требованиям способом представления контекста считается онтологическая модель представления информации и знаний. Контекстно-зависимыми представлениями онтологической модели являются:

- 1) концепты, представляющие объекты, вовлеченные в ситуацию;
- 2) свойства концептов, представляющие местоположение объектов и время;
- 3) фундаментальные ограничения на ситуацию.

В рамках задачи *сбора контекстной информации* контекст должен предоставлять релевантную текущую информацию для решения конкретной задачи. Включаемая информация должна содержать непосредственно получаемые данные, историю получения этих данных и знания, которые на данный момент известны взаимодействующим объектам.

В части *интерпретации* контекста управление им должно предусмотреть механизмы, применение которых позволяет использовать собранную информацию соответствующим образом при конкретных условиях для конкретных целей. Данные механизмы позволяют:

- 1) логически выводить новый контекст из имеющихся;
- 2) повторно использовать контекст с помощью применения контекстов более высоких уровней абстракции, их интеграции и конкретизации для данных условий и задач;
- 3) получать контекст более высокого уровня абстракции из данного рассматриваемого контекста;
- 4) разбивать контекст на субконтексты.

Хранение контекстов связано с постоянным, независимым от текущих задач, накоплением контекстной информации и хранением истории получения и изменений контекстов.

В части *распространения* контекстной информации последняя должна быть предоставлена всем заинтересованным объектам в виде, который обеспечивает одинаковую интерпретацию данными объектами.

Принимая во внимание *динамический* характер контекстной информации, управление контекстом должно предусматривать механиз-

мы, позволяющие модифицировать, обновлять и распространять контекстную информацию в реальном времени.

Сформулированные требования к управлению контекстом послужили основой для разработки модели управления контекстом в СИППР.

3. Двухуровневая модель управления контекстом в СИППР.

В работе предполагается, что СИППР получает информацию от информационных ресурсов, расположенных в открытой информационной среде, т. е. в едином информационном пространстве, в котором поддерживаются открытые технологии. Технология управления контекстом используется для создания модели текущей ситуации.

Рассматривается следующая формальная постановка задачи. Имеется открытая информационная среда, включающая в себя набор информационных ресурсов

$$R = \{r_n | n \in \mathbb{N}\}.$$

Для каждого ресурса определен набор его функций

$$FR_i = \{f_m | m = 1, \dots, M\},$$

где M – число функций ресурса r_i .

Значения, являющиеся результатом выполнения функций ресурсов, используются при моделировании текущей ситуации. Текущая ситуация описывается функциональной сетью

$$FN = \{f_n | n = 1, \dots, N\},$$

где N – число функций, которые должны быть выполнены в ходе моделирования, $FN \subseteq \bigcup_{i=1}^{NR} FR_i$, $NR = |R|$.

Требуется построить модель текущей ситуации, используя информацию, поставляемую ресурсами открытой информационной среды, по критерию минимизации количества вовлеченных ресурсов: $|R_u| \rightarrow \min$. При этом набор вовлеченных ресурсов должен обеспечить выполнение всех функций, описывающих модель текущей ситуации:

и данные ресурсы должны быть доступны на интервале $[t_0, T]$, где t_0 – время начала получения информации, используемой в качестве входных данных для функций, описывающих текущую ситуацию, T – время окончания моделирования.

$$FN \subseteq \bigcup_{j=1}^{NR_u} FR_j, \quad NR_u = |R_u|,$$

и данные ресурсы должны быть доступны на интервале $[t_0, T]$, где t_0 – время начала получения информации, используемой в качестве входных данных для функций, описывающих текущую ситуацию, T – время окончания моделирования.

3.1. Управление контекстом при моделировании текущей ситуации. В работе используется двухуровневое моделирование ситуации. На первом, семантическом, уровне строится модель текущей ситуации на основании знаний проблемной области. На втором, прикладном, уровне данная модель конкретизируется значениями переменных, получаемых от информационных ресурсов.

Знания проблемной области формализованы в прикладной онтологии, включающей в себя:

1) концептуальные знания (знания об объектах реального мира, представленные посредством спецификации свойств этих объектов (атрибутов) и существующих между данными объектами отношений);

2) знания о задачах, существующих в данной области, и методах их решения (онтология задач & методов).

Методы реализованы как вычислительные ресурсы открытой информационной среды. На эти ресурсы в прикладной онтологии установлены ссылки.

Для моделирования текущей ситуации предложено использовать два типа контекста: абстрактный и прикладной. Абстрактный контекст представляет собой онтологическую модель текущей ситуации, построенную на основании интеграции знаний проблемной области, релевантных для данной ситуации. Прикладной контекст является конкретизацией абстрактного контекста для реальных условий.

