

Яков Осипенков

GASEND



# Google Tag Manager

**ДЛЯ ГООГЛЯТ**

Руководство по управлению тегами

Примеры настроек от GaSend



Версия v0.2.2018 / osipenkov.ru



12+

Google Tag Manager

**Яков Осипенков**  
**Google Tag Manager для**  
**гооглят: Руководство**  
**по управлению тегами**

*[http://www.litres.ru/pages/biblio\\_book/?art=36063307](http://www.litres.ru/pages/biblio_book/?art=36063307)  
SelfPub; 2018*

**Аннотация**

Перед вами первая книга в русскоязычном сообществе по работе с самым популярным инструментом по управлению тегами в мире – Google Tag Manager. С помощью GTM вы сможете управлять всеми тегами в едином интерфейсе, сэкономить время в процессе настройки, снизить зависимость от разработчиков при внедрении кодов сторонних сервисов на ваш сайт, улучшить производительность путем снижения времени загрузки страниц, а также предотвратить возникающие в процессе работы ошибки через встроенный в Google Tag Manager режим отладки.

# Содержание

От автора	7
От автора / Команда GaSend.com	9
Благодарности	10
Предисловие	12
Подготовительные работы	17
Веб-аналитика	18
Контекстная реклама (Яндекс.Директ и Google AdWords)	20
Сдача работ	22
Глава 1. Введение в Google Tag Manager	24
История возникновения систем управления тегам	24
Google Tag Manager / 2012 год	30
Термины и определения, встречающиеся в работе с GTM	40
Навыки и знания, необходимые в работе с GTM	56
CSS-селекторы в GTM	65
Элемент	78
Селекторы по атрибутам	84
Псевдоклассы	91
Псевдоэлементы	94
Консоль разработчика	96

CSS Selector Tester	98
Селекторы в jQuery	103
Регистрация аккаунта	114
Установка Google Analytics и Яндекс.Метрика через GTM	125
Плагины и расширения для браузеров	138
Google Analytics Debugger	138
Google Tag Assistant	139
Web Analytics Solution Profiler (WASP)	141
Code Editor for GTM	142
Tag Manager Injector	144
Google Tag Manager Blocker	151
GTM Sonar	152
Глава 2. Разбор Google Tag Manager	162
Интерфейс	162
Управление пользователями	195
Версии	207
Администрирование	211
Импорт-экспорт контейнеров	215
Среды	225
Папки	240
Переменные	247
Страницы	256
Утилиты	259
Ошибки	263
Клики	268



История	279
Видео	286
Прокрутка	294
Видимость	300
Пользовательские переменные	306
Элементы страницы	322
Утилиты	331
Уровень данных	343
Триггеры	361
Теги	398
Отладчик	444
Глава 3. Google Tag Manager на примерах	466
Прослушивание кликов в GTM	467
Отслеживание кликов с помощью GTM	472
Как задать ценность покупки для пикселя Facebook	485
Передача Client ID в Google Analytics через Google Tag Manager	496
Передача ошибок JavaScript сайта в Google Analytics через Google Tag Manager	513
Настройка междоменного отслеживания через Google Tag Manager	519
Отслеживание прокрутки страниц	527
Отслеживание глубины скроллинга в GTM с привязкой к автору статьи и категории в блоге	536
Отслеживание YouTube видео в Google	548

Analytics с помощью GTM	
Подмена контента на сайте с помощью GTM	558
В заключение	572

# Яков Осипенков

# Google Tag Manager

## для гООГЛЯТ

### От автора

*«Информация принадлежит всем. Главная задача – создавать и распространять знания, а не держать их в секрете».* Так гласит одно из правил этического кодекса хакеров.

Я придерживаюсь данной точки зрения и хочу, чтобы в нашем сообществе интернет-маркетологов и веб-аналитиков создавалась среда, в которой мы общими усилиями двигали индустрию вперед на благо общего дела, не забывая при этом о рыночной конкуренции.

Чтобы в нашей сфере деятельности было больше созидателей, а не потребителей, чтобы специалисты делились своей точкой зрения с коллегами, опытом, рассказывали истории собственных успехов и неудач другим людям, не боялись выглядеть глупыми или неправыми. И чтобы у каждого человека на планете была возможность получить эту информацию в полном объеме и в любой момент времени.

В конце концов, мы с вами технические специалисты и очень тесно взаимодействуем в процессе работы с веб-раз-

работчиками и программистами. Посмотрите, какое количество литературы, статей, курсов, блогов есть у них и какое у нас... И сделайте выводы. Кто-то может возразить, что интернет-маркетинг более молодое направление, чем программирование, и что все это придет со временем. Думаете? А я вот не уверен, потому что отношение к делу *индифферентное*. Начинать надо с себя.

А чтобы создавать и распространять знания, нужно быть немного бунтарем, гиком, нарушителем спокойствия, отказываясь соблюдать формальные предписания общества, видеть мир по-другому, не любить правила и не ценить сложившийся порядок вещей. *«Потому что мир меняет именно тот, кто достаточно ненормален, чтобы считать себя способным на это»* © Вы готовы измениться? Если да, то приглашаю к себе на борт! ☺

Данное электронное руководство по GTM не было в издательстве, ее не проверял корректор, поэтому в ней могут содержаться как речевые, языковые, так и орфографические и пунктуационные ошибки. Буду признателен тем, кто в случае нахождения таковых укажет на неточности, чтобы с каждой последующей версией издания их становилось все меньше и меньше. Пишите на почту [ya.osipenkov@icloud.com](mailto:ya.osipenkov@icloud.com)

# От автора / Команда GaSend.com

Вы удивитесь, насколько просто и быстро можно работать с Google Tag Manager. Вы сможете самостоятельно устанавливать необходимые коды на сайт для выполнения разных задач, и только изредка прибегать к помощи программистов. В данной книге вы не только освоите теорию по GTM, но и закрепите свои знания на практике. Множество примеров в вашем распоряжении!

# Благодарности

Я благодарю всех читателей за то, что заинтересовались данным материалом, не остались равнодушными, нашли время и желание скачать электронное руководство по Google Tag Manager, а также за обратную связь в виде подписок, лайков, комментариев и отзывов.

Моя жена Анастасия пережила уже три издания и более 13 месяцев сумбура и постоянного клацанья по клавиатуре ноутбука. Я признателен ей за то, что она всегда поддерживает меня в трудные минуты и верит тогда, когда я на грани уже вот-вот все бросить... Недаром говорят, что за каждым мужчиной стоит великая женщина. Настена, я тебя люблю!

Персонально хочу поблагодарить своих родителей, Михаила Ивановича и Елену Владимировну Осипенковых, за поддержку и воспитание. Что выросло, то выросло. Мам, пап, люблю вас!

*Спасибо* Марии Котовой (за дизайн обложек и рекламных материалов), Антону Петроченкову из компании ConvertMonster (за вектор развития карьеры), Татьяне Михальченко и Евгении Дубровой-Аликсюк (за первых подписчиков и за то, что постите мои публикации у себя в сообществах), Андрею Осипову (за консультации по GTM), Ивану Иванову (за отличный блог [prometriki.ru](http://prometriki.ru)), команде GaSend (персонально Максиму Городнюку) за схожие взгляды

ды и огромную помощь в подготовке этой книги.

Отдельно хочу сказать *Спасибо* тем, кто помогал и помогает в распространении пособия в интернете – «по сарафанке», в социальных сетях, на форумах, торрентах и т.д. Ваш вклад в общее дело бесценен.

# Предисловие

Эта книга познакомит вас с инструментом Google Tag Manager. Вы пройдете путь от истории возникновения систем управления тегами, через теоретическую базу, которая так необходима в процессе работы с GTM, до практических знаний по настройке диспетчера тегов Google.

**Глава 1. Введение в Google Tag Manager.** Руководство начинается с истории возникновения систем управления тегами. В ней также даются определения встречающихся в GTM терминов, рекомендации по обучению и список рекомендуемой литературы из смежных областей, поскольку для максимального использования функционала системы управления тегами желательно знать основы HTML, CSS и язык программирования JavaScript.

**Глава 2. Разбор интерфейса Google Tag Manager.** Глава представляет из себя теоретическую часть, каждый элемент интерфейса GTM разобран со скриншотами и подробными комментариями авторов.

После прочтения этой главы у вас будет одно конкурентное преимущество по сравнению с другими пользователями – вы получите знания о том, что с помощью GTM можно делать и какие задачи он позволяет решать. А это более ценно, чем сама реализация. **«Знание ЧТО»** здесь котируется выше, чем **«Знание КАК»**. Ведь для **«Знания КАК»** мож-



но просто нанять специалиста, чтобы технически решить поставленную задачу. Но без «Знания ЧТО можно» вы и не знаете, стоит ли нанимать специалиста или нет. Как никогда важна теоретическая база, чтобы иметь представление о функционале Google Tag Manager, и чтобы проще контролировать подрядные организации.

**Глава 3. Google Tag Manager на примерах.** Отдельная глава, которая посвящена разбору различных примеров настройки GTM от команды GaSend.

**Глава 4. Дополнительно.** Список других моих книг, которые были выпущены и которые можно скачать в свободном доступе в интернете, а также раздел «Донат».

Для многих материал может показаться слишком банальным и простым, особенно кто работает с GTM не первый год. Однако я ориентирован не на узкую аудиторию веб-аналитиков и интернет-маркетологов, а на людей, которые хотят попробовать себя в этой специальности, но не знают с чего начать свой путь.

Я старался четко структурировать всю информацию, разложить ее по полочкам, как в библиотеке. Я пишу так, как сам поглощаю информацию, как считаю это правильным. Последовательно и небольшими порциями. Я ничего нового не придумал (и никто другой), все уже давно написано и есть в интернете в виде справочной информации и руководств. В той же справке Google или русскоязычном сообществе Google.

Но мой подход отличается от других авторов, и я убежден, что он в обучении для начинающего пользователя гораздо эффективнее, чем выдергивание различного количества данных из разных источников (агентств, новостных сайтов, фрилансеров). Как правило, там публикуют материал исходя из контент-плана или же из решаемых в данный момент задач. Нужно настроить расширенную электронную торговлю через GTM? Хорошо, сделаем. Заодно еще и статью для блога агентства напишем. Отличный вариант для продвижения агентства, но не для обучения.

Поэтому среди моих статей вы могли встретить как очень простые, так и более сложные. Информация централизована и последовательна. Если вы хотите начать свое обучение и познакомиться с веб-аналитикой поближе, то рекомендую сначала прочитать книгу **«Google Analytics для googлят: Практическое руководство по веб-аналитике. Издание 2»**, и лишь за тем переходить к изучению данного материала.

## **ВАЖНО**

В книге изложен мой полугодичный опыт использования Google Tag Manager для веб-сайтов. Я не считаю себя «гуру» веб-аналитики, наоборот, мой опыт ограничен, он небольшой, и я также, как и вы стараюсь оттачивать навыки из года в год на собственных проектах.

В первое издание я включил все самое необходимое для работы с GTM, разделив руководство на теоретическую и практическую части. Оно построено таким образом, что изучать приводимые в ней сведения можно с любого места и любой главы. Если в процессе чтения вы наткнетесь на материал, который вам известен, можете смело его пропустить. Примеры настроек GTM любезно предоставила команда GaSend.com

Еще раз хочу обратить внимание – вся информация в этой книге, включая текст и скриншоты, подготовлена на основании личного опыта, и отражает субъективный взгляд ее авторов (меня и команды GaSend). Поэтому гарантировать 100% достоверность предоставленных данных мы не можем. К тому же, Google все время выпускает обновления своих сервисов и то, что вы прочитали в этой версии, может быть не актуально уже через месяц-два или полгода.

Основная задача состояла в том, чтобы собрать весь накопленный опыт русскоязычного сообщества пользователей Google Tag Manager, структурировать это все и выдать готовый продукт в упрощенном виде для широкого круга читателей. А получилось у нас это или нет – каждый решит для себя после прочтения!

Рекомендую в качестве дополнительного материала по изучению диспетчера тегов Google:

- блог [gasend.com](https://gasend.com);
- [prometriki.ru](https://prometriki.ru) (Иван Иванов);

- [web-analytics.me](http://web-analytics.me) (Андрей Осипов);
- [simoahava.com](http://simoahava.com) (Симо Ахава) – на английском языке;
- [lunametrics.com/blog](http://lunametrics.com/blog) – на английском языке.

P. S. Все материалы из этой книги также представлены в блоге [osipenkov.ru](http://osipenkov.ru) (за исключением некоторых статей из практической части, поскольку данный материал размещен на [GaSend.com](http://GaSend.com))

# Подготовительные работы

На этом этапе идет знакомство с проектом, общение, сбор исходных данных, определение основных метрик для отслеживания, а также весь комплекс мер, направленных на создание комфортных условий для последующей работы, а именно:

- аудит существующих рекламных кампаний;
- сбор, учет и анализ ниши / конкурентная среда;
- технический анализ сайта (рекомендации);
- определение ключевых показателей эффективности (KPI);
- составление прогнозируемого! медиаплана;
- согласование исходных данных для работы (бриф);

# Веб-аналитика

Этот этап является одним из важнейших при заказе комплексной услуги. Поэтому я провожу его до или параллельно настройке самих рекламных кампаний.

Как бы это банально ни звучало, но перед запуском контекста обязательно нужно установить счетчики веб-аналитики (если этого не было сделано ранее) и проверить корректность сбора данных. Часто бывает, что и этого не делают. Также проводится:

- настройка целей, событий на кнопки / формы сайта;
- настройка электронной торговли (если e-commerce, для интернет-магазинов);
- настройка дополнительных форм (JivoSite, форма обратного звонка, callbackhunter и т.д.);
- настройка колтрекинга (как статического, так и динамического);

Яндекс.Метрика и Google Analytics – это тот минимум, который необходим для каждого проекта, направленного на получение прибыли. Благодаря инструментам веб-аналитики можно отслеживать ключевые показатели эффективности бизнеса – доход, средний чек, количество обращений, их стоимость, цикл продаж, а также общую информацию о посетителях сайта – интересы, пол, возраст, местоположение и т.д. Без него просто не имеет смысла запускать продвижение

и тратить рекламные бюджеты.

Помощь в настройке сквозной и омниканальной аналитики, интеграция с CRM, а также дополнительных сервисов автоматизации, управления ставками и рекламными кампаниями также присутствует в этом этапе работ. Все зависит от поставленных задач.

Для каждого проекта строится автоматический отчет со всеми необходимыми для заказчика показателями. Это могут быть как кастомные отчеты в Google Analytics, так и более сложные сводные в Google Таблицы, обычном Microsoft Excel, Google Data Studio, Microsoft Power BI и других инструментах. Опять же, все зависит от конкретной задачи.

Чтобы понять, почему веб-аналитика является отдельной услугой, стоимость настройки которой сопоставима с созданием рекламных кампаний, взгляните еще раз на схему выше из «Собираю команду».

# Контекстная реклама (Яндекс.Директ и Google AdWords)

На основании исходных данных, рекламного бюджета и ключевых показателей подбирается структура рекламного аккаунта (важно!), определяются типы создаваемых рекламных кампаний. И только после этого происходит:

- сбор ключевых слов и их последующая сегментация. Для кампаний в сетях – это поиск и согласование изображений;

- написание текстов объявлений (А/В тестирование);
- загрузка РК в рекламный кабинет;
- выставление основных настроек (гео, соцдемо, типы устройств, расписание показов и т.д.) согласно брифу;
- сопровождение рекламных кампаний и их последующая корректировка.

В настройку контекстной рекламы не входит (оплачивается отдельно):

- отрисовка графических баннеров в разных форматах (за исключением внутренних конструкторов Google и Яндекс);
- ведение рекламы.

