

09 декабря 2015

Типичные ошибки проектов по автоматизации систем бюджетирования

Создание автоматизированной системы планирования, бюджетирования и прогнозирования - сложный процесс, на успех которого влияет множество факторов. К сожалению, зачастую даже самые продуманные «на бумаге» модели в ходе реализации не достигают поставленных целей. В материале для портала Astera Максим Кудряшов, консультант-эксперт департамента CRM ГК «КОРУС Консалтинг», раскрывает наиболее распространенные ошибки, характерные для проектов по автоматизации систем бюджетирования.

Как "театр начинается с вешалки", так и эффективное управление любой организацией начинается с построения прозрачной системы бюджетов и прогнозов. Конечно, одно лишь наличие системы планирования и бюджетирования отнюдь не гарантирует достижения целей компании и финансового успеха, однако отсутствие плана почти всегда гарантирует провал. Построение автоматизированной системы планирования, бюджетирования и прогнозирования - очень сложный процесс, на успех которого влияет множество факторов. К сожалению, зачастую даже самые продуманные "на бумаге" модели в ходе реализации не достигают поставленных целей. Опираясь на опыт в сегменте CPM (Corporate



Performance Management), в данной статье мы попытались раскрыть наиболее распространенные ошибки, характерные для проектов по автоматизации систем бюджетирования, и обозначить их возможные пути решения. **Проблема 1: излишнее внимание к деталям** Слишком много внимания посвящено сбору данных на очень детальном уровне, что вызвано желанием получить как можно более точный прогноз. К сожалению, подобный подход приводит как раз к обратному - огромный объем данных ручного ввода снижает точность прогноза, сильно замедляет процесс сбора бюджетов, делает его подверженным человеческим ошибкам, крайне негибким и неэффективным. Казалось бы, тут возникает парадокс - почему планирование на более детальном уровне приводит к снижению точности? Дело в том, что большие модели зачастую теряют прозрачность, возникает проблема отслеживания взаимосвязей между бюджетами, при вводе данных допускаются ошибки. Да и при проектировании чересчур сложных моделей вычислений сложно обойтись без погрешностей, что вкупе с непрозрачностью модели приводит к потере доверия пользователей к полученным данным. Сбор детальных данных занимает слишком много времени, тем более что в процессе регламентного согласования данных пользователям приходится по несколько раз корректировать весь объем информации. В результате львиная доля времени уходит не на анализ информации и формирование управленческих решений, а на рутинные операции по сбору и согласованию данных. Решение: годовой бюджет должен содержать минимально необходимый объем информации, способный обеспечить возможность формирования гибкой системы бюджетов. Не следует пытаться планировать в той же детализации, которая свойственна набору данных по фактическому сценарию. Фактические данные

могут собираться по своим правилам и по более детальным аналитикам - это не противоречит данной концепции. Если для анализа плановых данных необходима большая детализация, то следует придумать механизмы распределения или расчета данных с помощью драйверов. **Проблема 2:**

отсутствие гибкости бюджетов Отчасти данная проблема связана с предыдущим вопросом - излишней детализацией данных. В ситуации, когда большие массивы данных планируются вручную, система становится слишком неповоротливой и негибкой. При необходимости составить новую версию бюджета, например, подготовить план развития событий при пессимистичном сценарии, пользователям приходится вручную корректировать все показатели. Мало того что этот процесс занимает слишком много времени, так еще и приводит к сильному снижению точности прогнозов. Решение: необходимо обеспечить максимальную параметризацию модели - все показатели, расчет которых можно алгоритмизировать, должны считаться системой автоматически. Можно спроектировать специальную настроенную таблицу, в которой для каждого показателя можно указать драйвер для расчета, например, процент от выручки или фактические данные, умноженные на коэффициент инфляции. Данные для новых периодов планирования не должны заполняться с нуля вручную (так называемый Zero-Based Budgeting), а должны инициализироваться специальным правилом в системе на базе набора предпосылок. **Проблема 3: излишний фокус на годовом бюджете**

Большинство организаций до сих пор фокусируют свои усилия на сборе годового бюджета, формирование которого может занимать до нескольких месяцев и требует значительных трудозатрат ключевых специалистов.