Так как прикладная онтология формализует два типа знаний, абстрактный и прикладной контексты также включают в себя два типа знаний. Наличие в контекстах формализованных задач и методов их решения делает прикладной контекст функциональным средством, которое может быть использовано для решения формализованных в данном контексте задач. В соответствии с ранее разработанной методологией контекстно-управляемой поддержки принятия решений эти задачи решаются как задачи удовлетворения ограничений (задачи поиска допустимого решения) [48].

Знания, представленные в прикладной онтологии, описаны средствами формализма объектно-ориентированных сетей ограничений (ООСО). Этот формализм обеспечивает совместимость используемой модели представления знаний (табл. 1):

1) с типовыми примитивами, используемыми при онтологическом моделировании [49];

2) с представлениями, поддерживаемыми системами моделирования [50];

3) с моделью задачи удовлетворения ограничений;

Таблица 1. Совместимость моделей

ООСО	Онтология	Моделирование	Открытая среда (XML, xmlns:xs="http://www.w3.org/2001/XMLSchema")	Решатели (задача удовлетворения ограничений)
Класс объектов	Класс объектов	Объект, процесс	xs:element	Множество переменных
Множество атрибутов	Множество свойств (атрибутов)	Множество атрибутов	<xs:complexType> <xs:sequence> <xs:element name = >	
Домен (область допустимых значений атрибута)	Диапазон значений атрибута	Домен	xs:anySimpleType, <xs: anySimpleType> <xs:restriction> <xs:enumeration>, xs:complexType	Домен
Множество значений атрибутов	Экземпляр класса	Экземпляр объекта	xmlns:i=name space <i:name>value</i:name>	Множество значений переменных
Ограничения	Аксиомы и отношения	Отношения, функции, правила	xmlns:mml="http://www.w3.org/1998/Math/MathML" mml:Relations	Ограничения

- 4) со стандартами открытых сред и систем;
 5) с внутренними представлениями знаний в решателях ограничений.

Согласно формализму ООСО онтология A описывается четверкой $A = (O, Q, D, C)$,

где O – множество классов объектов (далее «классы»), Q – множество атрибутов классов (далее «атрибуты»), D – множество доменов – областей допустимых значений атрибутов (далее «домены»), C – множество ограничений.

В соответствии с принятым формализмом в рамках общей постановки задачи введена следующая постановка задачи моделирования текущей ситуации. При заданных и формализованных знаниях проблемной области $A = (O, Q, D, C)$ требуется построить двухуровневую модель текущей ситуации S , представленную абстрактным контекстом

$$\text{Context}_a(T_a) = (S | O_a, Q_a, D_a, C_a, R_a, T_a),$$

$O_a \subseteq O$, $Q_a \subseteq Q$, $D_a \subseteq D$, $C_a \subseteq C$, $R_a \subseteq R$, T_a – время адекватности модели, и прикладным контекстом

$$\text{Context}_p(t) = (S | O_p, Q_p, D_p, C_p, R_p, t),$$

$O_p \subseteq O_a$, $Q_p \subseteq Q_a$, $D_p \subseteq D_a$, $C_p \subseteq C_a$, $R_p \subseteq R_a$, t – текущее время, и найти множество экземпляров объектов

$$I(t) = \{v_{q_i}^o \mid v \in D_p, o \in O_p, q \in Q_p, i = 1, \dots, N_q\},$$

удовлетворяющих всем ограничениям C_p , N_q – число атрибутов у класса o , при этом значения параметрам модели присваиваются ресурсами R открытой информационной среды в моменты времени t .

Онтологическая модель абстрактного контекста представляется как

$$\text{Context}_a(T_a) = (S | O_a, Q_a, D_a, C_a, WS_a, T_a),$$

где S – тип моделируемой ситуации; O_a – множество классов, требуемых в общем случае для моделирования ситуации S ; Q_a – множество атрибутов классов O_a ; D_a – множество доменов атрибутов Q_a ; C_a – множество ограничений, включенных в абстрактный контекст; WS_a – множество web-сервисов, которые моделируют функции информационных ресурсов, присваивающих значения атрибутам Q_a , $WS_a \subseteq WS$, где WS – множество зарегистрированных web-сервисов; T_a – прогнозируемое время адекватности модели.