Я занимаюсь настройкой всех типов рекламных кампаний в Яндекс.Директ и Google AdWords, в том числе торго-



вых, медийных, видеокампаний и рекламы мобильных приложений. Для последних необходим инструмент для трекинга и аналитики вашего приложения (AppMetrica, Adjust, Appsflyer и др.).

# Сдача работ

На выходе, после оказания всех перечисленных услуг, мы имеем несколько сценариев возможного развития событий:

1. вы довольны работой и остаетесь со мной;
2. вы недовольны работой и остаетесь со мной до тех пор, пока не окажетесь в пункте 1;
3. вы довольны работой, но не остаетесь со мной;

При всех раскладах вы получаете пожизненную online-поддержку по выполненным работам и возможность задать любой вопрос, связанный с проектом.

Не забывайте, что контекстная реклама – не таблетка от всех болезней, не панацея. На практике я все чаще убеждаюсь, что этот инструмент является вспомогательным. Да, вы можете получить от него результат в краткосрочной и долгосрочной перспективе, и он может быть, как положительным, так и отрицательным. Однако истинное «величие и выживаемость» вашего бизнеса определяется только комплексным подходом и умением приспосабливаться к окружающим реалиям.

Если у вас еще возникают сомнения, напишите мне на почту [ya.osipenkov@icloud.com](mailto:ya.osipenkov@icloud.com) с описанием проблемы или конкретных задач. Желательно сразу прикрепить ТЗ. Я свяжусь с вами в течение 24 часов, и мы вместе обсудим все вопросы. Вы всегда можете обратиться ко мне за помощью по

настройке контекстной рекламы и веб-аналитики.

# Глава 1. Введение в Google Tag Manager

## История возникновения систем управления тегами

**Маркетинговые теги** – это средства для сбора данных с веб-сайтов. Данные, собранные тегами, используются для проведения рекламных кампаний и веб-аналитики. В наши дни в среде веб-аналитики возникает много шумихи по поводу тегов, ведь в основе любого компетентного решения относительно рекламных кампаний в интернете лежат данные, собранные тегами.

В идеале теги могут преследовать различные цели, в том числе:

- сбор данных из веб-браузеров;
- взаимодействие между различными сайтами;
- интеграция стороннего контента;
- настройка файлов cookie и т.д.

Но каким образом это происходит? Когда браузер запрашивает веб-страницу с сервера сайта, содержимое страницы вместе с кодом тега возвращается браузеру, код тега выполняется и в результате осуществляется сбор данных. Помимо

этого, тег указывает браузеру отправить собранные данные на сервер сбора данных.

Данные могут собираться на стороннем сервере сбора данных или в центре сбора данных. Теги можно подразделить на две группы в зависимости от функций:

**1. основные теги (first-party tags)** запускают cookie для сбора данных в одном домене и облегчают внутренний сбор данных;

**2. сторонние теги (third-party tags)** запускают cookie для сбора данных в стороннем домене, обычно на стороннем сайте или сайте поставщика (вендора). Теги инициируют сбор данных сторонними серверами.

## Эволюция тегов

Приводим краткий обзор истории тегов, и как они превратились из пикселя отслеживания в сложную систему сбора данных.

### Сторонние теги (Third-Party Tags) – 90-ые

Первые теги появились примерно в 1995 году и их основной функцией было содействие рекламным серверам и аналитическим системам проведению оценки эффективности онлайн-маркетинговых кампаний и использования веб-

сайтов. С развитием цифровых технологий все больше сторонних поставщиков услуг начали создавать собственные теги для сбора данных, требуемых для функционирования их проектов.

Все началось с простых пикселей отслеживания, которые представляли собой небольшие прозрачные изображения, невидимые посетителям сайта, которые загружались с сервера, на котором они размещены. Эти теги-изображения представляли собой пиксель 1×1, который отображался после осуществления определенного действия, например, отправки формы или совершения покупки. Запросы пиксельного изображения у сервера подсчитывались в целях отслеживания действий пользователей на сайте. Изменив его исходный HTML-код, пиксель можно было легко добавить на веб-страницу.

Постепенно теги превратились из простых пикселей в сложный JavaScript код, который может собирать широкий спектр данных. Фрагмент JavaScript вставляется в код страницы сайта для отслеживания действий пользователей. После загрузки страницы этот код выполняется, затем он собирает большой объем данных и отправляет их в систему аналитики. JavaScript обладает рядом преимуществ, включая бесконечную кастомизацию, возможность осуществлять асинхронную загрузку, относительно низкие барьеры для реализации и простую настройку.

В результате упрощения реализации тегов и гибкости на-

стройки, количество тегов увеличилось до максимальных пределов. Однако, управлять тегами стало сложнее, поскольку в код сайта необходимо было добавлять коды от разных поставщиков.

## **Тег контейнера (Tag Container) – 00-ые**

Для решения проблемы добавления нескольких тегов на веб-сайт и управления ими через один интерфейс (в одном месте), в начале 2000-х годов крупные рекламные сети запустили теги контейнера. Теги контейнера содержат или выполняют роль своеобразной оболочки вокруг нескольких тегов. Запуск одного тега приводит к запуску других тегов, содержащихся в этом контейнере. Контейнеры, представляющие собой механизм доставки тегов, были разработаны для упрощения добавления, изменения или удаления нескольких тегов.

Несмотря на все преимущества централизации, у тегов контейнера есть один главный недостаток. Когда на сайт добавлялся сторонний код, его производительность резко снижалась. Это стало поворотным моментом, который привел к эволюции систем управления тегами, благодаря которой стало возможно заменить все теги на сайте единым JavaScript кодом. Все замененные теги запускаются на базе правил, установленных пользователем в рамках системы управления тегами. В результате чего производительность сайта повыси-

лась, а пользователь получил больший контроль над сайтом и тегами.

## **Хаос данных в результате «piggybacking»**

### **(объединения тегов)**

При реализации нескольких тегов на сайте необходимо расширить диапазон сбора данных. Данная потребность была реализована в основном за счет увеличения расходов на онлайн-рекламу, а рекламные сети объединили теги рекламного сервера в цепочку увеличения охвата аудитории на нескольких сайтах-издателях. Объединение тегов в цепочку называется «piggybacking» и заключается в возможности одного сервера инициировать теги с другого сервера, которые затем по цепочке запускают теги третьего сервера и так далее. То есть теги запускают друг друга по цепочке.

Объединение тегов «piggybacking» расширил аудиторию сайтов, что привело к инновационным подходам к покупке и продаже рекламных объявлений, в том числе тендеры в реальном времени (Real Time Bidding, RTB) и обмен данными. Однако, это привело к хаосу данных, так как владельцы сайтов обычно не могут контролировать теги по цепочке. Стремительный рост маркетинговых тегов поставил на пер-



вый план проблему конфиденциальности в связи с непреднамеренными передачами данных и утечки данных третьим сторонам в связи с тем, что владелец сайта не контролирует всю цепочку тегов. Помимо этого, затраты на инициирование тегов увеличиваются, что сказывается на производительности сайта и может привести к потере данных в результате проблем с загрузкой тегов.

*Источник информации: <https://taginspector.com/evolution-of-marketing-tags/>*

# Google Tag Manager / 2012 год



Рис. 1. Логотип Google Tag Manager

Осенью 2012 года на саммите **eMetrics** в Бостоне компания Google анонсировала выпуск своего нового продукта – **Google Tag Manager**. За неполные 6 лет он был установлен на 4 000 000+ сайтов, став самой популярной системой по управлению тегами.










	Technology	Websites	
1	 Google Tag Manager	4,435,041	Websites using Google Tag Manager
2	 Tealium	163,963	Websites using Tealium
3	 Adobe Dynamic Tag Management	81,046	Websites using Adobe Dynamic Tag Management
4	 Enlighten	37,044	Websites using Enlighten
5	 Yahoo Tag Manager	15,317	Websites using Yahoo Tag Manager
6	 QuBit OpenTag	9,611	Websites using QuBit OpenTag
7	 TagMen	7,683	Websites using TagMen
8	 Mezzobit	1,214	Websites using Mezzobit
9	 TagCommander	231	Websites using TagCommander

Рис. 2. Топ-9 технологий систем управления тегами по данным similartech.com

В десятку также входят: Tealium (163,963), Adobe Dynamic Tag Management (81,046), Ensignen (37,044), Yahoo Tag Manager (15,317), QuBit OpenTag (9,611), TagMan (7,683), Mezzobit (1,214) и TagCommander (231).

Диспетчер тегов Google (менеджер тегов, тег менеджер, таг менеджер, google tag, google tag manager, гугл тег), также известный как GTM (ГТМ), представляет собой бесплатный инструмент, который позволяет легко управлять тегами, размещенными на сайте или в мобильном приложении.

**Теги** – это небольшие фрагменты кода, как правило, на языке JavaScript, которые собирают информацию о посетителях сайта для их дальнейшего анализа.

В качестве тегов могут быть:

- Код счетчика Google Analytics;

```
<!-- Global site tag (gtag.js) - Google Analytics -->
<script async src="https://www.googletagmanager.com/gtag/js?id=UA-77456218-1"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'UA-77456218-1');
</script>
```

Рис. 3. Код счетчика Google Analytics

## ● Код счетчика Яндекс.Метрика;

```
<!-- Яндекс.Метрика counter -->
<script type="text/javascript" >
  (function (d, w, c) {
    (w[c] = w[c] || []).push(function() {
      try {
        w.yaCounter32630795 = new Ya.Metrika2({
          id:32630795,
          clickmap:true,
          trackLinks:true,
          accurateTrackBounce:true,
          webvisor:true,
          ecommerce:"dataLayer"
        });
      } catch(e) {}
    });
  })(document, window, ["script"]);

  var n = d.getElementsByTagName("script")[0],
      s = d.createElement("script"),
      f = function () { n.parentNode.insertBefore(s, n); };
  s.type = "text/javascript";
  s.async = true;
  s.src = "https://mc.yandex.ru/metrika/tag.js";

  if (w.opera == "[object Opera]") {
    d.addEventListener("DOMContentLoaded", f, false);
  } else { f(); }
```

Рис. 4. Код счетчика Яндекс.Метрика

## ● Отслеживание конверсий AdWords с тегом событий;

```
<!-- Global site tag (gtag.js) - Google AdWords: 853786644 -->
<script async src="https://www.googletagmanager.com/gtag/js?id=AW-853786644"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'AW-853786644');
</script>

<!-- Event snippet for test conversion page -->
<script>
  gtag('event', 'conversion', {send_to: 'AW-853786644/zhS5CKzrh4QIICPlwM'});
</script>
```

Рис. 5. Код отслеживания AdWords

## ● Google Оптимизация;

```
ga('require', 'GTM-56WNV7J');
```

Рис. 6. Код Google Оптимизация

## ● Facebook Pixel;

```
<!-- Facebook Pixel Code -->
<script>
!function(f,b,e,v,n,t,s){if(f.fbq)return;n=f.fbq=function(){n.callMethod?
n.callMethod.apply(n,arguments):n.queue.push(arguments)};if(!f._fbq)f._fbq=n;
n.push=n;n.loaded=!0;n.version='2.0';n.queue=[];t=b.createElement(e);t.async=!0;
t.src=v;s=b.getElementsByTagName(e)[0];s.parentNode.insertBefore(t,s)}(window,
document,'script','https://connect.facebook.net/en_US/fbevents.js');
fbq('init', '1710554435920136'); // Insert your pixel ID here.
fbq('track', 'PageView');
</script>
<noscript></noscript>
<!-- DO NOT MODIFY -->
<!-- End Facebook Pixel Code -->
```

Рис. 7. Код пикселя Facebook

## ● Код ретаргетинга Вконтакте;

```
<script type="text/javascript">(window.Image ? (new Image()) :  
document.createElement('img')).src = 'https://vk.com/rtrg?p=VK-RTRG-  
120904-erTZ2';</script>
```

Рис. 8. Код ретаргетинга Вконтакте

И таких сервисов, которые собирают информацию о пользователях посредством внедрения своего кода на сайт, очень много. Например, теги системы автоматизированной закупки рекламы **DoubleClick**, популярной ретаргетинг-платформы **AdRoll**, сервиса по А/В тестированию и оптимизации конверсии **Visual Website Optimizer (VWO)** и т.д. В качестве дополнительного инструмента веб-аналитики для более глубокого анализа аудитории владельцы сайтов ставят **comScore**, **KISSmetrics**, **Quantcast**, **Adobe Analytics** и др.

Последние очень распространены на иностранных рынках. Даже если не брать их в счет, для российского интернет-сегмента собирается немаленькое количество. Из примеров выше их уже как минимум 6. Не забываем и про продукты в виде чатов, коллтрекингов, форм обратного звонка, сервисов мультиканальной/омниканальной аналитики. У каждого из них свой JavaScript код, который необходимо добавить на сайт на те страницы, на которых вы хотите отслеживать действия пользователей. А чем больше таких сторонних служб (сервисов), тем выше вероятность замедления ра-

боты сайта.

Что же приходится делать владельцам сайтов, когда они хотят внедрить очередной код какого-либо сервиса к себе на сайт или внести изменения в существующий без использования Google Tag Manager? Все верно, как минимум писать разработчику ТЗ с подробными инструкциями того, куда нужно вставить код, а как максимум – внедрять его на сайт самостоятельно. При такой последовательности действий есть большая вероятность:

- самому ошибиться с внедрением различных кодов в силу незнания правил и разметки веб-страниц;

- получить от программиста перечень пунктов, которые были ему непонятны из ТЗ. В этом случае начнется игра в «настольный теннис»: я сделал все согласно ТЗ, но не работает. Присылайте новое ТЗ;

- сорвать все сроки и дедлайны из-за долгой обратной связи. Вытекает из предыдущего пункта.

Такой вариант внедрения можно представить в виде последовательности шагов:



Рис. 9. Внесение изменений в код сайта без использования Google Tag Manager

1. подготавливается ТЗ из некоторого количества пунктов;

2. список передается разработчику, который просматривает его;

3. если у него нет вопросов, то он внедряет эти пункты. В противном случае мы возвращаемся на шаг 1;

4. код устанавливается на сайт. Если по каким-то причинам это было сделано некорректно, все этапы придется начинать сначала (подготавливать ТЗ, назначать тикеты программисту, писать письма, отправлять фрагменты кода и т.д.).

Согласитесь, очень долгий и бесполезный процесс. При установке контейнера Google Tag Manager процесс сводится к следующему:





Рис. 10. Внесение изменений через Google Tag Manager

1. устанавливается код Google Tag Manager один раз;
2. внедряются изменения на сайт через рабочую область GTM без прибегания к помощи со стороны;
3. производится отладка всех процессов и публикуется рабочая версия тегов.

Когда Google выпускал свой продукт больше 5 лет назад, он хотел решить ряд задач, в числе которых:

- ускорение загрузки страниц и работоспособности сайтов путем объединения тегов в одном удобном инструменте;
- экономия времени разработчиков, маркетологов и веб-аналитиков;
- избегания дублирования и ошибок в работе тегов;
- снижение необходимости изменения исходного кода сайта при обновлении или добавлении тегов;
- завоевание доли рынка благодаря бесплатности GTM

(да, без этого никуда).

В качестве недостатка Google Tag Manager, да и вообще всех диспетчеров тегов, можно отметить зависимость от объектной модели документа (DOM) – верстки или исходного кода страниц. Поскольку все операции выполняются с привязкой к различным идентификаторам, атрибутам и классам, то в случае их изменения, сделанные ранее настройки могут перестать работать.

