



Разработка Web-приложений с учетом элементов качества данных – DQAWA

¹ Ц. Гьерра-Гарсия, ORCID: 0000-0002-9290-6170 <cesar.guerra@uaslp.mx>

¹ Г. Перес Гонсалес, ORCID: 0000-0003-3331-2230 <hectorgerardo@uaslp.mx>

¹ М. Рамирез-Торрес, ORCID: 0000-0002-7457-7318 <tulio.torres@uaslp.mx>

² Р. Хуарез-Рамирез, ORCID: 0000-0002-5825-2433 <reyesjua@uabc.edu.mx>

¹ Автономный университет Сан-Луис-Потоси,

Мексика, 78000, Сан-Луис-Потоси, Альваро Обрегон, 64

² Автономный университет Нижней Калифорнии (UABC),

Мексика, 21100, Нижняя Калифорния, Эсенана

Аннотация. В настоящее время обеспечение достаточного уровня качества данных является первостепенной задачей для поддержки успешной деятельности любой организации или предприятия. Поэтому внедрение практик управления качеством данных необходимо в ситуациях, требующих обеспечения определенного уровня качества данных для конкретной функциональности приложения или сервиса. Подобные специализированные практики управления качеством данных должны внедряться в процесс разработки программного обеспечения как можно раньше. На основе анализа существующих предложений в данной области было установлено, что все еще заметна нехватка методологических и технологических решений при разработке приложений, ориентированных на качество данных. Основываясь на достижениях области Web-проектирования под управлением моделей, данная работа представляет частичный результат наших исследований в этой области: предлагается использование метамодели и профиля UML как артефактов качества данных на стадии проектирования Web-приложений. Основной целью работы является предоставление инструментов для предотвращения проблем с качеством данных при проектировании Web-приложений.

Ключевые слова: качество данных; Web-приложения; проектирование под управлением моделей

Для цитирования: Гьерра-Гарсия Ц., Перес Гонсалес Г., Рамирез-Торрес М., Хуарез-Рамирез Р. Разработка Web-приложений с учетом элементов качества данных – DQAWA. Труды ИСП РАН, том 33, вып. 2, 2021 г., стр. 49-64. DOI: 10.15514/ISPRAS-2021-33(2)-2

Developing Web Applications with Awareness of Data Quality Elements – DQAWA

¹ C. Guerra-García, ORCID: 0000-0002-9290-6170 <cesar.guerra@uaslp.mx>

¹ H. Pérez González, ORCID: 0000-0003-3331-2230 <hectorgerardo@uaslp.mx>

¹ M. Ramírez-Torres, ORCID: 0000-0002-7457-7318 <tulio.torres@uaslp.mx>

² R. Juárez-Ramírez, ORCID: 0000-0002-5825-2433 <reyesjua@uabc.edu.mx>

¹ Universidad Autónoma de San Luis Potosí,

Álvaro Obregón #64, Col. Centro, San Luis Potosí, C.P. 78000, México

² Autonomous University of Baja California (UABC),

Ensenada, Mexico, 21100

Abstract. An acceptable level of quality in data is nowadays a paramount for any kind of organization or enterprise that wishes its business processes to prosper. Thus, introducing activities focused in the data quality management is a crucial requirement for the analysts if the level of quality of data for the functionality or service at hand is to be ensured. Such specialized data quality management activities should be presented as early as possible during the software development process. So far and having done a search for proposals in this field, there is still a lack of either methodological or technological proposals with which a developer could be able to design data quality aware applications in the specific field of Web application development. Considering the benefits offered in the field of Model Driven Web Engineering, this work presents a partial outcome of our research in this novel field: a metamodel and a UML profile, both able to be used as data quality artefacts during the design stage of Web applications. The main objective is to provide the designer with the tools needed to design Web applications, in order to prevent data quality issues.

Keywords: data quality; Web applications; Model Driven Web Engineering

For citation: Guerra-García C., Pérez González H., Ramírez-Torres M., Juárez-Ramírez R. Developing Web Applications with Awareness of Data Quality Elements – DQAWA. Trudy ISP RAN/Proc. ISP RAS, vol. 33, issue 2, 2021, pp. 49-64 (in Russian). DOI: 10.15514/ISPRAS-2021-33(2)-2.

1. Введение

С точки зрения пользователя можно было бы утверждать, что Информационная Система не может быть качественнее, чем используемые ею данные [1]. Общая проблема поиска информации в Сети не изменилась – требуется найти документы и данные, релевантные запросу, то есть документы, соответствующие информационной потребности пользователя [2]. Учитывая огромное количество данных в Интернете, для организаций как никогда важно иметь возможность собирать, хранить, обмениваться и отображать данные с адекватным уровнем качества через Web-приложения [3-5]. Одной из основных трудностей при извлечении данных с Web-страниц является автоматическое обнаружение интересующих объектов и их свойств и сохранение их в базе данных в едином формате [6]. В некоторых организациях использовались отдельные базы данных для каждого подразделения, несмотря на то что они содержали информацию об одних и тех же объектах, но различались схемами данных. Кроме того, данные часто содержат типографские ошибки, неточности в их описаниях и устаревшие значения [7]. Описание требований должно включать структуру, содержание и значения результатов, ожидаемых пользователем, объектом или системой в определенных условиях и при определенных исходных данных [8].

Наиболее распространенные требования к качеству данных (Data Quality, DQ) включают: полноту, корректность, полезность и доступность для систем [4]. Эти требования к DQ должны быть введены как можно раньше в процесс разработки Web-приложений для обеспечения работы приложения в соответствии с ними. Для этого важно изучить, как Web-приложения обеспечивают сохранность данных и как они могут адаптировать свое содержание под различные конфигурации визуализации [9].

