

Перспективы Workflow-систем. Паттерны – попытка навести порядок в «мире» workflow.

Андрей Михеев, Михаил Орлов,

В предыдущих публикациях мы рассказали, что такое WF-системы, охарактеризовали области их применения, а так же дали представление о «войне стандартов» в мире Workflow. Напомним некоторые положения, сформулированные в предыдущих статьях:

Деятельность большинства современных предприятий можно представить в виде набора бизнес-процессов.

Бизнес-процесс - это хронологически упорядоченный набор заданий, выполняемых как людьми, так и информационными системами предприятия, направленный на достижение заранее известной бизнес-цели за известное время.

Цель применения WF-системы - автоматизация бизнес-процессов предприятия.

WF-система предприятия должна решать две основные задачи:

Задача А. «Эсперанто Менеджмента» - Формирование единого языка описания бизнес-процессов для всех менеджеров предприятия, а также создание библиотеки бизнес-процессов предприятия.

Задача Б. «Универсальный Клей» - Быстрая интеграция («склеивание») в рамках единого процесса труда сотрудников и компьютерных систем предприятия. Быстрая сборка из разнородных "кирпичиков" связного процесса.

Данная публикация посвящена Workflow-паттернам (далее WF-паттерны, или просто паттерны), которые, по нашему мнению, могут стать «лакмусовой бумажкой» для оценки существующих WF-систем и WF-стандартов.

Зачем нужны WF-паттерны.

В настоящее время не существует универсальной общепризнанной теории, описывающей бизнес-процессы. Различными компаниями и сообществами разработано немало различных несовместимых друг с другом стандартов, таких как BPEL4WS, XPDL, BPML, WSCI, BPSS, YAWL...

Ситуация с программными продуктами тоже достаточно запутанная. Современные WF-системы в большинстве случаев соответствуют какому-либо из известных стандартов, однако также существует большое количество WF-систем, которые никаким стандартам не следуют.

В этих условиях Ван-дер-Альст, Хофстеде, Кипужевский и Баррос (<http://tmitwww.tm.tue.nl/research/patterns/download/wfs-pat-2002.pdf>) попытались разработать единую методологию систематизации и классификации WF-систем и WF-стандартов. Они проанализировали распространенные WF-системы и стандарты, выделили в них типичные элементы, выявили наиболее часто повторяющиеся структуры и

назвали их Workflow-паттернами (Workflowpatterns).

Согласно определению: **WF-паттерны - это характерные структуры внутри бизнес-процесса, рассматриваемые в его контексте.**

Видно, что определение содержит в себе субъективный элемент. Это не удивительно, т.к. для выявления WF-паттернов был применен феноменологический подход, в рамках которого набор WF-паттернов явно зависит как от исследователя, так и от набора исследуемых WF-систем. Тем не менее, в результате удалось выявить набор из двадцати WF-паттернов, на основе которых, как нам кажется, можно построить 99,99% всех реальных бизнес-процессов (см. www.workflowpatterns.com).

Паттерны помогают решить две задачи:

- Сравнить предлагаемые WF -стандарты и/или WF -системы. Составив таблицу, показывающую, какие паттерны в каких системах и стандартах поддерживаются, можно сравнить их друг с другом.
- Выбрать WF - систему для предприятия - определить множество паттернов, характерных для бизнес-процессов данного предприятия, далее сузить круг рассматриваемых систем, исключив те, в которых эти паттерны не поддерживаются.

Эволюция WF-паттернов.

К изначально предложенным 20 паттернам, относящимся в основном к управлению потоком (control-flowperspective), Ру, Маджинис и Браун (W. A. Ruh, F. X. Maginnis, W.J. Brown. Enterprise Application Integration: A Wiley Tekh Brief. JohnWileyandSons, Inc, 2001) добавили шесть дополнительных паттернов коммуникации, относящихся к взаимодействию WF-процессов и потоков управления.

