

Главное условие существования свободного программного обеспечения сегодня, как и всегда, это люди, которые готовы бесплатно делиться текстами своих программ и совершенствовать чужие. Свободное программное обеспечение унаследовало модель открытой научной разработки, а вместе с ней академическую модель взаимодействия между учёными. Модель эта инициировала организацию сообщества разработчиков и пользователей.

Взаимопомощь

У каждого пользователя любого программного обеспечения рано или поздно возникают вопросы, когда он пытается применить его для решения собственных задач. Пользователь несвободных программ платит за неё производителю. Тот, в свою очередь, иногда предоставляет пользователю некоторые гарантии, одна из которых — техническая поддержка данного программного продукта. Для этого организуется служба технической поддержки, которая по телефону, электронной почте и другим средствам связи отвечает на вопросы пользователей и помогает решить те или иные проблемы, возникающие с их программами.

Пользователь свободно распространяемой программы не получает никаких гарантий: автор создал её исходный текст открытым для общества, но при этом не взял на себя никаких обязательств. Стоит оговориться, что несвободное программное обеспечение в 99 % случаях тоже без гарантий. Так как сообщество пользователей большинства программ разбросано по всему миру, для взаимодействия в нём организуют форумы и другие средства общения в Интернете. Для накопления и категоризации информации по программе (списков часто задаваемых вопросов (FAQ — frequently asked questions) и других структур), а также организации совместной разработки, создаются веб-сайты, посвящённые программам.

Исправление ошибок

Практически в каждой достаточно сложной программе обязательно есть ошибки и дефекты, количество которых почти всегда неопределимо. Крупные производители программного обеспечения организуют для этого отдел контроля качества (QA — Quality assurance), который контролирует соответствие процессов разработки программ требованиям стандартов. Выполнение этих требований позволяет снизить вероятность появления ошибок в программном обеспечении. Однако, методы, позволяющие гарантировать отсутствие ошибок в сложных программах, в настоящее время отсутствуют. Для определения сложности программного обеспечения существуют формализованные критерии сложности.

Пользователь программы с закрытым исходным кодом, столкнувшись с ошибкой, не всегда может понять её причину и уж тем более исправить ее. Недоступны ни исходные коды программы, ни отладочная информация. Самое большое, что может сделать пользователь в рамках закона, это описать ошибку и условия, в которых она возникла.

Пользователь может обратиться всё в ту же службу технической поддержки, и, если там решат, что ошибка действительно в программе, а не в действиях пользователя, о ней будет сообщено разработчикам программного обеспечения.

Пользователь может долго ждать исправления ошибки в новых версиях программы. Нередко такое обновление приравнивается производителем к приобретению новой копии, что влечет за собой соответствующие материальные убытки и позволяет говорить о нарушении закона о защите прав потребителей. Диагностика ошибки, возникшей на машине пользователя, — не простая задача, ведь у сотрудников службы технической поддержки (и тем более программистов) нет доступа к пользовательскому компьютеру. Поэтому сейчас широко практикуются программы, собирающие разнообразную информацию о конфигурации компьютера пользователя и, часто, отладочную информацию. Эти данные можно отправить производителю для решения возникшей неполадки.

У традиционной свободной программы обычно нет оплачиваемого отдела контроля качества. Пользователь может столкнуться с ещё большим количеством ошибок, чем в коммерческой проприетарной программе. Тем важнее для него иметь возможность сообщить об ошибке разработчикам. Раньше в сопровождающей программу документации было принято указывать электронный адрес, на который можно было отправлять сообщения об ошибках. Некоторые разрабатывали форму для таких сообщений, чтобы автоматизировать их обработку. Уже это требует существенно более высокой степени коммуникации мирового сообщества, существенно большего, чем достаточно для закрытых программ.

Разработчики и контролёры частного программного продукта могут приходить на службу в один и тот же офис и там общаться. Такая организация труда эффективна, если круг разработчиков невелик, а организовать общую дисциплину относительно легко. Для открытого проекта круг и взаимное расположение потенциальных разработчиков ограничены только рамками планеты, поэтому эффективность разработки таких продуктов в значительно большей степени зависит коммуникаций

между разработчиками и «сознательности» пользователей.

Для упрощения и оперативной обработке сообщений об ошибках были созданы системы отслеживания ошибок (bug tracking system). Самые популярные из них разрабатывались для собственных проектов, но благодаря свободным лицензиям теперь используются повсеместно. Это такие системы как GNUTS (разработанная в GNU), Bugzilla (Mozilla Foundation), JitterBug (проект Samba), Debian BTS. Более ранние версии, как правило, используют для отправки сообщений электронную почту, более современные включают в свой функционал веб-интерфейс. Например, при помощи Bugzilla организуется сайт в Интернете, на котором пользователь может заполнить специальную форму сообщений об ошибках. Каждое сообщение имеет уникальный идентификатор, по которому можно попасть на страницу со всей информацией, связанной именно с этой ошибкой. Можно оставлять комментарии и скриншоты, а также получать на email сообщения, отражающие ход исправления ошибки.

Серьезным преимуществом пользователя свободного программного обеспечения является в возможности посмотреть исходный код программы. Если пользователь разбирается в программировании, зачастую, удастся самостоятельно обнаружить ошибку и даже исправить ее. В случае заинтересованности в развитии программы, пользователь может сообщить разработчику об ошибке и прислать ему свой вариант исправления.