По мере получения информации от окружающей среды (от информационных ресурсов, на которые есть ссылки в абстрактном контексте задачи) осуществляется присвоение значений атрибутам классов, включенных в абстрактный контекст, и порождение прикладного контекста. Прикладной контекст Context_p является моделью текущей ситуации в нотации объектно-ориентированных сетей ограничений с присвоенными значениями переменных. Данная модель интерпретируется как задача удовлетворения ограничений. Модель прикладного контекста представляется как

$$\text{Context}_p(t) = (S | O_p, Q_p, D_p, C_p, WS_p, T_a, \Delta T),$$

где t – текущее время; O_p – множество классов, реально используемых для моделирования ситуации S при конкретных условиях; Q_p, D_p, C_p и WS_p – используемые множества атрибутов, доменов, ограничений и web-сервисов, соответственно; $\Delta T = t - t_0$ – текущее время жизни прикладного контекста; t_0 – время создания абстрактного контекста.

Под задачей удовлетворения ограничений CSP понимается тройка множеств

$$CSP = (V_{CSP}, D_{CSP}, C_{CSP}),$$

где V_{CSP} – множество переменных, D_{CSP} – множество соответствующих областей переменных (домены переменных), C_{CSP} – множество ограничений на переменные.

Решением задачи удовлетворения ограничений называется набор значений переменных $v \in V_{CSP}$, удовлетворяющих всем ограничениям на переменные. Данный набор содержит множества экземпляров объектов

$$I(t) = \{v_{q_i}^o \mid v \in D_p, o \in O_p, q \in Q_p, i = 1, \dots, N_q\}.$$

Для обеспечения обмена информацией между разнородными ресурсами, а также ее обработки разработана модель ресурса, совместимая с онтологической моделью представления знаний. Совместимость обеспечивается за счет использования при моделировании информационного ресурса интерфейса web-сервисов.

3.2. Модель информационного ресурса. Для моделирования функций информационных ресурсов используется интерфейс web-сервисов. Для каждой функции f_r информационного ресурса существует отображение

$$\Phi_f : f_r \rightarrow f_{ws},$$

где f_{ws} – функция web-сервиса.

Web-сервис ws ($ws \in WS$) представляется в виде модели:

$$ws = (URI, IS, F, FI, FID, FO, FOD, Calls, Fails, Price, \\ Time_Access, Time_In),$$

где URI – адрес web-сервиса, представленный идентификатором URI (Unique Resource Identifier) данного сервиса; IS – идентификатор информационного ресурса; F – моделируемая функция, для F задано отображение $f_m : F \rightarrow O_m$ такое, что $f_m(F) = o_m, o_m \in O_m$, где O_m –

множество методов, формализованных в онтологии; FI – множество входных аргументов функции F , для FI задано отображение $f_{inp} : FI \rightarrow Q$ такое, что для всех $fInp \in FI$ верно выражение $f_{inp}(fInp) = q$, $q \in Q$, где Q – множество атрибутов классов онтологии; FID – множество типов данных входных аргументов функции F , для FID определено множество функций $\Phi_{inp} : FID \rightarrow D$, где D – множество доменов атрибутов Q , таких, что для всех $fType^{Inp} \in FID$, $Inp \in fInp$ существует функция

$$\varphi_{inp}(fType^{Inp}) = \{d_i^q \mid d_i \in D, q \in Q, i \in N\}, \varphi_{inp} \in \Phi_{inp};$$

FO – выходной аргумент функции F , для FO задано отображение $f_{oup} : FO \rightarrow Q$ такое, что $f_{oup}(FO) = q$, $q \in Q$; FOD – тип данных выходного аргумента функции F , для FOD определена функция $\varphi_{oup} : FOD \rightarrow D$ такая, что

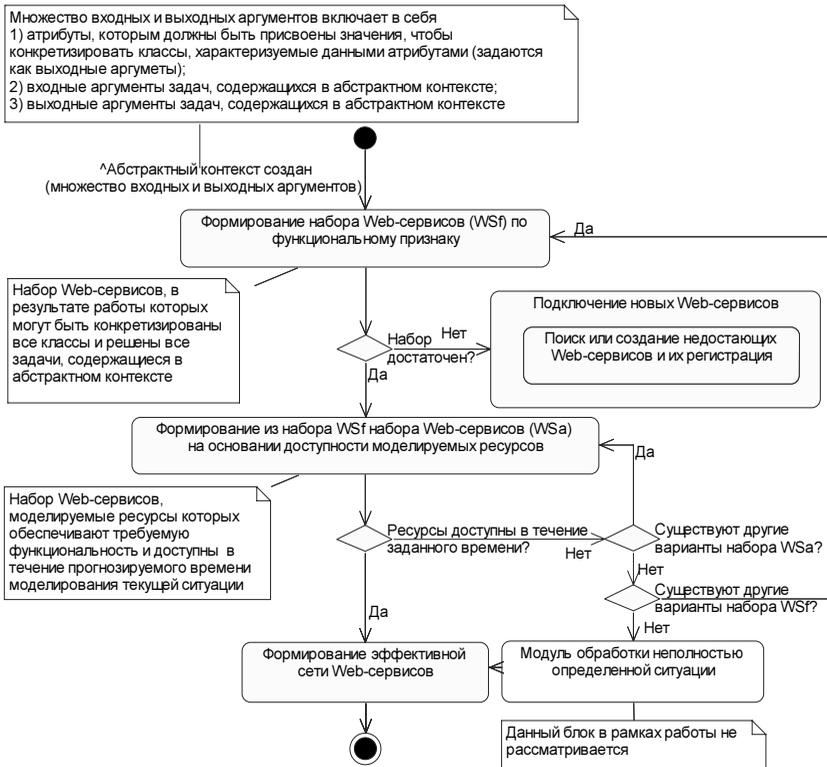
$$\varphi_{oup}(fType^{Oup}) = d^q \mid d \in D, q \in Q, fType^{Oup} \in FOD;$$

