

Разработка мультиаспектной онтологии для поддержки принятия решений в производственных системах

Н. Г. Шилов

Санкт-Петербургский Федеральный исследовательский центр РАН, Санкт-Петербург, Россия

Аннотация. В статье предложен подход к разработке мультиаспектной онтологии для поддержки принятия решений в области технического обслуживания производства. Мультиаспектная онтология основана на многоуровневом подходе к интеграции знаний о различных аспектах комплексной проблемной области (ее составляющих, «подобластях») с сохранением автономности исходных онтологий. Разработанная онтология обеспечивает взаимодействие аспектов с помощью механизмов логического вывода, повышая эффективность информационных потоков и степень автоматизации связанных с ними процессов. Приведенный пример показывает, что предложенный подход может значительно снизить степень вовлеченности операторов в процессы технического обслуживания на предприятии, а также снизить когнитивную нагрузку на операторов и специалистов по техническому обслуживанию.

Ключевые слова: мультиаспектная онтология, поддержка принятия решений, производственная система, техническое обслуживание производства.

DOI 10.14357/20718632240205

EDN MVPWUM

Введение

В настоящее время производственные системы переходят в новую цифровую эру, что делает возможным интеллектуализацию поддержки принятия решений и сопутствующую автоматизацию различных процессов на производстве. Одним из результатов этого перехода в производственном контексте является появление киберфизических производственных систем (КФПС). Киберфизические системы (КФС) – это системы, тесно интегрирующие физический мир (физические ресурсы и процессы) и виртуальный (кибернетический) мир (информационные технологии и сети). Таким образом, КФПС — это КФС, используемые в производственных системах [1].

В КФПС накоплены разнообразные базы данных и присутствуют информационные хранилища, которые могут быть использованы для принятия обоснованных решений. Однако они часто фрагментированы и распределены между различными отделами, имеют разную терминологию и описаны в разных нотациях, а их информационные модели зачастую несовместимы друг с другом. При этом процессы часто пересекаются и имеют значительные общие фрагменты информации, так что изменение информации одним процессом должно быть учтено в ряде других процессов, что приводит к существенной сложности интеграции информации в КФПС при поддержке принятия решений [2].

Способность систем взаимодействовать для достижения общих целей, включая обмен информацией и знаниями между ними посредством процессов, которые они поддерживают, и обмен данными между информационными системами" называется интероперабельностью [3]. Семантическая интероперабельность определяется как общая семантическая интерпретация информации, описываемой с помощью метамоделей. Сегодня нельзя сказать, что проблема обеспечения семантической интероперабельности для интеграции информации в многодоменных информационных экосистемах решена [4–6].

В настоящее время наиболее популярным инструментом решения этой проблемы являются онтологии. Онтологии определяются как «формальные концептуализации областей интересов, которыми могут обмениваться разнородные приложения» [7], и включают понятия, связи между ними и аксиомы, существующие в соответствующих областях. Хотя онтологии и являются популярной темой исследований [8–10], по-прежнему отсутствуют общие мультидисциплинарные онтологии, которые поддерживали бы распространение изменений и запросов между проблемными областями. Использование онтологий в мультидоменных экосистемах по-прежнему проблематично, поскольку последним приходится оперировать разными терминологиями и формализмами [11–13]. Таким образом, научной проблемой, решаемой в статье, является исследование применения мультиаспектной онтологии (МАО) для повышения эффективности процессов поддержки принятия решений в производственной системе на примере технического обслуживания.

1. Подходы к созданию онтологий для комплексных проблемных областей

Для обеспечения гибкой организации труда в контексте Индустрии 4.0 необходима совокупность стандартов для успешного взаимодействия типа человек-машина и машина-машина, что в свою очередь влечет необходимость разработки семантических стандартов, которые удовлетворяют требованиям КФПС [14]. Для семантического представления знаний в рамках

Индустрии 4.0 необходимы не только высоковолневые онтологии, но и предметные онтологии, охватывающие различные приложения, которые могут быть интегрированы для реализации механизмов логического вывода и обмена знаниями [15, 16]. Однако работа с несколькими независимыми областями знаний затрудняет построение общей онтологии. В результате, в мультидоменных областях применение онтологий осложнено, поскольку они должны иметь дело с междисциплинарной информацией.

Одним из направлений решения этой проблемы является построение одной большой онтологии или обогащение существующих онтологий дополнительной информацией, изначально представленной в другой нотации (например, семантические аннотации, расширения DAML+OIL) [17, 18]. Однако в рамках такого решения любое изменение может вызвать необходимость проверки всей онтологии, а формирование специфических моделей знаний путем наследования или интеграции онтологий в рамках общей модели может привести к созданию большого количества различных слабосвязанных расширений, что не позволит передавать знания между ними без потерь.

Описание знаний в мультидоменных системах часто выполняется на основе использования общей онтологии и ее расширения для конкретных проблемных областей. В области технического обслуживания есть решения, основанные на построении одной сложной онтологии (например, [19]). Однако, если ее компоненты являются динамичными, особенно, когда они регулярно обновляются с помощью автоматизированного обучения онтологий, такой подход требует постоянной адаптации общей онтологии и ее соответствия входящим в нее онтологиям, что делает подход неэффективным. Кроме того, перевод знаний из одного формализма в другой также может привести к потере информации и знаний.

Еще одним подходом к описанию мультидоменных областей является создание дополнительного уровня поверх имеющихся онтологий, оставляя их в исходном состоянии. Авторы работы [20] предлагают многоуровневую модель для интеграции информации из источников, разнородность которых вызвана использованием

различных моделей (например, реляционной модели, RDF или XML) и терминологий. Предложенная модель основана на глобальной концептуализации и определении метаданных для сопоставления понятий из разных источников. Подход, представленный в работе [21], ориентирован на многооблачные системы, в которых облака могут быть как семантически, так и синтаксически разными. Подход основан на промежуточной онтологической модели, которая определяет сопоставление между ней самой и моделями облаков, тем самым обеспечивая сопоставление между облаками. Необходимость рассмотрения проблемной области с разных точек зрения (представления знаний, используемых для отдельных процессов или задач) привела к созданию онтологии для различных точек зрения (Multi-ViewPoint ontology). Ее авторы также предложили и язык для описания таких онтологий, названный MVP-OWL [22].

