

<sup>1</sup>Гладун А.Я., к.т.н.  
<sup>2</sup>Андрюшевич А. А.  
<sup>3</sup>Курбацкий А.Н., д.т.н.

## ОНТОЛОГИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ИНФОРМАЦИОННЫХ ОБЪЕКТОВ, МОДЕЛЕЙ И СЕРВИСОВ В WEB OF THINGS

<sup>1</sup>Международный научно-учебный центр  
информационных технологий и систем НАН Украины и МОН Украины  
[glanat@yahoo.com](mailto:glanat@yahoo.com)

<sup>2</sup> ЗАО "Институт информационных, инновационных, инвестиционных технологий"  
[andrushevich@bsu.by](mailto:andrushevich@bsu.by)

<sup>3</sup>Белорусский государственный университет,  
[andrushevich@bsu.by](mailto:andrushevich@bsu.by)

*Создание и развитие инновационных информационных технологий Интернета вещей (Internet of Things, IoT) повлекло за собой развитие и ряда новых направлений в формировании информационного пространства, в частности – Web of Things (WoT). В связи с этим возникла необходимость в разработке новых семантических, ориентированных на использование знаний, подходов для идентификации, поиска и исследования, специфичных для этого пространства информационных объектов.*

*В данной работе мы представляем набор онтологических моделей - объектов, ресурсов и сервисов, а также использование элементов онтологического анализа при создании моделей и сервисов поиска на основе онтологий в информационном пространстве Web of Things, базирующемся соответственно на платформе Internet of Things. Предлагается также подход к аннотированию описаний WoT-объектов при помощи онтологических моделей и семантических Web-сервисов*

**Ключевые слова:** Web of Things, Internet of Things, семантический Web-сервис, онтология, семантическое аннотирование

### **Введение**

Мы находимся на пороге новой технической революции - развития Интернета вещей (*Internet of Things, IoT*), который предполагает подключение к Интернету множества окружающих человека вещей, снабженных определенными электронными датчиками [1, 2] и управлять этими вещами на расстоянии.

*IoT* не ограничивается связью с вещами, снабженными метками *RFID (Radio Frequency IDentification)*, радиочастотная идентификация), а рассматривается в контексте объединения таких современных концепций, как всепроникающие компьютерные системы (*Pervasive Computing, Ubiquitous Computing*), и интеллектуальная окружающая среда (*Ambient Intelligence*) [3]. Таким образом, конвергенция создает условия для развития но-

вых технологий, включающих в себя помимо традиционного Интернета людей (*Internet of People, IoP*) еще и Интернет медиаконтента (*Internet of Media, IoM*), Интернет сервисов (*Internet of Services, IoS*), Интернет вещей (*Internet of Things, IoT*) и Веб вещей (*Web of Things, WoT*).

В каждой из перечисленных технологий может использоваться такое понятие как «вещь». Вещь может быть материальной и нематериальной, одушевленной и неодушевленной, а в контексте *IoT* вещь можно считать нечто реальное или виртуальное, но обязательно так или иначе связанное с цифровым миром посредством беспроводных коммуникаций, и это нечто должно обладать способностью, быть так или иначе идентифицированным во времени и пространстве [4].

У каждой из вещей реального физического мира в *IoT* предполагается циф-

ровой двойник, ее виртуальное представление. Эти цифровые аналоги физических объектов могут воспринимать информацию из окружающего мира, вступать во взаимодействие, обмениваться данными и т.д.

В результате такого взаимодействия формируется новая среда, где интеллект, заложенный в приложения, позволит оценивать происходящее в физическом мире, учитывать накопленные ранее сведения и опыт для поддержки принятия решений в системе.

Однако *IoT* гетерогенная среда, и она должна развиваться в направлении создания более структурированного набора решений, где «вещи» должны быть представлены некоторым унифицированным способом и одинаково поддаваться обнаружению, иметь возможность общаться с другими объектами, а также быть непосредственно интегрированными с инфраструктурой Интернета и его сервисами, независимо от специфического способа их подключения. *IoT* нуждается в интероперабельности на множественных уровнях [5].

Последние тенденции в области *IoT* направлены на развитие методов и способов интегрирования «вещей» с существующей инфраструктурой *Web* и к представлению подключенных «вещей» однородным способом, как ресурсов *Web*, что в результате вылилось в новое направление *Web* вещей (*Web of Things, WoT*).

В настоящее время наблюдается усложнение структуры и способов получения информации, доступной пользователям через *Web*. Все большая часть информации поступает не от людей, а от различных физических устройств, и в дальнейшем отправляется также различным устройствам, а не непосредственно пользователям.

Поэтому сейчас можно говорить о новом поколении *Web*, имеющем свою специфику, базирующемся на новых технологиях (в частности, связанных с *IoT*) и требующем развития соответствующих способов и методов обработки.

### Эволюция *Web*

Если технологической основой традиционного *Web* стали Интернет-технологии, то базисом развития *Web of Things* является *Internet of Things (IoT)*. *WoT* – это некоторая надстройка над *IoT*, представляющая информацию, передаваемую и получаемую через *Web*-среду от элементов, составляющих *IoT*.

Объект в *WoT* – это встроенное устройство, соединенное с физическим объектом (датчик + объект). При этом сам физический объект не является датчиком или вычислительным устройством.

Например, в фонарный столб (физическое устройство) может быть встроен микроконтроллер с датчиками для управления освещенностью и мобильный телефон для связи. Другой пример – помещение в доме, оборудованное датчиками температуры и средствами для связи с пользователем для получения указаний о желаемой температуре.

