

Содержание

- Как именно триггеры покрывают бизнес-процессы** 3
- Когда начинать автоматизировать?** 3
 - Ожидания VS Реальность 4
- Будьте аккуратны!** 5
- Как это работает** 5
 - Отфильтровать 5
 - Отреагировать 6
- Кейсы и сценарии** 6
 - Перевод этапа жизненного цикла 6
 - Автоматическая запись в справочник 6
 - Кейс: Инициация проекта 7
- Технические требования** 7

Триггеры

Триггеры – это отдельный модуль, который общается с системой по интеграционной шине.

Цель триггеров как инструмента – **автоматизация бизнес-процессов**. Возможность сделать так, чтобы пользователю приходилось делать меньше действий в системе и не совершать ошибок на уровне «забыл проставить нужный реквизит».



Как именно триггеры покрывают бизнес-процессы



Идеальная картина: нажал одну кнопчку – и готово. Система магическим образом узнала, чего от неё хотел пользователь. Всё заполнилось-настроилось-появилось как надо.

Реальность: есть множество небольших кейсов, условий, каждый из которых влечёт за собой действия, которые раньше пользователь воспроизводил вручную.

«Большая» автоматизация – это набор небольших сценариев. Каждый «большой» бизнес-процесс декомпозируется на составляющие.

Таким образом первоначально триггеры **автоматизируют маленькие кусочки бизнес-процессов**.

Когда начинать автоматизировать?

Только когда процессы, которые вы перенесли в систему, обкатаны вручную. Когда вы «на коленке» убедились в их жизнеспособности.

В момент первоначального внедрения ИСУП, вы можете только *предполагать*, как это будет работать.

Автоматизация фиксирует процессы.

Прежде чем процесс зафиксировать, нужно его обкатать, убедиться в его жизнеспособности.

Предположение → проверка гипотезы → правки → снова проверка → автоматизация.

Сначала нужно научиться работать с тем, что есть, вручную. Потом в этом процессе найти «где болит» – и устранить это «болит» триггерами.

На этапе внедрения может быть некая идеальная картина, которую воспроизводят объекты и

справочники системы. Но уже через несколько месяцев бизнес-процессы могут внести свои коррективы в то, что было настроено. ⇒ Если будет на старте внедрена автоматизация, она либо перестанет работать, либо станет проблемой, которую пользователи вынуждены будут обходить.

Например, в отделе Сервиса приходилось делать двойную работу:

1. сначала записать в справочник в «Сделке» результат последнего контакта и что запланировано с этим клиентом далее;
2. вручную создать задачу на следующую активность, которая дублирует запись справочника.

Сейчас этот процесс автоматизирован. Когда триггер фиксирует событие о том, что в таком-то справочнике появилась запись из такого-то объекта, и там указан определённый классификатор, триггер через API инициирует создание нужного объекта.

Ожидания VS Реальность

Слабость автоматизации в том же, в чём и сила: в «бетонировании» бизнес-процессов.

Бизнес-процессы постоянно меняются.

Меняются требования, отчёты, данные, которые нужно вносить. Появляются новые идеи, которые конфликтуют со старыми.

Расхождения между реализованной автоматизацией и реальным бизнес-процессами могут показаться ошибками. Но на самом деле – это лишь *старые* бизнес-процессы, которые перестали быть актуальными.

Ошибка в контексте триггеров – это если реализация не соответствует требованиям из согласованного технического задания.

Если же прошло полгода-год-два, сами процессы изменились, а триггер работает по-старому, это не ошибка работы триггера. Программный код не в курсе ваших изменений – об этом можете знать только вы.

Почти любые изменения в скрипте через значимый промежуток времени по трудозатратам – это как написание нового скрипта.

Процессы изменчивы, и это нормально. Самое сложное в триггерах – это соблюсти баланс.

- Пользователь должен понимать, что система делает в ответ на его действия.
- Действия системы должны быть логичны и предсказуемы.
- Одинаковые в глазах пользователя действия не должны вести к разным последствиям.

- Необкатанные, но зафиксированные автоматизацией процессы создают гораздо больше неудобств, чем если бы эти процессы вообще не были автоматизированы.

Будьте аккуратны!



Триггер – это мощный инструмент в руках человека, понимающего настроенные в ADVANTA процессы и знающего, как они работают.

В руках же малоквалифицированного сотрудника с большим доступом – это страшная дубина, которой можно разнести весь настроенный порядок.

Триггер это микроавтомат (робот), выполняющий набор рутинных действий при наступлении определенного события в бизнес-процессе, и поэтому чувствителен к изменению условий в которых он работает. Если изменились условия или процесс, то триггер, выполняющий рутину под старый процесс, скорее всего станет выполнять неадекватные новому процессу действия, или сломается.