Далее количество относящихся к workflowпаттернов стало стремительно увеличиваться - появились паттерны, характеризующие взаимоотношения продавцов-покупателей, паттерны данных, паттерны, связанные с транзакциями, и т.д.

С какого-то момента эти новые WF-паттерны фактически переходят в разряд design-паттернов объектно-ориентированных языков программирования.

И это не случайно. Мы считаем, что WF-языки - частный случай высокоуровневых языков программирования, а, следовательно, к ним должны быть применимы многие общепрограммистские подходы.

Наиболее распространенные WF-паттерны

Ниже мы приведем описание наиболее распространенных WF-паттернов вместе с их графическими образами. Вообще выбор графической нотации WF-процессов представляет собой сложную задачу и мог бы стать темой отдельной статьи. Мы остановились на нотации UML (www.uml.org), как наиболее распространенной и изученной.

«Последовательность». (Англоязычный термин -Sequence).

Простейший «элемент» бизнес-процесса: Два узла соединены переходом. После того, как исполнитель выполнил действие первого узла, управление переходит ко второму.



Рисунок 1. Паттерн «Последовательность»

«Параллельное расщепление». (Англоязычные термины - AND-split, Fork).

Представляет собой узел, в который приходит только один переход и из которого исходит два или более переходов. Причем после того, как управление передано узлу, поток управления бизнес-процессом распадается на несколько потоков, которые выполняются параллельно. Для каждого исходящего перехода должен существовать «свой» поток управления.

Данный паттерн часто используется в связке с паттерном «синхронизация».

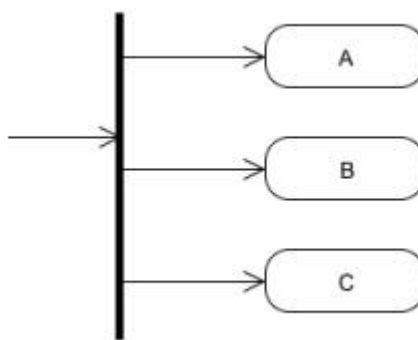


Рисунок 2. Пример паттерна «Параллельное расщепление»

«Синхронизация». (Англоязычные термины - AND-join, Join).

Узел, в котором соединяются два или более перехода, а выходит только один. Управление не будет передано дальше, пока все потоки управления бизнес-процессом (по количеству входящих переходов) не достигнут данного узла. После того, как в узел придут все потоки управления, будет инициирован только один поток, соответствующий исходящему переходу.

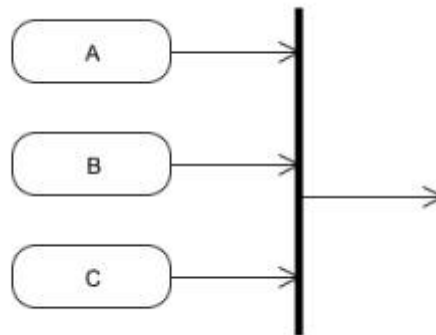


Рисунок 3. Пример паттерна «Синхронизация».

«Исключающий выбор». (Англоязычные термины - XOR-split, Decision).

Узел в графе WF-процесса, в который приходит только один и из которого исходит два или более переходов. Причем, после того, как управление перешло к данному узлу, в нем делается выбор, по какому из исходящих переходов оно будет передано далее.

Паттерн часто используется в «связке» с паттерном «простое соединение».

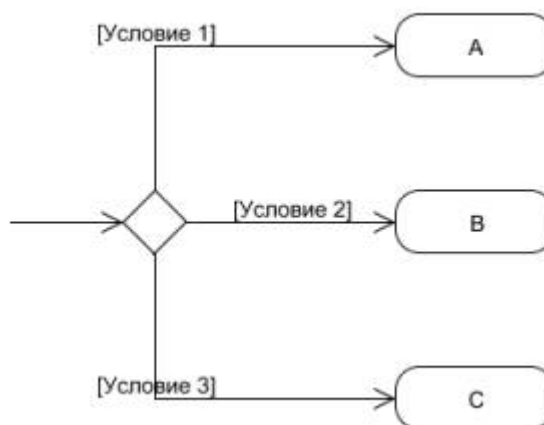


Рисунок 4. Пример паттерна «Исключающий выбор»

«Простое соединение». (Англоязычные термины - XOR-join, Merge).