Таким образом, объект в *WoT* – это совокупность датчиков и встроенных устройств, соединенных с тем или иным физическим предметом, предназначенным для отличных от вычислений функций и воспринимаемые с ним, как единое целое.

Технология *Web вещей (Web of Things, WoT)* возникла с начала 2000-ых годов, как попытка создания *Web*-ориентированной сервисной платформы для «умных» вещей, т.е. предметов, снабженных различными датчиками или электронными устройствами для приема/передачи данных и сигналов управления, связь между которыми поддерживается через *IoT*.

*Web of Things* можно рассматривать как новый этап развития *Web*, интегрирующий результаты его предыдущих поколений, в частности, *Semantic Web* и социального *Web*.

Если технологической основой традиционного *Web* являются Интернет-технологии, то базисом развития *Web of Things* является *Интернет вещей (Internet of Things – IoT)*. *WoT* – это информация, передаваемая и получаемая через *Web*-

среду от элементов, составляющих *IoT*.

Особенностями архитектуры современного *IoT* является наличие чрезвычайно большого количества разнообразных устройств и ресурсов, которые обмениваются информацией между собой и между прикладными программами, сообщают о состоянии ресурсов, обеспечивают наблюдение и измерение разных физических величин и т.п.

Пока трудно оценить перспективы развития *WoT*. Вероятно, некоторые приложения, такие как мобильные платежи и инициатива *IBM, Smarter Planet*, станут широко распространенным всего через несколько лет. Маркетинг также будет преобразован, так как потребители получат беспрепятственный доступ к цифровым продуктам от рекламы в физическом мире. Тем не менее, очевидно, что объем вычислений, связанных с обработкой данных от *WoT*, увеличится очень быстро и потребует технологий хранения и анализа потоковых данных.

При этом перспективным направлением представляется трансформация получаемых сведений в значительно более компактные знания и последующее использование именно таких «обработанных» знаний для решения прикладных задач.

*WoT* развивается в неразрывной связи с *IoT* и для повышения эффективности требует применения семантических, ориентированных на знания методов онтологического распознавания, выбора, поиска и управления объектами и сервисами доставки, хранения и обработки данных. В данной работе предлагается использование разработанной онтологии «умных» вещей для задач семантической интероперабельности (методы выравнивания онтологий), выбора сервисов и объектов в распределенном и гетерогенном пространстве *Web of Things*.

Отсутствие четкого и формального представления знаний в *WoT* может привести к двусмысленности в терминологии, а также препятствовать совместимости и в основном семантической совмести-

сти.

Особенностями архитектуры современного Интернета вещей является наличие чрезвычайно большого количества разнообразных устройств и ресурсов, которые обмениваются информацией между собой и между прикладными программами, сообщают о состоянии ресурсов, обеспечивают наблюдение и измерение разных физических величин и т.п.

Распределенная природа и неоднородный характер *IoT* делает взаимодействие между "вещами" сложной задачей и требует методов, которые могут облегчить автоматизированную машинную обработку.

Идея *IoT* начала успешно развиваться благодаря развитию следующего набора информационных технологий: широко распространения беспроводных сетей (*Wi-Fi, 3G, LTE, WiMAX*) и датчиков; активного перехода Интернет на *IPv6*; роста популярности облачных вычислений и появления технологий межмашинного взаимодействия (*Machine to Machine, M2M*) [6].

С помощью *M2M* осуществляется доступ к удалённым объектам для сбора данных и мониторинга состояния. *M2M* работает с протоколами *TCP/IP*, беспроводными сетями стандарта *IEEE 802.11*, технологиями сотовой связи и проводными сетями, такими как *Ethernet*. Эта технология незаменима в тех случаях, когда удалённое оборудование используется в труднодоступных местах или когда использование проводного соединения невозможно в принципе.

Набор технологий, разработанных в *Semantic Web*, таких как онтологии, семантические аннотации, связанные данные и семантические *Web*-сервисы могут быть использованы в качестве принципиальных решений в целях реализации *WoT*.

### **Постановка задачи**

Сегодня *Web*-сервисы представляют собой значительную часть инфраструктуры информационного пространства *Web*. Для того, чтобы обнаружить и использовать требуемые объекты *WoT* необходимо

использовать подходы и методы, разработанные для исследования семантических *Web*-сервисов. Для реализации этой цели предлагается рассматривать каждый *WoT*-объект, как семантический *Web*-сервис, функции которого соответствуют назначению физической составляющей *WoT*-объекта и связанных с ним датчиков и устройств.

Для семантического аннотирования *WoT*-объектов предлагается использовать как онтологии, специально разработанные для описания объектов *WoT*, *IoT* и *Web*-сервисов, так и произвольные онтологии предметных областей. При этом проблема исследования *WoT* сводится к задаче сопоставления онтологий, для которой на сегодня уже разработано некоторое количество алгоритмов и методов.

### **Информационная модель *Web of Things***

Если проследить историю зарождения *WoT*, то можно отметить следующее. В 2002 году при разработке проекта по «умному городу» [7] было предложено связать физические объекты с *Web*-страницами, содержащими информацию о городе, и сервисами для пользователей.

Для этого были использованы интерфейсы или штриховые коды на объектах, позволяющие пользователям при взаимодействии с физическими объектами легко находить *URI* страниц, связанных с этими объектами.