Проблема в том, что в тот момент, когда бюджет, наконец, полностью сверстан

и утвержден, предпосылки, которые использовались в нем, могут настолько устареть и измениться, что весь бюджет к этому моменту не будет представлять собой практически никакой ценности. Какой смысл проводить план-факт анализа отклонений, если уже на этапе утверждения бюджета всем понятно, что фактические результаты будут сильно отличаться от заложенных в бюджете? Решение: для решения этой проблемы некоторые компании применяют метод скользящего планирования. Скользящий бюджет - это бюджет, который подвергается регулярным корректировкам по прошествии периода путем добавления к периоду планирования одного интервала времени и вычитания одного истекшего периода. При таком подходе система бюджетов постоянно поддерживается в актуальном состоянии, а применение гибкой модели планирования позволяет автоматически пересчитывать все данные в системе на базе обновленной информации по факту и внесенных пользователями корректировок ключевых показателей модели (например, макроэкономических показателей). Применение скользящего планирования позволяет компаниям создавать постоянное "окно" в будущее шириной в интервал прогноза. Если говорить о системах бюджетирования, то некоторые продукты, например, Oracle Hyperion Planning, полностью поддерживают создание скользящих моделей планирования, причем позволяют максимально упростить поддержку системы - администратору модели достаточно раз в месяц изменить в настройках горизонт периодов планирования. **Проблема 4:**

нереалистичные рамки проекта Зачастую из-за желания автоматизировать сразу все бизнес-процессы бюджетирования и сделать это максимально детально, используя чуть ли не все доступные инструменты программного обеспечения, возникает "размытие" задач. Становится сложно отличить

наиболее критичные требования от второстепенных. При этом сроки проектов всегда ограничены, начинается спешка, попытка успеть реализовать сразу все, в результате чего страдает качество итогового решения. Решение: необходимо разделять требования на критичные и "было бы неплохо реализовать", и начинать проект с реализации и тщательного тестирования первой категории задач. Далее уже можно приступать к следующей фазе проекта, в ходе которой внедрять "красивости" и менее критичные бизнес-требования. По данной проблеме можно добавить еще один важный момент: нельзя составлять план проекта таким образом, чтобы передача любой важной функциональности на тестирование заказчику осуществлялась в конце этапа разработки. Такой подход - это залог неудачи и испорченных отношений между разработчиками и бизнесом. Разработка любого блока должна вестись итерационным способом, чтобы в ходе каждой итерации происходил совместный просмотр настроенной функциональности и была вовремя получена обратная связь. **Проблема 5: отсутствие требуемых фактических данных** Редко бывают ситуации, когда в организации уже есть полный набор фактических данных, полностью удовлетворяющих требованиям спроектированной системы бюджетирования. Чаще всего в ходе проекта формируются новые требования к данным, в результате чего иницируются новые проекты по доработке существующих систем, являющихся источниками данных для формирования планов и прогнозов. Доработка учетных систем может занимать весьма длительное время, которое превышает сроки, отведенные на внедрение системы бюджетирования. При этом сформировать достоверный план или прогноз без наличия фактических данных практически невозможно. Решение: о проблеме получения фактических данных необходимо задуматься

в самом начале проекта. Как минимум, сразу можно определить перечень информационных систем, из которых необходимо будет забирать данные, и определить, кто будет отвечать за настройку интеграционных связей. Как только станут понятны общие контуры бюджетной модели, необходимо составить требования к фактическим данным и инициировать процесс доработок систем-источников информации. **Проблема 6: спешка в подключении пользователей к системе** Тщательное тестирование настроенной системы бюджетирования силами как самих разработчиков, так и команды ключевых пользователей заказчика крайне важно - его нужно осуществлять прежде, чем будет принято решение подключать всех остальных пользователей. Первое впечатление - наиболее важное, и ни в коем случае нельзя допустить любого негатива от пользователей в ходе первого запуска системы. Проблема тут заключается еще и в том, что зачастую в плане проекта на финальное тестирование отводится не так много времени. Вкупе с тем, что в ходе проекта часто появляются все новые и новые требования, не зафиксированные в ходе предпроектного интервьюирования, это приводит к снижению качества процесса тестирования. Решение: при составлении плана проекта нужно резервировать дополнительное время на стадию тестирования. Заранее обговаривать с заказчиком условия таким образом, чтобы на этапе тестирования все его ключевые пользователи, участвующие в проекте, подключались к процессу максимально плотно, не отвлекаясь на решение текущих задач.