МЕЖФОРМАТНЫЕ МЕТА-ПРЕОБРАЗОВАНИЯ ГИПЕРТЕКСТА

М. Ю. Колодин

Санкт-Петербургский институт информатики и автоматизации РАН

СПИИРАН, 14-я линия В. О., д. 39, Санкт-Петербург, 199178

<myke@mail.ru>

УДК 004.91

Колодин М. Ю. **Межформатные мета-преобразования гипертекста** // Труды СПИИРАН. Вып. 6. — СПб.: Наука, 2008.

Аннотация. *Рассмотрены вопросы преобразования информации между важнейшими форматами, показана необходимость использования универсальных открытых форматов, предложен мета-подход к выполнению таких преобразований.* — Библ. 6 назв.

UDC 004.91

Kolodin M. Y. **Interformat meta-transformations of hypertext** // SPIIRAS Proceedings. Issue 6. — SPb.: Nauka, 2008.

Abstract. *Issues of information transformation between most important formats are studied, necessity of using universal open formats is shown, meta-approach to perform such transformations is suggested.* — Bibl. 6 items.

Современный этап развития информатики характеризуется наличием множества одновременно используемых форматов представления как простой текстовой, так и сложной гипертекстовой и гипермедийной информации [1]. Значительные ресурсы тратятся на преобразования информации между этими форматами, зачастую с потерей части информации или с достройкой ее по некоторым дополнительным (часто эвристическим) правилам. Такое состояние не может считаться удовлетворительным ни при ручной, ни, тем более, при автоматической обработке документов.

Сейчас для выполнения работ и оформления результатов научных исследований часто используются «офисные» форматы, типа MSO Word и т. п. Однако это связано со многими проблемами: технологическими, лицензионными, объема, сложности, переносимости и т. п. Современные «офисные» форматы (типа OpenOffice), особенно стандартизованные ISO, лучше, поскольку они четко документированы и имеется API для работы с ними для большинства распространенных языков программирования, но и они избыточно сложны. Форматы типа HTML недостаточны, поскольку помимо собственно текста фактически содержат только сведения о его текущем оформлении, да и то неполные, почти полностью игнорируя содержательную, смысловую часть. Форматы типа CHM, PDF, PS ориентированы более на статическое окончательное оформление документов; использовать их внутри рабочего цикла нецелесообразно. Форматы типа SGML (в т. ч. DocBook, TEI) интересны, универсальны, но зачастую избыточно сложны для обычных случаев, ориентированы на свою область применения (документация, литература).

Для повышения эффективности и единообразия преобразования информации, ее переносимости между различными платформами и системами, понижения сложности как собственно информационных блоков, так и их преобразователей, для удобства использования таких систем предпочтительно использовать универсальные открытые форматы и преобразователи, такие как XML и его производные. Заметим, что для первоначального ввода информации чело-

веком могут быть и даже желательны упрощенные форматы (на основе линейного текста с простой разметкой, например, класса wiki [2]), но для машинной обработки документы в них должны быть преобразованы к XML с попутной проверкой правильности введенного текста. Использование прочих упомянутых форматов нежелательно, но допустимо при условии обеспечения полноценного автопреобразования между ними (или их специально выбранными подмножествами) и базовым форматом типа XML, а также при помещении их в конце технологической цепочки (например, как окончательный документ, выдаваемый внешнему пользователю, без возможности дальнейшей обработки).

Хранение информации в системе при этом можно реализовать несколькими способами: хранить только исходный текст, или только полученный XML, или оба экземпляра документа. Оптимальным оказалось двойное представление: для редактирования пользователь получает простой размеченный им же текст, при вводе обновленного текста в систему таковой проверяется и переводится в XML, из которого дальше получаются все прочие производные; при активном редактировании или наличии сложных связей в системе документов можно пользоваться отложенной процедурой: каждый раз преобразование не выполняется, а перед использованием XML-формы проверяется ее актуальность и при необходимости выполняется сборка, то есть преобразование системы документов к XML и далее в соответствии с технологической цепочкой. Кроме очевидной трудоемкости набора в XML (даже в «упрощенных» его формах, типа YAML), следует учитывать, что вся информация, полученная от пользователя, по умолчанию считается потенциально некорректной по формату; соответственно, дополнительная проверка и переформатирование введенных данных, которые в соответствующих контекстах на последующих этапах работы становятся программами, необходимы.

