DOI: 10.15514/ISPRAS-2021-33(6)-10

Unidata: открытая компонентная платформа для разработки MDM-решений

[1,2] *С.В. Кузнецов, ORCID: 0000-0001-6752-6742 <sergey@unidata-platform.ru>*
[1] *А.В. Цырюльников, ORCID: 0000-0001-7509-7888 <alexey@unidata-platform.ru>*
[2] *Д.В. Кознов, ORCID: 0000-0003-2632-3193 <d.koznov@spbu.ru>*
[1] *ООО «Юнидата»,*
197110, Россия, г. Санкт-Петербург, ул. Красного Курсанта 25Б
[2] *Санкт-Петербургский государственный университет,*
199034, Россия, Санкт-Петербург, Университетская набережная 7/9

Аннотация. Управление мастер-данными (Master Data Management, MDM) является важной дисциплиной управления данными в современных компаниях. В настоящее время индустриальные решения в этой сфере активно развиваются, о чем свидетельствует, в частности, ежегодные обзоры консалтинговой компании Gartner. Однако проблема эффективной настройки стандартных MDM-продуктов под индивидуальные нужды организаций стоит очень остро. Естественным решением этой проблемы является компонентная архитектура стандартного продукта, а также его открытость. Однако, фактически, единственным индустриальным открытым компонентным MDM-продуктом является на сегодняшний день платформа Unidata, разработанная в российской компании ООО «Юнидата». В данной работе представлена архитектура этой платформы, а также проведён обзор имеющихся на рынке открытых компонентных разработок в области MDM.

Ключевые слова: корпоративные информационные системы; мастер-данные; открытые системы; компонентные системы

Для цитирования: Кузнецов С.В., Цырюльников А.В., Кознов Д.В. Unidata: открытая компонентная платформа для разработки MDM-решений. Труды ИСП РАН, том 33, вып. 6, 2021 г., стр. 149-160. DOI: 10.15514/ISPRAS–2021–33(6)–10

Unidata: open source component platform for Master Data Management

[1,2] *S.V.Kuznetsov, ORCID: 0000-0001-6752-6742 <sergey@unidata-platform.ru>*
[1] *A.V. Tsyryulnikov, ORCID: 0000-0001-7509-7888 <alexey@unidata-platform.ru>*
[2] *D.V. Koznov, ORCID: 0000-0003-2632-3193 <d.koznov@spbu.ru>*
[1] *Unidata Ltd.*
Kursanta Str., St Petersburg, 197110, Russia
[2] *Saint Petersburg State University,*
7/9 Universitetskaya emb., St. Petersburg, 199034, Russia

Abstract. Master Data Management (MDM) is an important data management discipline in modern companies. Currently, industrial solutions in this area are actively developing, as evidenced, in particular, by the annual reviews of the consulting company Gartner. However, the problem of effectively customizing standard MDM products for the individual needs of organizations is very acute. The natural solution to this problem is the component architecture of the standard product, as well as its openness. However, in fact, the only industrial open component MDM product today is the Unidata platform, developed by the Russian company Unidata LLC. In this paper, the architecture of this platform is presented, as well as an overview of open component developments in the field of MDM available on the market.

Keywords: enterprise applications; master data management; open source; component approach

For citation: Kuznetsov S.V., Tsyryulnikov A.V., Koznov D.V. Unidata: open source component platform for Master Data Management. Trudy ISP RAN/Proc. ISP RAS, vol. 33, issue 6, 2021, pp. 149-160 (in Russian). DOI: 10.15514/ISPRAS–2021–33(6)–10

1. Введение

Переход к цифровой экономике предъявляет особые требования к управлению данными в больших организациях: необходимо, чтобы эта дисциплина была столь же повсеместно используема, как, например, бухгалтерия [5]. В большой организации имеется особый вид данных, без которых трудно себе представить её нормальную работу. Речь идёт о юридических данных, клиентской базе, сведениях о поставщиках и контрагентах и пр. Пользователи этих данных рассчитывают на их согласованность в пределах организации. Эти данные часто дублируются в различных информационных системах организации, в связи с чем возникают разночтения и противоречия, которые порождают проблемы, задержки, коллизии, финансовые и имиджевые потери [3]. Таким образом, целесообразно консолидировать и сопровождать единую целостную версию таких данных, обеспечивая ею все информационные системы организации. Тогда эти данные называются мастер-данными (Master Data), а процесс их сопровождения – управлением мастер-данными (Master Data Management, MDM) [1, 3].