$Calls$ – число вызовов web-сервиса; $Fails$ – число неудачных вызовов; $Price$ – стоимость web-сервиса; $Time_Access$ – время доступа к web-сервису; $Time_In$ – интервал времени, когда информационный ресурс доступен.

3.3. Получение прикладного контекста. В данном разделе рассматривается проблема эффективного использования ресурсов открытой среды, которые могут быть вовлечены в процесс порождения прикладного контекста. Основой для построения прикладного контекста служит абстрактный контекст. Принципы построения этого контекста изложены в [51], и в данной работе процедура его построения не рассматривается.

Результатом построения абстрактного контекста является онтологическая модель текущей ситуации в нотации ООСО без значений переменных модели, т. е. абстрактный контекст не содержит экземпляров объектов. Этот контекст ссылается на множество web-сервисов, моделирующих функции ресурсов, информация от которых может быть использована для конкретизации знаний, представленных в контексте. Множество web-сервисов включает в себя альтернативные web-сервисы, т. е. те, которые моделируют функции альтернативных ресурсов.

Для порождения прикладного контекста предложена сценарная модель формирования сети web-сервисов из множества web-сервисов, на которые ссылается абстрактный контекст (см. рисунок). В соответствии с этой моделью вначале формируется набор web-сервисов WS_f , обеспечивающий минимизацию количества ресурсов за счет использования ресурсов с максимальным набором требуемых функций: $|FR_i \cap FN| \rightarrow \max$ для i -го ресурса.



UML-диаграмма сценарной модели формирования сети web-сервисов на основании абстрактного контекста.

Затем из набора WSf формируется набор web-сервисов WSa . Набор WSa состоит из web-сервисов, моделируемые ресурсы которых доступны на интервале

$$[t_0, T]: [t_0, T] \subseteq \bigcup_{j=1}^{NR} \bigcup_{i=1}^{nt} \Delta t_i$$

где Δt_i – интервал времени, в течение которого доступен j -й ресурс, nt – количество интервалов Δt на интервале $[t_0, T]$.

На последнем этапе в наборе WSa из альтернативных ресурсов выбираются наиболее эффективные на основании таких характеристик, как надежность ресурса, время доступа к нему и его стоимость.

4. Технологическая модель. Технологическая модель, позволяющая реализовать функции контекстно-управляемой СИППР, включает в себя технологии *управления онтологиями, управления контекстом, удовлетворения ограничений и профилирования* (табл. 2). Применение

Таблица 2. Технологическая модель

Решаемая задача	Способ решения	Технология	Результат в терминах ООСО
Построение прикладной онтологии	Интеграция существующих онтологий	Управление онтологиями	Сеть ограничений без значений переменных
Представление функций, выполняемых источниками информации	Установка соответствий между онтологией и описаниями web-сервисов	Управление онтологиями	Сеть ограничений без значений переменных
Сбор и организация знаний	Создание абстрактного контекста	Управление онтологиями	Модель задачи без значений переменных
Сбор и организация информации	Получение прикладного контекста	Управление контекстом	Модель задачи со значениями переменных
Поиск допустимых решений	Обработка сети ограничений решателем	Удовлетворение ограничений	Множество допустимых решений задачи
Повторное использование моделей задач	Сбор и организация контекстов в архиве	Управление контекстом	Сеть ограничений
Выявление предпочтений пользователя	Контекстно-зависимое накопление принятых решений	Профилирование	Множество ограничений пользователя

конкретной технологии зависит от стадии функционирования системы. Данные стадии определены в общем сценарии функционирования СИППР.

На этапе подготовки СИППР к работе экспертами проблемной области, инженерами по знаниям и инженерами по онтологиям создается прикладная онтология. В общем случае она может создаваться «с нуля» и посредством интеграции уже существующих онтологий. В первом случае для построения прикладной онтологии используется технология инженерии онтологий. Во втором случае, который использовался в рамках представленных здесь исследований, к технологии инженерии онтологий подключается технология *управления онтологиями*.