Применение многоуровневых онтологий представляется наиболее перспективным для решения рассматриваемой в статье проблемы, поскольку помимо разрешения терминологических проблем такие онтологии также позволяют сохранять оригинальные формализмы, используемые в онтологиях отдельных проблемных областей.

2. Предложенный подход к созданию мультиаспектной онтологии

Вышеизложенный анализ литературы говорит о возможном применении МАО на производстве при взаимодействии человек – машина и машина – машина. Одной из перспективных в этом отношении производственных задач является техническое обслуживание, где сбои оборудования требуют выполнения сложных бизнес-процессов по техническому обслуживанию, для которых необходимы системы, использующие методы искусственного интеллекта. Сегодня случаи простого оборудования по-прежнему в основном документируются людьми, что приводит к их неоднородным описаниям. Другой процесс устранения неисправностей включает документирование успешного решения проблем, проведенное назначенным специалистом по техническому обслуживанию (техником).

Этот управляемый человеком процесс очень трудоемкий и подвержен ошибкам. Кроме того, результаты документирования обычно не используются в дальнейшем и имеют разное качество, поскольку разные техники применяют разные формулировки и уровни детализации. Предложенная концепция интеллектуальной поддержки принятия решений на основе МАО призвана решить описанные проблемы (Рис. 1). Согласно ей люди и машины обмениваются данными через интерфейсы на локальном уровне, который состоит из онтологий различных проблемных областей: машины передают данные своих датчиков, а техник получает метаинформацию и предварительно заполненные отчеты по техническому обслуживанию.

Предлагаемая МАО (семантический уровень на Рис. 1) имеет три уровня: 1) глобальный уровень описывает понятия и отношения, относящиеся ко всем аспектам; 2) аспектный уровень описывает понятия и отношения, относящиеся к одному аспекту, но «видимые» из других аспектов; 3) локальный уровень описывает понятия и отношения, относящиеся к одному аспекту (как «видимые» из других аспектов, так и нет).

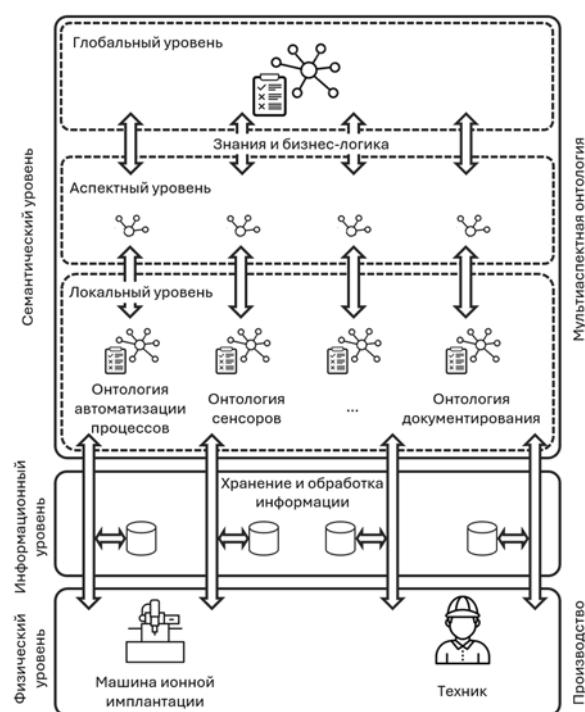


Рис. 1. Использование мультиаспектной онтологии при поддержке принятия решений в производственной системе

Аспекты обозначают различные области рассматриваемой сложной проблемы. Например, проблема интеллектуального технического обслуживания может иметь такие аспекты, как автоматизация процессов (описание производственного оборудования и процессов, происходящих на предприятии), датчики (описание вспомогательного оборудования [фактически, различные датчики], отвечающего за сбор данных, необходимых для принятия решений о требуемых вмешательствах в работу оборудования) и документирование (описание административных процессов, связанных с документированием различных процессов и действий, происходящих на предприятии).

Первые два уровня (глобальный и аспектный) описываются с помощью формализма, который является глобальным для системы, а третий (локальный) – во внутреннем формализме аспекта. Между понятиями разных уровней должны быть установлены связи.

При разработке нотации для описания МАО были проанализированы различные нотации, направленные на описание онтологий, объединяющих несколько концептуализаций (онтологии различных проблемных областей, модулей, точек зрения), а именно: распределенная дескрипционная логика (Distributed Description Logic /DDL), контекстный OWL (Context OWL), OWL с несколькими точками зрения (Multi-Viewpoint OWL / MVP-OWL), эпсилон-отношения (epsilon-relations), пакетная дескрипционная логика (Package-based Description Logic / P-DL) и другие. Разработанная нотация основана на MVP-OWL, поскольку она также ориентирована на создание единой онтологии, а не на координацию и сравнение различных онтологий, но отличается от нее набором используемых связывающих отношений.

Согласно данной нотации, все элементы онтологии (классы и свойства OWL) делятся на глобальные (видимые из двух и более аспектов) и локальные (видимые только из одного аспекта): C^G – множество глобальных классов; P^G – множество глобальных свойств. Кроме того, вводятся понятия аспекта (A) и связывающих отношений (R). Экземпляры объектов (индивидуи OWL) могут быть только локальными, но они могут быть отражены в других аспектах, если их классы связаны соответствующим отношением.

Таким образом, МАО описывается как: $O = (C^G, P^G, A, R)$. Каждый аспект описывается как:

$A_i = (C_{A_i}^A, P_{A_i}^A, E_{A_i}^L, R_{A_i} V_{A_i})$, где $i \in I$ – множество аспектов МАО;

$C_{A_i}^A$ – множество классов аспекта A_i ;

$P_{A_i}^A$ – множество свойств аспекта A_i ;

$E_{A_i}^L$ – множество локальных элементов аспекта A_i , специфичных для формализма аспекта;

R_{A_i} – множество отношений эквивалентности между элементами (классами и свойствами) аспекта A_i и его локальными элементами: $e_j^A \stackrel{\equiv}{\rightarrow} e_k^L$, где $e_j^A \in C_{A_i}^A \cup P_{A_i}^A$, $e_k^L \in E_{A_i}^L$; V_{A_i} – множество экземпляров аспекта A_i .