Несмотря на то, что с внедрением Google Tag Manager наша зависимость от разработчиков существенно снизилась, отказаться полностью от их помощи все же не удастся. Есть ряд задач, которые по-прежнему будет необходимо решать вместе с программистами. Сюда входят:

- фиксация транзакций;
- настройка User ID;
- добавление пользовательских параметров и показателей;
- внедрение уровня данных;
- прочие задачи.

Останавливаясь на функциях, описанных выше, мы не будем. Все они подробно разобраны в моей другой книге, которая называется **Google Analytics для googлят: Практическое руководство по веб-аналитике**.

Таким образом, из преимуществ Google Tag Manager можно выделить:

- **бесплатный инструмент** – лидер рынка;

● **экономит время** – не нужно искать разработчиков, которые внедрят изменения на сайт и нет необходимости ждать последующих правок, если первоначальные были сделаны с ошибками;

● **снижает зависимость от разработчиков**;

● **позволяет управлять тегами в едином пространстве** – не придется писать дополнительный код или вносить изменения в код отслеживания, вся работа выполняется через веб-интерфейс;

● **средства предотвращения ошибок** – режим предварительного просмотра (чтобы вы могли видеть предлагаемые изменения перед их внедрением);

● **работает быстро благодаря асинхронной загрузке тегов** – одновременная (параллельная) загрузка тегов, в результате которой более медленные загружающиеся теги никак не повлияют на скорость выполнения других, более быстрых тегов.

# Термины и определения, встречающиеся в работе с GTM

Основными понятиями, которые используются в Google Tag Manager, являются: *контейнер*, *среды*, *тег*, *триггер*, *переменная*, *рабочая область*, *версия*, *уровень данных*.

Под **контейнером** в GTM можно подразумевать две составляющие:

1. пространство, в котором мы работаем и совершаем какие-либо действия с остальными элементами (тегами, триггерами, переменными);

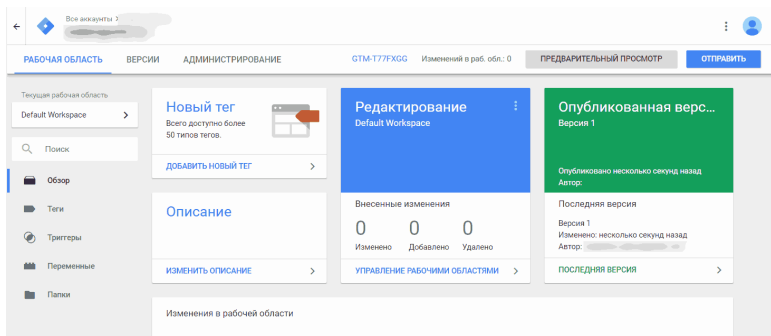


Рис. 11. Контейнер GTM

2. специальный код (скрипт), который размещается на

всех страницах сайта и который активирует все остальные теги этого сайта в соответствии с правилами, заданными в интерфейсе. Он состоит из двух частей – один из них размещается в раздел **<head>**, а второй после открывающего тега **<body>**.

Установите Диспетчер тегов Google ✕

Скопируйте приведенный ниже код и вставьте его на каждую страницу сайта.

Вставьте этот фрагмент в раздел **<head>** кода страницы как можно ближе к началу:

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-T77FXGG');
<!-- End Google Tag Manager -->
```

Также вставьте непосредственно после открывающего тега **<body>** следующий фрагмент:

```
<!-- Google Tag Manager (noscript) -->
<noscript><iframe src="https://www.googletagmanager.com/ns.html?id=GTM-T77FXGG"
height="0" width="0" style="display:none;visibility:hidden"></iframe></noscript>
<!-- End Google Tag Manager (noscript) -->
```

Подробнее о том, как добавить фрагмент кода Диспетчера тегов Google, читайте в [руководстве по началу работы](#).

Рис. 12. Диспетчер тегов Google

Таким образом, контейнер – это глобальный тег (самого верхнего уровня), который позволяет работать GTM с вашим сайтом.

**Тег** в Google Tag Manager – это фрагмент JavaScript кода,

который собирает данные о посетителях на сайте и в приложении, а затем пересылает их на сторонние сервисы – Google Analytics, Google AdWords, Facebook, Яндекс.Метрика и т.д.

```
<!-- Global site tag (gtag.js) - Google Analytics -->
<script async src="https://www.googletagmanager.com/gtag/js?id=UA-77456218-1"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());