В дополнение к уже использовавшимся форматам и мета-преобразованиям [3] следует применить мета-подход и к самим преобразователям, в частности, обобщить его на произвольное число уровней и направлений преобразований. При этом чем выше число повторных использований преобразователей, тем выше их полезность и эффективность [4]. Именно применение универсальных открытых форматов и свободно распространяемых преобразователей и средств построения таких преобразователей может существенно помочь в решении этой задачи, поскольку независимо от конкретного формата при наличии его мета-описания преобразователи получаются автоматически. Практические их реализации могут быть выполнены на большинстве современных языков и систем программирования, однако следует стремиться к максимальному абстрагированию программного ядра системы от конкретных документов, форматов, преобразователей в частности и их мета-описаний вообще; удобнее для работы языки интерпретирующего типа, где можно динамически построить программу и выполнить ее (perl, lisp, и т. п.); наиболее полно и удобно реализует эту идею язык форт [5].

Для выполнения этого условия будем рассматривать преобразователи и описания форматов как такие же мета-системы, как и исходные и результирующие системы документов, с применением к ним тех же способов обработки. При этом достигается почти полная унификация описания системы, включающая все ее элементы; степень повторного использования элементов существенно повышается. Известные реализации показывают перспективность данного направления. Для полноты исследования представляется полезным аналогичное рассмотрение метрик, оценок, контекстов и целей преобразований.

Действительно, как преобразуемая система документов (пассивная составляющая процесса), так и преобразователи (активная составляющая) могут быть описаны в одних и тех же терминах; для их разработки, создания, использования, дальнейшего преобразования могут быть использованы те же инструменты; из одних и тех же систем различными преобразователями или в различных контекстах — совокупностях задействованных объектов (например, параметров) и их значений — получаются различные результаты заданного типа. Это позволяет почти полностью унифицировать процесс, сводя к минимуму исключения из него; активные составляющие, однако, никогда не будут устранены, какой бы сильной многоуровневой свертке ни подвергались компоненты системы.

Эффективность описываемого подхода весьма высока, если ему следовать полностью; отклонения и «временные упрощения» сказываются крайне негативно на результате. На первых экземплярах систем и первых шагах внедрения часто наблюдается снижение производительности персонала вследствие малого опыта и ошибок в применении инструментария, его неготовности или недонастроенности, неправильной структуризации проекта и его документов, необходимости переписывать документы и программы по мере реструктуризации проекта и переформулирования требований. Однако на рабочих этапах и при последующей работе с аналогичными проектами производительность труда существенно (на величины порядка десятков процентов) возрастает [6]. При этом чем более сильно использованы мета-свойства, чем больше уровней работы вовлечены в процесс, тем больше последующих этапов работы и новых систем оказываются «аналогичными». Полезно сразу заводить системы метрик даже в не полностью готовые проекты, чтобы иметь возможность в дальнейшем оптимизировать работу, явно записывать опыт работы, готовить инструментарий к повторному применению (то есть документировать его, тестировать, модернизировать, и т. п.), вводя все эти деятельности как составные части процесса. Сам процесс работы также может быть мета-формализован и соответствующим образом технологизирован; это направление сейчас изучается теоретически и будет реализовано программно.

Литература

1. *Колодин М. Ю.* Многоформатные документы // Труды конференции CyrTeX-2000. Протвино, 2000. С. 40–45.
2. *Колодин М. Ю.* Сравнительный анализ гипертекстовых форматов и преобразований документов между ними [Электронный ресурс] // <http://ru.wikipedia.org> «Википедия:Википконференция 2007/Программа/Доклады/Колодин М.Ю. Сравнительный анализ гипертекстовых форматов и преобразований документов между ними» (по состоянию на 29.02.2008)
3. *Колодин М. Ю.* Мета-технология: Назначение и реализация // Информационные технологии и интеллектуальные методы. СПб: СПИИРАН, 1995. С. 83–86.
4. *Колодин М. Ю.* Произвольно-уровневые гипертекстовые системы // Современные проблемы информатизации в системах моделирования, программирования и телекоммуникациях. Сборник трудов. Вып. 9 (по итогам IX Международной открытой научной конференции). Воронеж: Научная книга, 2004. С. 358–359.
5. *Колодин М. Ю.* Мета-возможности в форте (на англ. яз.) // Международная конференция «ЕвроФорт-96». СПб, 1996. С. 41–44.
6. *Колодин М. Ю.* Мета-информационная модель поддержки распределённой коллективной деятельности. // X Российская конференция «Распределенные информационно-вычислительные ресурсы» (с участием иностранных ученых). Новосибирск, 2005. С. 32–34.