Определены следующие три типа связывающих отношений, соответствующие отношениям, определенным в распределенной дескрипционной логике:

$G: c \rightarrow A_i: d$ – связывающее отношение включения, означающее, что экземпляр глобального элемента c является экземпляром элемента d аспекта A_i ;

$G: c \rightarrow A_i: d$ – связывающее отношение включения, означающее, что экземпляр элемента d аспекта A_i является экземпляром глобального элемента c ;

$G: c \leftrightarrow A_i: d$ – двунаправленное связывающее отношение "эквивалентности", означающее, что множества экземпляров глобального элемента c и элемента d аспекта A_i равны.

Отличительными особенностями этой нотации являются независимость от локальных аспектных нотаций, совместимость с языком OWL, а также ориентация на интеграцию слабо связанных аспектов, в отличие, например, от DDL или Context OWL, ориентированных на координацию и согласование различных онтологий, или MVP-OWL, ориентированных на интеграцию сильно пересекающихся точек зрения на проблемную область.

3. Создание мультиаспектной онтологии для производственной системы

Проведенный анализ существующих методологий разработки онтологий [23–25] показал, что этот процесс обычно состоит из 5 шагов:

1) определение цели и проблемной области онтологии; 2) определение понятий и связей, которые должны быть включены в онтологию, а также терминов для их именования; 3) редактирование онтологии; 4) верификация онтологии; 5) валидация онтологии. Эти шаги были использованы в качестве руководства при разработке методологии создания МАО для интеллектуальной производственной системы [26]. В итоге были определены следующие шаги создания МАО:

1. Спецификация требований к интероперабельности для определения цели и проблемной области МАО.

2. Спецификация аспектов МАО.

3. Разработка онтологий проблемных областей (или повторное использование и адаптация существующих онтологий проблемных областей) для каждого из указанных аспектов ("онтологии аспектов").

4. Интеграция аспектов и формирование «глобального уровня» на основе понятий, которые считаются общими для нескольких аспектов.

5. Верификация онтологии.

6. Валидация онтологии.

Ниже подробно описана реализация данных шагов на примере МАО производственной системы.

3.1. Цель и проблемная область мультиаспектной онтологии

В рассматриваемом примере, выбранном совместно с экспертами, используются данные из полупроводниковой промышленности и демонстрируются типичные действия, предпринимаемые при отказе оборудования (машины). Полупроводниковая промышленность использует сложные машины, зачастую тесно связанные между собой посредством сетевых технологий. Документирование рабочих процессов в настоящее время все еще требует значительного участия со стороны человека. Обычно оператор машины уведомляется об отклонении значения какого-либо параметра с помощью сообщения об ошибке от машины. Если оператор может легко исправить проблему, она решается немедленно. Однако в большинстве случаев это невозможно из-за высокой степени сложности машин. Поэтому оператор машины регистрирует ее отказ и фиксирует краткое описание проблемы в отчете. Затем уведомляется один из

дежурных техников, имеющий необходимую компетенцию. Техник работает над устранением проблемы и составляет отчет о техническом обслуживании после его успешного выполнения. Этот процесс требует много времени и может привести к значительному увеличению среднего времени обнаружения отказа, если описание оператором проблемы является неточным. Целью использования МАО является поддержка данного процесса для повышения эффективности процессов принятия решений и сокращения среднего времени обнаружения отказа, а также времени, необходимого для документирования. В примере рассматривается машина для ионной имплантации со встроенным датчиком, который измеряет температуру при создании вакуума. Рабочая температура находится в диапазоне от 300°C до 400°C. Если температура превышает 400°C, генерируется сообщение об ошибке. В настоящий момент эта проблема не может быть решена без вмешательства оператора машины. МАО описывает элементы процесса и связи между ними для поддержки принятия решений: в случае превышения предельного значения температуры должен назначаться соответствующий техник для диагностики и устранения проблемы (непосредственно через систему, основанную на МАО, или на основе ее рекомендации), и генерироваться автоматический отчет посредством связей семантического и физического уровней с информационным.

3.2. Спецификация аспектов мультиаспектной онтологии

На основе приведенной выше задачи были определены следующие три основных аспекта МАО: 1) аспект автоматизации процессов (*process automation domain*), отвечающий за определение оборудования, участников производственных процессов, а также других необходимых концептов; 2) аспект датчиков (*sensors domain*) отвечает за описание и анализ данных, собираемых датчиками; 3) аспект документирования (*documenting domain*) отвечает за генерацию отчетов об ошибках в случае отказов оборудования.

3.3. Разработка аспектных онтологий

Для разработки аспектных онтологий был выбран подход, ориентированный на повторное

использование существующих онтологий. Было проанализировано несколько онтологий в качестве потенциальных кандидатов для определения аспектов, и на основе их соответствия рассматриваемой задаче и уровню детализации были выбраны следующие: 1) Core Ontology for Robotics and Automation [27] (базовая онтология для робототехники и автоматизации) была использована для формирования аспекта автоматизации процессов и адаптирована путем внесения незначительных изменений; 2) Semantic Sensor Network [28] (семантическая сеть датчиков) была выбрана для аспекта датчиков и адаптирована путем внесения незначительных изменений; 3) SD3 – Simulation Delivery and Documentation Deviations [29] (отклонения в получении результатов и документировании при моделировании) была выбрана для формирования аспекта документирования и значительно сокращена, поскольку изначально содержала большое число классов, специфичных для медицины.

Обзор полученной онтологии представлен на Рис. 2 и подробно описан ниже.