В данной работе проблема рассматривается как с методологической, так и с технологической точки зрения в рамках Web-проектирования под управлением моделей (Model-Driven Web Engineering, MDWE) [10]. Большая часть исследовательских работ в этой области была ориентирована преимущественно на этапы анализа и проектирования [11]. Кроме того, почти все существующие работы предлагают конкретные процессы для систематической и полуавтоматической разработки приложений. Но после анализа литературы [12-15] мы обнаружили, что ни одна из этих работ в области модельно-ориентированного проектирования не затрагивает вопросы, связанные с управлением DQ при проектировании и разработке Web-приложений.

Чтобы устранить выявленную проблему, в рамках данной работы предлагается *расширенная метамодель и соответствующий ей профиль, заданный с помощью языка UML, которые позволяют на стадии проектирования процесса Web-разработки объединить определенные функции и артефакты, совместно удовлетворяющие требованиям управления качеством данных*. Кроме того, предлагается набор правил отображения, способствующих улучшению качества проектирования *Web-приложений, ориентированных на качество данных*.

2. Родственные области

2.1. Разработка требований к управлению качеством данных

Читатели могут обратиться к нашей предыдущей работе [14], посвященной *требованиям к DQ и их анализу* для ознакомления с необходимыми элементами *управления качеством данных на стадии проектирования*. Введение элементов управления DQ позволяет учитывать требования к DQ и способствует их реализации на стадии проектирования.

В данной работе под DQ понимается *пригодность данных к использованию* [16-18]. С этой точки зрения, когда от пользователя требуется оценить уровень качества какого-либо элемента данных для решения конкретной задачи в конкретном контексте, необходимо построить модель DQ и определить несколько критериев DQ.

В данном исследовании ставится следующий вопрос: как должна разрабатываться Информационная Система (ИС), чтобы обеспечить адекватный уровень качества хранящихся в ней данных? Для этого пользователям должны быть предоставлены модели DQ с подходящими критериями DQ, которые подходят под их собственные требования к DQ. В данной работе используется стандарт ИСО/МЭК 25012 [19]. В нем определены характеристики DQ, значимых при оценке уровня DQ в информационной системе. Выделяются пятнадцать характеристик DQ (*точность, полнота, непротиворечивость, достоверность, актуальность, доступность и т.д.*), которые делятся на две группы: *неотъемлемые и зависящие от системы*.

Одной из целей данной работы является проектирование конкретных функций веб-приложения, позволяющих оценить характеристики, представленные в стандарте. В настоящее время не существует исследований в области перехода от анализа к проектированию, посвященных конкретным аспектам или характеристикам управления DQ на стадии проектирования Web-приложения. Единственной связанной с данной проблемой работой является [14], в которой для *спецификации требований к DQ в Web-разработке* были предложены расширенная метамодель и UML-профиль (названный *DQ_WebRE*).

2.2. Веб-проектирование

Наиболее важными методологиями, используемыми на этапе моделирования при проектировании Web-приложений, являются *OOWS* [20], *UWE* [21], *WebML* [22], *W2000* [23], *WSDM* [24], *SOD-M* [25], *WebSA* [26], *FACPL* [27] и *HiLA* [28].

Все эти методологии в основном посвящены выявлению и определению функциональных аспектов, которые в основном связаны с навигационными, концептуальными и

презентационными моделями. Однако ни одна из данных методологий не применяется для обеспечения DQ, за исключением подхода, предложенного Сери (Stefano Ceri) и др. (WebML) [22], где упоминаются некоторые конкретные цели и соображения, которые следует учитывать при разработке Web-приложений.

3. Расширение метамодели UWE для поддержки проектирования веб-приложений, ориентированных на качество данных

Проведя сравнительное исследование различных подходов, мы решили взять в качестве основы подход UWE. Подход UWE был впервые представлен Кох (Nora Koch) и Краусом (Andreas Kraus) в [21], а затем обновлен в [29-31]. Подход также был улучшен благодаря добавлению новых моделей, учитывающих соображения безопасности [32, 33]. В подходе UWE предлагается метамодель для описания концептов и отношений в Web-проектировании. Она определяет конкретные представления, графически изображаемые диаграммами классов UML, которые сгруппированы в четыре разных пакета: *Navigation*, *Presentation*, *Process* и *Content*. Мы решили использовать подход UWE в качестве основы, поскольку он предоставляет возможность расширения, в данном случае с помощью конкретных элементов DQ.

3.1 Спецификация стереотипов для элементов качества данных в проектировании

В этом подразделе описывается предлагаемый подход, который является расширением метамодели UWE с добавлением концепций, необходимых для управления DQ на стадии проектирования. Во-первых, мы проанализируем основные подходы, представленные в [34] и связанные с моделированием DQ, а затем рассмотрим следующие ключевые понятия *накема Process*: *DQProcessClass*, *DQProcessProperty* и *DQLink*, вместе с тремя новыми элементами в *накема Content* – *DQDim*, *DQDimProperty* и *DQ_Constraint*, а также один элемент *накема Presentation* – *UIE_DQVerifier*. Ниже представлено подробное описание данных метаклассов.

- ***DQProcessClass***. Представляет собой процесс, ответственный за управление метаданными DQ и связанный с данными, которые обрабатываются в каждом *ProcessClass*. Другими словами, *DQProcessClass* может использовать класс *DQDim* и определяет метаданные DQ каждого критерия DQ.
- ***DQProcessProperty***. Указывается в *DQProcessClass* и используется для определения конкретных критериев, связанных с DQ. Каждый элемент данных, управляемый соответствующим *ProcessClass*, должен охватывать какой-либо критерий.
- ***DQLink***. Специальная связь, соединяющая элементы *ProcessClass* и *DQProcessClass*.
- ***DQDim***. Этот метакласс представляет собой элемент контента; он соответствует всем критериям DQ: он будет в основном отвечать за сохранение различных метаданных DQ, называемых *DQDimProperty*.
- ***DQDimProperty***. Принадлежит критерию DQ и используется для определения всех метаданных DQ, связанных с каждым критерием DQ (*DQDim*).
- ***DQ_Constraint***. Представляет собой элемент контента. Соответствует всем *Constraints*, которые должны быть определены и связаны с различными критериями DQ вместе с их соответствующими границами (*upper_bound* и *lower_bound*).
- ***UIE_DQVerifier***. Этот элемент представляет класс, ответственный за проверку соответствия данных, управляемых каждым элементом пользовательского интерфейса различным характеристикам типовых критериев DQ (*DQDim*).