Существует обширный спектр MDM-стратегий и методов, структурированных в рамках энциклопедии DAMA-DMBOK (Data Management Body of Knowledge) [3]. Существует также большое количество готовых *MDM-продуктов* от таких известных компаний как IBM, Oracle, Informatica и др. (см., например, отчет Gartner за 2021 год [4]). Однако на практике встречается значительное количество случаев, когда требуются нестандартные решения (далее – *MDM-решений*), поскольку целевые организации, особенно крупные, имеют большое количество индивидуальных черт. Кроме того, часто MDM внедряется в контексте итеративной стратегии – это означает, что с его помощью решаются определённые задачи организации – улучшение каких-то бизнес-процессов, внедрение в деятельность организации новых бизнес-идей и т.д. [7]. Также организация может иметь специфичную ИТ-инфраструктуру, используя широкий спектр платформ и технологий, которые непросто интегрировать в рамках стандартных MDM-продуктов, и, следовательно, необходимо специальное MDM-решение. После успешного решения всех этих задач возможно продолжение реализации MDM в организации для решения других задач, а также в рамках других бизнес-сегментов.

Индивидуальные особенности крупных организаций, итеративная стратегия внедрения MDM, а также гетерогенный ИТ-ландшафт больших организаций требуют гибкого подхода к внедрению MDM. Этого можно достичь, используя компонентный подход. С его помощью удаётся использовать MDM для решения бизнес-задач компании, дополняя базовый продукт необходимой функциональностью, интегрируя его с готовыми сторонними компонентами для работы с данными – поиска, хранения, очистки данных и пр. Компонентный подход позволяет легко комплектовать итоговую поставку MDM-решения для организации только необходимыми ей возможностями, «отрезая» остальные. Компонентное решение оказывается существенно легче поместить в облако, нежели монолитный продукт, что оказывается дополнительным преимуществом в ряде бизнес-сценариев. Также компонентную систему существенно проще развивать в рамках популярного на данный момент открытого подхода (open source).

В данной статье представляется открытая компонентная платформа Unidata, разработанная и успешно применяемая для крупного бизнеса одноименной российской компанией.

Платформа разделена на уровни (пакеты), включает в себя механизм создания адаптеров для различно платформо-зависимых возможностей, а также предоставляет средства масштабирования.

Статья организована следующим образом. В разд. 2 представлен обзор существующих на настоящий момент открытых и компонентных MDM-продуктов. Разд. 3 является обзором платформы Unidata. Разд. 4-7 описывают основные пакеты платформы Unidata. В разд. 8 подводятся итоги и обозначаются направления для дальнейших исследований и разработок.

2. Обзор существующих MDM-продуктов

MDM-решение для конкретной организации реализуется на основе готового MDM-продукта, выбранного из имеющихся на рынке. С обзором самых известных MDM-продуктов можно ознакомиться в отчетах консалтингового агентства Gartner [4]. MDM-продукты реализуют типовую функциональность по работе с мастер-данными, в частности, хаб данных для консолидированного хранения и доступа мастер-данных, средства для создания метамодели мастер-данных, процедуры очистки, классификации иерархизации мастер-данных, а также средства доставки мастер-данных целевым информационным системам организации. Весь этот функционал не нужно реализовывать «с нуля», что существенно снижает риски данного MDM-проекта организации, а также повышает качество итогового решения. Вместе с тем очень существенным вопросом является гибкость базовых MDM-продуктов, т.е. то, на сколько легко их изменить и настроить под индивидуальные потребности конкретной организаций. Рассмотрим несколько ведущих MDM-продуктов с этой точки зрения.