Также на подготовительном этапе устанавливаются отношения соответствия между прикладной онтологией и web-сервисами, моделирующими функции информационных ресурсов среды. Установка соответствий – одна из задач, реализуемая технологией *управления онтологиями*.

Этап функционирования СИППР начинается с построения абстрактного контекста. С этой целью из прикладной онтологии извлекаются знания, релевантные текущей ситуации, а затем интегрируются в абстрактный контекст. Для этого используется технология *управления онтологиями*.

Для получения прикладного контекста (наполнения абстрактного контекста значениями переменных) используется технология *управления контекстом*.

Формализованные в прикладном контексте задачи решаются как задачи удовлетворения ограничений. Для решения этих задач используется технология *удовлетворения ограничений*.

Абстрактные и прикладные контексты сохраняются в архиве контекстов, а соответствующие этим контекстам принятые решения – в профиле пользователя. Организация и поддержка архива контекстов – одна из задач технологии *управления контекстом*. Накопленная в архиве контекстов и профилях пользователей информация используется для выявления предпочтений пользователя. Для этого используются методы и модели технологии *профилирования*.

5. Заключение. В статье проведен анализ известных подходов к описанию и формированию контекста в различных информационных средах. На основании выявленных в ходе анализа свойств контекста, а также анализа моделей контекста и задач, решаемых в процессе управления контекстом, сформулированы требования к управлению контек-

стом в СИППР. Данные требования послужили основой для разработки двухуровневой модели управления контекстом для организации интеллектуальной поддержки принятия решений в динамических структурированных областях, выбора модели описания ресурсов открытой информационной среды и определения технологической модели контекстно-управляемой СИППР.

Разработанная двухуровневая модель управления контекстом позволяет строить модель ситуации на абстрактном уровне, а затем ее конкретизировать для текущих условий. Это делает возможным получение моделей типовых ситуаций на абстрактном уровне и их повторное использование с конкретизацией для текущих условий.

Предложенный в статье способ моделирования функций информационных ресурсов при помощи интерфейса web-сервисов позволяет организовать обмен информацией между разнородными ресурсами открытой информационной среды, а также между ресурсами и СИППР.

Разработанная сценарная модель формирования сети web-сервисов позволяет эффективно использовать ресурсы открытой информационной среды в зависимости от выполняемых ресурсами функций, доступности ресурсов и наличия альтернативных ресурсов.

Предложенная технологическая модель, которая основана на интеграции технологий управления онтологиями, управления контекстом, удовлетворения ограничений и профилирования, обеспечивает требуемую функциональность контекстно-управляемых СИППР.

Литература

1. *Ajenstat J., Padovitz A., Zaslavsky A., Loke S. W.* Modeling and Adapting to Context Changes: Case of Stock Market Decision Making [Electronic resource] // Proc. Workshop on Context Modeling and Decision Support held in conjunction with the Fifth Intern. and Interdisciplinary Conf. on Modeling and Using Context. Paris, France, 2005. CEUR Workshop Proc. URL: http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-144/06_ajenstat.pdf (access date: 03.04.2009).
2. *Lyons J.* Linguistic Semantics: an Introduction. Cambridge University Press, 1995. 376 p.
3. *Brézillon P.* Context in Problem Solving: a Survey // The Knowledge Engineering Review. 1999. Vol. 14, № 1. P. 1–34.
4. *Brézillon P.* Context in Artificial Intelligence: I. A Survey of the Literature // Computer & Artificial Intelligence. 1999. Vol. 18, № 4. P. 321–340.
5. *Brézillon P.* Context in Artificial Intelligence: II. Key Elements of Contexts // Computer & Artificial Intelligence. 1999. Vol. 18, № 5. P. 425–446.
6. *Wiederhold G. Jannink J.* Composing Diverse Ontologies [Electronic resource]: Tech. Rep., Stanford: Stanford University, USA. 1998. URL: <http://www-db.stanford.edu/SKC/publications/ifip99.html> (access date: 24.06.2009).

