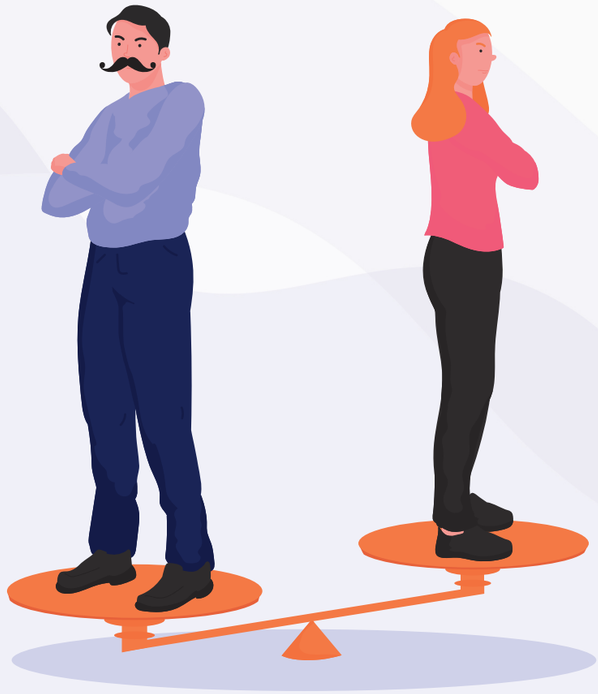
Vendor VS Inhouse, или Сами с усами

Для какого типа задач в банке выгоднее использовать внутреннюю разработку программных решений, а для каких — внешнюю? Что необходимо для самостоятельной разработки?



Текст
ВАДИМ ФЕРЕНЕЦ,
ОБОЗРЕВАТЕЛЬ **«Б.О»**



«Инхаус-разработка была временным решением, которое по каким-то причинам очень прижилось», — сделал довольно резонансное заявление в ходе конференции CNews «Цифровизация финансового сектора 2024» **Станислав Тульчинский,** управляющий директор департамента информационных технологий Россельхозбанка (РСХБ).

Слово «прижилось» имеет и численное выражение. **Марианна Данилина,** руководитель управления исследований и аналитики Ассоциации «ФинТех», выступая чуть позже, привела статистику: «По нашим данным, несмотря на кадровый голод, в крупных банках штат IT-подразделений в 2022 году составлял около 13% общей численности сотрудников, а в 2023 году — уже 17%. Причина — развитие собственной разработки».

Обнуленный open source

Как известно, нет ничего более постоянного, чем временное. Или все же на этот раз сработают объективные факторы, которые качнут маятник в обратную сторону?

А когда маятник, собственно, качнулся изначально в сторону инхаус-разработок? Общемировым трендом это стало, когда благодаря гибким методологиям, например Agile или Scrum, банкам удалось без оглядки на возможности вендоров резко сократить время time to market для новых продуктов, а сам набор этих продуктов увеличить за счет интеграции с нефинансовыми компонентами. Банкиры тогда даже заговорили о том, что они становятся IT-компаниями.

Вторым трендом, но уже с отечественной спецификой, стал процесс импортозамещения, прежде всего банковских core-систем. Специфика состояла в том, что некоторые модули АБС и сопоставимые с ней по сложности IT-решения крупные банки начали писать сами, не найдя ничего подходящего на рынке. А вот базы данных и операционные системы создавать с нуля никто решился, поэтому виден взлет отечественных вендоров, которые в свое время вложились в развитие «тяжелых» решений на базе open source.

Но этот этап обретения технологической независимости проходит. На повестке дня — создание решений, которые принято называть Applications (прикладные приложения среднего уровня): CRM, электронного документооборота, офисных компонентов, многочисленной обвязки вокруг АБС и аналитических платформ.

На этом этапе в экспертной среде стали слышны голоса практиков, говорящих о том, что написание ПО — это вообще-то бизнес, во главе которого находится задача создания надежного и безопасного кода промышленным способом. Это означает, что инхаус-разработчики кода столкнулись сразу с двумя мощными рыночными силами.

Во-первых, стоимость разработки одного прикладного решения, пусть даже со множеством непрерывных бизнес-доработок, выходит запредельно дорогой и неокупаемой. Это происходит из-за дороговизны труда программистов, введения Банком России требований по безопасной разработке ПО (БРПО), что обнуляет «бесплатность» open source, а также в большинстве случаев из-за невозможности реализации копий решения иным заказчикам в силу отсутствия рыночных компетенций по поддержке и интеграции, в том числе с государственными API.