Многолетним лидером в области MDM по версии Gartner является продукт Informatica MDM [8]. Этот продукт был создан в компании Siperian, которую в 2010 году выкупила компания Informatica. Как и многие другие системы этого класса, Informatica MDM двигается в сторону поддержки облачной инфраструктуры, но при этом более чем пятнадцатилетнее наследие не позволяет сделать это эффективно. На текущий момент Informatica MDM, хоть и является наиболее функционально богатым MDM-продуктом, но имеет многочисленные ограничения для кастомизации. В частности, продукт имеет «монолитную» архитектуру, а также обладает существенной сложностью – как внутренней (сложность архитектуры), так и внешней – сложность пользовательского функционала. Это является следствием того, что продукт во многом составлен из других готовых продуктов, появившихся после поглощения компанией Informatica других продуктовых компаний. Также отметим, что исходный код системы является закрытым.

Ещё одним яркий представителем классических MDM-продуктов является SAP MDG [9]. Так же, как и Informatica MDM, этот продукт обладает богатой функциональностью. SAP MDG реализован на основе проприетарной платформы SAP HANA, что обеспечивает хорошую производительность и упрощает поддержку, но существенно ограничивает потенциального пользователя при интеграции SAP MDG с технологиями и системами, основанными не на SAP. Также стоит отметить, что продукт имеет «монолитную» архитектуру и его исходный код является закрытым.

Стоит отметить, что многие производители MDM-продуктов заявляют о том, что их продукты являются компонентными. Однако в лучшем случае они поддерживают открытый программный интерфейс, а также средства кастомизации отдельных платформо-зависимых возможностей.

Рассмотрим, какие из существующих MDM-продуктов действительно являются компонентными и открытыми.

Talend MDM является первой полноценной попыткой сделать открытый MDM-продукт [10]. Его архитектура является «монолитной», но имеется набор публичных интерфейсов. Проект стартовал в 2014 году и просуществовал до 2019 года.

Egeria является полноценным открытым проектом и ориентирован на обмен метаданными разных вендоров в области корпоративных информационных системам [11]. В проекте уделяется внимание тематике Data Governance, имеются отдельные компоненты, относящиеся к MDM-сфере. Однако Eneria нельзя назвать полноценным MDM-продуктом.

Продукт Pimcore MDM является частично открытым – он имеет открытое ядро, которое поддерживает контейнеризацию и ориентировано на исполнение в облачной инфраструктуре [12]. Несмотря на то, что по заявлениям разработчиков продукт является 100% API-driven, его архитектура в основе является «монолитной».

Продукт AtroCore MDM имеет компонентную архитектуру, но не является открытой. Этому продукту не хватает полноценной поддержки облачной инфраструктуры и горизонтального масштабирования. Ещё одной особенностью продукта является использование языка PHP, что удобно при создании небольших Web-ориентированных расширений, но может оказаться проблемой при прохождении аудита безопасности целевых решений, созданных на его основе.

Интересной попыткой сделать полностью открытый MDM-продукт с гибкими возможностями по кастомизации является проект Fuyuko, стартовавший в 2020 году [13]. К сожалению, после первого выпуска проект не получил дальнейшего развития.

Из этого обзора очевидно, что открытые компонентные MDM-продукты являются актуальной темой исследований и разработки.

3. Обзор архитектуры платформы Unidata

Платформа UniData состоит из компонент, которые являются единицами пакетирования и распространения функциональности платформы. Компоненты, в свою очередь, состоят из сервисов, а последние – из Java-классов.

Компоненты платформы UniData распределены по четырём основным пакетам: **Platform Core**, **Storages**, **MDM**, **Extra MDM** (см. рис. 1). Этот набор пакетов имеет уровневую структуру – чем ниже находится пакет, тем в большей степени он является системным, чем выше – тем более прикладным. Коме того, «низлежащие» пакеты предоставляют сервисы для «вышележащих» и при этом редко бывает так, чтобы пакет непосредственно использовал сервисы несмежного с ним нижнего пакета.