  gtag('config', 'UA-77456218-1');
</script>
```

Рис. 13. Google Analytics – Global site tag (gtag.js)

В GTM есть n-ое количество шаблонов тегов, которые облегчают их установку на сайте. Они разделены на подгруппы:

- *рекомендуемые* – Universal Analytics, классический Google Analytics, отслеживание конверсий AdWords, ремаркетинг AdWords, счетчик DoubleClick Floodlight, тег продаж DoubleClick Floodlight, Google Optimize, Google Опросы

- *специальные* – пользовательский HTML и пользовательское изображение

- *еще* – сторонние сервисы, Twitter Universal Website Tag, Adometry, Crazy Egg, comScore Unified Digital Measurement, K50 tracking tag, Foxmetrics, Hotjar Tracking Code и другие.



## Выберите тип тега



**Стандартное отслеживание ClickTale**  
ClickTale



**comScore Unified Digital Measurement**  
comScore



**Crazy Egg**  
Crazy Egg



**Criteo OneTag**  
Criteo



**Ter DistroScale**  
DistroScale



**Универсальный пиксель Dstellery**  
Dstellery



**Eulerian Analytics**  
Eulerian Analytics



**FoxMetrics**  
FoxMetrics



**Google Trusted Stores**  
Google Trusted Stores



**Hotjar Tracking Code**  
Hotjar



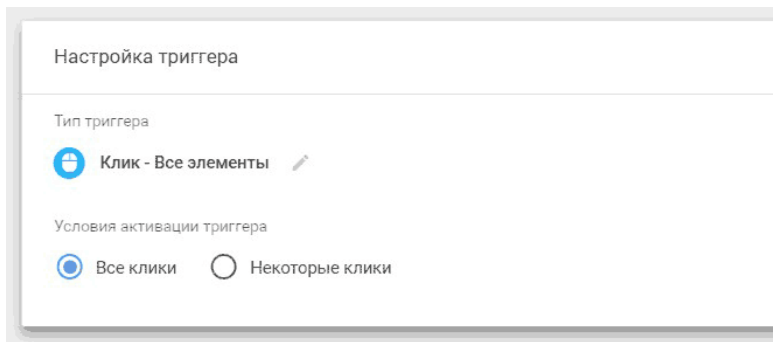
**Тег отслеживания звонков Infinity**  
Infinity Tracking



**Intent Media - Search Compare Ads**  
Intent Media

## Рис. 14. Сторонние теги

**Триггер** в GTM – это условие (правило), при котором активируется или блокируется тег. Для запуска тега должен быть хотя бы один триггер. Таким образом, нельзя создать тег без триггера. Каждый триггер может быть истинным (true), либо ложным (false). И он выполняется только в том случае, когда его значение является истинным и, если выполняются все условия триггера.



## Рис. 15. Условие активации триггера

Триггеры в Google Tag Manager делятся на **триггеры активации тегов** и **триггеры блокировки тегов**.

**Триггер активации** – это условие, при выполнении которого срабатывает тег.

Например, вы хотите отслеживать клики по определенной



кнопке на сайте. Зная значения переменной, вы можете настроить триггер активации, например, по **Click ID**:

Настройка триггера

Тип триггера

Клик - Все элементы

Условия активации триггера

Все клики  Некоторые клики

Активировать триггер при наступлении события и выполнении всех этих условий

Click ID ▾ равно ▾ menu-item-4306 - +

Рис. 16. Условия активации триггера – Некоторые клики

Таким образом, тег активируется только в том случае, когда идентификатор HTML-тега равен *menu-item-4306*.

Триггеры активации делятся на:

- **встроенные триггеры** – доступны к использованию готовые шаблоны;

- **пользовательские триггеры** – ручная самостоятельная настройка.

**Триггер блокировки** – это условие, при выполнении которого активация тега блокируется.

Например, вы хотите активировать тег на всех страницах сайта, кроме одной конкретной. Тогда необходимо создать триггер с условием активации на нужных страницах, а далее в соответствующем теге добавить этот триггер в исключение.

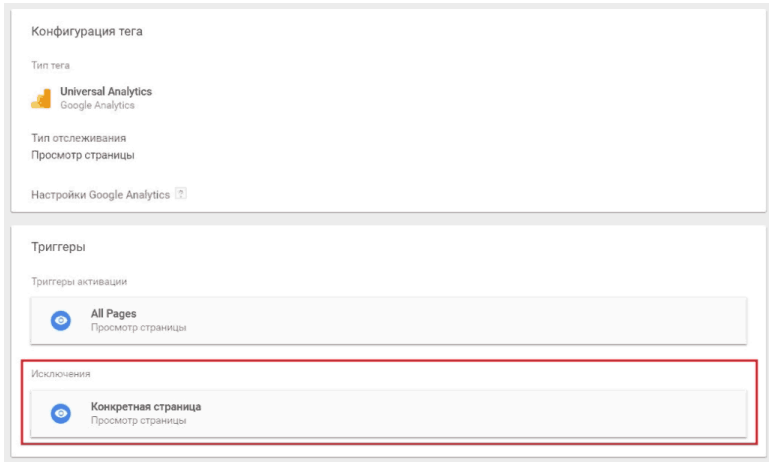


Рис. 17. Триггер блокировки, исключения

При конфликте триггеров активации и блокировки, приоритет отдается блокиратору.

Устаревшее название триггера в предыдущей версии диспетчера тегов Google – правило.

**Переменная** в Google Tag Manager – это объект, которому дано имя и который может принимать различные значения (вида пара «ключ-значение»). Она вызывается внутри тега и триггера. Имя переменной постоянно, а ее значение меняется в зависимости от того, какие действия происходят на сайте и в зависимости от того, какая выстроены процессы внутри самого GTM. Единственный случай, когда переменная имеет постоянное значение, это тип переменной – кон-

станта.

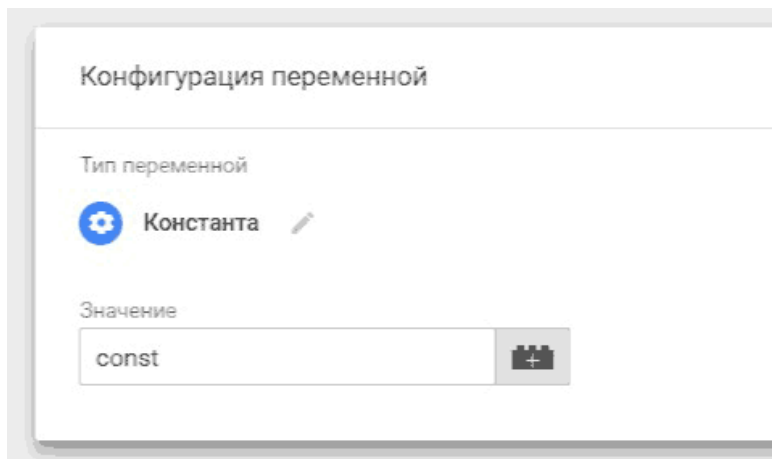


Рис. 18. Тип переменной – Константа

Переменные нужны для хранения данных, используемых при определении триггера и передачи информации в тег, например, в Google Analytics о сумме транзакции, ID заказа, идентификаторе продукта и т.д.

В Google Tag Manager они делятся на две категории:

● **встроенные переменные** – набор готовых, предустановленных в системе переменных. Включаются/отключаются путем простановки галочек (чекбоксов) напротив каждой из них;

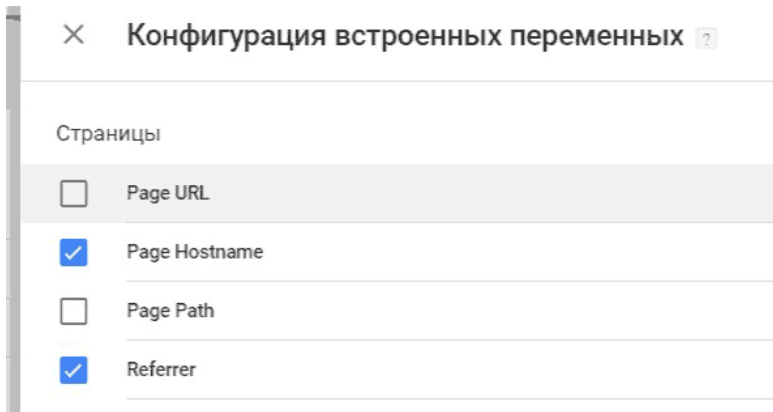


Рис. 19. Конфигурация встроенных переменных

● **пользовательские переменные** – самостоятельное создание с использованием существующих типов переменных.

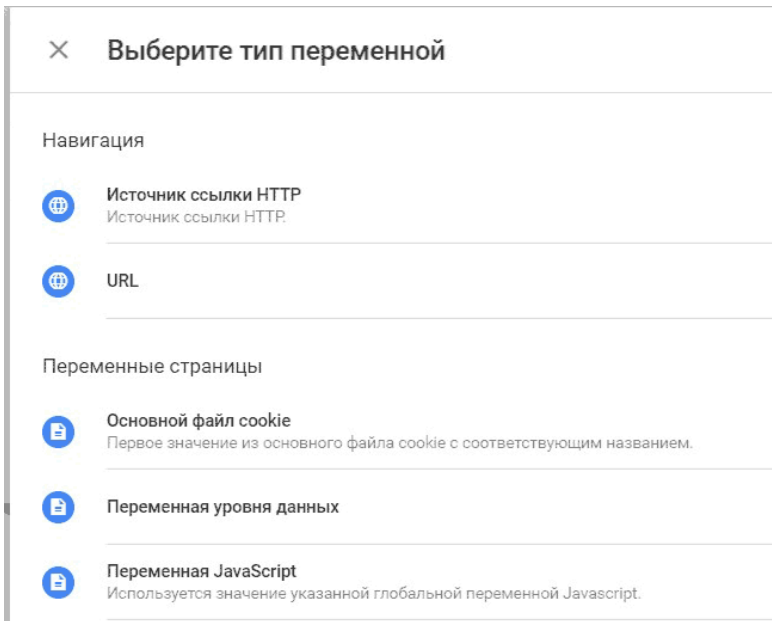


Рис. 20. Пользовательские переменные

Синтаксис для вызова и выполнения переменной в GTM – это двойные фигурные скобки `{{ }}`. Устаревшее название переменной в предыдущей версии диспетчера тегов Google – *макрос*.

**Рабочая область** в Google Tag Manager – это пространство для работы с изменениями в диспетчере тегов.

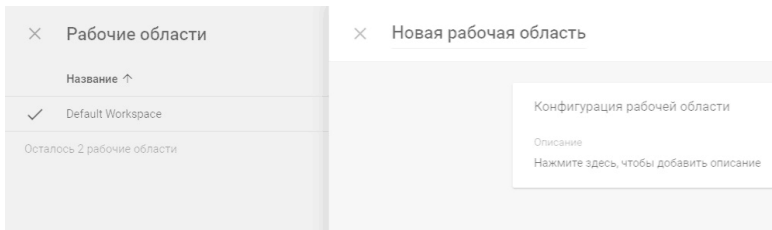


Рис. 21. Рабочие области в GTM

Несколько рабочих областей целесообразно использовать, если над проектом работает  $N > 1$  специалистов или команд, каждая из которых отвечает за свой перечень задач. Например, можно создать рабочую область для разработчиков, SEO-специалистов, специалистов по платному трафику или для тех, кого вы привлекаете в качестве независимых консультантов (исправить ошибки, помочь с настройками и т.д.).

Рабочие области в GTM можно грубо сравнить с представлениями в Google Analytics.

**Версия** в Google Tag Manager – сохраненная копия контейнера в данный момент времени. Она позволяет сохранять промежуточные версии работы и возвращаться к ним при необходимости.

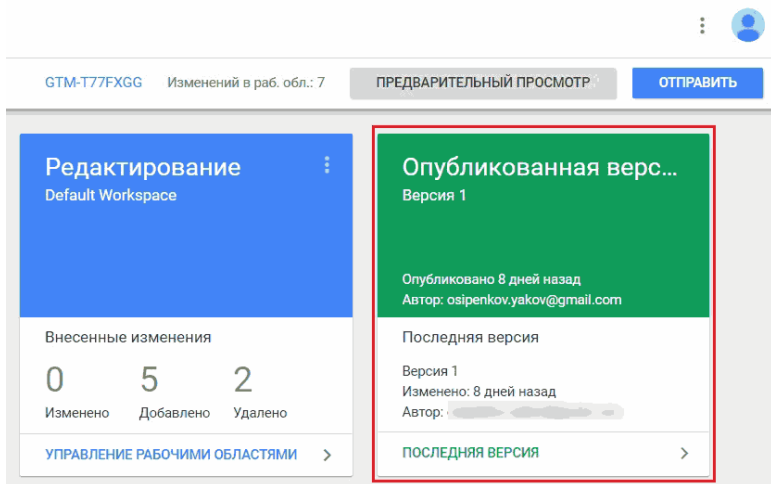


Рис. 22. Рабочие области в GTM

Условно, версии могут создаваться двумя способами:

- **Вручную** (сохранение текущего варианта рабочей области в виде версии без публикации);

- **Автоматически** (когда вы публикуете контейнер, записывается его версия. Таким образом, в вашем аккаунте будут находиться все опубликованные версии контейнера).

Версии в GTM создаются вручную с помощью кнопки «Отправить» – «Новая версия», добавив к конфигурации название и описание версии.

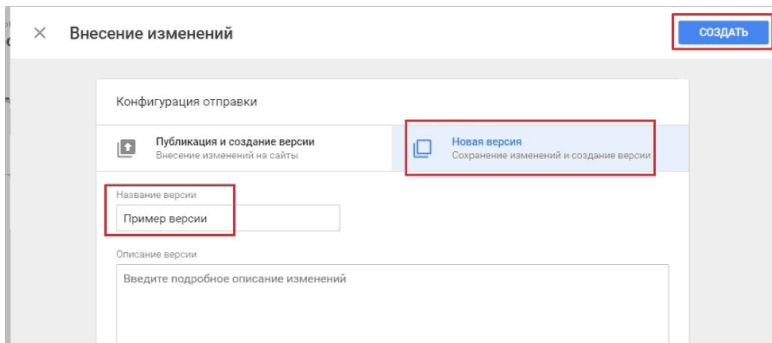


Рис. 23. Новая версия

В это случае мы сохраняем версию, которая не будет опубликована на сайте и ее не увидят пользователи. При выборе **«Публикация и создание версии»**, наоборот, все изменения в данной версии будут доступны посетителям.

Версии в Google Tag Manager также полезны тогда, когда необходимо откатиться (вернуться) назад в случае ошибок на предыдущем шаге.



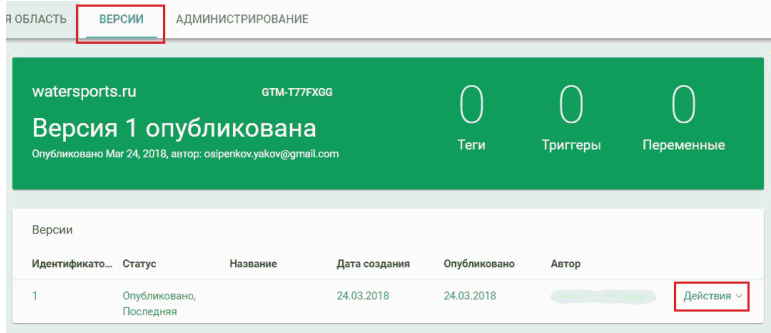


Рис. 24. Меню версий

С версиями можно выполнять различные действия. Доступны:

- Предварительный просмотр;
- Поделиться ссылкой для просмотра;
- Опубликовать;
- Переименовать;
- Редактировать примечания;
- Выбрать в качестве последней версии;
- Экспортировать;

**Уровень данных (он же dataLayer)** – это объект JavaScript, который необходим для того, чтобы передавать информацию из вне системы в Google Tag Manager. Например, информацию о транзакции, цене товара, количестве покупок в интернет-магазине для ее последующей фиксации в Google Analytics или Яндекс.Метрика.

Уровень данных описывается специальным образом в исходном коде страницы сайта. Для того, чтобы информация, которая в нем описана, попала в GTM, необходимо создать пользовательскую переменную, выбрав соответствующий тип **«Переменная уровня данных»**.

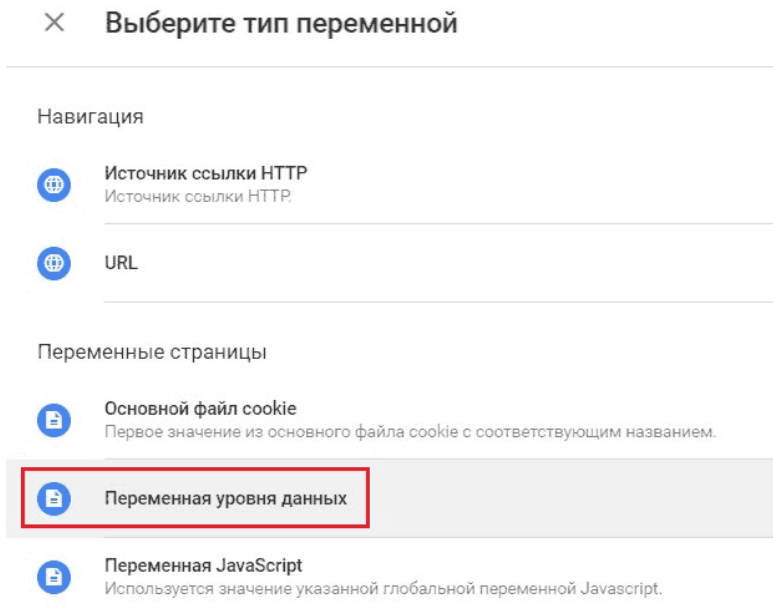


Рис. 25. Переменная уровня данных

**Среды** в Google Tag Manager позволяют протестировать изменения, внесенные в контейнер, перед их публикацией

на сайте. По умолчанию в контейнере создается две среды – **Live (реальная)** и **Latest (последняя)**.

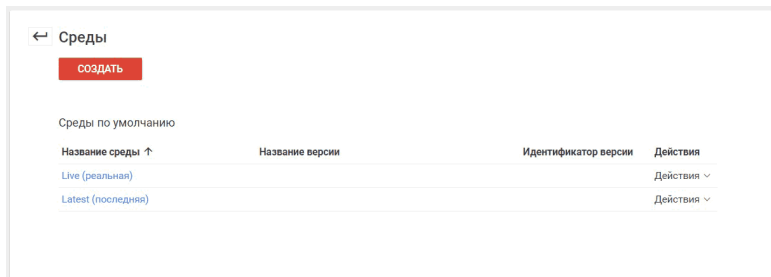


Рис. 26. Среды в GTM

Подробнее о каждом из вышеописанных терминов разберем в последующих разделах книги.

# Навыки и знания, необходимые в работе с GTM

На мой взгляд, наиболее ценные навыки и знания, которые необходимы веб-аналитику при работе с проектами, заключаются не в умении пользоваться интерфейсами Google Tag Manager, Google Analytics, Яндекс.Метрика и т.д., знать где расположены различные кнопки и функции, а в понимании принципов работы с данными – делать из них выводы, которые будут ценными для компании (клиента, заказчика) и в дальнейшем приведут к положительному результату.

Важно видеть неочевидные вещи, уметь извлекать пользу из данных, из которых, на первый взгляд, вообще никакого вывода сделать нельзя. Работа веб-аналитика не ограничивается построением отчетов и настройкой счетчиков аналитики. Он должен быть внимателен к деталям и быть в курсе всех событий внутри компании (акций, скидок, специальных предложений, расширении ассортимента и т.д.), чтобы находить зависимости между этими событиями и оперативно реагировать на изменения.

Помимо этого, аналитик должен понимать, как работает тот или иной канал трафика, по которым посетители приходят на сайт. Он должен уметь пользоваться не только базовыми и бесплатными инструментами, но и иметь возможность предложить заказчику более серьезные сервисы и ориенти-

роваться в них.

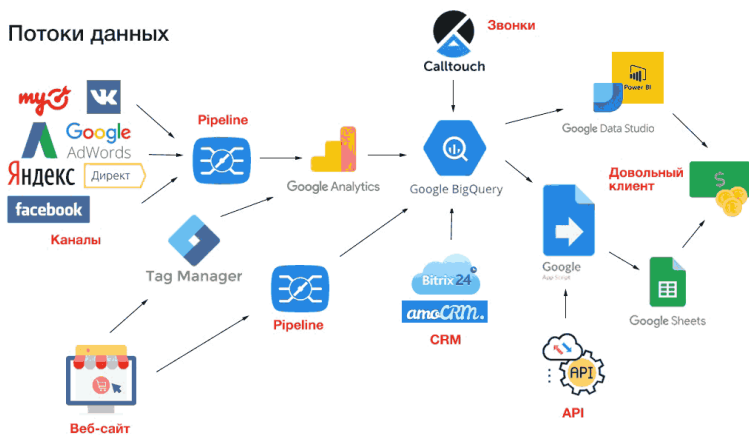


Рис. 27. Пример потоков данных проекта

Изображение выше иллюстрирует один из примеров настройки потоков данных для среднего интернет-магазина. Вы уже видели эту картинку в самом начале руководства в разделе «Собираю команду». Если вы считаете себя неплохим веб-аналитиком, задайте вопрос: «Сколько сервисов и технологий из перечисленных я»:

А. из этого списка знаю;

Б. пробовал внедрить для заказчика;

Разумеется, такая работа подразумевает под собой комплексный подход, ни один день работы и ни в одиночку.

Уметь делать все самостоятельно необязательно, но ориентироваться в механике будет существенным плюсом.

Не менее важно, чтобы аналитик мог не только откликнуться на запрос передачи данных от других команд, но и мог сам определить, какими данными следует поделиться. Для этого нужно постоянно смотреть на данные под разным углом, думать о том, могут ли они быть полезны.

Переходя к инструментарию веб-аналитика в целом, и в Google Tag Manager в частности, я бы выделил следующие:

- счетчики веб-аналитики Google Analytics или Яндекс.Метрика;
- диспетчер тегов Google Tag Manager;

Он позволяет упростить взаимодействие между вами и заказчиком, снизив при этом зависимость от команды разработчиков сайта. Чтобы максимально использовать функционал системы управления тегами, желательно знать основы HTML, CSS и язык программирования JavaScript. Именно в такой последовательности лучше всего начинать обучение: 1. HTML; 2. CSS; 3. JavaScript.

**HTML** (от англ. HyperText Markup Language – «язык гипертекстовой разметки») – стандартизированный язык разметки документов в интернете. Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного

устройства.

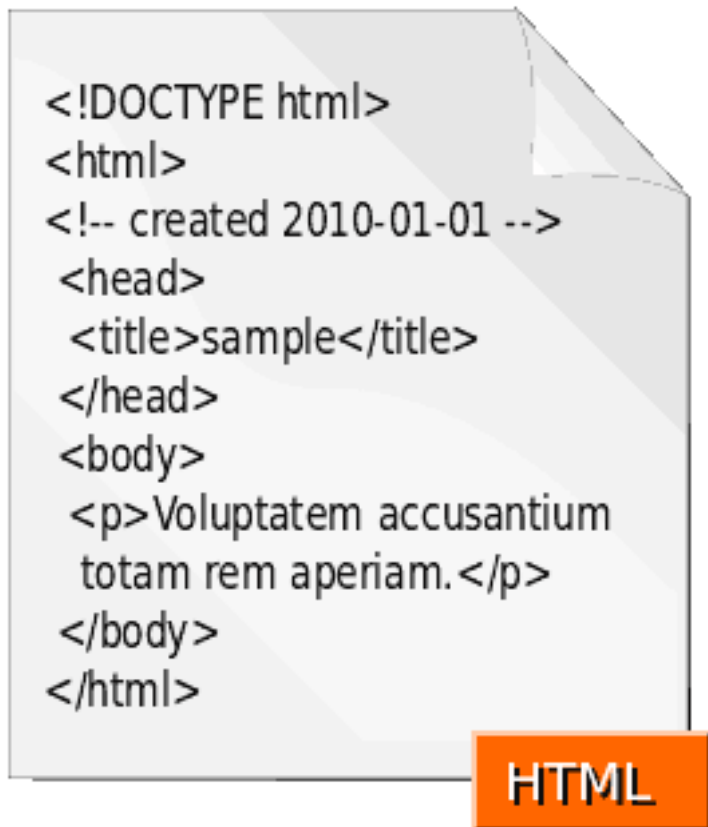


Рис. 28. HTML

Затем вы можете перейти к изучению CSS, и как исполь-

зовать их для оформления HTML документа (например, изменить шрифт и его размер, добавить границы и тени, разметить страницу в несколько столбцов, добавить анимацию и другие визуальные эффекты).

Один из самых важных механизмов CSS – селекторы. Это формальное описание того элемента или группы элементов, к которым применяется указанное правило стиля. В процессе работы с GTM вы будете очень часто сталкиваться с ситуацией, когда необходимо настроить отслеживание клика по какому-либо элементу на веб-странице, а у этого элемента нет идентифицирующих его атрибутов, таких как, например, **ID**, **class** или **URL**. Тогда на помощь приходят CSS-селекторы.

**Рекомендуемая литература:** [htmlbook.ru](http://htmlbook.ru) и книга Влада Мержевича «HTML и CSS на примерах».



The image features the letters 'JS' in a large, bold, black, sans-serif font. The letters are centered on a solid, bright yellow background. The 'J' is on the left and the 'S' is on the right, with a small gap between them. The overall appearance is clean and minimalist.

Рис. 29. JavaScript

Как правило, счетчики веб-аналитики представляют собой фрагмент JavaScript кода, который размещается на отслеживаемых страницах сайта. Код Google Analytics, Google Tag Manager, Яндекс.Метрика – в основе всех их лежит JavaScript.

На JavaScript также можно писать скрипты для Google AdWords для автоматизации определенных действий.



Рис. 30. Библиотека jQuery

Помимо самого JS, не помешают знания **jQuery**. jQuery – библиотека JavaScript, фокусирующаяся на взаимодействии

JavaScript и HTML. Она помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержанию элементов DOM, манипулировать ими.

**Рекомендуемая литература:** [learn.javascript.ru](http://learn.javascript.ru), Крис Минник и Ева Холланд «JavaScript для чайников», Марейн Хавербек «Выразительный JavaScript», Эрик Фримен и Элизабет Робсон «Изучаем программирование на JavaScript», Антон Шевчук «Учебник jQuery для начинающих».

Для работы с GTM вышеописанных знаний вполне достаточно. Однако если говорить о других инструментах и задачах веб-аналитика, например, передавать данные из CRM-системы в Google Analytics или наоборот, то здесь нужны навыки и опыт импорта данных и API.

Использование SQL-запросов может пригодиться при работе с BigQuery. В последнее время все чаще используются инструменты визуализации данных (**Google Data Studio, Microsoft Power BI, Tableau** и др.). В службе Power BI можно просматривать визуальные элементы, созданные с использованием R-скриптов, и взаимодействовать с ними. Они написаны на языке программирования **R** – самом популярном языке программирования среди специалистов по статистике, обработке и анализу данных.

Все большую популярность набирает и **Python**, который имеет встроенные библиотеки для анализа данных (**Anaconda – Pandas**). Не стоит забывать и про старый добрый Microsoft Excel с мощным функционалом для работы с

табличными данными.

Образование в ВУЗах оставляет желать лучшего. Не берусь назвать ни один из университетов, кто готовил бы и выпускал специалистов по профессии «веб-аналитик». Разве что слышал о дополнительном образовании в ВШЭ (Высшая школа экономики). Но курсы и онлайн-тренинги, как правило, не дают полных знаний. Однако решение практических задач и аналитический склад ума является преимуществом и ускоряет освоение нового материала. Веб-аналитик без высшего образования – редкость в наше время.

# CSS-селекторы в GTM

Если в книге «**Google Analytics для googлят**» одной из самых непростых тем для изучения по праву считают регулярные выражения (из-за сложного восприятия и запоминания синтаксиса команд, метасимволов), то в этой, пожалуй, ей равнозначны именно CSS-селекторы, поскольку знания в данной области лежат далеко за пределами анализа статистики и построения отчетов.

Но как только вы научитесь правильно использовать селекторы CSS в Google Tag Manager, то забудете о сторонней помощи программистов по многим задачам. Здесь больше именно веб-разработки, но без спецсимволов из регулярных выражений все равно не обошлось.

В процессе настройки аналитики часто приходится отслеживать клики по определенным элементам на странице сайта. Все просто, если у отслеживаемых элементов (кнопки, ссылки, формы и т.д.) есть атрибуты **id** или **class**. Например, как здесь (инспектирование кода веб-страницы в консоли разработчика):