- Аспект автоматизации процессов (A_0):

- класс *Object* (объект), который является подклассом *Physical* (физический объект),

представляет все физические объекты проблемной области;

- класс *Maintenance_Technician* (техник) является подклассом класса *Human* (человек), который, в свою очередь, является подклассом класса *Agent* (агент), который, также в свою очередь, является подклассом класса *Object*. Каждый экземпляр класса *Maintenance_Technician* может быть связан объектным отношением *isCompatibleWith* (совместим с) с экземпляром класса *Machine* (машина), который является подклассом класса *Agent*. Аналогичным образом определен и класс *Operator* (оператор).

- формирование экземпляров класса *Repair_Assignment* (задание на ремонт) инициируется возникновением неисправности, описываемой в другом аспекте (см. ниже).

- Аспект документирования (A_1):

- класс *Deviation_report* (отчет об отклонении показателя) автоматизирует генерацию отчетов при возникновении неисправности. Экземпляр данного класса формируется сразу после обнаружения отклонения значения температуры. Данная логика заменяет ручную работу по документированию неисправности оператором машины и предоставляет

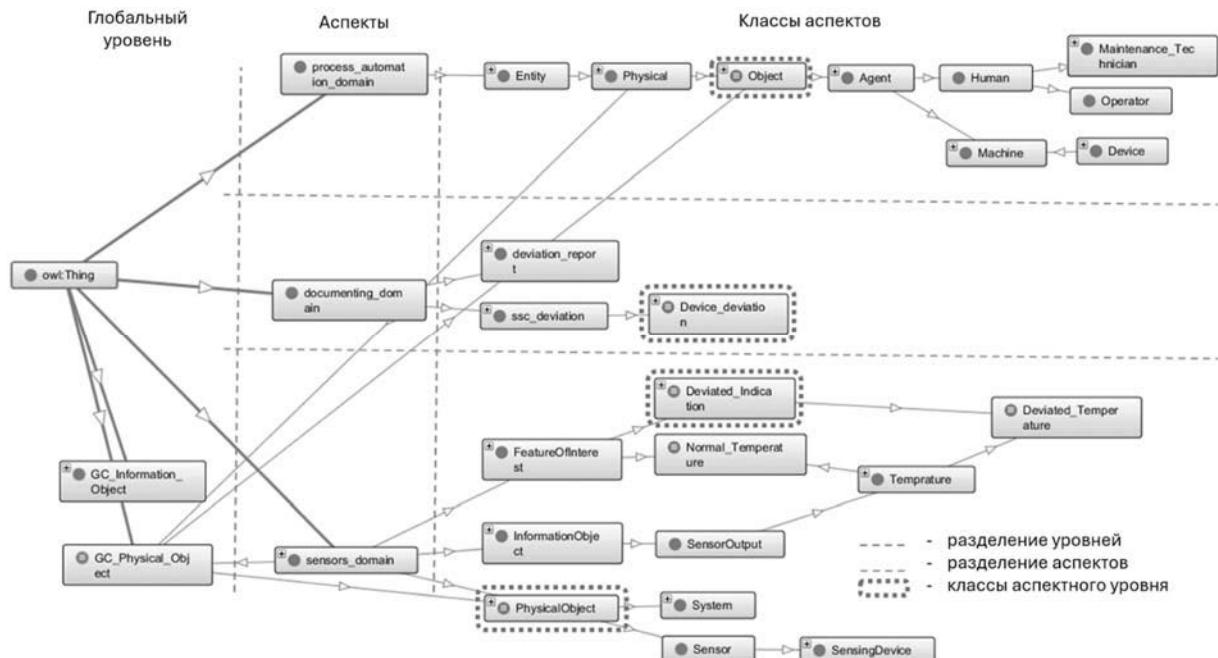


Рис. 2. Фрагмент таксономии разработанной мультиаспектной онтологии

ляет технику точный отчет о том, какой датчик измерил отклонение температуры, в какое время, а также значение отклонения;

- класс *Device_deviation* (устройство с показателем, имеющим отклонение) описывает конкретные неисправности, вызванные нештатной работой устройства (например, машины).

- Аспект датчиков (*A₂*) описывает структуру системы (класс *System*, являющийся подклассом класса *Physical*), включая датчики, которыми она оснащена, и их показания:

- класс *System* представлен экземпляром *Implantation_system* (система имплантации), который имеет объектные отношения *has_subsystem* (имеет подсистему) с экземплярами класса *Temp_sensor* (датчик температуры);

- класс *Sensing_Device* (датчик) имеет экземпляр *Temp_sensor*, который является подсистемой *Implantation_system*;

- класс *Temperature* (температура) является подклассом *Sensor_Output* (показание датчика). Его экземпляр *High_temperature* (высокая температура) формируется экземпляром класса *Temp_sensor* в случае превышения температуры 400°C и автоматически классифицируется как *Deviated_Temperature* (отклонение температуры) – подкласс класса *Temperature* (температура), определенный как *value some xsd:decimal[>= 400]* (значение больше или равно 400). *Deviated_Temperature* также является подклассом класса *Deviated_Indication*.

3.4. Интеграция аспектов и формирование "глобального уровня"

Не считая класса *Thing* («нечто» – родительский класс подавляющего большинства таксономий), только три класса были определены на глобальном уровне (*G*):

- *GC_Information_Object* (глобальный класс «информационный объект») описывает информационные объекты, представленные различными классами в различных аспектах

- *Deviation* (глобальный класс «отклонение», подкласс класса *GC_Information_Object*) описывает отклонения в различных аспектах. Он связан отношениями эквивалентности с классом

Deviated_Indication аспекта датчиков и классом *Device_deviation* аспекта документирования:

$$G: Deviation \xrightarrow{\equiv} A_2: Deviated_Indication$$

$$G: Deviation \xrightarrow{\equiv} A_1: Device_deviation$$

- *GC_Physical_Object* (глобальный класс «физический объект») описывает физические объекты в различных аспектах. Он связан отношениями эквивалентности с классом *PhysicaObject* аспекта датчиков и классом *Object* аспекта автоматизации процессов:

$$G: GC_Physical_Object \xrightarrow{\equiv} A_2: PhysicaObject$$

$$G: GC_Physical_Object \xrightarrow{\equiv} A_0: Object$$

3.5. Определение правил

SWRL (Semantic Web Rule Language / язык правил семантического веба, объединяющий OWL и RuleML, <https://www.w3.org/Submission/SWRL/>) – это язык, предназначенный для описания правил в онтологиях OWL. Правила, определенные в SWRL, состоят из антецедента (предпосылки), называемом «телом» правила, и следствия, называемом его «головой», каждый из которых, в свою очередь, состоит из набора атомов. После атома(ов) антецедента следует знак "->", за которым, в свою очередь, следует атом(ы) следствия.