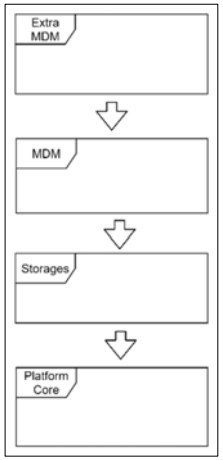

Рис. 1. Структура пакетов платформы Unidata
Fig. 1. Package structure of the Unidata platform

Кратко охарактеризуем эти пакеты.

- Пакет **Platform Core** включает в себя базовые компоненты платформы, минимально зависимые от других компонент и внешних продуктов и обеспечивающие системные сервисы платформы, такие как загрузка системы, выполнение цепочек задач, общие типы данных и пр. Компоненты этого пакета меняются крайне редко, и точно не меняются под конкретную задачу.
- Пакет **Storages** включает в себя различные хранилища данных, используемые платформой для разных функций и сервисов. Эти хранилища позволяют абстрагировать компоненты других пакетов от реальных хранилищ данных и работают с типами данных платформы (определены в Platform Core). Также большие сторонние хранилища упрощаются, так как MDM-функционал не нуждается во всех возможностях MySQL, Oracle и т.д. Платформа производит синхронизацию всех этих хранилищ на следующем уровне, в рамках пакета MDM.
- Пакет **MDM** включает компоненты, реализующие базовые функции управления мастер-данными. К этим функциям относятся управление метаданными, правила вычисления альтернативных вариантов мастер-данных, функциональность по управлению качеством данных, а также поиск дубликатов и реализация бизнес-процессов.
- Наконец, пакет **Extra MDM** включает в себя реализацию дополнительных функций управления мастер-данными, которые используются в сложных MDM-решениях.

В следующих разделах будет представлено описания каждого из пакетов. Скажем несколько слов об используемой ниже графической нотации в диаграммах, описывающих состав пакетов. В работе используется диаграмма компонент UML, но не некоторыми дополнениями.

- Зеленым цветом обозначаются компоненты, которые не меняются при создании MDM-решений на основе платформы Unidata.
- Желтым цветом обозначаются компоненты, которые существенно перерабатываются при создании MDM-решения.
- Серым цветом обозначаются сторонние компоненты, которые не являются частью платформы Unidata но используются при создании MDM-решений.

4. Пакет Platform Core

Рассмотрим пакет **Platform Core**. Компоненты этого пакета представлены на рис. 2.

Рис. 2. Пакет Platform Core
Fig. 2. Platform Core package

Платформа и создаваемые на ее основе целевые приложения организуются в виде компонент, реализующих один стандартный интерфейс, содержащий методы для инициализации, конфигурирования, запуска и проверки. Соответственно, компонента **System** обеспечивает контроль за загрузкой всех платформы проверяя, что у загружаемых компонент имеются все необходимые компоненты для нормальной работы[1].

[1] Существуют подобные системы, например, Apache OSGi/Spring DM, Eclipse Virgo, Java 9 Jigsaw и др. Мы создали упрощённый аналог, ориентированный на нашу платформу.

Компонента **Core** содержит основные типы данных платформы, такие как специализированные перечисления, словари, динамические ссылки (с постоянной и переменной частью), специальные массивы элементов, необходимых для MDM. Также типы данных поддерживают многоязычность (минимум, русский и английский языки). Также данная компонента поддерживает дополнительные сервисы, которыми могут пользоваться другие компоненты, например, логирование, аудит, проверка лицензий и пр.

Компонента **Draft** обеспечивает работу с черновиками. Черновики является важной частью MDM, поскольку мастер-данные нуждаются в синхронизации с данными ИД/ПД, и в рамках этого процесса часто создаются временные копии данных, т.е. черновики, которые проходят различные согласования прежде чем стать «чистовиками». Также черновики могут возникать при появлении конфликтов в процессе консолидации, при выполнении процедур Data Quality и т.д. Таким образом, важно абстрагировать работу с черновиком от деталей конкретной ситуации, так как имеется существенный общий функционал: создание, удаление и поиск черновика, назначение/переназначение владельца, публикация черновика (превращение его в «чистовик») по разным стратегиям – заменить текущий «чистовик», объединить (merge) черновика с текущим «чистовиком», отказ от публикации данного черновика и пр.