```
    </div>
  </div>
... <div class="thirdscreen dark-1" id="block_3"> == $0
  <div class="inner">
    <p class="all-titles">...</p>
    <ul class="tabs" id="razdel-tabs">...</ul>
    <div class="tabsfon-all">
      <div class="tabsfon">...</div>
      <div class="tabsfon active">
```

Рис. 31. Атрибуты id и class у элемента на сайте

В Google Tag Manager для активации тега на данный элемент остается только настроить триггер со встроенными переменными **Click Classes** и **Click ID**.

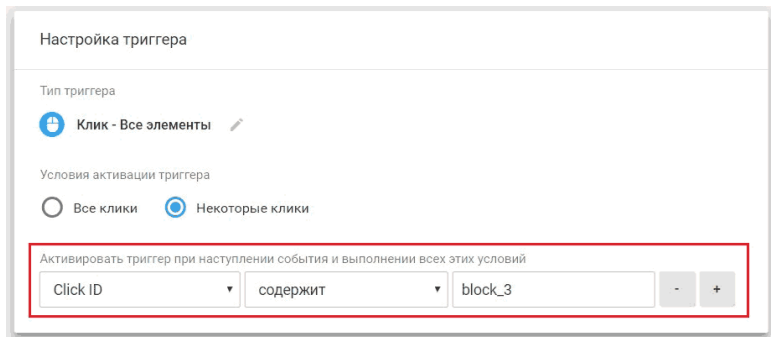


Рис. 32. Пример условия активации триггера по Click ID

Но часто бывает, что у необходимого элемента нет таких атрибутов. Тогда на помощь приходят CSS-селекторы.

Селектор – это часть CSS-правила, которая сообщает браузеру, к какому элементу (или элементам) веб-страницы будет применен стиль.

CSS-селекторы применяются в Google Tag Manager для условий активации триггеров, а также в пользовательских переменных, таких как «Видимость элемента», «Элемент DOM» и «Собственный код JavaScript».

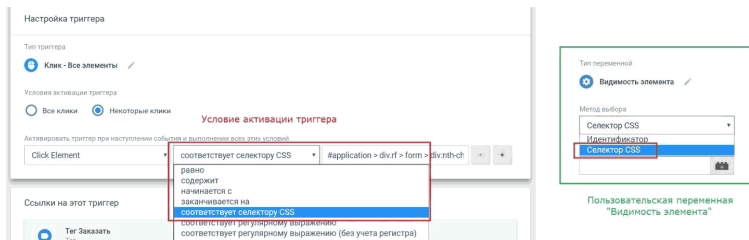


Рис. 33. CSS-селекторы в триггерах и переменных GTM

Если атрибут нужного элемента не повторяется на странице, то рекомендуется использовать именно встроенные переменные. Но если один и тот же класс используется в нескольких элементах на странице, тогда рекомендуется использовать селекторы CSS.

Как правило, большинство настроек с использованием селекторов CSS приходится на связку «соответствует селектору CSS» + **Click Element** или **Form Element**.

Но прежде чем разбираться в селекторах, рекомендую для

начала ознакомиться с глобальной структурой документа в формате HTML – какие бывают элементы, заголовки, атрибуты, метаданные и т.д.

Подробнее обо всех элементах HTML читайте в руководстве разработчиков Mozilla. Чаще всего вы будете встречаться с такими тегами, как `<p>`, `<a>`, `<div>`, `<span>`, `<h1>`... `<h6>`.

Кроме этого, вы должны познакомиться с объектной моделью документа (**Document Object Model, DOM**). Согласно DOM-модели, документ является иерархией, деревом. Каждый HTML-тег образует узел дерева с типом «элемент». Вложенные в него теги становятся дочерними узлами.

Прежде чем в Google Tag Manager настраивать триггеры на отслеживание прокрутки, взаимодействия пользователей с видео, электронную торговлю, вам необходимо понять, что из себя представляет дерево DOM.

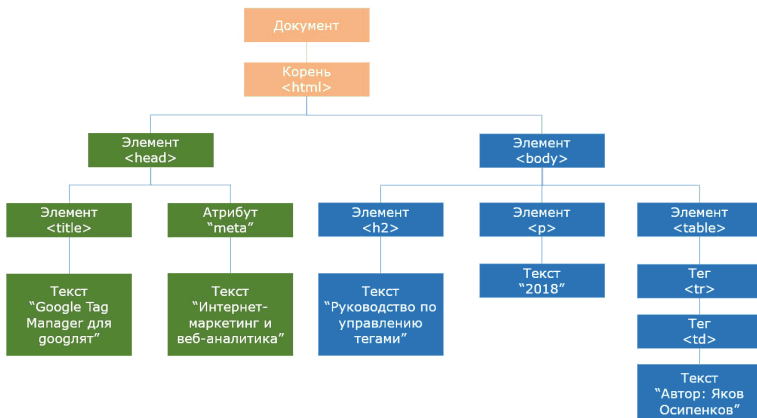
DOM определяет логическую структуру документа, способы доступа к элементам и их изменению. Документ может быть как HTML, так и XML. Когда DOM используется для определения логической структуры HTML-документа, значит речь идет о HTML DOM. Давайте рассмотрим пример:



```
HTML ▼
1 <html>
2 <head>
3 <title>Google Tag Manager для googля</title>
4 <meta name="description" content="Интернет-маркетинг и веб-аналитика"/>
5 </head>
6 <body>
7 <h2>Руководство по управлению тегами</h2>
8 <p>2018</p>
9 <table>
10 <tr>
11 <td>Автор: Яков Осипенков</td>
12 </tr>
13 </table>
14 </body>
15 </html>
```

Рис. 34. Пример HTML-документа

Представление DOM этого документа HTML выглядит следующим образом:



## Рис. 35. Представление DOM для документа HTML

Здесь `html`, `head`, `body`, `title`, `meta`, `h2`, `p`, `table`, `tr`, `td`, Google Tag Manager для googлят, Интернет-маркетинг и веб-аналитика, 2018 и т.д. являются примерами объектов DOM. Они также называются элементами или узлами. Нижние блоки с пометкой «Текст» являются текстовыми узлами.

Чтобы отслеживать клики/события через GTM по определенным элементам на странице, необходимо понимать взаимосвязь между различными узлами документа.

DOM нужен для того, чтобы манипулировать страницей – читать информацию из HTML, создавать и изменять элементы. Получив узел, мы можем что-то сделать с ним. Все узлы в документе имеют иерархическое отношение друг к другу. Существует несколько специальных терминов для описания отношений между узлами:

- **родительский узел (parent node)** – родительским узлом по отношению к рассматриваемому объекту является узел, в который вложен рассматриваемый объект.

На примере выше по отношению к узлам `<h2>`, `<p>` и `<table>` элемент `<body>` является родительским. Для узла `<title>` родительским является узел `<head>`.

- **узлы-потомки (child node)** – узлом-потомком по отношению к рассматриваемому объекту является узел, который вложен в рассматриваемый объект. На нашей схеме по отношению к узлу `<body>` `<h2>`, `<p>` и `<table>` являются

потомками. Для узла `<head>` потомками являются `<title>` и атрибут `“meta”`.

● **узлы-братья (sibling node)** – узлы, находящиеся на одинаковом уровне вложенности по отношению к их родительскому узлу. На нашей схеме узлами-братьями являются `<body>` и `<head>` по отношению к корневому `<html>`, а также `<h2>`, `<p>` и `<table>` по отношению к родительскому узлу `<body>`.

Самый верхний узел в DOM называется корневым. Так как объект `document` не является частью DOM, в нашем примере корневым является `<html>`. Он не имеет родителей и сам является родительским узлом по отношению к `<head>` и `<body>`.

В интернете в различных материалах можно встретить разные термины – предок, потомок (прямой и непрямой), сыновья, дочери, дети, ребенок, братья, сестры, правнуки и т.д. По сути, это одно и то же. То, как вы привыкнете их называть, не имеет значения. Главное – это правильно понимать взаимосвязь между ними и уметь к ним обращаться.

Каждый объект HTML, кроме текстовых узлов, имеет свойства (атрибуты) и методы. Например:

```
1 <a href="#web-analytics" class="purchase-btn tariff-order fancybox"
  onclick="dataLayer.push({'event': 'buttonClick', 'eventCategory': 'click',
  'eventAction': 'кнопка'});">Весь перечень</a>
```

Рис. 36. Фрагмент кода

- **<a>** – это элемент HTML;

- **href, class** – атрибуты элемента **<a>**;

- **dataLayer.push ()** – это метод, который выполняется в ответ на событие **onclick** (возникает при щелчке левой кнопкой мыши на элементе, к которому добавлен атрибут **onclick**).

Google Tag Manager по умолчанию отслеживает все основные элементы и атрибуты, но не отслеживает кастомные (пользовательские) атрибуты.

Когда мы используем CSS-селекторы, мы находим элементы в DOM. для управления внешним видом HTML-документа существует формальный язык CSS. CSS это аббревиатура «**Cascading Style Sheets**» или «**Каскадные Таблицы Стилей**». Обычно CSS называют просто «стили». На текущий момент существует несколько спецификаций (CSS1, CSS2, CSS2.1, CSS3, CSS4). Последняя содержит незначительные изменения, и поэтому самой популярной и часто используемой на данный момент является CSS3.

Как правило, все стили веб-сайта выносятся в отдельный файл с расширением **.css**, к которому впоследствии можно обратиться по определенному пути.

Селекторы могут быть:

- *простыми* (напрямую идентифицируют элемент или несколько элементов страницы на основании типа элемента (или его **класса (class)** или **id.**);

● *по атрибутам* (позволяют искать элементы по их атрибутам **attributes** и значениям этих атрибутов. Делятся на две категории: селекторы наличия и значения атрибута и селекторы подстроки в значении атрибута);

● *псевдоселекторами* (те, что выбирают только часть элемента или только элементы в определенном контексте. Бывают двух типов – псевдоклассы и псевдоэлементы);

● *комбинированными* (селекторы и их взаимосвязи между друг другом выражаются посредством комбинаторов);

Давайте рассмотрим каждый вид подробнее.

## Простые селекторы

К ним относятся: *селекторы классов* (*class selectors*) и *селекторы по ID*.

Селектор класса состоит из точки (.), за которой следует имя класса. Имя класса – это любое значение, не содержащее пробелов в HTML-атрибуте **class**.

```
ments Console Sources Network Performance Memory
<br>
<input type="hidden" class="hidden" value="main" name=":
<br>
<input type="submit" class="form_button" value="Получит
</p>
▶ <div class="form_label_checkbox">...</div>
```

Каскадные таблицы стилей  
(файл с расширением .css)

```
. convert-new_min_81&ver=4.9.6:1
form_horizontal.full_button
.form_button,
.form_horizontal.full_button
.form_button_inline {
display: block;
width: 100%;
margin-top: 15px;
}
```

Рис. 37. Пример селектора классов

Пример **.form\_button**. Это означает, что **css-селектор .class** соответствует всем элементам с классом **.form\_button** (**class="form\_button"**).

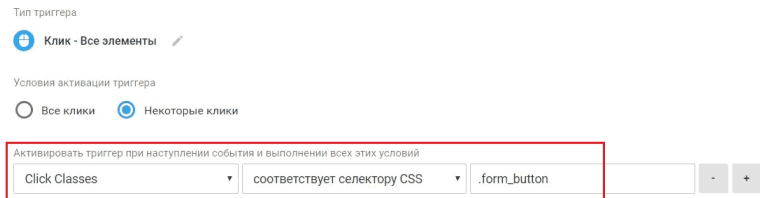


Рис. 38. Клик по элементам, соответствующим CSS-селектору класса **.form\_button**

ID-селектор состоит из символа решетки (**#**), за которым следует ID нужного элемента. Каждый элемент может иметь уникальный идентификатор, назначаемый атрибутом **id**.

```
::before
▼ <div class="w3-col 110 m12" id="main">
  ▶ <div id="mainLeaderboard" style="overflow:hidden;...</div> =
  ▶ <h1>...</h1>
  ▶ <div class="w3-clear nextprev">...</div>
  <hr>

```

Таблица стилей в .css

```
100 #main {padding:16px}
101 #mainLeaderboard {height:90px;}
102 #right {text-align:center;padding:16px}
103 #right a {text-decoration:none}
104 #right a:hover {text-decoration:underline}
105 #skyscraper {min-height:600px}

```

Рис. 39. Пример селектора по ID

Пример **#maiLeaderboard**. Это означает, что CSS-селектор **#id** соответствует всем элементам с идентификатором **maiLeaderboard** (`id="mainLeaderboard"`).

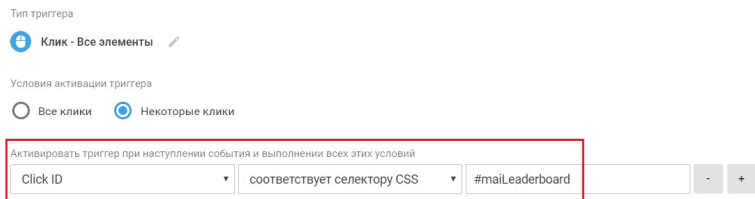


Рис. 40. Клик по элементам, соответствующим CSS-селектору

с идентификатором **#maiLeaderboard**

Для выбора всех элементов на странице существует универсальный селектор (\*). Помните этот спецсимвол в регулярных выражениях? Там он соответствует 0 или более предыдущих элементов, а здесь служит для одновременного установления одного стиля для всех элементов веб-страни-

цы. Например, чтобы задать шрифт или начертание текста.

Рассмотрим пример, в котором нам необходимо отследить клик по любому блоку внутри блока `<div class="block">`.

