

УДК 004.223.2

МЕТАДАННЫЕ И ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

К.В. СТАНКЕВИЧ*(Представлено: канд. физ.-мат. наук., доц. Ю.Ф. ПАСТУХОВ)*

В статье рассматривается состав и применение метаданных в прикладном программировании на различных этапах разработки программного обеспечения. Метаданные рассмотрены, как инструмент для отделения уровня работы с данными.

Ключевые слова: *информационные технологии, метаданные, информационные системы, информация, прикладное программирование.*

Самое простое для понимания определение: метаданные – это данные о данных. Чаще всего к метаданным относят информацию, описывающую структуры используемых программой данных, а также, способы их представления, методы их обработки, права доступа к ним и прочие настройки. Большинство более-менее качественных ресурсов в интернете созданы на базе некоторой системы управления содержанием сайта, которая, в свою очередь, формирует «контент» веб-страниц на основе определенной информации, извлекая ее, к примеру, из базы данных. Эта информация и является метаданными по отношению к содержимому управляемого сайта, потому как содержит перечень страниц, панелей, меню, профилей пользователей и прочих структурных элементов сайта – является данными, описывающими данные.

В настоящее время само по себе использование метаданных при разработке программ и ресурсов в интернете уже не является откровением, но то, как и на сколько широко и глубоко эти метаданные используются, существенно влияет не только на «читаемость» программного кода или html разметки, но и на возможность декомпозиции программы на отдельные функциональные компоненты, пригодные, в том числе, для повторного использования.

Многим известна трехуровневая архитектура программных продуктов, представленная уровнем работы с данными, уровнем бизнес – логики, который определяется назначением программы и третьим уровнем – уровнем пользовательского интерфейса. В этой статье рассмотрим метаданные, как инструмент для отделения уровня работы с данными от уровня бизнес – логики и, как средство «разгрузки» уровня пользовательского интерфейса от ненужных ему деталей.

Состав и применение метаданных в прикладном программировании. Большинство программных продуктов, выполняющих что-то осмысленное и полезное, используют для хранения своих данных реляционные базы данных – этот путь естественный и весьма хорошо «протопан». По этой причине выделить описание используемых программой таблиц в элемент метаданных кажется логичным и первым, что приходит в голову. Хранение в специальных структурах данных информации о таблицах (имя, перечень полей с указанием типа и прочих атрибутов) позволит инкапсулировать логику работы с отдельной таблицей (например, чтение, обновление, вставка и удаление записей) в специальном сервисе. Этот сервис мог бы предоставлять удобный программный интерфейс, а при инициализации получать на вход метаданные, описывающие структуру нужной вам таблицы – метаописание таблицы. Естественно, сразу приходит мысль в раздел метаданных включить метаописание соединения с базой данных. Такой подход поможет изолировать особенности работы с той или иной СУБД (Oracle, MS Access, MS SQL Server, MySQL и т.п.) от общего подхода к работе с данными.

Если вы мыслите категориями объектно-ориентированного подхода к проектированию программных продуктов, то согласитесь с аргументом, что работы с базой данных в терминах таблиц, пусть даже посредством сервисов, работающих на основе метаданных, - недостаточно для полноценного отделения уровня работы с данными от уровня бизнес-логики. Объекты, с которыми имеет дело программа, могут не описываться исключительно отдельными записями конкретной таблицы: элементы информационной модели могут быть представлены в базе данных совокупностью связанных записей, хранящихся в разных таблицах. Да и логика интерпретации данных одной и той же таблицы может отличаться в каждом конкретном случае, и определяться контекстом использования, например, текущим бизнес-сценарием или профилем пользователя.

Одним из следующих шагов в расширении структур метаданных может стать создание метаописания интерфейса таблицы, а именно, описание интерпретации каждого отдельного поля таблицы, например: простое значение, значение из справочника, ссылка на другой объект, автоматически вычисляемое значение и т.д. Метаописание интерфейса таблицы должно характеризоваться именем интерфейса, именем таблицы и набором характеристик для каждого поля, в состав которых входит описанный выше способ интерпретации, модификатор доступа «только для чтения», подпись и прочие полезные атрибуты.

Следующим шагом, логично завершающим первый этап разработки набора метаданных, является введение метаописания информационного объекта, которое, помимо уникального имени, будет иметь коллекцию ссылок на интерфейсы таблиц с указанием различных характеристик этих ссылок – компонентов объекта. Смысл такого подхода в том, что моделируемые программой объекты зачастую имеют множество функциональных аспектов, которые создаются по различным шаблонам проектирования и предоставляют информацию о своем состоянии через отдельные интерфейсы. Для каждого объекта, в таком случае, должен быть, как минимум, определен тип связи с экземплярами его компонентов, например: «один ко многим», «один к одному», «один к одному или нулю» (наличие информации необязательно) и т.д. Разработка и реализация сервиса работы с данными программы в терминах экземпляров информационных объектов – брокером информационных объектов – существенно поможет в решении поставленной выше задачи: отделении реализации уровня работы с данными от реализации уровня бизнес-логики. Уровень пользовательского интерфейса также будет упрощен, потому как User Interface будет иметь дело с экземпляром или коллекцией экземпляров информационных объектов и их брокером. Методы этого брокера будут позволять создавать, читать, удалять и сохранять изменения в информационном объекте в знакомом многим контексте транзакций. Также, проверки целостности и допустимости значений атрибутов компонентов (свойств интерфейсов) уместно частично возложить на сервисы метаданных.

Заключение. Расширений наборов метаданных придумано и разработано достаточно много с небольшими различиями в названиях, назначениях, структурах и т.д. В состав метаданных включают метаопределения запросов, метаопределения бизнес - сценариев и процедур контроля целостности данных, метаопределения профилей пользователя и другие. Большинство серьезных современных систем автоматизации построены на основе метаданных не только ради того, чтобы их код был прозрачным и разделяемым на уровни.

ЛИТЕРАТУРА

1. Метаданные и ваша безопасность [Электронный ресурс] – Режим доступа: <https://www.opengsm.com/blog/metadannie-i-bezopasnost/> – Дата доступа: 23.09.2019.
2. Доступные метаданные файла [Электронный ресурс] – Режим доступа: <https://suip.biz/ru/?act=mat> – Дата доступа: 23.09.2019.
3. Метаданные [Электронный ресурс] – Режим доступа: <https://ru.bmstu.wiki/%D0%9C%D0%B5%D1%82%D0%B0%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5>. – Дата доступа: 23.09.2019.