**Программные стандарты и их спецификации**

Сергей Кузнецов

Программные стандарты являются основой подхода Открытых Систем. По прошествию многих лет я не могу не согласиться с Юрием Николаевичем Знаменским (Привет, Юра!) в том, что для создания распределенных систем необходимо использовать стандартные транспортные протоколы. В той или иной степени, в зависимости от прикладной области, но по крайней мере, учитывать наличие стандартов нужно обязательно. Если... Если вы хотите сохранить возможность расширения своей системы путем вовлечения в нее компонентов, разработанных независимо, но с учетом стандартов. Если... Если вы хотите обеспечить интероперабельность (новорусское словечко, означающее возможность совместного функционирования независимо разработанных программных или аппаратных элементов) компонентов своей системы с компонентами других систем, разработанных независимо, но с учетом стандартов. Если... Если вы хотите сохранить возможность переноса приложений на платформы других производителей, разработанные независимо, но с учетом стандартов.

Следование набору общепринятых стандартов практически эквивалентно приверженности подходу Открытых Систем. Сегодня это уже всем понятно (конечно, тем людям, для которых это существенно). Непонятно другое: как должен быть оформлен стандарт, насколько он должен быть формализован, как проверить соответствие конкретной реализации тому или иному стандарту. Общее согласие по этому вопросу отсутствует. Имеется масса различных точек зрения, предлагаются различные решения. И понятно, что вряд ли удастся принять стандарт для составления стандартов. Эта заметка направлена на то, чтобы хотя бы частично разобраться с современными стандартами программных средств, с их спецификациями, уровнями формализации стандартов и возможностями проверки соответствия стандарту конкретной реализации. Я не претендую на общность и излагаю только собственные соображения без ссылок на авторитеты.

Начнем с положительных (и не очень) примеров. Для меня самым любимым стандартом является международный стандарт ANSI/ISO языка Си. Вот почему я его люблю. Этот стандарт опубликован в виде двух книг. Первая книга представляет собой формальное описание языка, включая Бекусовские определения синтаксиса и естественно-язычные (на английском языке) описания семантики соответствующих языковых конструкций. Вторая книга (Rational) включает подробные неформальные разъяснения смысла языковых конструкций, введенных в первой книге. Идея стандарта состоит в том, что параллельно читаются обе книги. Основная информация содержится в первом томе, но как только изложение на (полу)формальном уровне становится непонятным, можно обратиться к соответствующему месту второго тома и получить неформальные человеческие пояснения. Кроме определения языковых конструкций Стандарт Си содержит спецификации основных библиотек, которые должны поддерживаться в любой стандартной реализации языка Си. Наличие этих спецификаций исключительно важно само по себе, поскольку, как известно, язык Си не содержит конструкций, обеспечивающих связь с внешним миром (в частности, операторов ввода/вывода). Для этой заметки особенно важно то, что спецификации библиотечных функций в Стандарте Си вводятся с использованием ранее определенных конструкций языка Си. Конечно, эти спецификации носят только синтаксический характер, а семантика библиотечных функций определяется на естественном языке.

Вторым по качеству, с моей точки зрения, является стандарт языка баз данных SQL-92. По моему мнению, этот стандарт является лучшим в компьютерной истории стандартом языков баз данных. Синтаксические конструкции языка формально определяются Бекусовскими формулами. Семантика операторов описывается на естественном языке, но достаточно подробно и точно. Подобно стандарту языка Си стандарт SQL-92 содержит дополнительную часть, в которой средствами языка SQL специфицированы необходимые таблицы-каталоги, которые должны поддерживаться в любой SQL-ориентированной базе данных. По своей значимости наличие стандартизованных спецификаций таблиц-каталогов равносильно наличию стандартизованных спецификаций библиотек в стандарте языка Си. Еще раз заметим, что спецификации стандарта SQL-92 носят исключительно синтаксический характер. Весь смысл языковых конструкций и стандартизованных таблиц-каталогов объясняется на естественном языке.

Наверное, наиболее актуальный набор стандартов в мире операционных систем составляют стандарты, составленные рабочими группами POSIX. Первая рабочая группа POSIX (Portable Operating System Interface) была образована в IEEE в 1985 г. на основе UNIX-ориентированного комитета по стандартизации /usr/group (ныне UniForum). Отсюда видна первоначальная направленность работы POSIX на стандартизацию интерфейсов ОС UNIX. Однако постепенно тематика работы рабочих групп POSIX (а со временем их стало несколько) расширилась настолько, что стало возможным говорить не о стандартной ОС UNIX, а о POSIX-совместимых операционных средах, имея в виду любую операционную среду, интерфейсы которых соответствуют спецификациям POSIX.

Наиболее важной с практической точки зрения является деятельность рабочей группы POSIX 1003.1 "Интерфейсы системного уровня и их привязка к языку Си". В документах этой рабочей группы определяются обязательные интерфейсы между прикладной программой и операционной системой. С выпуска первой версии этого документа началась работа POSIX, и он в наибольшей степени связан с ОС UNIX, хотя в настоящее время интерфейсы 1003.1 поддерживаются в любой операционной среде, претендующей на соответствие принципам Открытых Систем.

Из числа прочих рабочих групп упомянем POSIX 1003.2 "Shell и утилиты", POSIX 1003.3 "Общие методы проверки совместимости с POSIX", POSIX 1003.4 "Средства, предоставляемые системой для прикладных программ реального времени", POSIX 1003.5 "Привязка языка Ада к стандартам POSIX", POSIX 1003.6 "Расширения POSIX, связанные с безопасностью" и т.д.

Рабочие группы POSIX в настоящее время находятся в ведении IEEE, и именно этот институт по мере готовности стандартов рекомендует их к принятию Международной организации по стандартизации (ISO).

Как показывает наличие POSIX 1003.3, POSIX-сообщество справедливо озабочено проблемой формальной проверки соответствия стандартам конкретных реализаций. К сожалению, несмотря на наличие целого ряда соответствующих программных продуктов, проверки носят только синтаксический характер. Как и большинство современных программных стандартов, все документы POSIX включают описание семантики только на неформальном уровне.

Приведенные примеры, конечно, затрагивают лишь небольшую часть современных программных стандартов. Однако этого достаточно, чтобы продемонстрировать основную проблему: мы научились (и уже давно) формально специфицировать синтаксис программных конструкций, но не умеем на том же уровне простоты специфицировать их семантику. И дело не в том, что отсутствуют языки спецификации семантики (например, существует красивый язык алгебраических спецификаций SDL). Беда в том, что при использовании любого такого языка семантические спецификации получаются слишком сложными. Сложность семантической спецификации программной конструкции приближается к сложности реализации этой конструкции на языке программирования. Поэтому, в частности, возникает задача проверки правильности (или отладки?) самих спецификаций. А на что при этом опираться? Снова на неформальное описание семантики?

Поэтому, как это не печально, в ближайшем будущем нам придется принимать на веру заявления производителей программных продуктов об их соответствии стандартам. Некоторую уверенность может дать процедура сертификации программного продукта, производимая авторитетной и независимой организацией. Но и эта уверенность может быть только относительной, поскольку эксперты, выполняющие процедуру, тоже опираются на неформальные спецификации семантики.