В MDM имеется достаточное количество действий, которые представляют собой цепочку операций, которая должна выполняться как единое целое. Например, к хабу с мастер-данными подключается новая система, которая хочет получить имеющиеся мастер-данные. Для осуществления этого действия нужно совершить следующие операции: выполнить подключение системы к хабу, обойти хранилище мастер-данных и выбрать нужные данные, переслать эти данные системе. Компонента **Jobs** реализует интерфейс для выполнения таких действий, состоящих из цепочки операций посредством интерфейса, содержащего следующие методы: конфигурирование цепочки операций, старт, просмотр результатов, перезапуск и т.д. Поддерживается также распределённое выполнение таких цепочек в облаке.

Компонента **Search.** В MDM-задачах операция поиска используется чрезвычайно часто: например, нужно найти определенную запись на хабе данных, или нужный атрибут в метамодели, или определённый бизнес-процесс и т.д. При этом важно иметь унифицированный интерфейс для выполнения поиска и использовать существующие поисковые машины (Elasticsearch, Apache Solr, Apache Lucene); связь с конкретным используемым средством поиска осуществляется с помощью специально реализованного для этого средства варианта компоненты **Search Adapter**. Несмотря на то, что Search адаптирован для использования компонентами платформы, значительной MDM-специфики у него нет. Отметим, что при решении данной задачи не существует такого широкого спектра технологий, как, например, для хранения данных или поиска дубликатов. В силу этих причин обе компоненты располагаются, во-первых, в пакете **Platform Core**, а во-вторых, сам поиск реализован с помощью двух видов компонент, а не в четырех, как, например, для поиска дубликатов (Match), и не в трёх, как для поддержки работы с графовыми данными.

5. Пакет Storages

Теперь рассмотрим пакет **Storages.** Компоненты этого пакета представлены на рис. 3.

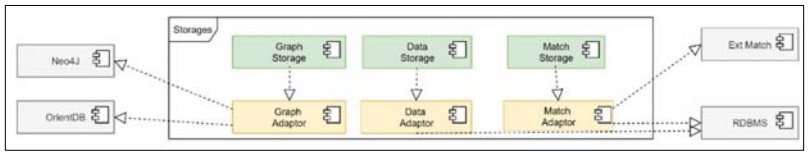

Рис. 3. Пакет Storages
Fig. 3. Package Storages

Компонента **Data Storage** представляет абстракции доступа к данным в терминах хранения (но не расчета, как компонента **Data** из пакета **MDM**). Эта компонента позволяет другим

компонентам платформы не зависеть от способа хранения мастер-данных (Oracle, MySQL, NoSQL, облако и пр. варианты). Также эта компонента реализует механизм транзакций, которые не поддерживаются, например, NoSQL.

Match Storage – компонента для абстрактного представления данных при поиске дубликатов: образца, который необходимо искать, а также кластер записей, в которых ищутся дубликаты. При этом кластер записей отличается от реальных записей: не все свойства реальных записей необходимы для поиска дубликатов, также несущественны детали хранения. Кроме этого компонента Match Storage осуществляет перевод исходного запроса в языки запросов для имеющихся готовых технологий по поиску дубликатов (matching engines), таких как Elasticsearch, Senzing и нек. др., а также в реляционные базы данных.

Graph Storage – компонента, которая абстрагирует графовую структуру данных для MDM-функционала. Графы используются в таких задачах, как каталог данных (DataCatalog) и другие специфичные для компаний сервисами, оперирующими разными иерархиями. При этом платформа использует существующие графовые хранилища, такие как Neo4J, OrientedDB и др. Таким образом, более высокоуровневые компоненты получают возможность не заботиться о деталях реализации графовых хранилищ и единообразно, в рамках всей платформы, создавать и использовать необходимые для их работы графы.