Другая попытка использовать *Web* для связи с объектами реального мира заключалась в том, чтобы объединить «умные вещи» со стандартизированной архитектурой *Web*-сервиса, использующей такие традиционные стандарты, как *SOAP*, *WSDL*, *UDDI* [8]. Но на практике это оказалось слишком сложно и неоправданно для простых объектов.

Поэтому вместо *SOAP* и *WSDL*, используемых в *Web*-сервисах, была предложена технология "*Web* вещей" на основе вложенного протокола передачи гипертекста (*HTTP*) для серверов и *Web 2.0* [9].

Необходимые для этого *Web*-серверы с продвинутыми функциями (такими как

альтернативные соединения или назначение сервера для уведомления о событиях) могут быть реализованы всего с 8 КБ памяти и без поддержки операционной системы. Кроме того, благодаря эффективной оптимизации стека протоколов *TCP/HTTP*, эти *Web*-серверы могут быть установлены на таких миниатюрных встроенных системах, как смарт-карты и др.

Так как внедренные *Web*-серверы, которые используются в Интернете вещей, как правило, имеют в наличии значительно меньше ресурсов по сравнению с такими клиентами *Web*, как браузеры или мобильные телефоны, то асинхронный *JavaScript* и *XML (AJAX)* оказался хорошим способом передавать часть рабочей нагрузки от сервера к клиенту.

Рассмотрим главные абстракции и понятия, которые лежат в основе домена *IoT* и описывают отношения между ними.

Основной принцип *IoT* и *WoT* - расширение Интернета в область физического мира, чтобы вовлечь взаимодействие физических объектов из окружающей среды. Объекты являются «вещами» в Интернете вещей и могут быть как человеком, животным, так и автомобилем, электронным прибором, закрытой или открытой средой. «Объект» - главный центр взаимодействий между людьми и/или программными агентами. Это взаимодействие дает возможность посредством аппаратных компонент или «устройств», которые или подключаются к объекту или является частью среды объекта, управлять объектом и контролировать его.

Устройство позволяет объекту быть частью цифрового мира, посредством взаимодействия. Фактически программный компонент предоставляет информацию об объекте или позволяет управлять устройством, «ресурсом».

Сервисы представляют функциональные возможности устройства, подключаясь к его управляемым ресурсам. При этом «сервис» обеспечивает четкий и стандартизированный интерфейс для

пользователя, предлагая все необходимые функциональные возможности для того, чтобы взаимодействовать с объектами и связанными процессами.

Другие сервисы могут вызвать сервисы нижнего уровня и компоновать их для обеспечения функциональных возможностей на более высоких уровнях, например, выполнить поддержку работы указанного бизнес-процесса. Отношения между сервисами и объектами должны быть смоделированы как ассоциации. Эти ассоциации могли быть статичными, например, в случае, если устройство внедрено в объект; и динамичными, например, если устройство в среде контролирует мобильный объект [10].

На рис. 1 представлены эти идентифицированные понятия домена *WoT* и отношения между ними. Идентифицированные понятия должны быть смоделированы в формате, который предоставляет интерактивное и автоматизированное представление, поддающееся толкованию, как человеку, так и машине.

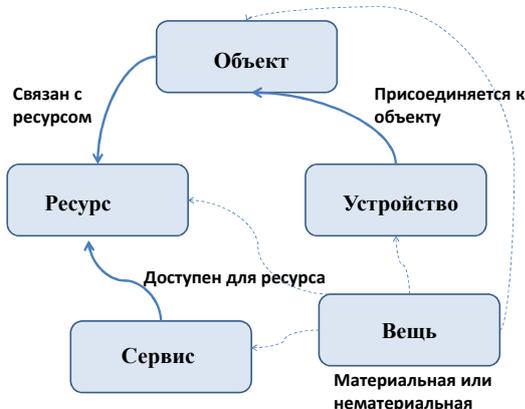


Рис. 1. *WoT* модель: ключевые концепты и взаимодействия

Концепция *Semantic Web* вводит формальные определения, называемые онтологиями, что позволяет строить модели гетерогенных объектов в домене, совместно использовать знания и поддерживать автоматизацию формулирования логических выводов из этих знаний. Специфицирован язык *Web* онтологии – дескриптивная логика (*OWL-DL*), внедренная в модели логика первого порядка, что обеспечивает платформу для фор-

мального и машинно-обрабатываемого знания.

### Физические компоненты *WoT*

Основная цель *WoT* заключается в том, чтобы обеспечить возможность людям общаться с объектами и физическими объектами, которые находятся вокруг них. Эти объекты могут также взаимодействовать друг с другом, а затем отправлять людям информацию на их смартфоны. Взаимодействие людей с физическими объектами осуществляется с помощью датчиков, которые могут получать информацию, передавать информацию и выполнять действия, требуемые человеком или компьютерами и объектами. Если поместить датчик в районе жилого дома или прикрепить его к физическому объекту или вещи, то он сможет отправлять информацию на телефон пользователя или на любой другой объект или объект.

Объектом, в данном случае, может быть любая вещь (материальная или нематериальная) в доме, в офисе (например, дверь, светильник, окна, ключи от квартиры), устройство (кофеварка, стиральная машина, автомобиль) или же абстрактный объект (человек, животное). Если поместить датчик на любой такой «объект» и иметь возможность общаться с ним, то он становится «умной» вещью (*smart thing*).

На рис. 2 представлена обобщенная схема взаимодействия пользователя с объектами через смартфон посредством *Web*-интерфейса.