7. *Shahar Y.* Dynamic Temporal Interpretation Contexts for Temporal Abstraction // *Annals of Mathematics and Artificial Intelligence*. 1998. № 22. P. 159–192.
8. *Dey A.K., Salber D., Abowd G. D.* A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications [Electronic resource] // *Context-Aware Computing. A Spec. Triple Issue of Human-Computer Interaction*. Vol. 16. Lawrence-Erlbaum, 2001. URL: <http://www.cc.gatech.edu/fce/ctk/pubs/HCIJ16.pdf> (access date: 24.06.09).
9. *Winograd T.* Architectures for Context // *Human-Computer Interaction*. 2001. Vol. 16. P. 2–3.
10. *Brézillon P., Cavalcanti M.* Modeling and Using Context: Rep. on the First Intern. and Interdisciplinary Conf. CONTEXT-97 // *The Knowledge Engineer Review*. 1997. Vol. 12, № 4. P. 1–10.
11. *Motschnig-Pitrik R.* Contexts as Means to Decompose Information Bases and Represent Relativized Information [Electronic resource] // *Proc. CHI2000 Workshop on “The What, Who, Where, When, Why, and How of Context Awareness”*, The Hague, The Netherlands, 2000. URL: <ftp://ftp.cc.gatech.edu/pub/gvu/tr/2000/00-18n.pdf> (access date: 24.06.2009).
12. *Dey A. K.* Understanding and Using Context // *Personal and Ubiquitous Computing J.* 2001. Vol. 5, № 1. P. 4–7.
13. *Pomerol J.-Ch., Brézillon P.* About Some Relationships between Knowledge and Context // *Modeling and Using Context (CONTEXT-01)*, Dundee, Scotland, 2001. *Lecture Notes in Computer Science*, Springer Verlag. 2001. P. 461–464.
14. *Brézillon P.* Context-based Modelling of Procedures and Practices in Business Environments [Electronic resource] // *Rep. 2 COCONET Workshop “Roadmap Elaboration and Construction”*, Finland, Helsinki-Espoo, 2002. URL: https://doc.telin.nl/dscgi/ds.py/Get/File-28084/COCONET_D3.2_Report_of_Workshop_2.pdf (access date: 24.06.2009).
15. *Pomerol J.-Ch., Brézillon P.* Context proceduralization in decision making // *Modeling and Using Context (CONTEXT-03)*. *Lecture Notes in Artificial Intelligence*. 2003. Vol. 2680. P. 491–498.
16. *Ozturk P., Aamodt A.* A Context Model for Knowledge-Intensive Case-Based Reasoning // *Intern. J. Human-Computer Studies. Spec. Issue on Using Context in Applications*. 1998. Vol. 48, № 3. P. 331–355.
17. *Tiberghien G.* Context and cognition: Introduction // *Cahier de Psychologie Cognitive*. 1986. Vol. 6, № 2. P. 105–119.
18. *Barrett K., Power R.* State of Art Surveys [Electronic resource]: M-Zones Deliverable No. 1, Release 2. Cork Institute of Technology; Trinity College Dublin; Waterford Institute of Technology. Deliverable 1.1. 2003. URL: http://www.m-zones.org/deliverables/d1_1/d1_1-final-consolidated.pdf (access date: 08.06.2009).
19. *Gwizdka J.* What's in the context? [Electronic resource] // *Proc. CHI2000 Workshop on “The What, Who, Where, When, Why, and How of Context Awareness”*, The Hague, The Netherlands. 2000. URL: <ftp://ftp.cc.gatech.edu/pub/gvu/tr/2000/00-18e.pdf> (access date: 24.06.2009).
20. *Ebling M. R., Hunt G. D. H., Lei H.* Issues for Context Services for Pervasive Computing [Electronic resource] // *Proc. Workshop on Middleware for Mobile Computing*, Heidelberg, Germany, 2001. URL: <http://www.cs.arizona.edu/mmc/13/Ebling.pdf> (access date: 24.06.2009).
21. *Ranganathan A., Lei H.* Context-Aware Communication // *Computer*. 2003. № 4. P. 90–92.

