

Шутливые размышления о кроссархитектурости

Nikolay N. Ivanov, Четверг, 21 Апрель 2005, 11:47

"Бей, но выслушай!"

Фемистокл.

"И замерещилась мне... история - в каких-то темных углах..."

Ф. М. Достоевский.

Preface

На размышления о кроссархитектурости меня натолкнула детская англоязычная книга "Alice in wonderland" (Lewis Carrol). Прошу не считать доподлинно серьезным все написанное ниже, поскольку это, как следует из названия, всего лишь размышления, но никак не мое личное мнение.

Абстракция

"Что есть архитектура? Когда меня спрашивают, я знаю о чем идет речь, но стоит мне начать объяснять, я не знаю, что и ". Так сказал бы бл. Августин, будь он жив в наши дни.

Существует комплексное определение архитектуры по Одинцову, достаточно точное и очень скучное. В рамках этой статьи предлагаю использовать договоренность о том, что архитектура - это набор команд и внутренних правил той или иной аппаратной платформы.

Представьте себе, что некоторый естественный язык является платформой со своей архитектурой. Тогда на определенной стадии абстракции (читай: медитации) можно сказать, что есть, например, русская и английская языковая архитектура.

Квинтэссенция

В книге "Алиса в стране " есть отрывок, наталкивающий на размышления: а можно ли перевести эту книгу на русский язык? Вот этот отрывок:

"My tale is long and sad," began the Mouse.

"Oh, yes, it's a long tail," said Alice and looked at the Mouse's tail. "But why do you call it sad?"

"Oh, you are not trying to understand me. You are a very silly girl!"

And the Mouse got up and walked away.

С точки зрения программиста - этот кусок текста - ассемблерная вставка. Ассемблерные вставки жестко привязаны не только к аппаратной платформе, но и к операционной системе

Шутливые размышления о кроссархитектурности

<http://www.osrc.info/plugins/content/content.php?content.91>

и даже к компилятору. Именно из-за этого любой здравомыслящий программист прежде чем
вставить ассемблерный код в свою программу десять раз подумает. Действительно,
попробуйте-ка найти в русском языке два созвучных слова, одно из которых означает
"история" или "", а другое - "". Но ведь перевели! И вот как это ухитрились сделать:

- Я расскажу. Но только продолжение мое очень длинное и печальное, - сказала Мышь, повернувшись к Алисе и вздыхая.

- Это длинное продолжение, несомненно, - заметила Алиса, глядя с удивлением вниз, на мышиный хвост. - Но почему вы называете его печальным?

Да, был искажен текст, но остался СМЫСЛ, осталась ИДЕЯ! И я не знаю, насколько нужно быть извращенно-педантичным, чтобы отказаться от хорошего замысла в угоду точности.

А хорошая ИДЕЯ - это уже искусство. А искусство не знает границ. Стихотворения, поэмы - все это так тесно связано с конкретным языком понятиями рифмы и слога; но переводятся и будут переводиться! Возьмем, например, популярное английское стихотворение про Шалтая-Болтая:

Шалтай-Болтай сидел на стене.
Шалтай-Болтай свалился во сне...

В оригинале это выглядит так:

Humpty Dumpty sat on a wall.
Humpty Dumpty had a great fall,...

Дословным такой перевод не назовешь, но как красиво передан СМЫСЛ!

Отсюда вывод: искусство не может быть привязано к конкретной архитектуре. Вы можете написать сверхадаптированный к другому языку рассказ, избегая двусмысленности национальной привязанности; но если такой рассказ не несет в себе ИДЕЮ, то и перевод вряд ли понадобится. С другой стороны - поэмы, вдоль и поперек пестрящие национальными фразеологизмами и фолклорными подтекстами, - переводятся на сотни языков мира и читаются миллионами людей, удовлетворяя их естественную потребность в искусстве.

То же самое происходит и в мире программного обеспечения. В книге Л. Торвальдса и Д.

Шутливые размышления о кроссархитектурности

<http://www.osrc.info/plugins/content/content.php?content.91>

Даймонда "Just for fun" можно найти интересную цитату Линуса в конференции относительно новоиспеченной операционной системы:

P.S. Она свободна от кода minix и включает мультизадачную файловую систему. Она НЕ переносима (используется переключение задач 386 и пр.) и, возможно, никогда не будет поддерживать ничего, кроме AT-винчестеров - потому что у меня больше ничего нет :(

Чуть позже:

Как я уже писал, она использует MMU как для страничной подкачки (пока не на диск), так и для сегментации. Именно из-за сегментации она ПО-НАСТОЯЩЕМУ зависит от 386-го

Потом была знаменитая виртуальная перепалка Торвальдса с Э. Таненбаумом. В этом споре Таненбаум утверждал, что его Minix много лучше Linux, потому что Minix мало зависит от конкретной архитектуры и легко переносится. Прошло почти 14 лет. Бэстселлер Таненбаума "Современные операционные " вдоль и поперек утыкан словом "Linux", как образцом для разработчиков операционных систем. А команда `"ls /usr/src/linux/arch"` выводит следующее:

alpha/ i386/ ia64/ ppc/ um/ x86 64/

Linux используется в гоночных автомобилях, мобильных телефонах и т. п., а кроссархитектурная Minix если и используется, то лишь по своему прямому назначению: в учебных целях. Это яркий пример того, как отличная ИДЕЯ победила предрассудки.

В заключение хочу упомянуть фундаментальные алгоритмы. Вряд ли кто-нибудь будет оспаривать то, что они будут выполняться на любой вычислительной машине, отвечающей принципам фон-Неймана. А все оттого, что такие алгоритмы несут в себе **ХОРОШУЮ ИДЕЮ!**