Рис. 2. Объединение физических объектов через *WoT*

Если поставить датчик на объект, то он может отправлять сообщение, содержащее информацию о его состоянии, местоположении, включен он или выключен, открыт он или закрыт и т.д. Пользователь может использовать смартфон для отправки различных команд к своим объектам и указывать им, что делать. Например, при помощи датчиков можно узнать, когда ребенок возвратится домой из школы и выключен ли свет в гостиной, включить электроплиту, заблокировать или разблокировать входную дверь.

Важно, что при этом объекты могут взаимодействовать друг с другом: например, когда ребенок приходит со школы домой, то датчик на его рюкзаке автоматически посылает сообщение на смартфон об этом, а тот посылает сообщение о разблокировке замка на входной двери.

На рис. 3 представлена иерархическая структура электронных устройств, используемых пользователями для подключения, мониторинга и управления окружающими объектами.



Рис. 3. Иерархическая структура устройств, используемых для подключения объектов в IoT и WoT

Чтобы сообщество «умных» устройств или объектов было жизнеспособным, следует наделить каждую объект определенным интеллектом, помогающим ей распознать контекст своего окружения (местоположение, собственное состояние и положение контролируемого объекта), а также наладить диалог с окружающими объектами и управляющим устройством [11].

Данные, собранные разными датчиками и устройствами по обыкновению являются многомодульными (температура, индикатор, звук, видео и т.п.). Разнообразие, изменчивость и разнотипность данных реального мира *WoT* вызывает

многие проблемы с их обработкой, интегрированием и интерпретацией. Поэтому динамическая и ограниченная ресурсами природа *WoT* требует специальной архитектуры представления знаний и обработки, которая должна быть учтена технологиями семантического *Web*, такими как онтологический анализ, семантические метаданные, связанные данные (*linked data*), семантические *Web*-сервисы и социальные сети.

### **Аспекты реализации Web of Things**

Одним из первых прототипов *Web of Things* был проект, в котором датчики использовались для мониторинга и контроля

за потреблением электроэнергии бытовых приборов [12], чтобы на домашнем компьютере пользователь мог визуально контролировать потребление электроэнергии.

Реализация функций человеко-машинного интерфейса при этом осуществляется на основе интерфейса прикладного программирования *RESTful API* (*Representational State Transfer* - «Передача Репрезентативного Состояния») [13].

*Web*-сервис *REST* реализует метод взаимодействия компонентов распределённого приложения в *IoT*, при котором вызов удаленной процедуры представляет собой обычный *HTTP*-запрос (обычно *GET* или *POST*; такой запрос называют *REST*-запрос), а необходимые данные передаются в качестве параметров запроса [Pautasso, 2008]. Этот способ является альтернативой более сложным методам, таким как *SOAP*, *CORBA* и *RPC*.

Другими словами *REST* означает концепцию построения распределённого приложения, при которой компоненты взаимодействуют наподобие взаимодействия клиентов и серверов в *Web*. Системы, поддерживающие *REST*, называются *RESTful*-системами.

Концепцию построения распределённого приложения на основе *REST* предполагает, что каждый запрос (*REST*-запрос) клиента к серверу содержит в себе исчерпывающую информацию о желаемом ответе сервера (желаемом репрезентативном состоянии), и сервер не обязан сохранять информацию о состоянии клиента («клиентской сессии»).

Можно выделить ряд важных преимуществ *Web*-сервисов *REST* (сбор, доставка и передача данных от/к умных объектов в *WoT*) в отличие от традиционных *SOAP Web*-сервисов:

- надёжность (за счет отсутствия необходимости сохранять информацию о состоянии клиента, которая может быть утеряна);

- производительность (за счет использования специального кэша);

- масштабируемость; прозрачность системы взаимодействия, особенно необ-

ходимая для приложений по обслуживанию сети;

- простота интерфейсов; портативность компонентов; легкость внесения изменений;

- способность эволюционировать, приспособиваясь к новым требованиям (на примере *Web*).

Таким образом, в архитектура *REST* очень проста в плане использования. По виду пришедшего запроса сразу можно определить, что он делает, не разбираясь в форматах (в отличие от *SOAP*, *XML-RPC*). Данные передаются без применения дополнительных слоев, поэтому *REST* считается менее ресурсоемким, поскольку не надо просить (автоматически обрабатывать или разбирать с целью получения нужных данных) запрос чтоб понять, что он должен сделать и не надо переводить данные из одного формата в другой.

Но важным моментом здесь остается информационная безопасность – недопустимо кому угодно позволять изменять информацию, то есть нужна еще авторизация и аутентификация. Эта проблема решается различными методами, например, при помощи различного типа сессий или просто *HTTP Authentication*.

### 1. Концептуальная модель архитектуры *Web of Things*

*WoT* является составной частью *IoT*: он предоставляет возможность мониторинга и управления объектами с помощью страниц *WWW*. На рис. 4 приведена структура организации *WoT*, из которой видно, что ключевую роль в нем играют два уровня: интеллектуальный уровень и уровень приложений. Это позволяет использовать через *Web* как изначально приспособленные к этому объекты, так и те объекты, для которых необходимы соответствующие согласующие устройства (шлюзы), например, объекты, функционирующие по протоколам *ZigBee* или *Bluetooth*.