Для достижения желаемого уровня интероперабельности экспертами были изучены проблемные области и определены и добавлены в соответствующие аспекты правила SWRL. Некоторые из них представлены ниже:

Присваивание значения свойству *hasProblem* (имеет проблему) экземпляру класса *Deviation_report*, при наличии отклонения (аспект документирования):

$$A_1: Device_deviation(? x) \wedge$$

$$A_1: Deviation_report(? y) \rightarrow$$

$$A_1: hasProblem(? y, ? x)$$

Присваивание значения свойству *hasProblem* экземпляру класса *Sensing_Device*, если он предоставляет *Deviated_Indication* (аспект датчиков):

$$A_2: Deviated_Indication(? x) \wedge$$

$$A_2: isProducedBy(? x, ? y) \wedge$$

$$A_2: SensingDevice(? y) \rightarrow$$

$$A_2: hasProblem(? y, ? x)$$

Присваивание значения свойству *hasProblem* экземпляра класса *Machine*, если у ее компонента есть проблема (аспект автоматизации процессов):

$$A_0: Machine(?x) \wedge A_0: hasPart(?x, ?y) \wedge$$

$$A_0: hasProblem(?y, ?z) \rightarrow$$

$$A_0: hasProblem(?x, ?z)$$

Присваивание значения свойству *hasAssignment* экземпляру класса *Maintenance_Technician*, если у объекта есть проблема и техник (экземпляр класса *Maintenance_Technician*) совместим с этим объектом (аспект автоматизации процессов):

$$A_0: Maintenance_Technician(?x) \wedge$$

$$A_0: isCompatibleWith(?x, ?y) \wedge$$

$$A_0: hasProblem(?y, ?z) \rightarrow$$

$$A_0: hasAssignment(?x, ?y)$$

3.6. Верификация онтологии

Верификация онтологии была проведена сначала вручную, а затем с использованием двух машин логического вывода, а именно HermiT и Pellet. Были выявлены и исправлены незначительные проблемы, возникшие при интеграции аспектов. Обе машины логического вывода приводили к идентичным результатам без заметной разницы в скорости их работы.

3.7. Валидация онтологии

Целью валидации разработанной онтологии является проверка ее соответствия исходным требованиям. Для этого был реализован описанный ранее сценарий.

Рост измеренной температуры выше 400°C приводит к ее классификации как *Deviated_Indication* в аспекте датчиков и на глобальном уровне. Это, в свою очередь, вызывает появление экземпляра класса *Deviation_report* в аспекте документирования, который используется для генерации соответствующего отчета. В то же время машина, оснащенная датчиком, предоставившим данные о повышенной температуре, идентифицируется в аспекте автоматизации процессов, и генерируется задание на ремонт технику, обладающему соответствующей компетенцией.

В результате новый процесс, поддерживающий разработанной МАО, выглядит как показано на Рис. 3 (справа), по сравнению с исходным, представленным на Рис. 3 (слева). Видно, что количество шагов рабочего процесса с участием человека (показаны светлыми прямоугольниками), необходимых для выполнения задач технического обслуживания, было сокращено на 37%: три шага рабочего процесса (диагностика причины, документирование и назначение техника) теперь могут выполняться автоматизировано, а не оператором машины (шаги, выполняемые автоматизировано, показаны темными прямоугольниками). Предложенный подход может значительно сократить необходимость вмешательства человека в процессы технического обслуживания. Помимо сокращения трудозатрат, этот автоматизированный процесс ускоряет техническое обслуживание оборудования и снижает когнитивную нагрузку на операторов и техников в задачах документирования. Оператор освобождается от всех рутинных шагов при сообщении о неисправности, в то время как техник для принятия решений обеспечивается четкими и подробными журналами



Рис. 3. Сравнение процессов без (слева) и с (справа) использованием разработанной мультиаспектной онтологии

неисправностей, которые помогают ему понять проблему и решить ее.

4. Технологическая основа прототипа

Поскольку приведенные аспектные онтологии описаны с использованием OWL, для разработки аспектного и глобального уровней применялся инструмент редактирования онтологий Protégé. Глобальный уровень онтологии импортирует онтологии аспектного уровня, которые в свою очередь импортируют онтологии локального уровня. Такая архитектура позволяет независимо модифицировать локальные онтологии, если элементы аспектного уровня не изменяются. В противном случае аспектный уровень должен быть перепроверен и (при необходимости) скорректирован.

Все правила реализованы в локальных онтологиях с использованием SWRL, а обмен знаниями осуществляется только с помощью связывающих отношений. Это позволит в будущем выполнять логический вывод как в локальных онтологиях, так и на аспектном и глобальном уровнях независимо (например, с использованием различных механизмов логического вывода). При этом факты (экземпляры концептов) могут передаваться между уровнями через связывающие отношения с помощью соответствующего инструмента программирования.

Как упоминалось ранее, логический вывод в прототипе выполняется с помощью Hermit или Pellet. Сценарий активируется путем определения необходимых экземпляров классов (объектов) в соответствующих онтологиях (например, температуры в аспекте датчиков) с помощью специально разработанного сервиса, который предоставляет соответствующую информацию из связанных систем мониторинга производства и систем управления производственными процессами, где хранится информация о работе оборудования.

При появлении новых объектов в онтологии активируется машина логического вывода. Появление объектов определенных классов в локальных онтологиях (в результате работы машины логического вывода) активирует дополнительные программные модули. Например, появление экземпляра класса *Deviation_report* активирует

инструмент, который заполняет шаблон отчета информацией, взятой из этого экземпляра, и сохраняет его в центральной базе данных хранения документов, а появление экземпляра класса *Repair_Assignment* активирует инструмент, который формирует рекомендацию по назначению техника, которая предоставляется соответствующему менеджеру для принятия окончательного решения.