22. *Rakotonirainy A., Loke S., Fitzpatrick G.* Context-Awareness for the Mobile Environment [Electronic resource]: position paper for CHI2000 Workshop on “The What, Who, Where, When, Why, and How of Context Awareness”, The Hague, The Netherlands, 2000. URL: <http://goanna.cs.rmit.edu.au/~swloke/writings/context-andry.ps> (access date: 08.06.2009).
23. *McCarthy J.* Notes on Formalizing Context // Proc. Intern. Joint Conf. on Artificial Intelligence, Chambéry, France, 1993. Vol. 1. Morgan Kaufmann, 1993. P. 555–560.
24. What is Cyc? [Electronic resource] / Cycorp, Inc.: [site]. [2002–2009]. URL: <http://www.cyc.com/cyc/technology/whatiscyc> (access date: 21.05.09).
25. Protégé [Electronic resource] / USA, Stanford Center for Biomedical Informatics Research: [site]. [2009]. URL: <http://protege.stanford.edu/> (access date: 28.05.2009).
26. *Pincus Z., Musen M.* Contextualizing Heterogeneous Data for Integration and Inference // AMIA Ann. Symp. Proc. 2003. P. 514–518.
27. *Brézillon P.* Context dynamic and explanation in contextual graphs // Modeling and Using Context (CONTEXT-03. Lecture Notes in Artificial Intelligence. 2003. Vol. 2680. P. 94–106.
28. *Гаверилова Т. А.* Использование онтологий в системах управления знаниями [Электронный ресурс] / Бизнес Инжиниринг Групп: [сайт]. [2003]. URL: http://bigc.ru/publications/bigspb/km/use_ontology_in_suz.php (access date: 28.05.2009).
29. *Berners-Lee T., Hendler J., Lassila O.* The Semantic Web // Scientific American. 2001. Vol. 284, № 5. P. 34–43.
30. *Gruber T.* Ontology // Encyclopedia of Database Systems. Springer, 2009. To appear.
31. Ontology [Electronic resource] / Semantic Web: [site]. [2009]. URL: <http://semanticweb.org/wiki/Ontology> (access date: 28.05.2009).
32. *Wiederhold G.* An Algebra for Ontology Composition // Proc. 1994 Monterey Workshop on Formal Methods. U.S. Naval Postgraduate School, Monterey CA, 1994. P. 56–61.
33. *Mitra P., Wiederhold G.* An Ontology Composition Algebra // Handbook on Ontologies; eds. S. Staab, R. Studer. Berlin; Heidelberg; New York: Springer-Verlag, 2003. P. 93–113.
34. *Firat A.* Information Integration Using Contextual Knowledge and Ontology Merging [Electronic resource]: PhD Thesis: 27.08.03: defended Sep. 2003. MIT, 2003. 151 p. URL: <http://www.mit.edu/~bgrosopf/paps/phd-thesis-aykut-firat.pdf> (access date: 29.05.2009).
35. *Firat A., Madnick S., Groszof B.* Contextual Alignment of Ontologies in the eCOIN Semantic Interoperability Framework // Information Technology and Management. 2007. Vol. 8, № 1. P. 47–63.
36. *Hitzle P., Krötzsch M., Ehrig M., Sure Y.* What Is Ontology Merging? A Category-Theoretical Perspective Using Pushouts // Proc. First Intern. Workshop on Contexts and Ontologies: Theory, Practice and Applications (C&O) held in conjunction with the 20th National Conf. on Artificial Intelligence (AAAI-05), Pittsburgh, Pennsylvania, 2005. AAAI Press, Menlo Park, California, 2005. P. 104–107.
37. *Noy N. F., Musen M. A.* Algorithm and Tool for Automated Ontology Merging and Alignment // Proc. Seventeenth National Conf. on Artificial Intelligence (AAAI-2000). USA, Austin, 2000. P. 450–455.
38. *Richardson B., Mazlack L. J.* Approximate Ontology Merging for the Semantic Web // Proc. 2004 IEEE Ann. Meet. NA Fuzzy Information Processing Society, 2004. Vol. 2; IEEE Cat. № 04TH8736. P. 641–646.

39. *Stumme G., Maedche A.* Ontology Merging for Federated Ontologies on the Semantic Web // Proc. Intern. Workshop for Foundations of Models for Information Integration (FMII-2001). Viterbo, Italy, 2001. P. 413–418.
40. *Noy N. F., Musen M. A.* The PROMPT Suite: Interactive Tools For Ontology Merging And Mapping // Intern. J. Human-Computer Studies. 2003. Vol. 59, № 6. P. 983–1024.
41. OpenCyc.org. [Electronic resource] / Cycorp, Inc.: [site]. [2001—2009]. URL: <http://opencyc.org/> (access date: 24.06.2009).
42. *Lenat D. B.* Cyc: A Large-Scale Investment in Knowledge Infrastructure // Communications of the ACM. 1995. Vol. 38, № 11. P. 33–38.
43. *Kahlert B. R. C., Sullivan J.* Microtheories [Electronic resource] // First Intern. Workshop “Ontology Based Modeling in the Humanities”. Germany, University of Hamburg, 2006. URL: <http://clio-knows.sourceforge.net/Microtheories-v2.pdf> (access date: 21.05.2009).
44. *Masters J.* Structured Knowledge Source Integration and its Applications to Information Fusion [Electronic resource] // Proc. Fifth Intern. Conf. on Information Fusion, USA, Annapolis, 2002. P. 1340–1346. URL: http://www.cyc.com/doc/white_papers/fusion2002.pdf (access date: 25.05.2009).
45. *Понов Э. В.* Экспертные системы. М.: Наука. 1987. 288 с.
46. Искусственный интеллект. В 3-х кн. Кн. 2. Модели и методы: справочник. М.: Радио и связь, 1990. 304 с.
47. *Riedl R.* Knowledge Management for Interstate E-Government [Electronic resource] // Proc. Workshop on Distributed Knowledge and e-Government, Siena, Italy, 2001. URL: <http://www.ifi.unizh.ch/egov/Sienanew.pdf> (access date: 24.06.2009).
48. *Смирнов А. В.* и др. Формирование контекста задачи для интеллектуальной поддержки принятия решений // Фундаментальные основы информационных технологий и систем: сб. науч. тр. / Ин-т системного анализа Рос. акад. наук. М.: ИСА РАН. 2004. Т. 9. С. 125–138.
49. *Smirnov A. et. al.* Ontology-Driven Knowledge Logistics Approach as Constraint Satisfaction Problem // On the Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE. Proc. Intern. Conf. on Ontologies, Databases and Applications of Semantics Information Reuse and Integration (ODBASE’2003), 2003. Catania, Sicily, Italy.. Lecture Notes in Computer Science. Vol. 2888. P. 635–652.
50. *Levashova T., Lundqvist M., Pashkin M.* Provision of Context-Sensitive Enterprise Knowledge for Decision Support: An Approach Based on Enterprise Models and Information Demand Contexts // Proc. 9th Intern. Conf. on Enterprise Information Systems (ICEIS 2007), Funchal, Madeira, Portugal, 2007. Vol. AIDSS. P. 88–93.
51. *Левашова Т. В.* Модель контекста в системах интеллектуальной поддержки принятия решений // Поддержка принятия решений: сб. науч. тр. / Ин-т системного анализа РАН. Т. 35. М., 2008. С. 33–42.