На рис.4 представлена пятиуровневая модель концептуальной архитектуры платформы [14] *WoT* для взаимодействия человека с «умными» вещами. Уровень

интеллектуальных приложений и умных вещей содержит базирующиеся на семантических технологиях (*Semantic Web*) программы, реализующие функции, поиска, сопоставления онтологий, логического вывода.

Для того, чтобы эффективно использовать преимущества, которые предоставляет сообщество "умных" вещей, нужно наделить каждую объект, входящую в это сообщество, определенным интеллектом (знаниями и правилами их использования), который должен помочь ей распознать контекст своего окружения (местонахождение, собственное состояние и положения объекта, контролирующего эту объект), а также наладить диалог с окружающими объектами и управляющим устройством [15].



Рис. 4. Архитектура *Web of Things* как части *Internet of Things*

Онтологический подход становится ядром, которое обеспечивает семантическое описание (виртуальную модель знаний) об этих объектах. При этом онтологии могут многократно использоваться разными приложениями.

Кроме того, использование онтологий для объектов *WoT* обеспечивает всю необходимую семантику для спецификации различных устройств, используемых в технологиях *IoT/WoT*, равно как и для спецификаций задач, решаемых непосредственно в приложениях *WoT* (ввод, вывод, логика управления), которые со-

здаются на основе использования информации об этих устройствах.

Для создания целостной системы развертывания приложений *WoT* в определенной сфере деятельности необходима разработка формальной онтологической модели объектов (вещей), входящих в состав *WoT*.

Архитектура модели объектов *WoT* должна включать:

1. Анализ структуры знаний предметной области *WoT*, основных объектов и отношений:

- использование мереологических (*mereological*) и онтологических методов для формирования терминологии домена *WoT* и структуры знаний;

- структура *OWL*-онтологии.

2. Многократное использование существующих тезаурусов, таксономий и онтологий домена *WoT*:

- семантический поиск релевантных объектов и анализ средств представления знаний и стандартов в *WoT*;

- краткий обзор соответствующих онтологий и других структур знаний;

- интеграция существующих таксономий и онтологий домена.

3. Архитектура методов для автоматизированного усовершенствования формальной онтологической модели *WoT*:

- архитектура методов для автоматизированной добычи знаний (термины и отношения) из текстов естественного языка, которые касаются домена *WoT*;

- методы автоматизированной лингвистической обработки текстов естественного языка;

- усовершенствованная *OWL*-онтология.

4. Семантический поиск в домене *WoT* на основе онтологии домена:

- семантический поиск объектов *WoT*;

- методы семантического поиска объектов *WoT*;

- методы семантического поиска сервисов *WoT (RESTfull)*;

– рекомендации относительно использования *WoT* объектов.

Данные датчиков, которые связаны с разными событиями и ситуациями, могут быть проанализированы и преобразованы в активное знание, которое позволяет нам лучше понимать физический мир и создавать больше продуктов и сервисов с добавленной стоимостью. Например, учет расстояний линий связи в *Smart GRID* может быть использован для того, чтобы лучше предусмотреть и балансировать потребляемой мощностью; анализируя комбинацию транспортных потоков, загрязнение, состояние погоды и перегрузки, можно создать знание на основе данных от датчиков, позволяя при этом обеспечить лучший трафик транспорта и муниципальное управление; контроль и обработка данных от датчиков и объектов, прикрепленных пациентам или людям преклонного возраста, могут обеспечить лучшее медицинское обслуживание пациентов на расстоянии.

### Архитектура *WoT*

Основная цель *WoT* – обеспечить наиболее эффективное принятие решений за счет более глубокого анализа наблюдения за окружающей средой.

Чтобы достигнуть этого, необходимо ввести несколько информационных уровней между датчиками (объектами) и блоком принятия решений и приложением (рис.5).

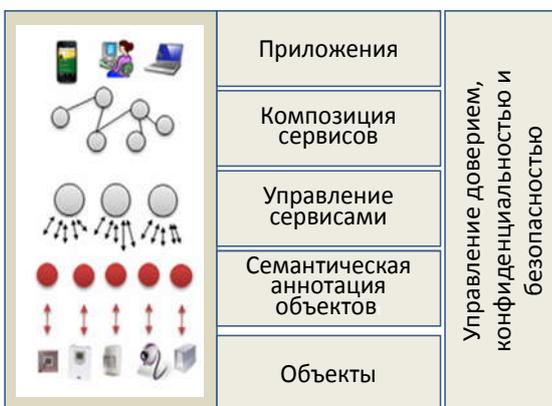


Рис. 5. Концептуальная архитектура *WoT*

При этом возникает проблема кон-

цептуализации домена датчиков (сенсоров). Ее решение связано с использованием онтологий и требует исследования семантической сети сенсоров.

Некоторые из существующих онтологий покрывают домен сенсоров, но большинство из них – только его часть. Онтология *W3C Semantic Sensor Network (SSN)* представляет собой попытку покрыть весь этот домен полностью.

Кроме того, необходимые знания могут извлекаться из различных онтологий предметных областей, доступ к которым обеспечивают различные репозитории онтологий [16].

Но, несмотря на наличие онтологий, извлечение полезных для пользователей сведений из потоков данных передаваемых датчиками остается сложной задачей.

Выделим несколько отдельных проблем в этой сложной задаче:

- поиск *WoT*-объектов, обладающих необходимыми пользователю возможностями;
- исследование свойств *WoT*-объектов, обладающих необходимыми пользователю возможностями;
- исследование потоков данных, передаваемых сенсорами и встроенными устройствами *WoT*-объектов, обладающих необходимыми пользователю возможностями;
- анализ свойств передаваемых потоков данных.