5. Обсуждение результатов

Предложенный подход реализует поддержку принятия решений с помощью МАО и разработанной нотации для интеграции знаний из различных аспектов рассматриваемой области. Эффективность подхода проиллюстрирована на примере из полупроводниковой промышленности, связанном с задачей технического обслуживания производства.

В частности, было сокращено среднее время обнаружения отказа благодаря диагностике причины, поддерживаемой МАО, что облегчило диагностику неисправностей техником. Это непосредственно влияет и на среднее время ремонта, которое дополнительно сокращается за счет сокращения времени, необходимого для документирования, поскольку отчеты частично или полностью предварительно заполняются с помощью предложенных онтологических механизмов. Однако эти преимущества также представляют собой и потенциальные ограничения, поскольку в случае неправильной диагностики, например, из-за неправильно распознанной или неизвестной причины, может потребоваться больше усилий для диагностики, и предварительно заполненные отчеты потребуют корректировки. Однако эти ограничения в значительной степени компенсируются масштабированием предложенного подхода на нескольких производственных площадках, поскольку в этом случае система получает обратную связь быстрее и при необходимости дополняется новыми знаниями.

Проведенный эксперимент подтверждает обоснованность подхода. В отличие, например, от чистого машинного обучения или других подходов, предложенный подход выигрывает от использования семантики, поддерживающей интероперабельность между существующими

системами технического обслуживания и позволяет выполнять логический вывод над всеми онтологиями, интегрированными в МАО. С другой стороны, в отличие от подходов, основанных на использовании единой общей онтологии, предлагаемый подход обладает большей масштабируемостью, поскольку логический вывод выполняется поэтапно: сначала на локальных онтологиях, а затем на глобальном уровне, и сохраняет автономию локальных онтологий. В результате обновление или расширение локальной онтологии (вручную или с помощью автоматического обучения онтологии) в большинстве случаев не влияет на глобальный уровень и другие локальные онтологии, и наоборот, обновление глобального уровня (например, введение новой задачи) не требует обновления локальных уровней. Расширение МАО также упрощается, поскольку добавление нового аспекта (новой локальной онтологии) не требует изменения других локальных онтологий, а только разработки соответствующего аспектного уровня и, возможно, некоторых изменений на глобальном уровне. Однако данный процесс раскрывает еще одно ограничение подхода, а именно, создание МАО является ручным процессом, требующим значительных усилий. Его можно упростить с помощью применения методов сопоставления онтологий, но более высокий уровень автоматизации все еще невозможен и может быть предметом будущих исследований.

Заключение

В статье представлен подход к совершенствованию поддержки принятия решений на основе использования МАО, который позволил сократить время и количество ручного труда в процессе технического обслуживания на производстве. Предложенный подход к созданию МАО позволяет описывать знания о сложной проблемной области с помощью разработанной нотации. Она способна интегрировать различные аспекты проблемной области (например, автоматизация процессов, документирование и датчики) путем использования существующих онтологий, сохраняя их независимость и внутренние формализмы. Это достигается за счет применения многоуровневой архитектуры МАО.

Использование подхода проиллюстрировано на примере из полупроводниковой промышленности, связанном с задачей технического обслуживания. Результаты показали, что подход может значительно сократить необходимость вмешательства человека в процессы технического обслуживания, а также снизить когнитивную нагрузку на операторов машин и техников за счет автоматизации нескольких этапов рабочего процесса.

Литература

1. Kumar R. et al. A cyber physical production system framework for online monitoring, visualization and control by using cloud, fog, and edge computing technologies // International Journal of Computer Integrated Manufacturing. 2023. Vol. 36, № 10. P. 1507–1525.
2. Ansari F. Knowledge Management 4.0: Theoretical and Practical Considerations in Cyber Physical Production Systems // IFAC-PapersOnLine. 2019. Vol. 52, № 13. P. 1597–1602.
3. European Commission. New European Interoperability Framework: Promoting seamless services and data flows for European public administrations [Electronic resource]. 2017. P. 48. URL: https://ec.europa.eu/isa2/sites/isa/files/eif_brochure_final.pdf.
4. Hafidi M.M. et al. Semantic web and machine learning techniques addressing semantic interoperability in Industry 4.0 // International Journal of Web Information Systems. 2023. Vol. 19, № 3/4. P. 157–172.
5. Атаева О.М., Калёнов Н.Е., Серебряков В.А. Онтологический подход к описанию единого цифрового пространства научных знаний // Электронные библиотеки. 2021. Том 24, № 1. С. 3–19.
6. Дородных Н.О., Николайчук О.А., Юрин А.Ю. Использование онтологических шаблонов содержания при построении баз знаний для технического обслуживания и ремонта авиационной техники // Онтология проектирования. 2022. Том 12, № 2. С. 158–171.
7. Gruber T.R. A translation approach to portable ontology specifications // Knowledge Acquisition. 1993. Vol. 5, № 2. P. 199–220.
8. Семенова В.А., Смирнов С.В. Модели и методы онтологического анализа данных в задаче структурного анализа и синтеза технических решений // Онтология проектирования. 2023. Том 13, № 4. С. 531–547.
9. Грибова В.В., Паршкова С.В., Федорищев Л.А. Онтологии для разработки и генерации адаптивных пользовательских интерфейсов редакторов баз знаний // Онтология проектирования. 2022. Том 12, № 2. С. 200–217.
10. Кудрявцев Д.В., Гаврилова Т.А., Смирнова М.М., Головачева К.С. Построение онтологии знаний потребителя в маркетинге: кроссдисциплинарный подход // Искусственный интеллект и принятие решений. 2021. № 3. С. 19–32.