Узел в графе WF-процесса, в котором соединяются два или более перехода, а выходит только один. После того, как в узел пришло управление от любого из входящих потоков, поток управления передается на единственный исходящий переход.

Предполагается, что управление может прийти в узел только по одному из входящих переходов.

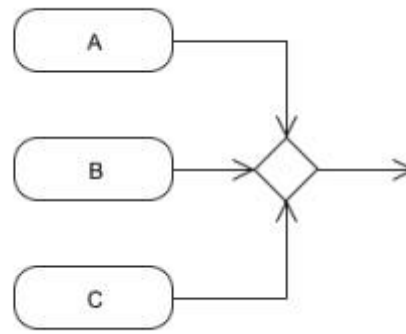


Рисунок 5. Пример паттерна «Простое соединение».

Пять вышеописанных паттернов поддерживаются всеми известными нам WF-стандартами и WF-системами и являются основой описания абсолютного большинства существующих WF-процессов. Однако в некоторых достаточно редких бизнес-процессах используются и паттерны, включенные в группу «дополнительных». Эти паттерны либо нельзя свести к комбинации и суперпозиции основных, либо при таком сведении получаются крайне громоздкие, неудобные для восприятия схемы. Для реализации подобных паттернов нужны дополнительные элементы WF-языков и, соответственно, дополнительная функциональность WF-систем. Приведем несколько дополнительных паттернов:

«Дискриминатор». (Англоязычный термин - Discriminator).

Узел в графе WF-процесса, в котором соединяются два или более перехода, а выходит только один переход. Как только узлу передается управление (по любому из входящих переходов), сразу активизируется исходящий переход, то есть начинает выполняться следующее действие бизнес-процесса. Выполнение других активных потоков не прерывается, однако передача управления каждого из этих потоков в узел дискриминатора игнорируется и пришедший поток завершает свое существование.

Данный паттерн часто используется в связке с паттерном «Параллельное расщепление».

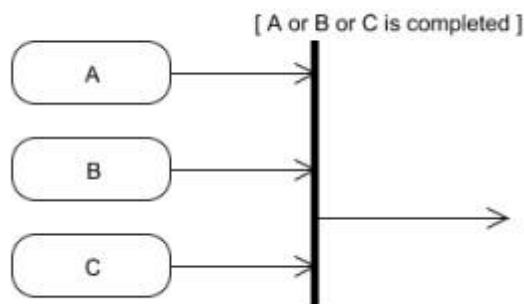


Рисунок 6. Пример паттерна «Дискриминатор».

Пример. Бизнес-процесс «Подбор ключа криптозащиты». Для проверки стойкости криптозащиты часто применяются параллельные вычисления, построенные по следующему принципу: каждому участнику выдается пакет исходных данных, причем (при большом количестве участников) эти пакеты не велики и обсчитываются за несколько дней. Как только один из участников находит ключ, он должен его сразу

прислать в исследовательский центр. Результат работы остальных участников может быть проигнорирован.

«Произвольный цикл». (Англоязычный термин - Arbitrary cycle).

Набор узлов в графе WF-процесса, в котором один или несколько узлов могут проходиться многократно, то есть - в эти узлы много раз может приходиться (и, соответственно уходить) управление. Причем, в отличие от регулярных циклов, точки «входов» в набор повторяющихся узлов (или «выходов» из них) могут быть различными.