### Средства вывода над онтологиями

Анализ современных методов и средств обработки (добычи) знаний и языков, для формирования запроса к онтологии позволил разработать подсистему интеллектуальных приложений на четвертом уровне модели *WoT*. Для добычи и обработки знаний, которые хранятся в онтологиях, используются блоки логического вывода (ризонеры, *reasoners*), которые играют ключевую роль в интеллектуальных системах, ориентированных на знание.

Существует множество реализаций процессоров логического вывода

(*reasoning engine*) для *OWL* онтологий, которые различаются по возможностям, областям применения и качества выполнения задач. Обобщенный анализ позволяет разделить их на три группы в зависимости от метода реализации:

1. Табличные *DL*-процессоры. Традиционно были разработаны первыми для решения подобных задач. Имеют низкую производительность, но способны делать умозаключения на сложных онтологиях с множеством нетривиальных конструкций. До этого класса относятся резонеры *Pellet*, *RacerPro*, *Fact++*, а также *Hermit* и *SHER*.

2. Дизъюнктивные *Datalog*-процессоры. Трансформируют онтологию в дизъюнктивную *Datalog* программу и используют методику дедуктивных баз данных и правило резолюций. Такие процессоры имеют удовлетворительное быстродействие при использовании некоторых оптимизаций, однако не поддерживают определенные *OWL*-конструкции, в частности кардинальные ограничения и номиналы. К этой группе принадлежит *KAON2*.

3. Процессоры правил. Используют системы обработки правил для умозаключений на онтологиях. Имеют высокую производительность, но могут обрабатывать лишь простые онтологии, лишённые многих важных конструкций. Представители этой группы: *Sesame/OWLIM*, *Jena*, *Owljesskb*.

Существуют два подхода к реализации логического вывода: на базе правил (с использованием алгоритмов *forward-chaining* и/или *backward-chaining*) и на базе семантического табло (*semantic tableau*). На базе правил реализованы *Semantics*, *SDK* и *Owlim*, а на базе семантического табло - *Pellet*.

В качестве системы логического вывода для дескриптивной логики нами была избрана свободно распространяемая система *Pellet* версии 2.0.0. Система *Pellet* реализует логический вывод для дескриптивной логики класса *ALCQHI (D)*, которая расширяет атрибутивный язык (*AL*)

такими возможностями, как произвольное выражение отрицания, транзитивные отношения, инверсные отношения, иерархия отношений, количественные ограничения на отношения и некоторые конкретные домены.

На сегодняшний день система *Pellet* реализует наиболее выразительную дескриптивную логику с использованием высокопроизводительного алгоритма (*tableau-based algorithm*) логического вывода, который используется для обработки онтологий, описанных языком *OWL DL*.

*OWL* позволяет представлять знания о предметной области в виде онтологий, которые можно использовать и обрабатывать в разных приложениях; язык *OWL* запросов *SPARQL* позволяет создавать метаописания *RDF* и онтологий.

Разработанная *WoT*-онтология имеет цель поддержки процесса автоматизированного развертывания интеллектуальных приложений в гетерогенных средах *WoT*. *WoT*-онтология используется для поддержки семантического реестра *WoT*-объектов и позволяет:

а) скрыть технологическую разнородность, которая характерна множеству гетерогенных объектов *WoT*;

б) скрыть семантическую разнородность, присущую используемым гетерогенным онтологиям домена для того, чтобы семантически аннотировать данные умных объектов *WoT*.

Рассмотрим более детально первый этап этой задачи – поиск в пространстве *WoT* устройств, необходимых пользователю. При этом пользователь может специфицировать как функции или свойства самого физического объекта, связанного с соответствующей *WoT*-объектом, так и характеристики связанных с ним сенсоров и получаемых от них сведений либо предоставляемых ими сервисов.

При этом пользователь может изменять термины из каких-либо доступных ему онтологий, описывающих пространство *WoT* и соответствующую предметную область (со ссылкой на сами онтологии). Но в описании *IoT*-объекты могут

использоваться термины из других онтологий, относящихся к близким доменам.

Поэтому возникает проблема сопоставления онтологий. В общем случае эта задача крайне сложна и трудоемка, но при наличии ряда ограничений на сопоставляемые онтологии ее можно решить за приемлемое время.

### **WoT-объект как семантический Web-сервис**

Web-сервисы являются наиболее современной попыткой реконструировать крупномасштабные распределенные вычисления. Они базируются на стандартах, которые действуют на синтаксическом уровне и не имеют возможностей для представления семантики. Семантика обеспечивает более качественные и масштабируемые решения для таких областей, как интероперабельность сервисов, обнаружение и компоновка сервисов, и оркестровка процессов [17].

Мы предлагаем рассматривать WoT-объект как семантический Web-сервис. При этом основное внимание уделяется не использованию стандартов Web-сервиса (что слишком сложно для WoT-объектов), а принципам создания и обработки семантической разметки на основе онтологий и поиска в информационном пространстве этих описаний [18].

Web-сервисы представляются перспективным базисом, обеспечивающим решение для интероперабельности в разнородных средах. Они базируются на расширяемом языке разметки (XML), который представляет собой базовую технологию Web-сервисов. Тем не менее, описываются Web-сервисы (как правило, синтаксически) по таким стандартам, как UDDI, SOAP и WSDL.