В табл.1 представлен предлагаемый набор новых стереотипных элементов DQ в соответствии со спецификацией UML [35].

Табл. 1. Спецификация стереотипов для элементов DQ в проектировании
Table 1. Specification of stereotypes for DQ design elements

Имя	UIE DQVerifier
Базовый класс	Класс
Описание	Класс, ответственный за определение и проверку соответствия данных, управляемых каждым элементом пользовательского интерфейса, указанным критериям DQ (DQDim).
Ограничения	Должен быть связан хотя бы с одним элементом UIElement
Используется в модели	Модель Presentation
Имя	DQProcessClass
Базовый класс	navigationNode::Class
Описание	Процесс, ответственный за контроль метаданных DQ, связанных с данными, обрабатываемыми в каждом элементе ProcessClass
Ограничения	Должен быть связан хотя бы с одним элементом типа ProcessClass
Используется в модели	Модели Navigation / Process
Имя	DQProcessProperty
Базовый класс	navigationProperty::Property
Описание	DQProcessProperty принадлежит классу DQProcessClass и используется для указания характеристик DQ, которым должен удовлетворять каждый элемент данных, управляемый в соответствующем классе ProcessClass.
Используется в модели	Модели Navigation / Process
Имя	DQLink
Базовый класс	link::Association
Описание	Специальная связь, соединяющая два типа конкретных элементов ProcessClass и DQProcessClass
Используется в модели	Модель Navigation
Имя	DQDim
Базовый класс	Класс
Описание	Соответствует всем критериям DQ: он будет в основном отвечать за сохранение различных метаданных DQ, называемых DQDimProperty.
Используется в модели	Модель Content
Имя	DQDimProperty
Базовый класс	Property
Описание	DQDimProperty принадлежит критерию DQ и используется для определения метаданных DQ
Используется в модели	Модель Content.
Имя	DQ_Constraint
Базовый класс	Класс
Описание	Соответствует набору всех Constraints, которые должны быть связаны с различными критериями DQ.
Используется в модели	Модель Content.

3.2 Правила отображения

Поскольку при получении диаграмм для проектирования на основе диаграмм требований важно поддерживать высокий уровень трассируемости, был создан набор правил отображения, приведенных в табл. 2, обеспечивающий корректный процесс создания диаграмм. Предложенные правила отображения должны быть учтены при моделировании соответствующей диаграммы (Content, Navigation, Presentation) на основе результатов, полученных на этапе анализа требований.

Табл. 2. Правила отображения DQ_WebRE в DQAWA
Table 2. Mapping rules DQ_WebRE to DQAWA

DQ_WebRE	DQAWA	Описание	Используется в модели
Add_DQ_Metadata	DQProcessClass	Каждый элемент Add_DQ_Metadata будет преобразован в DQProcessClass. Отношение элемента типа UserTransaction к другому элементу типа Add_DQ_Metadata будет изменено на конкретное отношение типа DQLink между элементами типа “ProcessClass” и DQProcessClass. Имена элементов типа DQ_Metadata и DQ_Validator, связанных с Add_DQ_Metadata, будут преобразованы в атрибуты типа DQDim в элементе DQProcessClass.	Navigation
DQ_Metadata	DQDim	Каждый элемент DQ_Metadata будет преобразован в DQDim, а его соответствующие определенные атрибуты (свойства DQDim Content im) будут скопированы. Каждое отношение между элементами типа DQ_Metadata и Contents будет преобразовано в отношения между элементом DQDim и элементом ContentClass.	Content
DQ_Validator	UIE_DQVerifier	Каждый элемент DQ_Validator будет преобразован в элемент UIE_DQVerifier. Отношение между элементами типа DQValidator и WebIU будет изменено на отношение между UIE_DQVerifier и PresentationPage.	Presentation
DQConstraint	DQ_Constraint	Каждый DQConstraint преобразуется в другой DQ_Constraint. Этот элемент будет отвечать за хранение конкретных данных, с которыми будут связаны ограничения, а также за переменные, используемые для определения максимальной и минимальной границ (upper_bound и lower_bound). Отношение между DQ_Validator и DQConstraint изменяется на отношение между DQDim и DQ_Constrain”.	Content

Для поддержки нашего подхода мы предлагаем новый UML-профиль для проектирования Web-приложений с ориентацией на DQ (DQAWA). Профиль был реализован с помощью инструмента Enterprise Architect, который поддерживает определение профилей на основе UML. После того, как профиль DQAWA был реализован, мы получили подходящую рабочую среду, в которой можно моделировать диаграммы с новыми элементами (рис. 1).

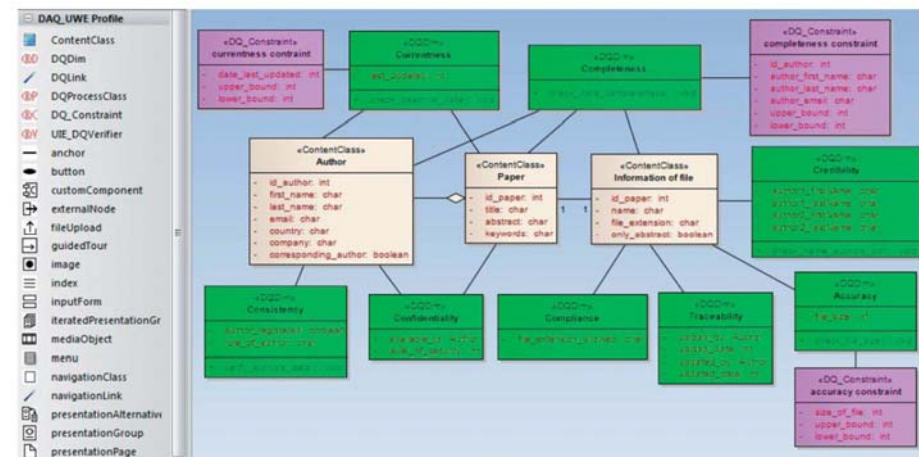


Рис. 1. Модель контента с элементами управления качеством данных
Fig. 1. Content model with elements for DQ management