Компоненты **Data Storage Adaptor, Match Storage Adaptor** и **Graph Storage Adaptor** реализуют переходники к конкретным СУБД, средствам поиска дубликатов и графовым хранилищам, используемым в целевом MDM-решении. Для каждой из этих сторонних технологий реализуется (используется) специальный адаптер.

6. Пакет MDM

Опишем пакет MDM (рис. 4).

Рис. 4. Пакет MDM
Fig.4. MDM package

Компонента **Meta** предназначена для управления метамоделью (схемой, структурой) мастер-данных. Компонента позволяет создавать и редактировать типы сущностей, атрибутов и свойств атрибутов, а также виды связей. Она также поддерживает версионирование метамодели и графический интерфейс для создания и редактирования метамодели пользователем итогового MDM-решения, а также автоматически генерирует объектный программный интерфейс для доступа к мастер-данных в терминах метамодели.

Компонента **Data** отвечает за предоставление мастер-данных (в соответствии с метамоделью) другим компонентам платформы, а также для модификации мастер-данных. При этом важной чертой является функционал по расчету мастер-данных. Дело в том, что платформа хранит не просто окончательный, однозначно консолидированный вариант мастер-данных, но разные варианты для одной и той же мастер-записи. Финальный расчёт состава атрибутов и их значений производится при обращении к данным, в связи с контекстом и в соответствии с правилами. Например, могут существовать разные названия одной и той же организации – текущее и прежние; при запросе данных о юрлице выдается текущее название этой организации, а при запросе ее выполненных договоров целесообразно выдавать прежнее название, которое было у компании на момент выполнения этих договоров.

Match – компонента, которая предназначена для поиска дубликатов в некотором MDM-хранилище данных. Для этих целей не хватает компоненты Match Storage: поиск дубликатов

может производиться в хранилище мастер-данных, в черновиках, в бизнес-процессах – об особенностях таких хранилищ Match Storage не имеет информации. Более детально, компонента Match реализует следующую функциональность: задание match-правил в терминах MDM с порядком выполнения и с привязкой этих правил к различным MDM-сущностям (например, к элементам метамодели); отображение результатов выполнения Match-процедуры обратно, в термины MDM; оркестрирование поиска дубликатов, т.е. средства для задания расписания Match-процедуры и её выполнение согласно данному расписанию. Эта компонента может быть модифицирована по особенности MDM-решения, потому что может меняться подмножество, где осуществляется поиск (и так будет быстрее), а также могут появляться новые задачи.

Компонента **Data Quality** отвечает за проверку данных (поиск ошибок и определение того, что делать с ошибочными записями), а также выполняющий обогащение данных. Например, если нет названия юрлица, то такая запись вообще не допускается. А если, например, нет СНИЛС, то запись принимается, но помечается как содержащая ошибки. Вариантом обогащения может быть, например, добавление СНИЛС, извлечённого из открытых источников. Также компонента выполняет оценку интегрального качества записи, тем самым определяя степень доверия к записи.

Компонента **Workflow** поддерживает бизнес-процессы: платформа Unidata реализует подмножество стандарта BPMN 2.0 [14], включая процессы и задачи, а также события. Этот функционал важен, поскольку в рамках MDM требуется реализовывать различные бизнес-процессы, задействующие разных должностных лиц и разные департаменты, например, процесс консолидации данных. Данная компонента может использовать различные сторонние движки бизнес-процессов, такие как Activiti BPMN и Camunda BPMN.

7. Extra MDM

Наконец, перейдём к рассмотрению пакета **Extra MDM** (рис. 5).

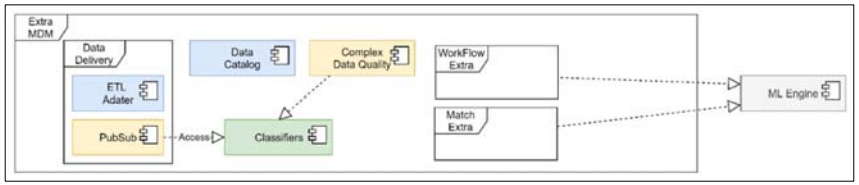

Рис. 5. Пакет Extra MDM
Fig. 5. Extra MDM package

Пакет **Data Delivery** реализует функциональность по доставке мастер-данных системам-потребителям. Компонента **PubSub** отвечает за доставку мастер-данных в режиме реального времени, пакетно и по подписке. При этом можно настроить необходимый объем данных, который требуется каждой целевой системе, а также расписание, по которому она будет получать эти данные. Также эта компонента может быть использована для закачки данных на хаб. Компонента **ETL Adapter** предназначена для передачи мастер-данных в ETL-цепочку трансформации данных[2].