```
<div class="block">  
  <div class="firstscreen dark-1"> ... </div>  
  <div class="secondscreen dark-1"> ... </div>  
  <div class="thirdscreen dark-1"> ... </div>  
  <div class="thirdscreen dark-1"> ... </div>  
  <div class="fourthscreen dark-1"> ... </div>  
  <div class="fifthscreen dark-1"> ... </div>  
</div>
```

Рис. 41. Пример использования универсального селектора \*

Настроить в GTM триггер на клик по элементу с классом "block", используя встроенную переменную Click Classes = "block", не получится, так как сам клик будет приходиться на один из узлов-потомков блока block. А настраивать на каждый внутренний элемент (firstscreen dark-1, secondscreen dark-1 и т.д.) не целесообразно.

По этой причине будем использовать универсальный селектор \* для класса .block:



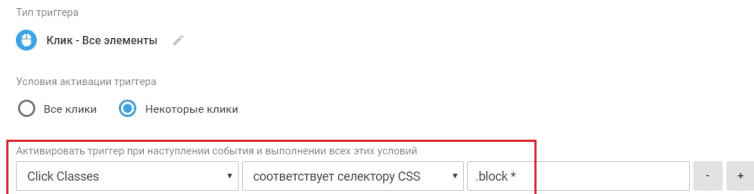


Рис. 42. Клик по элементам, соответствующим CSS-селектору

класса `.block` и всех узлов-потомков с помощью универсального селектора `*`

Такая запись означает, что необходимо отслеживать клик по всем узлам-потомкам элемента с классом "block".

В CSS есть такое понятие, как «комбинаторы». Они позволяют объединить множество селекторов, что дает возможность выбирать элементы внутри других элементов, или смежные элементы.

# Элемент

Если необходимо выбрать все определенные элементы на страницы, используйте конструкцию «элемент».

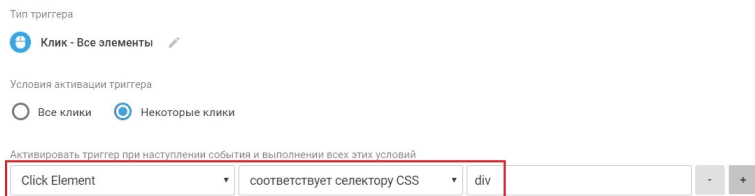


Рис. 43. Клик по элементам, соответствующим CSS-селектору по div

Триггер в GTM будет активироваться при выполнении события по всем элементам “**div**” на странице.

## Элемент, Элемент

Если необходимо выбрать все элементы “**div**” и “**p**”, то используйте конструкцию «элемент, элемент».

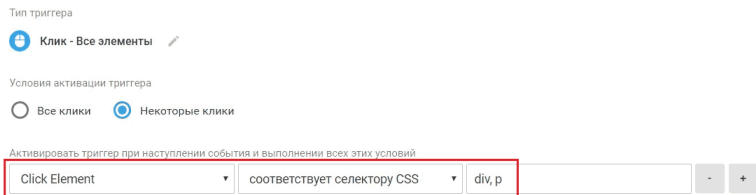


Рис. 44. Клик по элементам, соответствующим CSS-селектору по div, p

Триггер в GTM будет активироваться при выполнении события по всем элементам “div” и “p” на странице.

## Элемент Элемент

Если необходимо выбрать элемент, вложенный в другой элемент, то используйте конструкцию «элемент элемент».

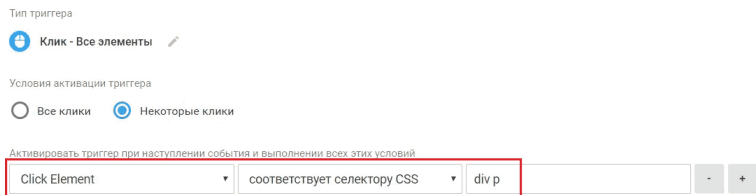


Рис. 45. Клик по элементам, соответствующим CSS-селектору div p

Триггер в GTM будет активироваться при выполнении события по всем элементам “**p**” на странице внутри элементов “**div**”.

Не обязательно должен быть «родительский узел» – «узел потомка», но необходимо, чтобы первый элемент был перед вторым.

## Элемент>Элемент

Пример `div>span`. Конструкция «элемент>элемент» выбирает все дочерние элементы “**span**”, у которых родитель – элемент “**div**”.

Разберем это на конкретном примере:

```
HTML ▾  
1 ▾ <div>  
2 ▾ <p>Мороз и солнце; день чудесный!  
3 ▾ </p>  
4 ▾ <span>  
5 ▾ <span>  
6 ▾ <p>Еще ты дремлешь, друг прелестный -</p>  
7 ▾ </span>  
8 ▾ </span>  
9 ▾ <p>Мороз и солнце; день чудесный!</p>  
10 ▾ <span>  
11 ▾ <p>Еще ты дремлешь, друг прелестный -</p>  
12 ▾ </span>  
13 ▾ <p>Мороз и солнце; день чудесный!</p>  
14 ▾ </div>
```

Рис. 46. Пример конструкции «элемент>элемент»

Добавляем в Google Tag Manager условие активации триггера:

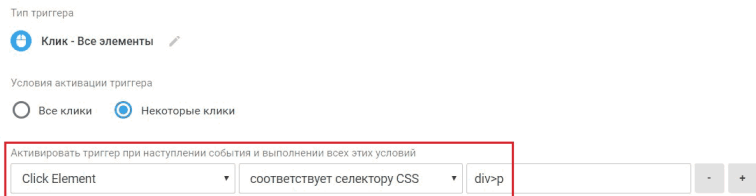


Рис. 47. Клик по элементам, соответствующим CSS-селектору div>p

Триггер в GTM будет активироваться при выполнении события по дочерним элементам “p” на странице внутри элемента ”div”, то есть тех, которые на скриншоте выше выделены зеленым цветом и соответствуют строке «Мороз и солнце; день чудесный!».

## Элемент+Элемент

Пример div+a. Конструкция «элемент+элемент» позволяет выбрать все элементы <a>, которые расположены сразу после закрывающегося тега <div>.

```
▼ <div class="first-buttons">
▼ <a href="#popup-audit" class="button green">Консультация</a>
▼ <a href="#price" class="button blue">Заказать рекламу</a>
  </div>
▼ <a class="policy-a fancybox" href="#policy-2">Пользовательское
  соглашение</a>
```

Рис. 48. Конструкция «элемент+элемент» на примере div+a

Таким образом, если мы зададим условие активации триггера в GTM по Click Element и соответствию селектору CSS “div+a”, то тег будет активироваться по событию на странице по тегу <a>, который идет сразу после закрывающегося тега ”div” (выделено зеленым), игнорируя другие теги <a> внутри “div” (выделены красным).

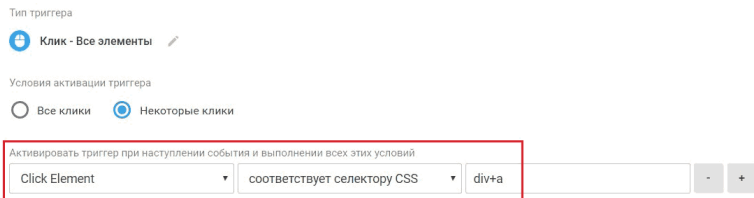


Рис. 49. Клик по элементам, соответствующим CSS-селектору `div+a`

## Элемент~Элемент

Пример `p~ol`. Конструкция «элемент~элемент» позволяет выбрать все элементы “**ol**”, которым предшествует элемент “**p**”.

# Селекторы по атрибутам

У элементов HTML есть атрибуты – это дополнительные значения, которые настраивают элементы или регулируют их поведение различным способом. Справочная информация по HTML атрибутам доступна по ссылке: <https://developer.mozilla.org/ru/docs/Web/HTML/Attributes>

Существует особый вид селекторов, позволяющий искать элементы по их атрибутам и значениям. Для их записи используются квадратные скобки [].

## Селекторы наличия и значения атрибута

Эти селекторы выделяют элементы на основе точного значения атрибута:

● **[атрибут]** – выбирает все элементы с атрибутом ”**атрибут**”, вне зависимости от его значения;

Пример `[target]` – выбирает все элементы на странице с атрибутом **target**.

**Примечание:** по умолчанию, при переходе по ссылке документ открывается в текущем окне или фрейме. При необходимости, это условие может быть изменено атрибутом `target` тега `<a>`.

● **[атрибут=”значение”]** – выбирает все элементы с атрибутом “**атрибут**”, которое имеет значение “**значение**”.



Пример [target=\_blank] – выбирает все элементы с атрибутом “target”, который имеет значение “\_blank”.

● [атрибут~=”значение”] – выбирает все элементы с атрибутом “атрибут”, в значении которого (в любом месте) встречается “значение” в виде отдельного слова. Если говорить простыми словами, то является аналогом условия «содержит».

```
1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>Пример</title>
6 <style>
7 [alt~="flower"] { border: 5px double red; }
8 </style>
9 </head>
10 <body>
11 <p>Картинка с атрибутом alt, содержащим слово "flower", будет иметь красную рамку.</p>
12 <p>
13 
14 
15 
16 </p>
17 </body>
18 </html>
```

Рис. 50. Пример использования [атрибут~=”значение”]

Тильда (~) в данном селекторе является ключом для осуществления выбора элемента на основании наличия в значении атрибута нужного слова, отделенного пробелом. Если тильда будет пропущена, то получится требование к точному значению.

## Селекторы подстроки в значении атрибута

Селекторы по атрибутам этого типа еще называют «Селекторы типа регулярных выражений (RegExp-like selectors)», поскольку они предлагают способ проверки условий, подобно тому, как это делают регулярные выражения. Подробно о регулярных выражениях и примерах их использования написано в моей другой книге «Google Analytics для гооглят: Практическое руководство по веб-аналитике (2018)».

Селекторы типа регулярных выражений:

- **[атрибут|=”значение”]** – выбирает все элементы с атрибутом “атрибут”, имеющие значение в точности равное “значение” или начинающееся с “значение-“ (обратите внимание, что символ | – это не ошибка, он нужен для работы с языковой кодировкой.)

Значение может быть единственным словом в атрибуте, например, lang="ru" или с добавлением других слов через дефис, например, lang="ru-region".

На примере ниже для “lang|=ru” задается стиль (color:green). Результатом будет измененный цвет текста у атрибута “ru” Привет! и “ru-region” Здравствуйте. (выделено зеленым), поскольку **[атрибут|=”значение”]** подразумевает «точное» значение или же «начинающееся с».

```
1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>Пример</title>
6 <style>
7 [lang|=ru] { color:green; }
8 </style>
9 </head>
10 <body>
11 <p lang="ru">Привет!</p>
12 <p lang="en">Hi!</p>
13 <p lang="en">Hello!</p>
14 <p lang="ru-region">Здравствуйе.</p>
15 </body>
16 </html>
```

Рис. 51. Пример использования [атрибут|=”значение”]

Далее идут три CSS-селектора, которые используют регулярные выражения и в их конструкции присутствуют специальные символы, такие же, как в Google Analytics при фильтрации или поиске. Это ^, \$ и \*.

● [атрибут^=”значение”] – выбирает каждый элемент с атрибутом “атрибут”, значение которого начинается с “значение”.

Пример `a[href^="https"]` – выбирает каждый элемент `<a>` с атрибутом `href`, значение которого начинается с `"https"`.

● [атрибут\$=”значение”] – выбирает все элементы с

атрибутом “**атрибут**”, значение которого заканчивается на “**значение**”.

```
1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>Пример</title>
6 <style>
7 <div[class$="test"] {color: red;}
8 </style>
9 </head>
10 <body>
11 <div class="1_test">Первый элемент div.</div>
12 <div class="2">Второй элемент div.</div>
13 <div class="test">Третий элемент div.</div>
14 <p class="test">Абзац с текстом.</p>
15 </body>
16 </html>
```

Рис. 52. Пример использования [атрибут\$=”значение”]

На примере выше для элемента `div` и класса с атрибутом, значение которого заканчивается на “**test**”, применяется стиль `{color:red}`; (задается красный цвет). Таким образом два элемента из четырех на странице будут отображены шрифтом красного цвета. Это `div class =”1_test` (Первый элемент `div.`) и `div class =”test` (Третий элемент `div.`). Последний тоже имеет `class = “test` (Абзац с текстом), однако он входит в элемент `<p>`, а не `div`, и поэтому не будет выделен

красным цветом.

Или еще один пример `a[href$=".pdf"]`. В этом случае вы можете отслеживать каждый элемент тега `<a>` с атрибутом `"href"`, значение которого оканчивается на `".pdf"`. В Google Tag Manager с помощью данного атрибута можно отслеживать скачивание файлов на сайте, клики по определенным картинкам формата `".png"`, `".jpg"`, `".gif"` и т.д.

● **[атрибут\*="значение"]** – выбирает все элементы с атрибутом `"атрибут"`, в значении которого (в любом месте) встречается подстрока (в виде отдельного слова или его части) `"значение"`.

Проиллюстрируем это на следующем примере:

