**Концепции построения ERP-систем на предприятии**

Румянцев Константин, Директор ООО "Софтоматика"

В этой статье пойдет речь о концепциях построения ERP-систем на предприятиях, а именно имитационной и аддитивной, причем под ERP-системами будем понимать не столько планирование именно ресурсов, сколько создание корпоративной информационной системы на предприятии.

Для чего вообще нужны концепции?

Попробуем ответить на этот вопрос другим вопросом: каким Ваше будет предприятие через 10 лет? Какие свойства ERP системы будут тогда востребованы? В общем, никто не сможет четко ответить на задачу с такой отдаленной перспективой. При этом предприятие должно быть уверено, что и в отдаленном будущем имеющееся у него программное обеспечение (ПО) будет способствовать развитию бизнеса, и только такое ПО востребовано. Так вот, для того, чтобы конструировать систему, в т.ч. для целей, которые неясны, и задач, появление которых мы не можем предвидеть, и существуют концепции.

Пример: Начинаем автоматизировать маленькое предприятие, в нем три человека, один компьютер. Количество клиентов - 10, наименований товара - 20. Вопрос - какую архитектуру данных выбрать? Обратимся к концепции. Во-первых, определимся, что предприятие будет развиваться (т.е. мы его не закроем через месяц). Во-вторых, отрасль, в которой мы сейчас работаем - это тысячи наименований, хотя на данный момент у нас всего 10 (например, мы продаем исключительно мыло, видов мыла у поставщика может быть 100, ну 200 максимум, однако, если смотреть шире - зубная паста, щётки, гели и т.п., что называется хозтоварами, а это уже десятки тысяч наименований). Следовательно, исключить то, что мы будем представлять весь ассортимент, мы не можем так же как нельзя исключить, что со временем предприятие превратится в большую структуру. Тогда добавляем в концепцию постулат, что наша ERP-система должна поддерживать произвольное количество наименований товара и рабочих мест, плюс должна быть нетребовательна к сети. Иначе, как создавать филиалы, если каждый из них потребует прокладки высокоскоростной сети на многие километры? Теперь очевидно, что самая подходящая архитектура - Клиент-Сервер. Вот и решение!

Рассмотрим внимательно, что мы сейчас сделали. Сперва мы определили некие общие параметры, которые присущи или будут в присущи будущем нашей системе (концепция), затем приняли решение относительно конкретного вопроса (выбрали архитектуру данных). В дальнейшем, во время решения конкретных задач, мы должны будем применять такие методы, которые бы не противоречили концепции. Заметьте: мы уже решаем задачи, о которых еще ничего не знаем.

Имитационная концепция: имитирует реальное строение предприятия.

ERP системы, построенные в рамках имитационной концепции, повторяют (имитируют) реальное строение предприятия с выделением модулей, соответствующих отделам. Например, плановый, финансы, бухгалтерия, склад, продажи, логистика и т.п. Модули пишутся как отдельные задачи для автоматизации конкретных отделов, затем обмениваются необходимыми данными дабы увязать все в единую систему.

Аддитивная концепция: информационная система дополняет человека (отсюда и название - аддитивная).

В этом случае ERP системы строятся таким образом, что первоочередной задачей становится корректный сбор данных. Здесь задача разбивается на подзадачи в другой плоскости: корректный сбор данных, их правильное устройство и хранение, вывод данных и их анализ, предоставление информации в таком виде, который позволяет человеку легко принимать правильные решения (обратная связь). Понятно, что в этом случае информационные потоки существенно упорядочиваются. Принцип "Если данные устроены правильно, они могут быть выбраны любым желаемым образом" является основополагающим.

Как ведет себя энтропия при росте системы?

Давайте отвлечемся и посмотрим, для чего человек (человечество) все более специализируется и почему большие системы менее устойчивы, чем малые.

Во-первых, базовым понятием для систем вообще является ЭНТРОПИЯ, мера беспорядка. Чем выше организация, тем меньше энтропия. Критерием организованности является следующее определение: уровень организации тем выше (а энтропии тем меньше), чем больше свойства целого отличаются от простой суммы свойств. Т.е., если труд организован (энтропия минимальна), то два человека, работающие вместе, сделают больше, чем оба, но работая по отдельности. С этой самой энтропией мы и намерены бороться.

Во-вторых, основным средством для уменьшения энтропии является ИНФОРМАЦИЯ. Приведем пример: на складе есть товар, но продавцы не имеют информации об остатках на складе. Нет информации - нет отгрузок, система не работает, хотя и кладовщики, и продавцы готовы выполнить свои обязанности.