Смирнов Александр Викторович — д-р техн. наук, проф.; заместитель директора по научной работе Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН), заведующий лабораторией интегрированных систем автоматизации. Область научных интересов: интеллектуальное управление конфигурациями виртуальных и сетевых организаций, логистика знаний, поддержка принятия решений. Число научных публикаций — 304. Адрес: smir@iiias.spb.su; СПИИРАН, 14-я линия В. О., д. 39, Санкт-Петербург, 199178, РФ; раб. тел. +7(812)328-2073, факс +7(812)328-4450.

Smirnov Alexander Victorovich — D.Sc., Prof.; a Deputy Director for Research and a Head of Computer Aided Integrated Systems Laboratory at St. Petersburg Institute for Informatics

and Automation of the Russian Academy of Sciences (SPIIRAS), a full professor of St. Petersburg State Electrical Engineering University. Research interests: intelligent configuration management of virtual and network organizations, knowledge logistics, decision support. The number of publications — 304. Address: smir@iias.spb.su; SPIIRAS, 39, 14th Line V. O., St. Petersburg, 199178, Russia; tel. +7(812)328-2073, fax: +7(812)328-4450.

Левашова Татьяна Викторовна — научный сотрудник лаборатории интегрированных систем автоматизации Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Область научных интересов: проблемы представления знаний, управление онтологиями, управление контекстом, web-сервисы. Число научных публикаций — 122. Адрес: oleg@iias.spb.su; СПИИРАН, 14-я линия В. О., д. 39, Санкт-Петербург, 199178, РФ; раб. тел. +7(812)328-8071, факс +7(812)328-4450.

Levashova Tatiana Victorovna — researcher, Computer Aided Integrated Systems Lab. at St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: knowledge representation, ontology management, context management, Web-Services. The number of publications — 122. Address: oleg@iias.spb.su; SPIIRAS, 39, 14th Line V. O., St. Petersburg, 199178, Russia; tel. +7(812)328-8071, fax: +7(812)328-4450.

Пашкин Михаил Павлович — канд. техн. наук, ст. науч. сотр. лаборатории интегрированных систем автоматизации Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Область научных интересов: управление данными и знаниями, многоагентные системы, принятие решений, web-сервисы. Число научных публикаций — 129. Адрес: michael@iias.spb.su; СПИИРАН, 14-я линия В. О., д. 39, Санкт-Петербург, 199178, РФ; раб. тел. +7(812)328-8071, факс +7(812)328-4450.

Pashkin Mikhail Pavlovich — PhD in Tech. Sciences, senior researcher, Computer Aided Integrated Systems Lab. at St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). Research interests: data and knowledge management, multiagent systems, decision support, web-services. The number of publications — 129. Address: michael@iias.spb.su; SPIIRAS, 39, 14th Line, St. Petersburg, 199178, Russia; tel. +7(812)328-8071, fax: +7(812)328-4450.

Поддержка исследований. Работа выполнена при финансовой поддержке РФФИ (проект № 08-07-00264) и Президиума РАН (проект № 213).

Рекомендовано СПИИРАН, директор Юсупов Р.М., чл.-корр. РАН.
Статья поступила в редакцию 25.06.2009.