```
1 <!doctype html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Пример</title>
6     <style>
7     <div[class*="test"] {color: blue;}
8   </style>
9 </head>
10 <body>
11 <div class="1_test">Первый элемент div.</div>
12 <div class="testirovshik_2">Второй элемент div.</div>
13 <div class="testing">Третий элемент div.</div>
14 <p class="test">Абзац с текстом.</p>
15 </body>
16 </html>
```

Рис. 53. Пример использования `[атрибут*="значение"]`

В стиле для элемента `div` и класса с атрибутом, значение которого “test”, применяется стиль `{color:blue}`; (задается синий цвет). Таким образом будут выбраны первые три элемента вне зависимости от того, «точное» ли было вхождение условия **class\*="test"** или нет, «начиналось с» или «заканчивалось на». Сам факт наличия значения “test” во всех трех элементах `div` селектора подстроки в значении атрибута позволяет нам выбрать все эти элементы. Последний вариант (Абзац с текстом) тоже имеет значение “test”, однако он не входит в элемент `div`, а является составляющей тега `<p>`, поэтому он не будет выделен синим цветом.

# Псевдоклассы

Псевдоклассы – это дополнения к обычным селекторам, которые делают их еще точнее и мощнее. Они определяют динамическое состояние элементов, которое изменяется с помощью действий пользователя, а также положение в дереве документа. Примером такого состояния служит текстовая ссылка, которая меняет свой цвет при наведении на нее курсора мыши. Это реализуется с помощью псевдокласса **:hover**.

Псевдокласс добавляется к селектору с помощью символа двоеточия **: (селектор:псевдокласс)**.

В Google Tag Manager псевдоклассы также можно применять для активации переключателей элементов интерфейса, например, *checkbox* и *флажки (radio)*, или для отслеживания кликов по определенным элементам из выпадающего списка.

Псевдокласс – это селектор, который используется для выбора того, что не может быть выбрано с помощью других селекторов или может быть выбрано, но достаточно трудным способом. Их существует большое количество, но на практике в GTM используются некоторые. Вот одни из самых популярных:

● **:checked, :selected, :disabled, :enabled** – различные состояния элементов *input*;

● **:nth-child(n)** – позволяет отследить определенный эле-

мент списка. Например, “**ul>li:nth-child(4)**” позволяет отследить четвертый элемент;

● **:nth-last-child(n)** – псевдокласс, противоположный предыдущему, который позволяет отследить определенный элемент списка, отчет элементов идет с конца. Например, селектор CSS “**:nth-last-child(2)**” вернет предпоследний элемент списка;

● **:not(селектор)** – псевдокласс отрицания. Выбирает все элементы, кроме того элемента, что в скобках.



Псевдокласс	Пример	Описание
<code>:link</code>	<code>a:link</code>	Выбор всех не посещенных ссылок
<code>:visited</code>	<code>a:visited</code>	Выбор всех посещенных ссылок
<code>:active</code>	<code>a:active</code>	Выбор активной ссылки
<code>:hover</code>	<code>a:hover</code>	Выбор ссылки при наведении курсора мышки
<code>:focus</code>	<code>input:focus</code>	Выбор элемента <code>&lt;input&gt;</code> , который находится в фокусе
<code>:first-child</code>	<code>p:first-child</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является первым дочерним элементом своего родителя
<code>:lang(язык)</code>	<code>p:lang(ru)</code>	Выбор каждого элемента <code>&lt;p&gt;</code> с атрибутом <code>lang</code> , значение которого начинается с "ru"
<code>:first-of-type</code>	<code>p:first-of-type</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является первым из элементов <code>&lt;p&gt;</code> своего родительского элемента
<code>:last-of-type</code>	<code>p:last-of-type</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является последним из элементов <code>&lt;p&gt;</code> своего родительского элемента
<code>:only-of-type</code>	<code>p:only-of-type</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является единственным элементом <code>&lt;p&gt;</code> своего родительского элемента
<code>:only-child</code>	<code>p:only-child</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является единственным дочерним элементом своего родительского элемента
<code>:nth-child(n)</code>	<code>p:nth-child(2)</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является вторым дочерним элементом своего родительского элемента
<code>:nth-last-child(n)</code>	<code>p:nth-last-child(2)</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является вторым дочерним элементом своего родительского элемента, считая от последнего дочернего элемента
<code>:nth-of-type(n)</code>	<code>p:nth-of-type(2)</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является вторым дочерним элементом <code>&lt;p&gt;</code> своего родительского элемента
<code>:nth-last-of-type(n)</code>	<code>p:nth-last-of-type(2)</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является вторым дочерним элементом <code>&lt;p&gt;</code> своего родительского элемента, считая от последнего дочернего элемента
<code>:last-child</code>	<code>p:last-child</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который является последним элементом своего родительского элемента
<code>:root</code>	<code>:root</code>	Выбор корневого элемента в документе
<code>:empty</code>	<code>p:empty</code>	Выбор каждого элемента <code>&lt;p&gt;</code> , который не содержит дочерних элементов (включая текст)
<code>:target</code>	<code>:target</code>	Выбор текущего целевого элемента на странице, то есть элемента, к которому был осуществлен переход по ссылке внутри страницы
<code>:enabled</code>	<code>input:enabled</code>	Выбор каждого включенного элемента <code>&lt;input&gt;</code>
<code>:disabled</code>	<code>input:disabled</code>	Выбор каждого выключенного элемента <code>&lt;input&gt;</code>
<code>:checked</code>	<code>input:checked</code>	Выбор элемента <code>&lt;input&gt;</code> , выбранного по умолчанию или пользователем
<code>:not(селектор)</code>	<code>:not(p)</code>	Выбор всех элементов, кроме элемента <code>&lt;p&gt;</code>

Рис. 54. Таблица псевдоклассов с примерами

# Псевдоэлементы

Помимо псевдоклассов, существуют еще и псевдоэлементы. Псевдоэлемент – это виртуальный элемент, который не существует в явном виде в дереве элементов документа. Они используются для выбора тех частей элемента, которые не могут быть выбраны с помощью других селекторов, а также для стилизации не всего элемента, а отдельных его частей.

Псевдоэлемент	Пример	Описание
<code>::first-letter</code>	<code>p::first-letter</code>	Выбор первой буквы каждого элемента <code>&lt;p&gt;</code> .
<code>::first-line</code>	<code>p::first-line</code>	Выбор первой строки каждого элемента <code>&lt;p&gt;</code> .
<code>::before</code>	<code>p::before</code>	Добавляет элемент с содержимым перед содержимым каждого элемента <code>&lt;p&gt;</code> .
<code>::after</code>	<code>p::after</code>	Добавляет элемент с содержимым после содержимого каждого элемента <code>&lt;p&gt;</code> .

Рис. 55. Таблица псевдоэлементов с примерами

Рекомендую перейти на сайт [w3schools.com](http://w3schools.com) и изучить два материала для лучшего понимания данной темы:

**CSS-селекторы:** [w3schools.com/cssref/css\\_selectors.asp](http://w3schools.com/cssref/css_selectors.asp)

**Демонстрация селекторов:** [w3schools.com/cssref/trysel.asp](http://w3schools.com/cssref/trysel.asp)

По первой ссылке представлены все CSS-селекторы и псевдоклассы с примерами и описанием их работы (на английском языке). По второй вы можете наглядно увидеть,

# как работают различные селекторы.

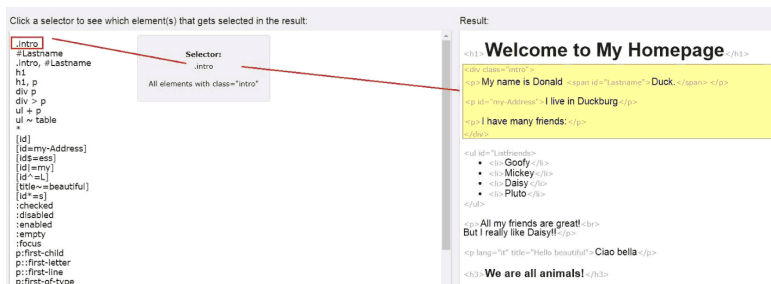


Рис. 56. Демонстрация селекторов на сайте w3schools.com

Выбрав в левой части экрана один из селекторов, справа вы увидите подсвеченный результат.

Также для закрепления пройденного материала рекомендую прочитать статью «Основы CSS-селекторов на примере котиков» и несколько разделов из справки разработчиков [developer.mozilla.org](https://developer.mozilla.org).

Для того, чтобы правильно обращаться к элементам на странице и применять их в Google Tag Manager для отслеживания определенных событий, нужно просто глубже разобраться в теме CSS-селекторов, понять их взаимосвязь друг с другом и использовать инструменты, которые упрощают их обнаружение.

Разберем три варианта определения CSS-селекторов.

# Консоль разработчика

Традиционный способ, с помощью консоли разработчика любого браузера. Я пользуюсь Google Chrome, поэтому разберем пример на нем. Выбрав отслеживаемый элемент на странице, нажмите на него правой кнопкой мыши и **«Просмотреть код»**.

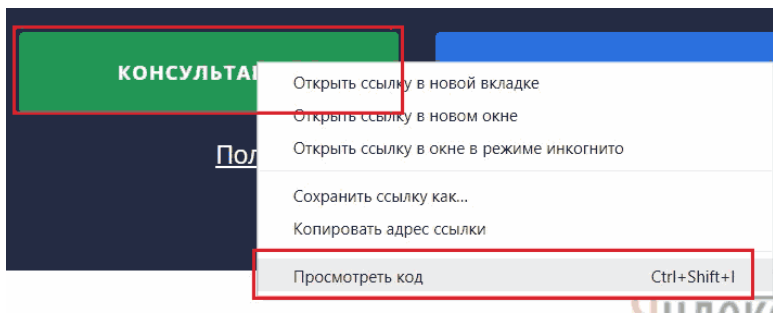


Рис. 57. Правой кнопкой мыши – Просмотреть код

После того, как мы удостоверились, что выбрали нужный элемент, в открывшемся коде страницы еще раз нажимаем правой кнопкой мыши по нужному элементу. Далее **«Copy» – «Copy selector»**.

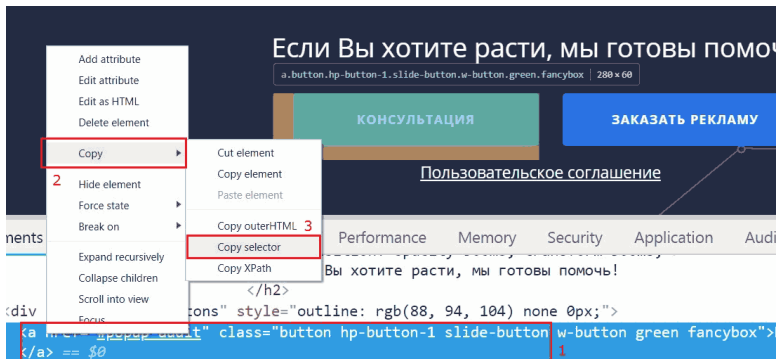


Рис. 58. Правой кнопкой мыши – Copy – Copy selector

В буфер обмена у вас скопировался селектор. Можно вставить его в любой текстовый документ и посмотреть, как он будет выглядеть. В моем примере он выглядит так: *#home > div > div > div > div > div > a.button.hp-button-1.slide-button.w-button.green.fancybox*

Получилась очень длинная цепочка взаимосвязей элементов друг с другом. Несмотря на это его можно использовать в Google Tag Manager для настройки отслеживания события. Однако тот же селектор можно записать и в таком виде: *a.button.hp-button-1.slide-button.w-button.green.fancybox*, просто убрав общую часть в начале.

# CSS Selector Tester

Чтобы проверить правильность выбора определенного элемента для настройки триггера и его отладки в GTM, можно воспользоваться специальным расширением для браузера Google Chrome, которое называется CSS Selector Tester. Скачать его можно по ссылке.

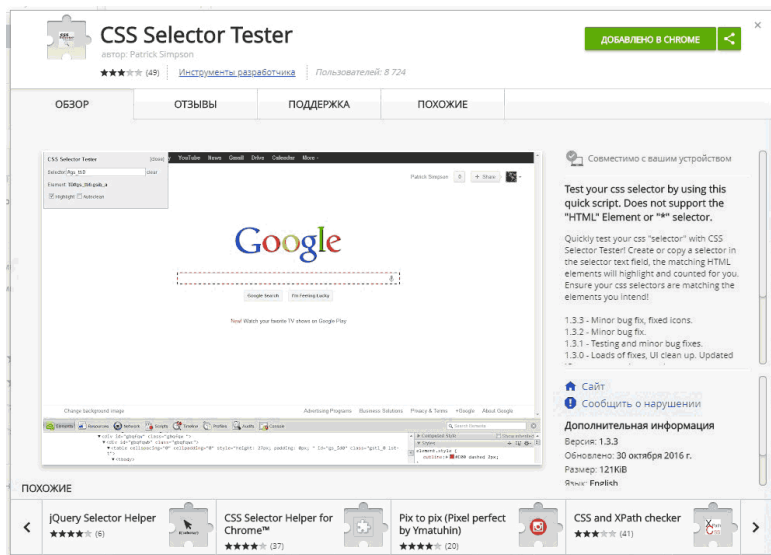


Рис. 59. Расширение для браузера CSS Selector Tester

После установки в правом верхнем углу у вас должен

отобразиться новый значок. Перейдите на необходимую страницу где хотите найти селектор и нажмите на иконку:



Рис. 60. Иконка CSS Selector Tester

У вас откроется дополнительное окно со строкой ввода селектора. Просто вставьте ваш селектор, и он подсветится на странице. В качестве примера я добавлю *a.button.hp-button-1.slide-button.w-button.green.fancybox* – укороченную версию селектора, который мы определили с помощью консоли разработчика.

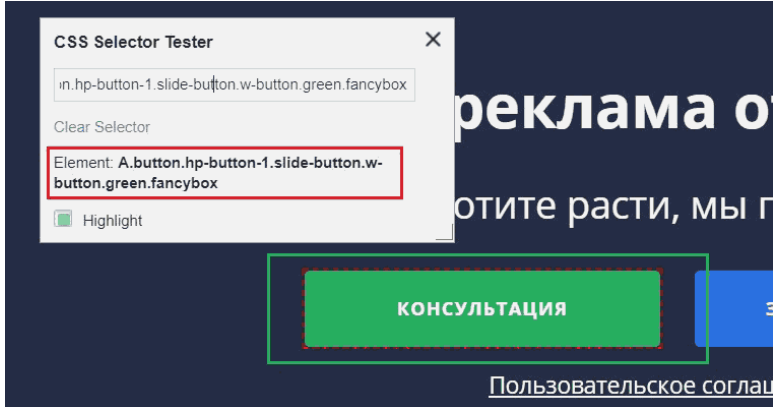


Рис. 61. Поиск селектора на странице

CSS Selector Tester нашел такой элемент на странице и выделил его границы красной пунктирной линией. Вот так он выделил все элементы div на странице:

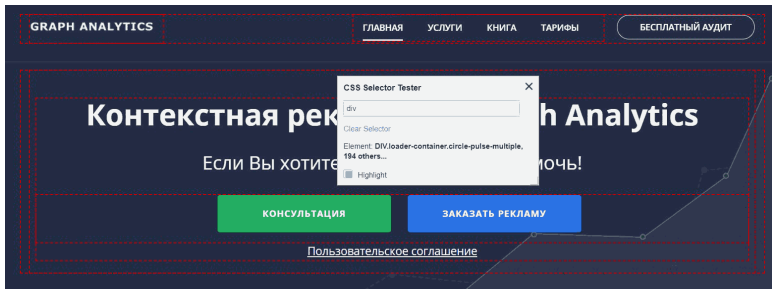


Рис. 62. Элементы div на странице



Если бы мы ввели неверный селектор, то сообщение было бы «**undefined**» (не определено) и элемент не был бы найден на странице.

## SelectorGadget

Для тех, кто не хочет лазать в консоль разработчика, инспектировать там элементы страницы и копировать конечный вариант селектора, может воспользоваться еще одним расширением для браузера Google Chrome «**SelectorGadget**».

Он позволяет легко определять селекторы CSS (при наведении на элемент в отдельном поле показывает его значение) и для наглядности выделяет выбранный элемент зеленым, а схожие — желтым.

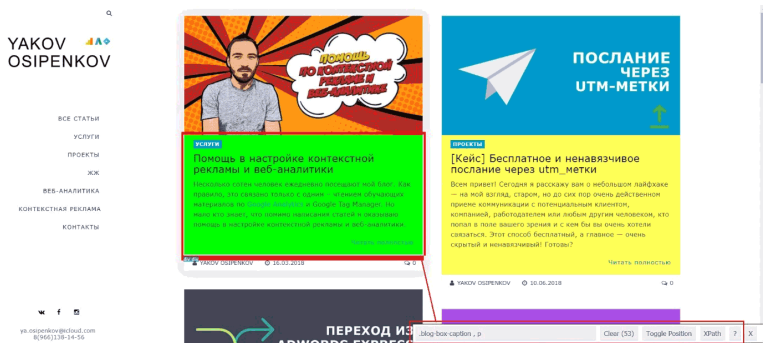


Рис. 63. Пример работы расширения SelectorGadget

Скачать расширение SelectorGadget можно по ссылке. После установки в правом верхнем углу у вас должен отображаться новый значок.



Рис. 64. Иконка SelectorGadget

# Селекторы в jQuery

Как мы узнали ранее, jQuery – это библиотека JavaScript, которая фокусируется на взаимодействии JavaScript, HTML, CSS и служит для облегчения работы разработчика. На данный момент она является самой распространенной библиотекой JS в мире.

В природе существуют и другие JS-библиотеки, фреймворки и инструменты, например, **React**, **AngularJS**, **Backbone.js**, **Ember.js** и т.д. Но в свое время именно jQuery произвела революцию в программировании клиентской части веб-приложений, введя селекторы CSS для доступа к узлам DOM-дерева, обработчики событий, анимации и AJAX-запросы.

JavaScript является фундаментальным языком и для того, чтобы создать всплывающее окно или выпадающий список, разработчику потребуется написать большое количество строк кода. В jQuery это уже все реализовано, достаточно всего лишь использовать готовую функцию.

Библиотека jQuery на сайте по умолчанию отсутствует. Для ее подключения необходимо скачать актуальную версию с официального сайта [jquery.com/download](http://jquery.com/download), загрузить ее на сервер, а затем подключить с помощью фрагмента кода:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <!--Подключаем библиотеку-->
7   <script src="js/jquery-3.3.1.min.js"></script>
8 </head>
9 <body>
10 </body>
11 </html>
12
```

Рис. 65. Подключение библиотеки jQuery через путь к файлу (версия 3.3.1)

Существует альтернативный способ подключить jQuery на страницы вашего сайта (не закачивая библиотеку на сервер). Можно подключить библиотеку, которая находится не на вашем сервере, а на серверах CDN. Существуют несколько таких хранилищ, наиболее известные и надежные из них Google CDN, Microsoft CDN, а также CDN, который организовали создатели jQuery.