В-третьих, человек, существо хоть и разумное, но склонное к ошибкам. Оказывается, что корректно выполнить 100 простых операций проще, чем одну, но в 100 раз более сложную. Это свойство человека и заставляет людей специализироваться, а предприятия состоять из отделов, ибо быть специалистом по маркетингу еще можно, и то в какой-нибудь отрасли, а вот быть одновременно специалистом по маркетингу, документообороту и управлению очень трудно, даже невозможно.

В-четвертых, люди способны работать с ограниченным набором данных (объектов). В среднем, человек нормально воспринимает 5-9 объектов. Если объектов больше, человек гораздо чаще ошибается. Применительно к ERP-системам, это означает, что интерфейс пользователя должен содержать как можно меньше окошек с данными, опираясь на которые пользователь принимает решение.

Пример: Пусть предприятие продает леденцы. Пришел покупатель, говорит: вот деньги, хочу леденцов на 50 тыс. руб. Продавец видит остаток на складе и, если товара достаточно, то он отгружается; если товара нет, об этом сообщается клиенту, и как-то этот вопрос улаживается - это нормальная работа. Если же продавец видит остаток на складе на 1 число текущего месяца и движение по данному товару за две недели, то чтобы принять решение об отгрузке он должен будет вычесть всё, что напродавал он сам и его коллеги за эти две недели и прибавить то, что пришло за это время на склад. Представьте, сколько места на экране должны занимать все эти цифры! Причем продавцу придется вычислять количество леденцов либо в уме, либо на калькуляторе, но в любом случае, количество ошибок возрастёт многократно. Заметим, что и в первом и во втором примере информация, в общем, эквивалентна, но человеку гораздо проще работать, если все представлено в удобном виде.

Итак, мы выяснили, что люди организовываются в группы (системы) для того, чтобы их деятельность была более эффективной. Сложную задачу разбивают на более узкие и простые и эти подзадачи выполняют специалисты в своей области. Для того, чтобы каждый человек корректно выполнял работу на своем участке, ему необходима информация о состоянии дел во всей организации или её части. И, наконец, чтобы снизить ошибки, информация должна быть представлена в максимально удобном виде.

Теперь посмотрим, что происходит, если человеческая система начинает укрупняться. Представим одного рабочего. Допустим, он ошибается крайне редко и выпускает 1 бракованную деталь к машине на 1000 не бракованных. В общем, мелочь. Но как только узлов в машине становится порядка 1000, и их так же делают рабочие с таким же процентом брака, то окажется, что в каждой машине что-нибудь, да не работает. Этот пример говорит о том, что если такая система увеличивается, её энтропия тоже возрастает, и нужно принимать меры к её уменьшению. Иначе система просто умрет - кому нужны 100% бракованных машин?

Применительно к аддитивной и имитационной концепциям.

Посмотрим, как все это относится к нашим концепциям.

Имитационная. Предположим, есть абсолютно неавтоматизированное предприятие, оно состоит из отделов, сотрудники обмениваются бумажками, считают на калькуляторах и т.п. Вот мы решили автоматизировать это предприятие исходя из имитационной концепции. Для этого мы выясняем, что делает каждый отдел, и, таким образом, получаем некоторый набор задач по количеству отделов, которые и решаем. Хочу четко обозначить, что мы понимаем под словом "отдел". Совсем не обязательно, что это будет именно ОТДЕЛ, в том понимании, как это принято на предприятиях. Это могут быть и несколько отделов, и группы людей в разных отделах и подразделениях. Важно, чтобы была возможность сгруппировать задачу таким образом, чтобы при автоматизации предприятия она разбивалась на подзадачи. К примеру, если в предприятии 10 складов, то пишется 1 модуль под названием "Склад", ибо он решает задачу складского учета.

Как такая автоматизация влияет на работу предприятия?

Конечно, в определенной степени энтропия системы уменьшается. В первую очередь, за счет того, что компьютер - не человек, ему все равно, сколько цифр сложить (это не совсем так, но предположим, что у нас очень быстрый и стабильно работающий компьютер) и как долго хранить. Но посмотрим, как ведет себя информация в имитационной концепции: отделы не могут работать обособленно, и если раньше они обменивались бумажками, то теперь эту роль взяли на себя информационные потоки между модулями системы. По сути, энтропия перераспределилась в эти потоки. Если они хорошо налажены, система работает хорошо, энтропия уменьшилась, информации достаточно, чтобы принять правильное решение. И тем не менее...