В предлагаемом подходе данные используются для того, чтобы найти соответствие (*matching*) между терминами, используемыми в запросе, касающемся поиска WoT, и терминами аннотации соответствующего WoT-объекта, представленной его поставщиком. Этот подход основан на архитектуре представления WoT-объектов как Web-сервисов путем добав-

ления семантики в их описание, а также в запрос клиента [19].

Предлагаемый подход к исследованию WoT основан на процессе аннотации, которая состоит в двух этапов:

- этапа категоризации, который позволяет классифицировать WoT-объект в соответствующий им домен;

- этапа сопоставления, который позволяет связать каждую сущность из WoT с соответствующей сущностью в онтологии домена.

Процесс аннотирования WoT-объектов опирается на методы сопоставления онтологий, которые, в свою очередь, используют некоторые меры подобия (*similarity measures*).

И категоризация, и сопоставление используют методы сопоставления онтологий – как специфичных для WoT, так и связанных с предметной областью. Например, при описании подключенной через IoT кофеварки может использоваться онтология бытовой техники, онтология WoT-датчиков и персональная онтология пользователя, описывающая, какие напитки он предпочитает в разное время суток.

Категоризация в данном случае – структурирование схожих онтологий, представляющих WoT-объекты по темам, формирование обобщающих, классифицирующих категорий, реализуемое в целях установления соотношений между объектами.

Цель сопоставления всех этих онтологий – найти отношения между сущностями, выраженными в различных онтологиях, и определить, насколько они семантически близки. Очень часто эти отношения являются отношениями эквивалентности, которые обнаруживаются через меру сходства между сущностями онтологий.

Задача алгоритма сопоставления заключается в использовании знаний о реальном мире для нахождения степени соответствия между семантическими аннотациями и запросами, а также в нахождении тех аннотаций сервисов, которые наиболее близки к запросу. Профили сер-

висов предназначенны для описания функциональностей, которые *Web*-сервис хочет предоставлять всему сообществу. *Web*-сервисы могут иметь много функциональностей, но при этом не все они могут быть представлены. Профили сервисов описывают как запросы сервисов, так и предоставляемые сервисы. В частности, в [20] предлагается реализовать гибкое сопоставление (потому что трудно ожидать их точного совпадения) описаний сервисов и запросов на основе операции дифференцирования над описаниями сервисов.

Именно сопоставление онтологий является основой для поиска в пространстве объектов, семантически размеченных терминами различных онтологий.

Для завершения процесса сопоставления онтологий соответствия используются меры сходства (*similarity measure*) между объектами. Мера сходства направлена на количественную оценку того, насколько две сущности подобны.

Для описания семантики *Web*-сервисов используются относительно небольшие и простые по структуре онтологии, терминами которой являются слова одного и того же естественного языка (если описание *Web*-сервиса дается на языке, не знакомом пользователю, то он не сможет эффективно применить его для своих целей). Предлагается следующий алгоритм нахождения соответствия между запросом пользователя и описанием *Web*-сервиса. На вход алгоритма поступают формализованные описания онтологий (на языке *OWL*) – множества онтологий *Web*-сервисов  $O_{si}$ , онтологии запроса пользователя  $O_q$  и онтологии предметной области (ПрО), для которой  $O_{si}$  и  $O_q$  являются расширениями. Следует отметить, что в большинстве случаев  $O_q$  совпадает с онтологией ПрО, т.е. пользователь просто выбирает одну из онтологий ПрО, на которые ссылаются онтологии доступных *Web* сервисов, и только в некоторых случаях может формировать собственную онтологию (как правило, с помощью объединения нескольких онтологий или

упрощения одной из предложенных – добавлять новые термины нецелесообразно, так как они не используются в описании семантики *Web*-сервисов, данных их разработчиками. Необходимо найти такой сервис  $O_{sm}$ , для которого  $\text{Podob}(O_q, O_{sm}) = \max_{i=1..n} \text{Podob}(O_q, O_{s_i})$  – коэффициент подобия с запросом пользователя будет максимальным. Вычисление такого коэффициента для каждого из доступных *Web*-сервисов состоит из следующих этапов:

1. Построение пересечения множеств терминов онтологий *Web*-сервиса и запроса  $T(O) = T(O_{s_i}) \cap T(O_q)$ .

2. Если это пересечение не пусто, то для каждого термина из  $T(O)$  строятся два множества  $T_{si}$  и  $T_q$  – термины, которые связаны с ним в каждой онтологии различными отношениями.

3. Для каждого термина из  $T(O)$  строится пересечение множеств  $T_{si}$  и  $T_q$ .

4. Анализ типов отношений между терминами из  $T(O)$  и пересечения множеств  $T_{si}$  и  $T_q$  (все отношения онтологии делятся на три типа – иерархические, синонимические и прочие). Для этого используется два словаря отношений –  $R_{ierarh}$  и  $R_{sinonim}$ .

5. После этого строится коэффициент сходства онтологий, который является количественным отображением сходства семантики двух онтологий. При этом учитываются следующие факторы: вхождение одного и того же термина в обе онтологии; то, что два термина находятся в разных онтологиях в одном и том же отношении; то, что два термина находятся в разных онтологиях в отношениях одного типа или разных.

6. Строится коэффициент подобия запроса и *Web*-сервиса – аналогично п.5, но учитываются только те термины из  $T(O) = T(O_{s_i}) \cap T(O_q)$ , на которые ссылаются имена параметров *Web*-сервиса. Если полученный коэффициент выше определенной пользователем константы – коэффициента доверия, то считается, что *Web*-сервис удовлетворяет потребностям поль-

зователя и может использоваться при компоновке составного *Web*-сервиса.