4. Практическое применение

Чтобы проиллюстрировать использование предложенного профиля, в этом разделе рассматривается пример системы управления конференциями Easy Chair [36]. Это приложение позволяет подавать документы, управлять и контролировать членов Программного комитета (ПК) и назначать статьи рецензентам, а также имеет некоторые другие функции. Хотя данная работа посвящена стадии проектирования, для создания более полного понимания всего процесса ниже приведены результаты, полученные на предыдущих стадиях. Поэтому мы начнем с моделирования конкретной диаграммы использования (рис. 2) для каждого субъекта, использующего веб-приложение, а также смоделируем соответствующие конкретные требования к качеству данных с помощью элементов, которые определены в *профиле DQ_WebRE*, предложенном в [14] (подраздел 4.1). В подразделе 4.2 мы создаем соответствующие диаграммы для проектирования, используя элементы, определенные в *профиле DQAWA*, и соответствующие правила отображения.

4.1 Сбор и моделирование требований к качеству данных с помощью DQ_WebRE

На диаграмме использования представлены функциональные возможности каждого типа пользователя при использовании профиля *DQ_WebRE* (рис. 2). Таким образом, сначала необходимо смоделировать основные сценарии использования Автором, а затем с помощью диаграмм деятельности дать подробное описание каждого сценария использования. Для простоты описания было решено сосредоточиться исключительно на сценарии использования *Новое представление* (*New Submission*), для которого *Авторы* понадобятся использовать приложение Easy Chair.

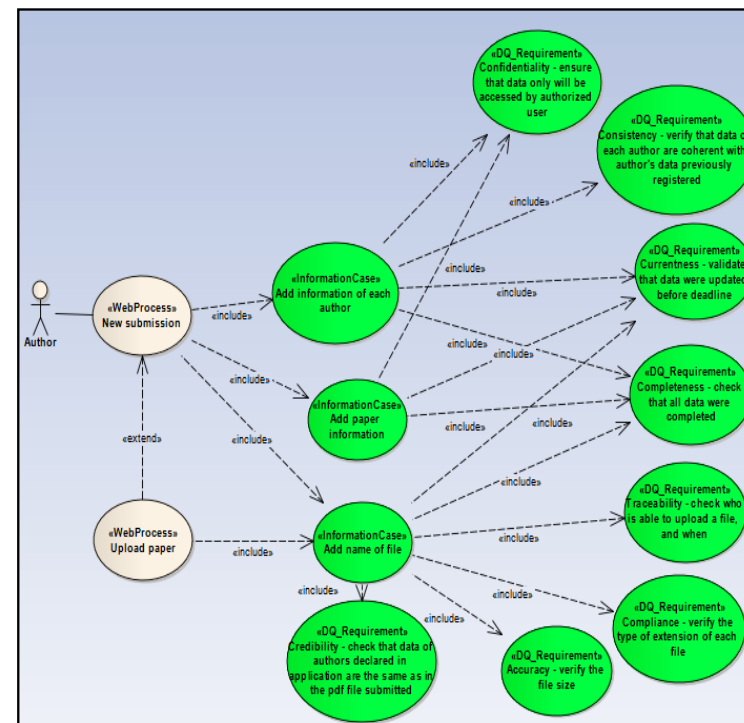


Рис. 2. Диаграмма использования с указанием требований к DQ
Fig. 2. Use case diagram specifying DQ requirements

Основываясь на работе, представленной ранее в [37], мы сформулировали требования к DQ, которые, по всей видимости, связаны с каждой из функциональных возможностей, реализованных в приложении. Учитывая, что сценарий использования *Новое представление* может быть отнесен к категории функциональности типа *Управление содержанием* (*Content Management*), можно видеть, что эта функциональность может быть связана со следующими характеристиками DQ: *точность, полнота, согласованность, достоверность и т.д.* Принимая во внимание эти соображения, смоделируем диаграмму использования (рис. 2) и обратим внимание на спецификацию конкретных сценариев использования *Добавить информацию о каждом авторе* (*Add information of each author*), *Добавить информацию о статье* (*Add paper information*) и *Добавить имя файла* (*Add name of file*), стереотипизированных как *InformationCase*, которые будут отвечать за управление всеми данными в сценарии использования *Новое представление* (см. комментарий на рис. 3).

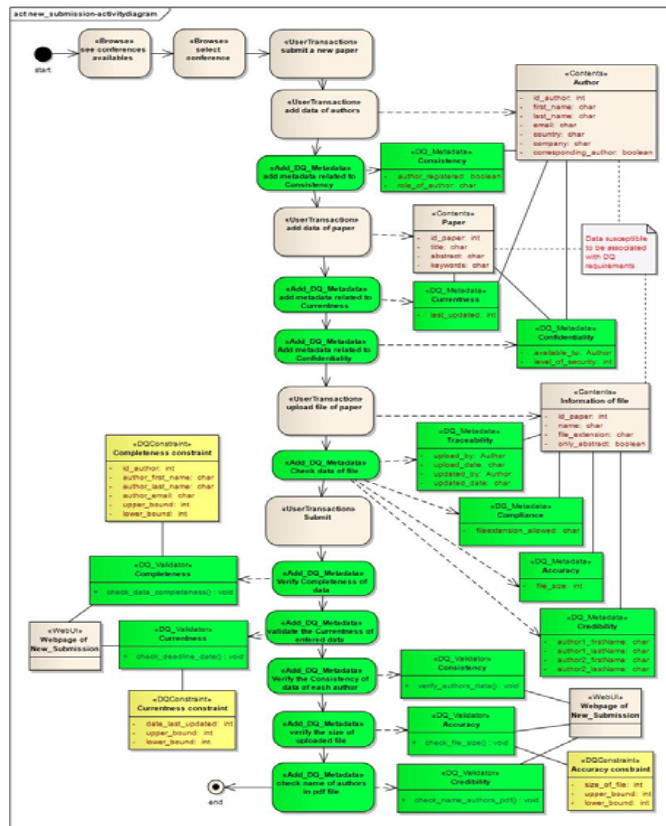


Рис. 3. Диаграмма активности с управлением DQ
Fig. 3. Activity diagram with data quality management