Напоминаю, речь идет о человеческой системе. Причем это открытая система, которая завязана на состоянии рынка, поставщиках, клиентах, законодательстве, конкурентах и т.п. При таком количестве воздействий система не может быть не изменяться. Это значит, что налаженный вчера поток данных между модулями системы сегодня уже будет недостаточен и приведет не к уменьшению, а увеличению энтропии. Этот поток и так-то очень трудно наладить, всем знакома проблема со слабыми связями модулей ERP-систем, вплоть до того, что данные из одного модуля заносятся в другой руками. Так к чему может привести постоянное изменение информационных потоков?! Как видим, в имитационной концепции уменьшение энтропии на начальном этапе проектирования привело к её перераспределению в информационные потоки между модулями и к дальнейшей нестабильной работе системы.

Почему же очевидный на первый взгляд способ решать большую и сложную задачу по частям не гарантирует стабильной работы системы?

Чтобы ответить на этот вопрос, давайте рассмотрим предпочтительную и более современную аддитивную концепцию построения ERP систем.

Поставим задачу:

Необходимо, чтобы система "человек - компьютер" функционировала с минимальной энтропией.

Энтропия на уровне компьютера.

Компьютер обладает недостижимой для человека скоростью работы с данными, не ошибается, не забывает, делает в точности то, что ему велит программа, а это значит, что энтропия на стороне компьютера минимальна (разумеется, если все корректно работает). В общем, компьютер - это инструмент и ему все равно, какие данные с чем складывать, какие делать выводы, как и что анализировать. Для него существуют только инструкции, которые он выполняет. И если для человека выполнять разные по характеру виды работы - проблема, то компьютеру все равно, была бы соответствующая программа. Из всего этого делаем вывод - компьютер требует принципиально иного к себе подхода, чем человек, для компьютера разбивать задачу на подзадачи не нужно, компьютер - это инструмент, он дополняет работу человека.

Энтропия на уровне человека.

Т.к. мы уже определили, что на уровне компьютера энтропия минимальна, то ответ однозначен - источником роста энтропии в системе является человек! Это он ошибается, не может быстро переключаться с одного дела на другое и плохо воспринимает более 9-ти объектов одновременно.

Как свести к минимуму человеческие ошибки (уменьшить энтропию системы)?

Во-первых, количество возможных решений должно быть максимально ограничено.

Давайте сравним количество информации, которую должен обработать продавец чтобы отгрузить товар с одного склада и с нескольких складов. В первом случае - это слежение за остатком на одном складе и доставка с одного склада клиенту, тут все просто и понятно. При отгрузке же с нескольких складов дополнительно возникает масса вопросов: как будет доставляться товар клиенту, с каждого склада отдельными машинами или сперва формировать заявку на каком-то одном складе, а потом доставлять? Одинаковы ли условия отгрузки на всех складах (возможно, один склад отгружает упаковками, другой штуками)? Как поступить, если большая часть товара для клиента есть на одном складе, а то, чего не хватило и есть на другом складе, стоит 100 руб. и есть ли смысл собирать и доставлять оставшееся количество товара через весь город и т.д. и т.п.? Можно быть уверенным, что количество ошибок существенно возрастет, и руководство предприятия должно решить для себя, что целесообразнее: увеличить продажи (что далеко не факт!) за счет резкого увеличения ошибок в системе (энтропия возросла!) или стабильность системы важнее? Нужно отдавать себе отчет, что каждое человеческое движение - это ошибки и количество возможных решений необходимо ограничивать.

Во-вторых, совершенно необходимо, чтобы принимая решение человек имел ПОЛНУЮ и АБСОЛЮТНО ДОСТОВЕРНУЮ информацию, которая ему необходима (помните, в имитационной модели именно неполные информационные потоки между модулями разваливали систему) и в наиболее удобном для него виде. И только общее информационное пространство способно предоставить такие гарантии.

Приведем еще один пример, когда маленькая задержка при передаче данных может обернуться бедой. Пусть в 1000 бухгалтерия распечатывает для своих продавцов дебиторскую задолженность по клиентам. После этого приходит клиент, который перечислил предоплату 100 тыс.р., и это отражено в бухгалтерском документе. Продавец, естественно, отгружает предоплаченный товар. Всё, казалось, хорошо. Но покупатель оказался жуликом, и через час идет к другому продавцу и получает товар еще на 100 тыс., т.к. продавец видит ту же распечатку, что и первый. Затем покупатель идет дальше - к третьему, четвертому и т.д. продавцам. В этом случае убытки могут оказаться весьма серьезными. Вот к чему может привести запаздывание информации всего на 1 день!