11. Bader S.R. et al. A Knowledge Graph for Industry 4.0 // Lecture Notes in Computer Science. Springer, 2020. Vol. 12123. P. 465–480.
12. Шахнов В.А. и др. Представление знаний в информационных системах с учетом свойств наноразмерных объектов и материалов // Информационные технологии и управляющие системы. 2014. № 4. С. 89–96.
13. Боргест Н.М. Онтология проектирования научного направления: формирование, развитие, примеры // Онтология проектирования. 2022. Том 12, № 2. С. 136–157.
14. El Zaatari S. et al. Cobot programming for collaborative industrial tasks: An overview // Robotics and Autonomous Systems. 2019. Vol. 116. P. 162–180.
15. Kumar V.R.S. et al. Ontologies for Industry 4.0 // Knowledge Engineering Review. 2019. Vol. 34. P. e17.
16. Meriem H., Nora H., Samir O. Predictive Maintenance for Smart Industrial Systems: A Roadmap // Procedia Computer Science. 2023. Vol. 220. P. 645–650.
17. Загорулько Ю.А. и др. Автоматизация разработки онтологий научных предметных областей на основе паттернов онтологического проектирования // Онтология проектирования. 2021. Том 11, № 4. С. 500–520.
18. Lim S.C.J., Liu Y., Lee W.B. A methodology for building a semantically annotated multi-faceted ontology for product family modelling // Advanced Engineering Informatics. 2011. Vol. 25, № 2. – P. 147–161.
19. Karray M.H., Chebel-Morello B., Zerhouni N. A formal ontology for industrial maintenance // Applied Ontology. 2012. Vol. 7, № 3. P. 269–310.
20. Cruz I.F., Xiao H. Ontology Driven Data Integration in Heterogeneous Networks // Complex Systems in Knowledge-based Environments: Theory, Models and Applications. Studies in Computational Intelligence. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. Vol. 168. P. 75–98.
21. Quinton C. et al. Towards multi-cloud configurations using feature models and ontologies // Proceedings of the 2013 international workshop on Multi-cloud applications and federated clouds - MultiCloud '13. New York, New York, USA: ACM Press, 2013. P. 21.
22. Hemam M., Boufaïda Z. MVP-OWL: a multi-viewpoints ontology language for the Semantic Web // International Journal of Reasoning-based Intelligent Systems. 2011. Vol. 3, № 3/4. P. 147.
23. Fernández-López M., Gómez-Pérez A. Overview and analysis of methodologies for building ontologies // Knowledge Engineering Review. 2002. Vol. 17, № 2. P. 129–156.
24. Noy N.F., McGuinness D.L. Ontology Development 101: A Guide to Creating Your First Ontology. 2001. 25 p.
25. Хорошевский, В.Ф. Проектирование систем программного обеспечения под управлением онтологий: модели, методы, реализации // Онтология проектирования. 2019. Том 9, № 4. С. 429–448.
26. Smirnov A. et al. Methodology for Multi-Aspect Ontology Development: Ontology for Decision Support Based on Human-Machine Collective Intelligence // IEEE Access. 2021. Vol. 9. P. 135167–135185.
27. Fiorini S.R. et al. Extensions to the core ontology for robotics and automation // Robotics and Computer-Integrated Manufacturing. 2015. Vol. 33. P. 3–11.
28. Giustozzi F., Saunier J., Zanni-Merk C. Context Modeling for Industry 4.0: an Ontology-Based Proposal // Procedia Computer Science. 2018. Vol. 126. P. 675–684.
29. SD3 - Simulation Delivery and Documentation Deviations [Electronic resource]. 2015. URL: <http://aber-owl.net/ontology/SD3>.

Шилов Николай Германович. Федеральное государственное бюджетное учреждение науки «Санкт-Петербургский Федеральный исследовательский центр Российской академии наук», Санкт-Петербург, Россия. Старший научный сотрудник, кандидат технических наук, доцент. Область научных интересов: онтологический инжиниринг, управление знаниями, машинное обучение. E-mail: nick@iias.spb.su

Development of a Multi-Aspect Ontology for Decision Support in Production Systems

N. G. Shilov

St. Petersburg Federal Research Center of the Russian Academy of Sciences, St. Petersburg, Russia

Abstract. Development of new data-driven concepts based on artificial intelligence methods leads to the appearance of new approaches to decision support in production systems aimed at increasing their efficiency. However, the use of existing uncoordinated data and knowledge to improve the quality of decision-making processes remains a challenging task due to the diversity of their terminologies and cognitive models. The paper proposes an approach to development of a multi-aspect ontology for decision support in production maintenance. The multi-aspect ontology is based on a layered approach to integrating knowledge about various aspects of a complex problem domain (its constituents or subdomains) while preserving the autonomy of the original ontologies. The developed multi-aspect ontology

supports interaction between aspects using inference mechanisms what increases the efficiency of information flows and the degree of automation of related processes. The given example shows that the proposed approach can significantly reduce the involvement of human workers in maintenance processes in an enterprise, as well as the cognitive load on operators and maintenance technicians.

Keywords: multi-aspect ontology, decision support, production system, maintenance/