Компонента **Classifiers** реализует функциональность для создания древовидных классификаторов мастер-данных, которые очень распространены в мастер-данных,

[2] ETL (Extract, Transform, Load) является обобщённой процедурой, которая состоит из следующих шагов-этапов: извлечение данных из исходных информационных систем, трансформация данных, последующая доставка данных в целевые информационные системы [15]. MDM может быть рассмотрен как один из шагов ETL-процедуры.

например, иерархия юридических лиц, иерархия видов товаров и пр. Кроме того, мастер-данные должны быть интегрированы с многочисленными внешними классификаторами. Компонента позволяет строить дерево из узлов, имеющих различные атрибуты. Также поддерживается версионирование деревьев.

MatchExtra – компонента, предоставляющая дополнительные, расширенные возможности по поиску дубликатов. Стандартная компонента **Match** базируется на понятии правила поиска дубликатов, т.е. на возможности описания правила в терминах атрибутов и свойств данных. Но в настоящее время развиваются подходы, основанные на других методах (например, машинное обучение), которые не могут быть описаны в виде правил, и стандартный подход не подходит. Для такого рода задач в платформе предполагается замена и/или расширение базовых возможностей, путем реализации дополнительных компонент. В частности, «референсной» реализацией такого подхода для решения задачи поиска дубликатов является компонента **MatchExtra**, использующая сторонний Machine Learning движок. Данная компонента не предоставляет интерфейса по заданию правил поиска дубликатов, но реализует функцию отображения результатов Match-процедуры в термины MDM. Реализация использует принцип дообучения моделей поиска дубликатов на основе решений пользователей. Данный метод желательно рассматривать как дополнительный к базовой функциональности, так как требует наличия размеченного набора данных для просторечия модели принятия решений. Что достигается путем накопления информации о решениях пользователей по дедупликации данных.

WorkflowExtra – компонента, предоставляющая дополнительные возможности для реализации сценариев по работе с матер-данными, основанных на бизнес-процессах. Эта компонента использует стандартную компоненту Workflow, но существенно расширяет её возможности за счет использования техник машинного обучения. В итоге компонента обеспечивает следующие дополнительные функции: выбор согласующего на основе предыдущей истории согласований, прогнозирование соблюдения временных регламентов при обработке запросов по работе с данными[3], поиск «совпадающих» заявок и пр.

Complex DataQulity – компонента, которая предназначена для проведения проверок мастер-данных, ориентированных на группу записей, в то время как компонента Data Quality позволяет проверять лишь одну запись. Например, проверка набора связанных записей одного или разных типов на соответствие атрибутов разных записей в наборе между собой, проверка агрегатов значений атрибутов разных записей набора на соответствие агрегата условию, или проверка соответствия значений атрибутов мастер и классификационных данных. Компонента обеспечивает дополнительный уровень представления композитного набора записей. Это представление вносит дополнительный слой агрегации изолированных моделей данных, хранящихся в платформе, и, зачастую, зависит от набора используемых компонент платформы.

Компонента **DataCatalog** реализует инвентаризированный набор знаний о видах данных компании, местах их происхождения, использования, хранения, предметных областях, и о взаимосвязях. Данная компонента нужна почти в любой крупной организации, являясь единым справочником имеющихся данных, что крайне важно для цифрового управления. Соответствующую справочную информацию нецелесообразно выяснять у авторов соответствующих ИС, поскольку таких ИС в организации может быть много (до 1000 и более).

[3] SLA (Service Level Agreement) – устоявшийся на сегодняшний день термин, определяющий скорость реагирования сервисных компаний, подсистем и пр. на запросы клиентов.