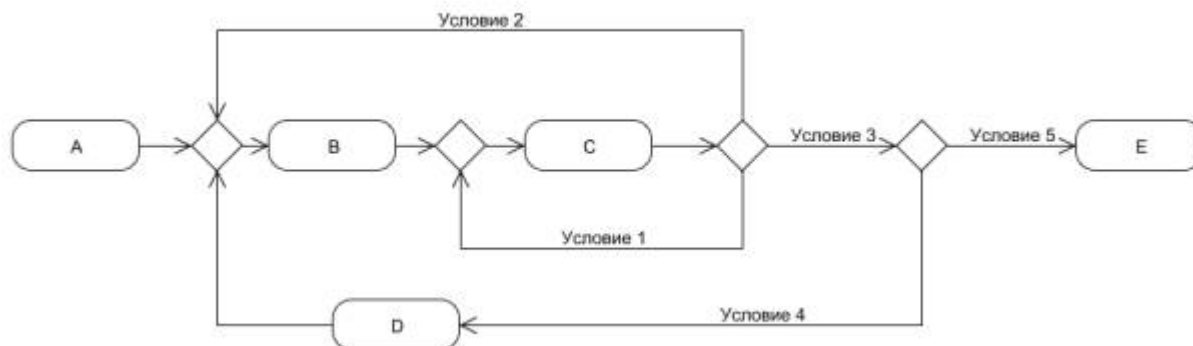


Рисунок 7. Пример произвольного цикла.

«Отложенный выбор». (Англоязычный термин - Deferred choice).

Узел в графе workflow-процесса, в который приходит только один и из которого исходит два или более переходов. После прохождения узла поток управления бизнес-процессом распадается на несколько потоков, число которых равно количеству исходящих переходов. Далее все потоки, соответствующие исходящим переходам становятся активными. Однако после того, как первый узел для исполнения выбран, активным остается только поток этого узла, а все остальные принудительно завершаются.

Паттерн часто используется в связке с паттерном «Простое соединение».

Пример. По электронной почте приходит письмо от клиента с запросом на поставку товара. Это письмо отправляется всем менеджерам отдела продаж. Тот менеджер, который первым подтвердит желание работать с данным заказом, и будет его исполнять, у остальных менеджеров запрос будет отозван.

Применение паттернов для сравнения различных WF-языков.

Авторы данной концепции произвели классификацию наиболее известных WF-языков относительно возможности «прямой» (в оригинале - «direct») поддержки WF-паттернов. По нашему мнению, произведенная ими классификация весьма субъективна и сильно зависит от определения понятия «прямая поддержка». Во многих случаях, когда по их мнению, тот или иной стандарт не обеспечивает «прямой» поддержки данного паттерна, на самом деле существует синтаксически допустимая конструкция языка, реализующая паттерн, но кажущаяся им слишком громоздкой (что в ряде случаев является спорным). В таблице 1 приведены опубликованные авторами концепции результаты сравнения трех WF-стандартов. (В таблицу включены только те из них, что имеют отношение к

изложенным в настоящей статье паттернам.)

Таблица 1.

Паттерн	WF-языки		
	XPDL	BPML4WS	BPML
Последовательность	+	+	+
Параллельное расщепление	+	+	+
Синхронизация	+	+	+
Исключающий выбор	+	+	+
Простое соединение	+	+	+
Дискриминатор	-	-	-
Произвольный цикл	+	-	-
Отложенный выбор	-	+	+

Из таблицы следует, что ни один из рассматриваемых WF-языков не поддерживает всех паттернов, и из них нельзя выбрать «наилучший» в этом смысле язык. Всегда существует паттерн, который данным языком не поддерживается, но поддерживается, по крайней мере, одним из оставшихся языков.

Заключение

В условиях «войны стандартов» WF-паттерны, несмотря на некоторые отмеченные в настоящей статье их недостатки, являются хорошим компасом в море WF-систем и стандартов, а также полезным «пробным камнем» при выборе предприятием конкретного решения.

С авторами статьи, сотрудниками консалтинговой группы «Руна», можно связаться по адресу: wf@runa.ru.