Теперь можно определить требования к DQ, которые будут связаны с каждым из элементов *InformationCase*. Если мы хотим гарантировать адекватный уровень *конфиденциальности* (*Confidentiality*) используемых данных, следует обеспечить выполнение следующего требования к DQ: *обеспечить доступ к данным только авторизованным пользователям* (*ensure that data will only be accessed by authorized users*). Аналогично, чтобы гарантировать *полноту* (*Completeness*) данных, следует также учесть следующее требование к качеству данных: *проверить, что все элементы данных заполнены* (*check that all the data items are completed*). Должны быть введены конкретные требования к *согласованности*, *актуальности*, *трассируемости* и *соответствию* данных на основе спецификации требований к DQ, показанных на рис. 2.

После корректной детализации диаграммы использования моделируется диаграмма действий с использованием соответствующих элементов, определенных с помощью *профиля DQ_WebRE*, которая также детализирует описания сценария использования *Новое представление*. Для диаграммы активности (рис. 3) можно смоделировать конкретные действия в соответствии с требованиями к DQ. Эти конкретные действия тесно связаны с требованиями к DQ, определяемые каждым пользователем для данных, управляемых каждым элементом *InformationCase*. На диаграмме активности первые действия, а именно *добавить метаданные, связанные с согласованностью* (*add metadata related to Currentness*) и *добавить*

метаданные, связанные с актуальностью (*add metadata related to Currentness*), которые стереотипизируются как *Add_DQ_Metadata*, будут отвечать за сбор метаданных, связанных с конкретной характеристикой *согласованности* (*author_registered*, *role_of_author*) и *актуальности* (*last_updated*). Эти метаданные будут храниться в экземпляре класса *DQ_Metadata* и использоваться для удовлетворения требований к DQ по критериям *согласованности* и *актуальности* соответственно.

Кроме того, активность *добавить метаданные, связанные с конфиденциальностью* (*add metadata related to Confidentiality*) будет собирать соответствующие метаданные, относящиеся к *available_to* и *level_of_security*, которые хранятся в конкретном классе *DQ_Metadata*. Заметим, что все классы *DQ_Metadata* связаны с данными, управляемыми предыдущими действиями *добавить данные авторов* и *добавить данные статьи* (стереотипизированные как *UserTransaction*). Активность *Проверить данные файла* (*Check data of file*) аналогично отвечает за сбор метаданных, связанных с выполнением конкретных требований к DQ по критерию *трассируемости* (*upload_by*, *upload_date* и т.д.), *соответствия* (*fileextension_allowed*), *точности* (*file_size*) и *достоверности* (*author1_firstName*, *author1_lastName* и т.д.).

Наконец, действия *Проверить полноту данных* (*Verify Completeness of data*), *Проверить актуальность введенных данных* (*Validate the Currentness of entered data*), *Проверить размер загруженного файла* (*Verify the size of uploaded file*) и *Проверить имена авторов и организации в поданном PDF-файле* (*Check authors names and affiliations in the submitted pdf file*) будут отвечать за добавление конкретных операций *check_data_completeness()*, *check_deadline_date()*, *check_file_size()* и *check_name_of_authors_in_pdf_file()*, чтобы проверить *полноту*, *актуальность*, *точность* и *достоверность* управляемых данных в элементе *Webpage New Submission*.

Данные, управляемые в элементе *Ограничение по полноте* (*Completeness constraint*), стереотипизированном как *DQConstraint*, будут использоваться для указания данных, необходимых для завершения представления, и спецификации границ критерия полноты, которые варьируются от *lower_bound* до *upper_bound*. Данные, хранящиеся в элементе *Ограничение по актуальности* (*Currentness constraint*), будут служить для определения допустимых границ и подтверждения актуальности даты обновленных данных *Автора* (*date_last_updated*). Действие *Проверить согласованность данных каждого автора* (*Currentness constraint*) будет отвечать за проверку этих элементов данных, введенных с помощью *Webpage New Submission*. Наконец, данные, хранящиеся в элементе *Ограничение по точности* (*Accuracy constraint*), будут использоваться для проверки того, что поданный файл не является пустым, а также для ограничения максимального допустимого размера (*upper_bound* и *lower_bound*).

4.2 Моделирование диаграмм для проектирования с помощью DQAWA

Следует отметить, что авторы метамодели UWE предложили набор правил преобразования [38, 39], с помощью которых можно получить различные элементы, используемые в моделях проектирования из элементов, которые определены в подходе *WebRE* для уточнения требований [40]. Если взять за основу стереотипные элементы диаграммы деятельности (рис. 3) и применить наши *правила отображения*, становится возможным создать соответствующие модели для проектирования. Диаграмма, содержащая *модель Content*, показана на рис. 1 вместе с классами, определенными для хранения основной информации, касающейся *Автора*, *Статьи* и *Информации* файла (стереотипизированной как *ContentClass*).