Таким образом, когда доступен набор онтологий, сходство между двумя наборами вычисляется путем сравнения набора сущностей описания *WoT*-объекты и набора сущностей каждой онтологии.

Для логического вывода на основе *DL* используют отношения категоризации (*subsumption*) и выполнимости (*satisfiability*) для поиска соответствия потенциальных соответствий.

### **Выводы**

Главным преимуществом представления семантических моделей для описания компонентов *WoT* является обеспечение интероперабельности данных об этих объектах и сервисах.

Современный этап *Web*-технологий и, в частности, перспективы развития *WoT*, требуют формирования соответствующих методов и технологий для их эффективного использования. Представляется целесообразным применять для этого уже существующие и доказавшие свою полезность разработки, такие как *Semantic Web* и семантические *Web*-сервисы, модифицируя их в соответствии со спецификой использования объектов реального мира, связанных через *IoT*.

Разработанные алгоритмы и модели семантического распознавания информационных объектов на основе онтологического представления знаний об этих объектах для дальнейшего их использования в интеллектуальных информационных технологиях и приложениях требуют дальнейшего развития и усовершенствования с учетом гетерогенности среды *Web of Things*.

### **Список литературы:**

1. Rob van Kranenburg The Internet of Things. A critique of ambient technology and the all-seeing network of RFID.- Waag Society Publishing, Amsterdam University, Holland. – 182 p.

2. Atzori L., Iera A., Morabito G. The Internet of Things: A survey// Computer Networks 54 (2010). – P. 2787–2805.

3. ITU Internet Reports, The Internet of Things, November 2012.

4. Гриценко В.И., Гладун А.Я., Рогущина Ю.В. Семантическое распознавание информационных объектов на основе онтологического представления знаний о предметной области в задачах интеллектуального управления // Кибернетика и вычислительная техника, 2014, №4, вып. 178. – С. 5-22.

5. Gladun A., Rogushina J., Andrushevich A., Kurbatski A. Use of the ontological approach to semantic search in environments of the Internet of Things; February 20-22, 2014, IV International Scientific and Technical Conference Open Semantic Technologies for Intelligent Systems, OSTIS 2014, Minsk, Belarus. – P. 70-75.

6. Machine-to-machine (M2M) – the rise of machine// White Paper, Juniper, 2011.

7. Kindberg T, Barton J, Morgan J, Becker G, Caswell D, Debaty P, Gopal G, Frid M, Krishnan V, Morris H, Schettino J, Serra B, Spasojevic M. (2002) People, places, things: web presence for the real world. *Mob Netw Appl* 7:365-376.

8. Guinard D, Fischer M, Trifa V (2010a) Sharing Using Social Networks in a Composable Web of Things. Proceedings of the 1st IEEE International Workshop on the Web of Things (WoT 2010) at IEEE PerCom, Mannheim, Germany.

9. Luckenbach T, Gober P, Arbanowski S, Kotsopoulos A, Kim K (2005) TinyREST – A protocol for integrating sensor networks into the internet. Proceedings of the Workshop on Real-World Wireless Sensor Network: SICS. Stockholm, Sweden.

10. Gladun A., Rogushina J. Intelligent Techniques of User-Oriented Recognition of Objects from the Web Informational Resources // Journal of Emerging Technologies in Web Intelligence, Vol 6, No 3, 2014. – P. 348-353.

11. Vermesan Ovidiu and Friess Peter Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems. – River Publishers, 2012.– 363 p.

12. Web вещей - визуализация энергопотребления – <http://www.webofthings.org/?s=energie+visible>.
13. Fielding Roy. Architectural Styles and the Design of Network-based Software Architectures (2000), Dissertation – <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>.
14. Andrushevich A., Tomek S. Klapproth A. The Autonomic Computing Paradigm in Adaptive Building // In Proceeding of the 2<sup>th</sup> Inter. Conference on Ambient Intelligence, Amsterdam, Netherlands, 2011.
15. Gladun A., Rogushina J. Use of Semantic Web technologies in design of informational retrieval systems // in Book “Building and Environment”, 2009 Nova Scientific Publishing, New-York, USA. – P. 89-103.
16. Гладун А.Я., Рогушина Ю.В. Репозитории онтологий как средство повторного использования знаний для распознавания информационных объектов // Онтология проектирования, № 1 (7), 2013. – С.35-50.
17. Levin M., Andrushevich A., Klapproth A. Composition of management system for smart homes // Information Processes, 2010, 10(1). P. 78-86.
18. Рогушина Ю.В., Гладун А.Я. Онтологическая модель интеллектуализации сервис-ориентированных вычислений в распределенной среде Интернет // Проблемы програмування, №2-3, 2006. – С. 526-536.
19. Гладун А.Я., Рогушина Ю.В. Онтологический подход к поиску веб-сервисов в распределенной среде Интернет // Информатика, Минск, № 4, 2006. – С. 116-127.
20. Gladun A., Rogushina J. Intelligent Techniques of User-Oriented Recognition of Objects from the Web Informational Resources// Journal of Emerging Technologies in Web Intelligence, Vol 6, No 3 (2014), 348-353, Aug 2014.- doi:10.4304/jetwi.6.3.348-353.- <http://www.ojs.academypublisher.com/index.php/jetwi/article/view/jetwi0603348353>

Статью представлено к печати 19.12.2014