Соответственно самый верный способ «завалить» работу триггеров – это создать хаос в настройках: забыть, зачем и что было настроено, смело добавлять, удалять и переименовывать реквизиты, объекты, справочники, менять ограничения по созданию дочерних объектов и т.п. Впрочем, такие действия обязательно создадут массу неадекватных данных и сломают отчетность, OLAP-кубы и прочие аналитические инструменты системы и без всяких триггеров. С ними же сломается все еще быстрее.

Как это работает

Триггеры пишутся на языке C Sharp.



Отфильтровать

Половина работы триггера: это понять, на какие события он должен реагировать.

Фильтрация событий проходит в несколько этапов с помощью 3 инструментов, которые дополняют друг друга.

1. Простой фильтр на события, который есть прямо в интерфейсе модуля триггеров.
2. LINQ-запрос(ы). Может как использоваться, так и не использоваться. Удобно, чтобы

выбрать из большой базы данных какие-то специфичные объекты.

3. В рамках написания прописываются дополнительные условия – как раз здесь используется настроенный LINQ-запрос для последней тонкой фильтрации событий.

Как это выглядит в интерфейсе модуля триггеров:

Фильтрация на уровне интерфейса модуля триггеров:



Тонкая фильтрация в коде триггера:



Отреагировать

События, которые попали под условия фильтров, запускают написанный скрипт. Результатом работы скрипта могут быть небольшие вычисления, создание новых записей в справочнике, новых объектов, изменение статусов, изменение значений реквизитов, их блокировка и т.д.

Триггеры работают через [интеграционное API ADVANTA](#). ⇒ Всё, что можно сделать через API, триггер может запустить в системе.

Как это выглядит в интерфейсе модуля триггеров:



Кейсы и сценарии

Перевод этапа жизненного цикла

Действия пользователя:

- изменяет реквизиты на карточке проекта или дочерних объектах.

Работа триггера (может быть как одно, так и все перечисленные действия):

- меняет статус проекта;
- меняет значения реквизита(ов) на карточке проекта;
- меняет фазу жизненного цикла.

Автоматическая запись в справочник

Действия пользователя:

- завершает контрольную точку (статус – «Завершено»)

Работа триггера (может быть как одно, так и все перечисленные действия):

- создаёт запись в справочник у проекта;
- меняет значения реквизита(ов) на карточке проекта;
- меняет фазу жизненного цикла.

Кейс: Инициация проекта

Действия пользователя:

1. создает карточку проекта с заполненными реквизитами в папке «Инициативы»;
2. создает дискуссию или согласование, в котором обосновывает проект
 1. в качестве реквизита указывает нового РП



Когда согласование завершено/дискуссия закрыта, система **ADVANTA** с помощью триггеров **автоматически**:

1. перемещает проект из папки «Инициативы» в папку «Проекты в работе»;
2. меняет фазу ЖЦ проекта с «идея» на «проект утвержден»;
3. блокирует реквизиты проекта для защиты от изменений;
4. делегирует данный проект согласованному РП;
5. меняет статус у КТ проекта «Назначение РП проекта» на «Завершен»;
6. отправляет уведомление владельцу компании о том, что в компании стартовал новый проект.



Технические требования

Триггеры – это кроссплатформенный модуль. Можно поставить на Windows Server или на Linux. Может использовать как базу MS SQL или PostgreSQL.

Всё зависит от количества событий в системе и того, насколько много скриптов в этом модуле работают.

Триггеры состоят из 2 частей:

1. Веб интерфейс
2. База данных

	Веб интерфейс	База данных
Требования к месту на жестком диске	100мб	Зависит от количества триггеров, частоты их использования. От 100мб
Требования к оперативной памяти	1Гб	1Гб
Требования к процессору	1-2 ядра	1-2 ядра

	Веб интерфейс	База данных
Где может быть запущено	IIS (windows 2012r2)	Windows, Linux mssql версии не ниже 2012 postgresql версии не ниже 10+ Версия SQLServer, на которой должна быть запущена ADVANTA - 2008R2 с последним сервис паком обновлений. Без обновлений может не работать.
Минимальные требования	Windows server 2012r2	Windows, Linux (работоспособность проверялась на Ubuntu 18)

Исходя из текущего опыта эксплуатации¹⁾, оптимальный вариант:

1. Веб-интерфейс запускается на том же сервере, где и стоит ADVANTA. В этом случае не понадобится никаких изменений в инфраструктуре.
2. База данных на том же сервере, где расположена основная база данных системы.

Если версия SQL не соответствует минимальной необходимой для триггеров (2012), то оптимальным вариантом будет установка на сервер баз данных PostgreSQL. В отдельном сервере нет необходимости.

¹⁾
май 2020

From:
<https://wiki.a2nta.ru/> - **Wiki [3.x]**

Permanent link:
<https://wiki.a2nta.ru/doku.php/product/triggers/about?rev=1605271102>

Last update: **13.11.2020 12:38**