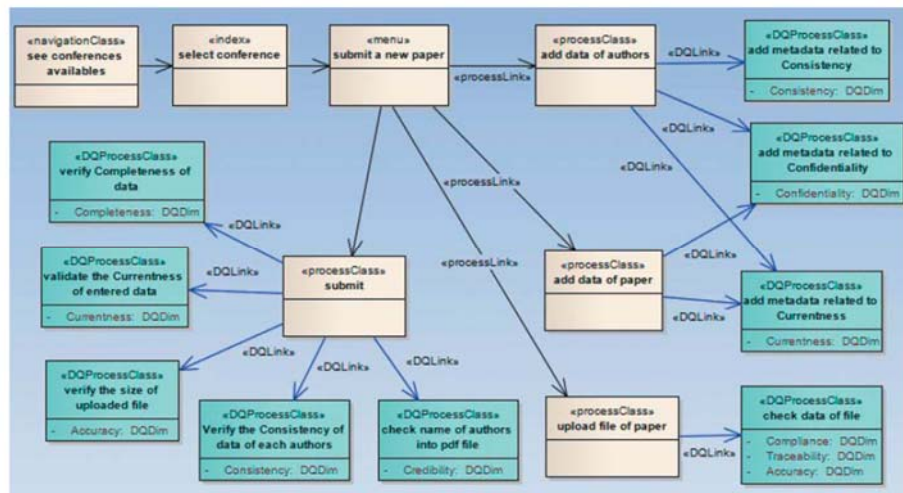


Рис. 4. Навигационная модель с управлением DQ
Fig. 4. Navigation model with DQ management

На этой диаграмме также показаны классы, связанные с определением каждого критерия качества данных (стереотипизированного как *DQDim*): *полноты*, *соответствия*, *конфиденциальности* и *т.д.*; а также их атрибуты (типа *DQDimProperty*) и соответствующие им операции. Также можно смоделировать классы *ограничение по актуальности*, *ограничение по точности* и *ограничение по полноте* (*DQ_Constraint*), которые будут хранить данные, используемые для определения ограничений и границ различных критериев качества данных.

Также возможно смоделировать гипертекстовую структуру с помощью модели *Navigation* (рис. 4). Она показывает узлы и связи (узлы именуются обозначенные как узлами процессов (*Process nodes*) и интегрированные в навигационный поток с помощью связей процессов). Эта диаграмма демонстрирует навигационный класс (*смотреть доступные конференции – see conferences available*), индексный класс (*выбрать конференцию – select conference*) и класс меню (*представить новую статью – submit a new paper*). Все данные, вводимые в веб-приложение, моделируются с помощью классов *добавить данные авторов*, *добавить данные статьи* и *загрузить файл статьи* типа *ProcessClass*.

Управление DQ моделируется с помощью элементов типа *DQProcessClass* (например, *добавить метаданные, относящиеся к согласованности – add metadata related to Consistency*, *добавить метаданные, связанные с конфиденциальностью – add metadata related to confidentiality* и *т.д.*), относящихся к разным классам типа *ProcessClass* с использованием конкретного отношения *DQLink*. Элементы *DQProcessClass* отвечают за указание характеристик DQ для данных в каждом узле процесса.

На рис. 5 показана диаграмма модели *Presentation* для нашего сквозного примера. Страница выдачи *представить новую статью* содержит три группы выдачи: *добавить данные авторов*, *добавить данные статьи* и *загрузить файл статьи*. Для каждой группы в модели выдачи показаны различные элементы. Например, в группе выдачи *добавить данные авторов* моделируются следующие элементы: *first_name*, *last_name*, *email*, *country*, *company* (тип *textInput*) и *corresponding author* (типа *selection*), а группа *добавить данные статьи* содержит поля *title*, *abstract* и *keywords* типа *textInput*. В этой модели выдачи мы видим, что класс *Полнота* (*Completeness*), стереотипизированный как *UIE_DQVerifier*, относится к

странице выдачи *представить новую статью*. Это означает, что этот класс будет отвечать за проверку полноты данных (с помощью операции *check_data_completeness*), введенных в каждый элемент, содержащийся в этой группе.

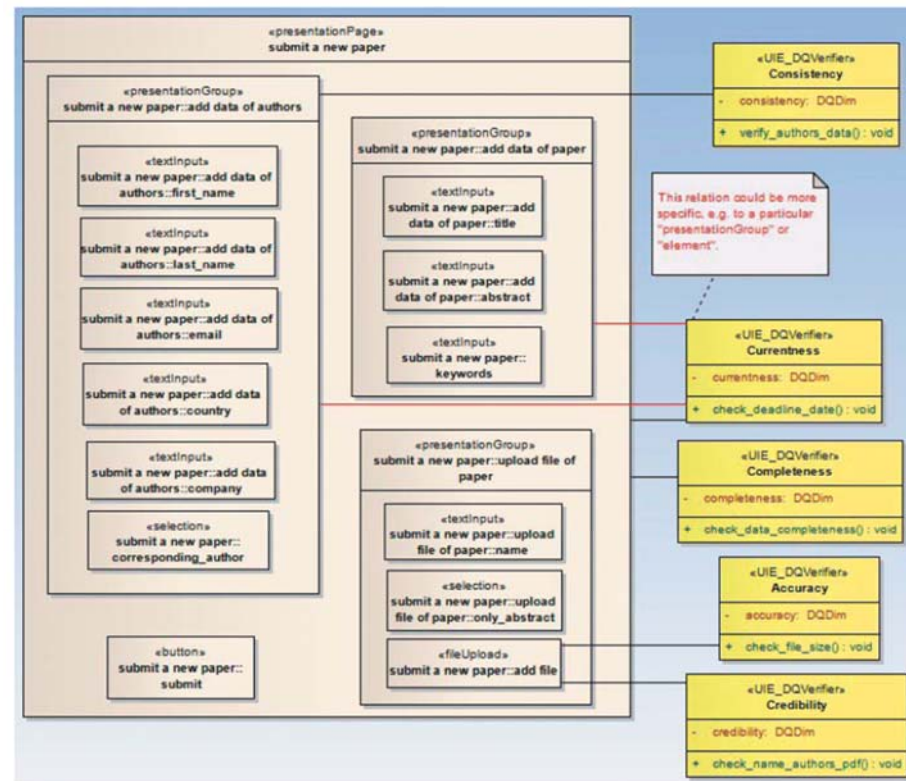


Рис. 5. Модель Presentation с элементами управления качеством данных
Fig. 5. Presentation model including elements for DQ management

Класс *Актуальность* (*Currentness*), стереотипизированный как *UIE_DQVerifier*, аналогичным образом будет отвечать за выполнение конкретной операции *check_deadline_date* для проверки актуальности каждого из элементов на странице выдачи *представить новую статью*. Класс *Согласованность* (*Consistency*) будет отвечать за проверку того, что все вводимые данные, принадлежащие каждому автору, поддерживаются согласованным образом с помощью операции *verify_authors_data*. Класс *Точность* (*Accuracy*) будет отвечать за выполнение операции *check_file_size* для проверки того, что размер файла находится в разрешенных границах. Наконец, класс *Достоверность* (*Credibility*) будет отвечать за проверку соответствия имени автора в файле формата pdf с именем, введенным в приложении.