Во-вторых, слой прикладного ПО — вотчина вендоров, которые занимаются этим не одно десятилетие и приучили пользователей к особенностям своих продуктов, включая интерфейсы. Вокруг них сложились мощные сообщества, в которых можно быстро найти экспертов самых разных направлений. А главное, их продукты дешевле собственных, особенно если речь идет о «коробках» или элементах конструкторов, из которых можно быстро и незатратно собрать удовлетворяющие регуляторов с точки зрения ИБ и рисков продукты.

Аргументы «за»

Оценив все плюсы и минусы самостоятельной разработки, Станислав Тульчинский пришел к выводу: «Самостоятельная разработка наиболее подходит для крупных компаний с соответствующими амбициями и бюджетами, организаций со сложной, гибкой, уникальной бизнес-моделью или в очень сложном и изменчивом бизнес-окружении. Выгодна она также организациям, которые строят свое конкурентное преимущество в том числе на обладании IT-компетенциями или считают, что нашли уникальную нишу, не занятую никем».

Таким образом, можно привести обобщенные выводы о преимуществе собственной разработки ПО.

- **Профессиональный контроль.** Разработчики имеют полный контроль над процессом разработки ПО, что позволяет им настраивать процесс с учетом специфики проекта и мгновенно реагировать на изменения, вносить коррективы и оптимизировать процесс.
- **Экономическая эффективность.** Самостоятельная разработка ПО позволяет сэкономить средства, которые обычно уходят на оплату услуг сторонних разработчиков. Кроме того, в долгосрочной перспективе компания сохраняет контроль над продуктом и не зависит от внешних поставщиков услуг.
- **Уникальность продукта.** При самостоятельной разработке ПО можно создать уникальный продукт, полностью соответствующий потребностям и требованиям компании.

Но у самостоятельной разработки ПО есть и недостатки.

- **Ограниченные ресурсы.** Самостоятельная разработка ПО требует значительных временных и финансовых ресурсов. Компания должна иметь квалифицированных специалистов и поддерживать их квалификацию, обеспечивать необходимое оборудование и программное обеспечение.
- **Риск срыва сроков.** Самостоятельная разработка ПО может вызвать срыв сроков из-за нехватки ресурсов или непредвиденных технических проблем, что отразится на успехе проекта.
- **Отсутствие опыта.** В случае отсутствия опыта разработки ПО у команды разработчиков самостоятельная разработка может привести к ошибкам и замедлению процесса разработки.

Деньги-то важнее

А какие аргументы можно услышать со стороны вендоров? Наверное, главное, что стало звучать публично, — практически ни одно вендорское решение сегодня, в условиях непрерывного изменения регуляторных требований, не способно полностью удовлетворить уникальные потребности конкретного банка. Всегда необходимы дополнительные настройки и доработки, а значит, времена «бронзовых» вендоров прошли, даже самые именитые бренды готовы к активному диалогу с заказчиком. Да, остались некоторые пока незаменимые решения на базе аналитики SAS или разработчиков западного процессингового ПО. Но исключение только подтверждает правило. Это, во-первых.

Во-вторых, стало заметно, что это изменения, связанные с тестированием ПО. Локальное самописное решение так или иначе можно пропустить через систему статических и динамических анализаторов, замерить производительность под нагрузкой и так далее, а вот как могут выкрутиться внутренние разработчики, если тестового стенда нет, а решение уже есть? Ждать, когда отдел закупок приобретет и привезет из Китая нужно количество оборудования? А как же «священная корова» в лице time to market? Выход уже давно найден в виде использования аутсорсинга и (или) облачных сред. А то, что это стало востребованным, видно по двузначным числам роста (в процентах) услуг независимых ЦОД.

Наконец, уж что-что, а экономику производства ПО вендоры научились считать лучше банкиров. **Наталия Оржевская,** директор Центра управления производственными командами компании «Диасофт», на конференции «Заказная разработка ПО 2024», в частности, заявила: «В последнее время мы занимаемся тем, что помогаем командам разработки быть эффективными и выдавать понятный объем работ. Мы научились измерять процессы производства на всех этапах работы команд: от написания концепции до развертывания решения на стороне заказчика. В итоге имеем управляемое сокращение стоимости разработки».

Что в финале? Похоже, экономические доводы об эффективности различных способов разработки ПО сводятся к аргументам в пользу вендоров. А вот антропологические показатели востребованности все новых и новых финансовых продуктов падают, люди не успевают их даже опробовать. Похоже, наш маятник застрял где-то посередине. Б.О