Итак, выпишем цепочку, при которой энтропия системы минимальна, а система, соответственно, наиболее стабильна и организованна.

Все просто: достоверная аналитическая информация - человек - правильное решение - корректное заведение данных по этому решению - хранение данных - аналитическая информация

Как видим, круг замкнулся и превратился в обозначенную ранее аддитивную концепцию.

В этой концепции уменьшение энтропии на уровне человека происходит за счет участков "аналитическая информация" и "корректное заведение данных". Очевидно, что при таком построении человеку легко принимать правильные решения, т.к. он имеет всю необходимую информацию. Неправильные же решения принимать становится труднее, т.к. при заведении данных он ограничен только корректными вариантами.

Тогда приведенный выше пример с непорядочным покупателем выглядел бы следующим образом. Пришел покупатель, продавец видит в компьютере сумму, на которую он может отгрузить клиенту (аналитическая информация) и принимает решение об отгрузке (корректное решение). Допустим, продавец недоглядел и пытается выписать товар на большую сумму. В этом случае компьютер предупредит продавца (аналитическая информация), а возможно и запретит такую операцию (корректное решение). Как только операция прошла, информационное поле, соответственно, изменилось, и если клиент попытается выписать товар у другого продавца, то тот, основываясь на правильных данных, взятых из уже измененного информационного поля, ему откажет (аналитическая информация).

Аддитивная концепция устойчива и эффективна при постоянно меняющихся внешних и внутренних условиях.

Пусть во внешней среде произошли некоторые изменения, и мы не можем прогнозировать, как они отразятся на конкурентоспособности системы. В рамках аддитивной концепции задача решается путем правильного сбора данных. Когда с данными все в порядке, то через произвольное время система может ими воспользоваться, что означает адекватную реакцию на изменение внешней среды. Заметьте, собирая данные, мы можем не предполагать, как они нам в дальнейшем понадобятся, но когда это необходимо, в системе уже все готово для их использования.

Пример: для того, чтобы увеличить свое присутствие на рынке, предприятие решило улучшить сервис при поставке товара клиентам. Для этого нужно точно планировать отгрузку товара, доставку, собирать информацию о конкретном заказе и много чего еще, что внешне позволяет сделать общение клиента с предприятием непринужденным, в то время как на самом деле внутри предприятия происходит масса процессов, каждый из которых норовит сорвать сделку. Но все это от клиента должно быть скрыто, нужна логистика.

Какие наши действия?

Во-первых, для того, чтобы отправить товар клиенту, нужно, как минимум, знать, что все этапы формирования заказа пройдены, т.е. должна присутствовать информации об исполнении каждого этапа.

Во-вторых, наличие информации о состоянии этапов сильно снижает вероятность где-нибудь ошибиться (например, отгрузить недоукомплектованный товар), а это, очевидно, снижает энтропию системы.

В-третьих, мы понятия не имеем, какие конкретно этапы мы будем отслеживать, и сколько их может появиться через год, следовательно, этапов должно быть произвольное количество.

В-четвертых, этапы должны относится к какому-либо документу (документ "заказ", этапы - сборка товара, его упаковка, отгрузка со склада, доставка до клиента; документ "счет" - этапы распечатан, отправлен и т.д.).

Далее, создаем таблицу видов этапов, таблицу в которой указывается, какой документ какие этапы содержит и таблицу, в которую заносим исполнение этапов при прохождении заказа. Естественно, все таблицы соответствующим образом связаны. Теперь система готова отследить движение каждого документа на каждом этапе его жизни, причем мы можем произвольным образом добавлять или удалять этапы, и если наши законодатели нас обяжут каждому клиенту выдавать по пирожку, то нам нужно будет лишь добавить в список видов этапов вид "пирожок", добавить его в этапы по заказу и контролировать процесс. Обратите внимание, не важно, какие именно этапы мы хотим отследить. Мы относимся к ним как к данным и просто храним их в нормализованном виде. В рамках аддитивной концепции совершенно простыми и красивыми действиями мы создали систему логистики, которая будет хорошо работать. В имитационной же модели нам пришлось бы создавать модуль логистики, настраивать обмен данными между остальными модулями и т.п., что отняло бы у нас кучу времени и стабильность системы осталась бы при этом весьма сомнительной.

Итак, в этой статье мы провели сравнение концепций построения ERP-систем. Автор намеренно не приводил конкретные примеры, читателю предлагается самостоятельно исследовать рынок корпоративных информационных бизнес-систем на предмет их принадлежности к той или иной концепции.