5. Заключение

За последние два десятилетия Web-приложения стали первостепенными информационными ресурсами для большинства организаций. Поэтому важно, чтобы приложения могли обеспечивать адекватный уровень предоставляемых данных. Управление элементами

качества данных на этапе проектирования поможет избежать некоторых возможных проблем с данными, а также позволит внедрить соответствующие механизмы для требуемого уровня управления данными. Для решения этой проблемы на основе подхода MDWE представлены расширенная метамодель и соответствующий ей UML-профиль (*DQAWA*) для управления элементами качества данных при проектировании Web-приложений. В настоящее время данный подход реализуется в различных реальных проектах для проверки предложенных положений и определения его достоинств и недостатков, а также для получения количественных данных в управлении проектами (например, время, необходимое для внедрения требований к качеству данных). Результаты практического использования помогут оценить предлагаемый подход, как упоминалось в [41].

Список литературы / References

- [1] Strong D.M., Y.W. Lee, and R.Y. Wang. Data Quality in Context. Communications of the ACM, vol. 40, no. 5, 1997, pp. 103-110.
- [2] Некрестьянов И.С., Пантелеева Н.В. Системы текстового поиска для Веб. Программирование, том 28, no. 4, 2002 г., стр. 33-67 / Nekrestyanov I.S. and N.V. Panteleeva. Text Retrieval Systems for the Web. Programming and Computer Software, vol. 28, no. 4, 2002, pp. 207 - 225.
- [3] Akoka J., Berti-Equille L., Boucelma O et al. A Framework for Quality Evaluation in Data Integration Systems. In Proc. of the 19th International Conference on Enterprise Information Systems, ICEIS. 2007, pp.170-175.
- [4] Bertino E., Maurino A., and Scannapieco M. Guest Editors' Introduction: Data Quality in the Internet Era. IEEE Internet Computing, vol. 14, no. 4, 2010, p. 11-13.
- [5] ISO/TS--8000-1, ISO/TS 8000-1:2011 Data Quality - Part 1: Overview. 2011.
- [6] Варламов М.И., Турдаков Д.Ю. Обзор методов извлечения информации из Веб-ресурсов. Программирование, том 42, no. 5, 2016 г., стр. 39-48 / Varlamov M.I. and Turdakov D.Yu. A survey of methods for the extraction of information from Web resources. Programming and Computer Software, vol. 42, no. 5, 2016, p. 279-291.
- [7] Недумов Я.Р., Турдаков Д.Ю., Майоров В.Д., Овчинников П.Е. Автоматизация процесса нормализации информации при внедрении систем управления основными данными. Программирование, том 39, no. 3, 2013 г., стр. 3-14 / Nedumov Y.R., D.Yu. Turdakov, V.D. Maiorov, and Ovchinnikov P.E. Automation of data normalization for implementing master data management systems. Programming and Computer Software, vol. 39, no. 3, 2013, pp. 115-123.
- [8] Липаев В.В. Методология верификации и тестирования крупномасштабных программных средств. Программирование, том 29, no. 6, 2003 г., стр. 7-24 / Lipaev V.V. A Methodology of Verification and Testing of Large Software Systems. Programming and Computer Software, vol. 29, no. 6, 2003, pp. 298-309.
- [9] Aguilar J.A., Garrigós I., Mazón J.-N., Trujillo J. An MDA Approach for Goal-oriented Requirement Analysis in Web Engineering. Universal Computer Science, vol. 16, issue 17, 2010, pp. 2475-2494.
- [10] Moreno N. and A. Vallecillo. Towards interoperable Web engineering methods. Journal of American Society for Information Science and Technology, vol. 59, no. 7, 2008, pp. 1073-1092.
- [11] Escalona M.J., Torres J., Mejías M. et al., The treatment of navigation in web engineering. Advances in Engineering Software, vol. 38, no. 4, 2007, pp. 267-282.
- [12] Guerra-García C., Juárez-Ramírez R., Menéndez-Domínguez V. et al. Improving the Project Planning Process Considering Artifacts with Quality. In Proc. of the 4th. International Conference in Software Engineering Research and Innovation, 2016, pp. 15-20.
- [13] Guerra-García C., Llamas R. and Montaña-Rivas O. et al. QUACOP: An approach to Increase the Quality of Artifacts considered in a Project Planning Process. In Software Engineering: Methods, Modeling and Teaching. Universidad San Buenaventura Medellín, 2017.
- [14] Guerra-García C., Caballero I., and Piattini M. Capturing data quality requirements for Web applications by means of DQ_WebRE. Information Systems Frontiers, vol. 15, issue 3, 2013, pp. 433-445.
- [15] Guerra-García C., Caballero I., and Piattini M. A Survey on How to Manage Specific Data Quality Requirements during Information System Development. Communications in Computer and Information Science, vol. 230, 2011, pp. 16-30.
- [16] Ge M. and Helfert M. A Review of Information Quality Research - Develop a Research Agenda. In Proc. of the 12th International Conference on Information Quality, 2007, pp. 76-91.