8. Заключение

В данной статье представлена открытая компонентная платформа Unidata, предназначенная для создания MDM-решений, ориентированных на обеспечение конкретных потребностей крупных бизнес-организаций. Данная платформа была создана на основе продукта Unidata и уже прошла успешную апробацию в ряде промышленных проектов. В качестве дальнейших направлений исследований и разработки можно указать доработку платформы для использования в облачных инфраструктурах, а также создание и интеграцию в архитектуру отдельных компонент, основанных на машинном обучении.

Список литературы / References

[1]. Gartner Glossary, Available at: https://www.gartner.com/en/glossary, accessed 01.12.2021.
[2]. Silvola R., Jääskelainen O. et al. Managing one master data – Challenges and preconditions. Industrial Management & Data Systems, vol. 111, issue 1, 2011, pp. 146-162.
[3]. DAMA-DMBOK: Data Management Body of Knowledge, Technics Publications, Second edition, 2017, 590 p.
[4]. Parker S., Hawker M., Walker S. Magic Quadrant for Master Data Management. Gartner 2021.
[5]. Khatri V., Brown C.V. Designing data governance. Communications of the ACM, vol. 53, issue 1, 2010, pp. 148-152.
[6]. Zmud R.W. An Examination of 'Push-Pull' Theory Applied to Process Innovation in Knowledge Work. Management Science, vol. 30, issue 6, 1984, pp. 727-738.
[7]. Кузнецов С.В., Кознов Д.В. Управление мастер-данными в рамках итеративного подхода. Онтология проектирования, том 11, no. 2, 2021, pp. 170-184.
[8]. Cloud-first, AI-powered Master Data Management. Available at: https://www.informatica.com/products/master-data-management.html, accessed 01.12.2021.
[9]. SAP Master Data Governance. Available at: https://www.sap.com/cis/products/master-data-governance.html, accessed 01.12.2021.
[10]. Clean, complete, uncompromised data for everyone. Available at: https://www.talend.com/, accessed 01.12.2021.
[11]. Egeria. Available at: https://odpi.github.io/egeria-docs/, accessed 01.12.2021.
[12]. Pimcore. Available at: pimcore.com, accessed 01.12.2021.
[13]. Fuyuko. Available at: fuyuko.org, accessed 01.12.2021.
[14]. Business Process Model and Notation (BPMN). Version 2.0. OMG, 2011, 538 p.
[15]. Vassiliadis P., Simitsis A., Skiadopoulos S. Conceptual modeling for ETL processes. In Proc. of the 5th ACM International Workshop on Data Warehousing and OLAP, 2002, pp. 14-21.

Информация об авторах / Information about authors

Сергей Викторович КУЗНЕЦОВ, исполнительный директор Юнидата, преподаватель кафедры прикладной кибернетики СПбГУ. Научные интересы: инженерия мастер-данных, требования к мастер-дата проектам, средства реализации управления мастер-данными, технологии доставки консолидированных данных потребителям.

Sergey Viktorovich KUZNETSOV, Executive Director of Unidata, Lecturer at the department of Applied Cybernetics of SPbU. Research interests: master data engineering, requirements for master data projects, means of implementing master data management, technologies for delivering consolidated data to consumers.

Алексей Владимирович ЦЫРЮЛЬНИКОВ, главный архитектор. Научные интересы: управление мастер-данными, архитектура средств управления мастер-данными, средства разработки и архитектура enterprise-приложений.

Alexey Vladimirovich TSYRYULNIKOV, chief architect. Research interests: master data management, master data management tools architecture, development tools, and enterprise application architecture

Дмитрий Владимирович КОЗНОВ, доктор технических наук, профессор кафедры системного программирования. Научные интересы: программная инженерия, модельно-ориентированная разработка программного обеспечения, программные данные, машинное обучение.

Dmitry Vladimirovich KOZNOV, Doctor of Technical Sciences, Professor of the System Programming Department. Research interests: software engineering, model-driven software development, program data, machine learning.