DOI 10.14357/20718632240205

EDN MVPWUM

References

1. Kumar R., Sangwan K.S., Herrmann C., Thakur S. A cyber physical production system framework for online monitoring, visualization and control by using cloud, fog, and edge computing technologies. *International Journal of Computer Integrated Manufacturing*. 2023;36(10):1507–1525. doi: 10.1080/0951192X.2023.2189312
2. Ansari F. Knowledge Management 4.0: Theoretical and Practical Considerations in Cyber Physical Production Systems. *IFAC-PapersOnLine*. 2019;52(13):1597–1602. doi: 10.1016/j.ifacol.2019.11.428
3. European Commission: New European Interoperability Framework: Promoting seamless services and data flows for European public administrations; 2017. Available from: https://ec.europa.eu/isa2/sites/isa/files/eif_brochure_final.pdf [Accessed 13 March 2024].
4. Hafidi M.M., Djezzar M., Hemam M., Amara F.Z., Maimour M.: Semantic web and machine learning techniques addressing semantic interoperability in Industry 4.0. *International Journal of Web Information Systems*. 2023;19:157–172. doi: 10.1108/IJWIS-03-2023-0046
5. Ataeva O.M., Kalenov N.E., Serebryakov V.A. Ontological approach to the description of a common digital space of scientific knowledge. *Russian Digital Libraries Journal*. 2021;24(1):3–19. doi: 10.26907/1562-5419-2021-24-1-3-19 (In Russ.).
6. Dorodnykh N.O., Nikolaychuk O.A., Yurin A.Yu. Using ontological content patterns in knowledge base engineering for maintenance and repair of aviation equipment. *Ontology of Designing*. 2022;12(2):158–171. doi: 10.18287/2223-9537-2022-12-2-158-171. (In Russ.).
7. Gruber T.R. A translation approach to portable ontology specifications. *Knowledge Acquisition*. 1993;5(2):199–220. doi: 10.1006/knac.1993.1008
8. Semenova V.A., Smirnov S.V. Models and methods of ontological data analysis in the problem of structural analysis and synthesis of technical decisions. *Ontology of Designing*. 2023; 13(4):531–547. doi: 10.18287/2223-9537-2023-13-4-531-547. (In Russ.).
9. Gribova V.V., Parshkova S.V., Fedorishev L.A. Ontologies for development and generation adaptive user interfaces of knowledge base editors. *Ontology of Designing*. 2022; 12(2):200–217. doi: 10.18287/2223-9537-2022-12-2-200-217. (In Russ.).
10. Kudryavtsev D.V., Gavrilova T.A., Smirnova M.M., Golovacheva K.S. Building Ontology of Consumer Knowledge in Marketing: Cross-Disciplinary Approach. *Artificial Intelligence and Decision Making*. 2021; 3:19–32. doi: 10.14357/20718594210302. (In Russ.).
11. Bader S.R., Granel-Gonzalez I., Nanjappa P., Vidal M.-E., Maleshkova M. A Knowledge Graph for Industry 4.0. *Lecture Notes in Computer Science*. 2020;12123:465–480. doi: 10.1007/978-3-030-49461-2_27
12. Shakhnov V.A., Averyanikhin A.E., Vlasov A.I., Zhuravleva L.V., Zinchenko L.A. Nanotechnology Knowledge Representation in Information Systems. *Information Technologies and Control Systems*. 2014;3: 89–96. (In Russ.).
13. Borgest N.M. Ontology of designing a scientific direction: formation, development, examples. *Ontology of designing*. 2022; 12(2):136–157. doi: 10.18287/2223-9537-2022-12-2-136-157. (In Russ.).
14. El Zaatar S., Marei M., Li W., Usman Z. Cobot programming for collaborative industrial tasks: An overview. *Robotics and Autonomous Systems*. 2019;116:162–180. doi: 10.1016/j.robot.2019.03.003
15. Kumar V.R.S. et al. Ontologies for Industry 4.0. *Knowledge Engineering Review*. 2019;34:e17. doi: 10.1017/S0269888919000109
16. Meriem H., Nora H., Samir O. Predictive Maintenance for Smart Industrial Systems: A Roadmap. *Procedia Computer Science*. 2023;220:645–650. doi: 10.1016/j.procs.2023.03.082
17. Zagorko Yu.A. Automation of the development of ontologies of scientific subject domains based on ontology design patterns. *Ontology of Designing*. 2021;11(4):500–520. doi: 10.18287/2223-9537-2021-11-4-500-520. (In Russ.).
18. Lim S.C.J., Liu Y., Lee W.B. A methodology for building a semantically annotated multi-faceted ontology for product family modelling. *Advanced Engineering Informatics*. 2011;25(2): 147–161. doi: 10.1016/j.aei.2010.07.005
19. Karray M.H., Chebel-Morello B., Zerhouni N. A formal ontology for industrial maintenance. *Applied Ontology*. 2012;7(3):269–310. doi: 10.3233/AO-2012-0112
20. Cruz I.F., Xiao H. Ontology Driven Data Integration in Heterogeneous Networks. In: *Complex Systems in Knowledge-based Environments: Theory, Models and Applications. Studies in Computational Intelligence*. Springer; 2009. p. 75–98. doi: 10.1007/978-3-540-88075-2_4
21. Quinton C., Haderer N., Rouvoy R., Duchien L. Towards multi-cloud configurations using feature models and ontologies. In: *Proceedings of the 2013 international workshop on Multi-cloud applications and federated clouds - MultiCloud '13*. ACM Press. 2013; p. 21. doi: 10.1145/2462326.2462332

22. Hemam M., Boufaïda Z. MVP-OWL: a multi-viewpoints ontology language for the Semantic International Journal of Reasoning-based Intelligent Systems. 2011;3(3/4):147. doi: 10.1504/IJRIS.2011.043539
23. Fernández-López M., Gómez-Pérez A. Overview and analysis of methodologies for building ontologies. *Knowledge Engineering Review*. 2002;17(2):129–156. doi: 10.1017/S026988902000462
24. Noy N.F., McGuinness D.L. *Ontology Development 101: A Guide to Creating Your First Ontology*. 2001. 25 p.
25. Khoroshevsky V.F. Ontology Driven Software Engineering: Models, Methods, Implementations. *Ontology of Desuigning*. 2019;9(4):429–448. doi: 10.18287/2223-9537-2019-9-4-429-448. (In Russ.).
26. Smirnov A., Levashova T., Ponomarev A., Shilov N. Methodology for Multi-Aspect Ontology Development: Ontology for Decision Support Based on Human-Machine Collective Intelligence. *IEEE Access*. 2021;9:135167–135185. doi: 10.1109/ACCESS.2021.3116870
27. Fiorini S.R. et al. Extensions to the core ontology for robotics and automation. *Robot. Robotics and Computer-Integrated Manufacturing*. 2015;33:3–11. doi: 10.1016/j.rcim.2014.08.004
28. Giustozzi F., Saunier J., Zanni-Merk C. Context Modeling for Industry 4.0: an Ontology-Based Proposal. *Procedia Computer Science*. 2018;126:675–684. doi: 10.1016/j.procs.2018.08.001
29. SD3 - Simulation Delivery and Documentation Deviations. Available from: <http://aber-owl.net/ontology/SD3> [Accessed 13 March 2024].

Shilov Nikolay G. PhD. St. Petersburg Federal Research Center of the Russian Academy of Sciences, St. Petersburg, Russia.
E-mail: nick@iias.spb.su