- [17] Pipino L.L., Wang R.Y., Funk J.D., Lee Y.W. Journey to Data Quality. The MIT Press, 2006, 240 p.
- [18] Wang R., Pierce E., Madnick S. et al., eds. Information Quality. Advances in Management Information Systems. Routledge, 2005, 265 p.
- [19] ISO-25012, ISO/IEC 25012: Software Engineering-Software Product Quality Requirements and Evaluation (SQuaRE)-Data Quality Model. 2008.
- [20] Fons J., et al. Development of Web Applications from Web Enhanced Conceptual Schemas. Lecture Notes in Computer Science, vol. 2813, 2003, pp. 232-245.
- [21] Koch N. and A. Kraus. The Expressive Power of UML-based Web Engineering. In Proc. of the 2nd International Workshop on Web-oriented Software Technology (IWOST '02), 2002, pp. 105-119.
- [22] Ceri S., Fraternali P., and Bongio A. Web Modeling Language (WebML): a modeling language for designing Web sites. Computer Networks, vol. 33, issues 1-6, 2000, pp. 137-157.
- [23] Baresi L., Garzotto F., Mainetti L., Paolini P. Meta-modeling Techniques Meet Web Application Design Tools. Lecture Notes in Computer Science, vol. 2306, 2002, pp. 182-206.
- [24] De Troyer O.M.F. and Leune C.J. WSDM: a user centered design method for Web sites. Computer Networks and ISDN Systems, vol. 30, issues 1-7, 1998, pp. 85-94.
- [25] De Castro V. and Marcos E. Towards a Service-Oriented MDA-Based Approach to the Alignment of Business Process with IT Systems: from the Business Model to a Web Service Composition Model. International Journal of Cooperative Information Systems, vol. 18, no. 2, 2009, pp. 225-260.
- [26] Meliá S. and Gómez J. Applying Transformations to Model Driven Development of Web applications. Lecture Notes in Computer Science, vol. 3770, 2005, pp. 63-73.
- [27] Busch M., Koch N., Masi M. et al. Towards Model-Driven Development of Access Control Policies for Web Applications. In Proc. of the Workshop on Model-Driven Security, 2012, pp. 1-6.
- [28] Zhang G. and Hölzl M. Aspect-Oriented Modeling of Web Applications with HiLA. Lecture Notes in Computer Science, vol. 7059, 2012, pp. 211-222.
- [29] Koch N., Knapp A., Zhang G., and Baumeister H. Uml-Based Web Engineering. In Web Engineering: Modelling and Implementing Web Applications, Springer, 2008, pp. 157-191.
- [30] Koch N. and Kraus A. Towards a Common Metamodel for the Development of Web Applications. Lecture Notes in Computer Science, vol. 2722, 2003, pp. 497-506.
- [31] Koch N. and Kroib C. UWE Metamodel and Profile. User Guide and Reference. Technical Report 0802, Institute for Informatics. Ludwig-Maximilians-Universität München (LMU), 2008, 35 p.
- [32] Busch M. Evaluating and Engineering: an Approach for the Development of Secure Web Applications. Dissertation, Ludwig-Maximilians-Universität München, 2016, 215 p.
- [33] Busch M., Koch N., and Suppan S. Modeling Security Features of Web Applications. Lecture Notes in Computer Science, vol. 8431, 2014, pp. 119-139.
- [34] Guerra-García C., Caballero I., and Piattini M. A Systematic Literature Review of How to Introduce Data Quality Requirements into a Software Product Development. In Proc. of the 5th International Conference on Evaluation of Novel Approaches to Software Engineering, 2010, pp. 12-19.
- [35] OMG. Unified Modeling Language: Superstructure. Versión 2.0. 2005.
- [36] EasyChair. EasyChair Conference System. Available at: <http://www.easychair.org/>.
- [37] Guerra-García C., Caballero I., and Piattini M. DQ-VORD: A Methodology for Managing and Integrating Data Quality Requirements into Software Requirement Specification. In Proc. of the IADIS International Conference on WWW/INTERNET, 2009, pp. 392-399.
- [38] Koch N., Zhang G., and Escalona M.J. Model transformations from requirements to Web system design. In Proc. of the 6th international conference on Web engineering. 2006, pp. 281-288.
- [39] Kraus A., Knapp A., and Koch N. Model-Driven Generation of Web Applications in UWE. In Proc. of the 3rd International Workshop on Model-Driven Web Engineering, 2007, pp. 1-16.
- [40] Escalona M.J. and Koch N. Metamodeling the Requirements of Web Systems. Lecture Notes in Business Information Processing book series, vol. 1, 2006, pp. 267-280.
- [41] Batini C. and Scannapieco M. Data Quality: Concepts, Methodologies and Techniques. Springer, 2006, 281 p.

Информация об авторах / Information about authors

Сезар Артуро ГУЕРРА-ГАРСИЯ, Ph.D., профессор. Область научных интересов: программная инженерия, качество данных и информации, инженерия требований.

César Arturo GUERRA GARCÍA, Doctor of Computer Science, Full Time Professor. Research interests include Software Engineering, Data and Information Quality, Requirements Engineering.

Гектор Херардо ПЕРЕС-ГОНСАЛЕС, кандидат наук, профессор. Область научных интересов: программная инженерия, проектирование программного обеспечения, инженерия требований.

Hector Gerardo PEREZ-GONZALEZ, Ph.D., Full Time Professor. Research interests include Software Engineering, Software Design, Requirements Engineering.

Марко Тулио РАМИРЕС-ТОРРЕС – кандидат прикладных наук, профессор. Область научных интересов: криптография, вейвлет-анализ, виртуальные приборы.

Marco Tulio RAMÍREZ-TORRES, Ph.D. in Applied Sciences, Full time Professor. Research interests include Cryptography, Wavelet Analysis, Virtual Instrumentation.

Рейес ХУАРЕС-РАМИРЕС, кандидат наук, профессор. Область научных интересов: разработка программного обеспечения, оценка неопределенности программного обеспечения и взаимодействие человека с компьютером.

Reyes JUÁREZ-RAMÍREZ, Ph.D., Full Time Professor. Research interests include Software Engineering, software uncertainty estimation, and human-computer interaction.