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
```

Рис. 66. Подключение библиотеки jQuery через CDN

Можно подключить jQuery и через GTM. Для этого необходимо создать пользовательский HTML-тег и добавить туда фрагмент кода выше, а в качестве триггера активации выбрать «All Pages» (Все страницы).

Селекторы в jQuery базируются на CSS селекторах (см. выше), они также используются для определения элемента или группы элементов.

Все селекторы в jQuery начинаются со знака доллара и круглых скобок \$ (), например, \$('div'). В этом случае будет осуществлен поиск всех элементов div на странице, \$('.className') найдет все элементы с классом className, \$('#sidebar') найдет все элементы на странице с идентификатором sidebar и т.д.

**Примечание:** при использовании метасимволов (#;&,+\*~':"!^\$[]()=>|/) в значении любого идентификатора, класса или названия атрибута, необходимо экранировать эти символы в селекторах с помощью двух обратных слэшей \

```
$('#input[value = 3.5]') /* неправильно */  
$('#input[value = 3\\.5]') /* правильно */
```

Рис. 67. Экранирование специальных символов с помощью \

В процессе работы с Google Tag Manager jQuery позволяет удобнее решать множество задач. Задачи можно разделить

на два типа:

1. взять элемент, который у нас находится на сайте, например, значение цены, название продукта, категорию продукта и т.д.;

2. отследить факт того, что у нас произошел клик по какому-либо элементу, зафиксировано событие или пользователь заполнил и отправил форму на сайте.

При настройке некоторых событий в GTM иногда возникают ситуации, когда необходимо получить дополнительную информацию, связанную с элементом, над которым произошло событие. Но эту информацию так просто нельзя получить через CSS-селекторы, поскольку она может находиться в структуре других элементов, которые логически связаны с рассматриваемым элементом. В это случае можно воспользоваться функциями библиотеки jQuery.

Давайте разберем на нескольких примерах работу библиотеки jQuery с набором элементов. Для этого я перейду на свой сайт [osipenkov.ru](http://osipenkov.ru) и вызову консоль разработчика, а затем перейду на вкладку **Console**.

С помощью `$("div").css("border", "1px solid red");` выберем все элементы div на странице и обведем их в красную рамку в 1 пиксель.

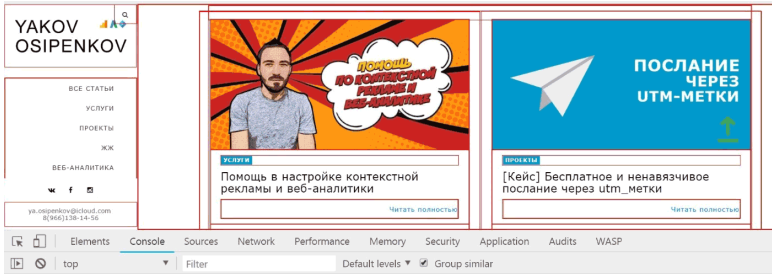


Рис. 68. Все div на странице обведены в красную рамку шириной в 1 пиксель

**Примечание:** мой блог работает на WordPress. В нем стандартный способ обращения к элементам через `$ ()` отключен (в конце файла `.../wp-includes/js/jquery/jquery.js` прописана строчка `jQuery.noConflict();`, которая отключает работу с элементами страницы через `$.`), так как другие библиотеки также могут использовать данный механизм обращения к элементам. Поэтому вместо знака доллара я использовал альтернативную конструкцию `jQuery ()`. Для изменения способа обращения на привычный `$` необходимо изменить фрагмент кода. Решение проблемы ищите в интернете с пометкой «*jQuery не работает в WordPress*».

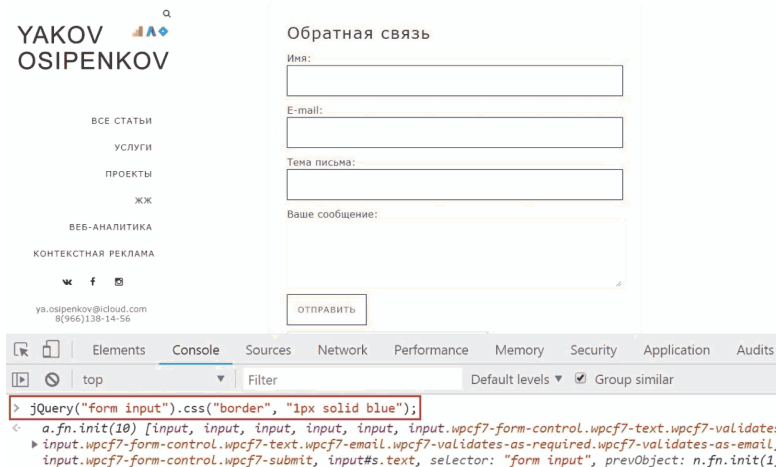
Теперь давайте поработаем с формами. Перейдем на страницу Контакты <https://osipenkov.ru/contacts/> где есть форма

обратной связи.

Добавим в консоль такую строчку:

```
$("form input").css("border", "1px solid blue");
```

Данная конструкция добавит рамку всем input-ам, которые являются потомками элемента form. Это пример так называемого «*parent child*», когда выбираются все элементы input, являющиеся узлами-потомками для родительского элемента form.



The image shows a browser window with a contact form titled "Обратная связь". The form includes fields for "Имя:", "E-mail:", "Тема письма:", and "Ваше сообщение:", and a "ОТПРАВИТЬ" button. The browser's developer console is open, showing a jQuery command: `jQuery("form input").css("border", "1px solid blue");` which has been executed. The console also displays the DOM tree structure for the form elements.

Рис. 69. Все div на странице обведены в красную рамку шириной в 1 пиксель

В качестве еще одного простого примера разберем вывод



значения заполненного поля «Имя» с `name="your-name"`.

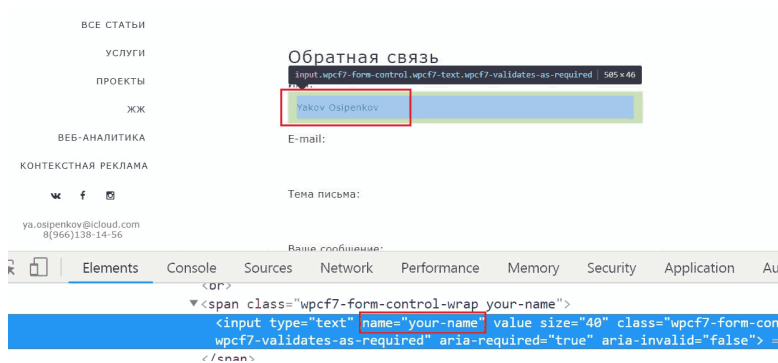


Рис. 70. Поле «Имя» с `name="your-name"`

В консоли разработчика вводим такую конструкцию:

```
$('[name="your-name"]').val()
```

Метод `.val()` позволяет получать и изменять значения элементов форм. Для элементов `input` это значение атрибута `name`, то есть в нашем случае `your-name`.

Получим результат **Yakov Osipenkov**.

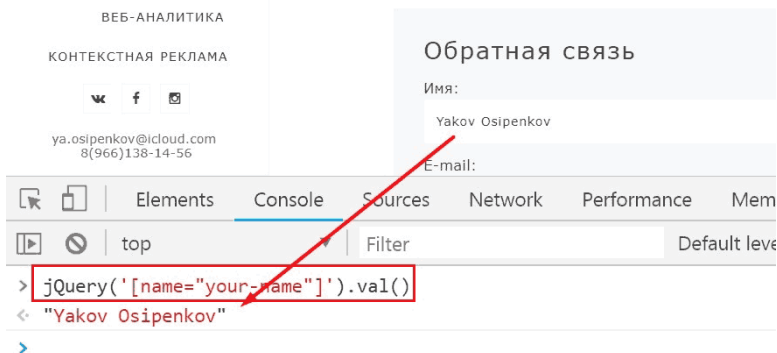


Рис. 71. Результат `$('[name="your-name"]').val()`

Разобранные выше примеры являются самыми простыми в освоении. При работе с Google Tag Manager вы будете использовать гораздо более сложные конструкции с различной цепочкой методов и другим набором элементов. Приведу пример некоторых функций jQuery перемещения по дереву DOM:

Функция	Описание
<code>.children()</code>	Находит все дочерние элементы у выбранных элементов. При необходимости, можно указать селектор для фильтрации
<code>.closest()</code>	Находит ближайший, соответствующий заданному селектору элемент, из числа следующих: сам выбранный элемент, его родитель, его прародитель, и так далее, до начало дерева DOM
<code>.find()</code>	Находит элементы по заданному селектору, внутри выбранных элементов
<code>.next()</code>	Находит элементы, которые лежат непосредственно после каждого из выбранных элементов
<code>.nextAll()</code>	Находит элементы, которые лежат после каждого из выбранных элементов
<code>.nextUntil()</code>	Находит элементы, которые лежат после каждого из выбранных, но не дальше элемента, удовлетворяющего заданному селектору
<code>.offsetParent()</code>	Возвращает ближайшего предка с позиционированием, отличным от <code>static</code> (позиционирование по умолчанию)
<code>.parent()</code>	Находит родительские элементы у всех выбранных элементов
<code>.parents()</code>	Находит всех предков у выбранных элементов, т.е. не только прямых родителей, но и прародителей, прапрародителей и так далее, до начало дерева DOM
<code>.parentsUntil()</code>	Находит предков, как и <code>.parents()</code> , но прекращает поиск перед элементом, удовлетворяющим заданному селектору
<code>.prev()</code>	Находит элементы, которые лежат непосредственно перед каждым из выбранных элементов
<code>.prevAll()</code>	Находит элементы, которые лежат перед каждым из выбранных элементов
<code>.prevUntil()</code>	Находит элементы, которые лежат перед каждым из выбранных, но не дальше элемента, соответствующего заданному селектору
<code>.siblings()</code>	Находит все соседние элементы (под соседними понимаются элементы с общим родителем)

Рис. 72. Список функций jQuery перемещения по дереву DOM

Функции фильтрации набора элементов:

Функция	Описание
<code>.eq()</code>	Возвращает элемент, идущий под заданным номером в наборе выбранных элементов
<code>.filter()</code>	Фильтрует набор выбранных элементов с помощью заданного селектора или функции
<code>.first()</code>	Возвращает первый элемент в наборе
<code>.has()</code>	Фильтрует набор выбранных элементов, оставляя те, которые имеют потомков, соответствующих селектору
<code>.is()</code>	Проверяет, содержится ли в наборе, хотя бы один элемент удовлетворяющий заданному селектору
<code>.last()</code>	Возвращает последний элемент в наборе
<code>.not()</code>	Возвращает элементы, не соответствующие заданным условиям
<code>.slice()</code>	Возвращает элементы с индексами из определенной области (например от 0 до 5)

Рис. 73. Функции фильтрации набора элементов

Также, как и с CSS-селекторами, рекомендую перейти на сайт [w3schools.com](http://w3schools.com) и изучить два материала для лучшего понимания темы селекторов jQuery:

**Селекторы jQuery:** [w3schools.com/jquery/jquery\\_selectors.asp](http://w3schools.com/jquery/jquery_selectors.asp)

**Демонстрация селекторов jQuery:** [w3schools.com/jquery/trysel.asp](http://w3schools.com/jquery/trysel.asp)

На сайте [jquery-docs.ru](http://jquery-docs.ru) есть перевод официальной документации API jQuery на русский язык.

После того, как вы познакомитесь с CSS-селекторами и уделите им достаточное количество времени на изучение, вы сможете выбрать любой элемент для отслеживания в Google Tag Manager на вашем сайте, даже если у него нет никаких

отличительных знаков, а также по достоинству оцените всю мощь и функциональность диспетчера тегов Google.

Данная статья написана с использованием следующих ресурсов – [habr.com](http://habr.com), [puzzleweb.ru](http://puzzleweb.ru), [jquery.page2page.ru](http://jquery.page2page.ru), [odesseo.com.ua](http://odesseo.com.ua), [jquery-docs.ru](http://jquery-docs.ru), [learn.javascript.ru](http://learn.javascript.ru), [htmlbook.ru](http://htmlbook.ru) и [developer.mozilla.org](http://developer.mozilla.org).

# Регистрация аккаунта

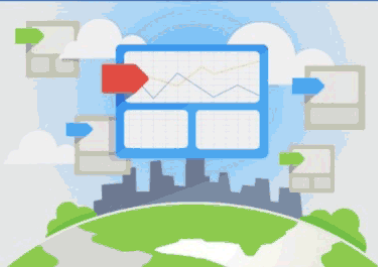
Для того чтобы начать работу с инструментом Google Tag Manager, необходимо зарегистрировать обычную почту в Gmail.com.

Однако если вы когда-либо работали с другими продуктами Google и уже имеете почту, то заводить отдельную не требуется. Авторизовавшись в аккаунте Google, просто перейдите на сайт диспетчера тегов Google или на обновленную англоязычную версию [google.com/analytics/tag-manager/](https://google.com/analytics/tag-manager/)

В первом случае нажимаем **«Регистрация»**

## Быстрый и гибкий цифровой маркетинг

В современном бизнесе скорость имеет значение. С помощью Диспетчера тегов Google можно быстро создавать и обновлять теги для веб-сайта или мобильного приложения.

[Регистрация](#)

Удобные и эффективные средства



Надежные данные



Надежный инструмент

Рис. 74. Регистрация Google Tag Manager

Во втором по любой из кнопок «**Sign up for free**» – разницы нет абсолютно никакой, кроме языка веб-страницы. Если еще не авторизовались, то сначала войдите в аккаунт (кнопка в правом верхнем углу), а затем приступайте к добавлению нового аккаунта.

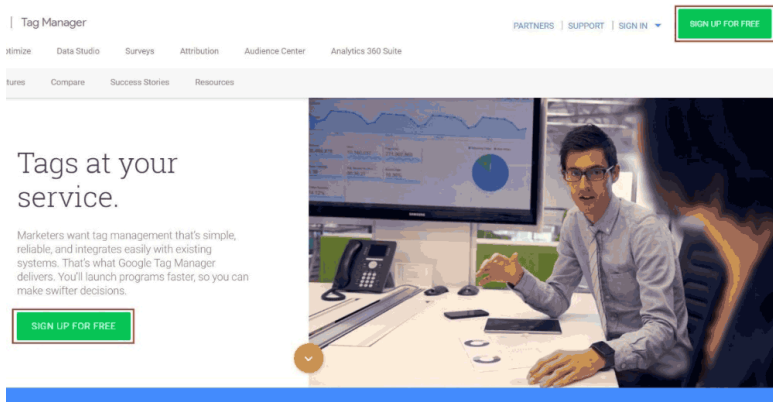


Рис. 75. Англоязычная версия Google Tag Manager

Создание аккаунта состоит из двух этапов:

- ввод названия аккаунта (лучше всего вводить простое, запоминающееся название, которое вы с легкостью сможете найти, если в учетной записи этих аккаунтов станет много);

Также Google просит вас разрешить передавать основные данные в свои продукты анонимно.

Включите функцию сравнения, разрешив передачу анонимных данных с вашего сайта. Google удалит всю идентифицирующую информацию, добавит ваши сведения в базу с данными сотен других анонимных ресурсов и будет использовать их для определения общих тенденций в службе сравнения.

Рис. 76. Согласие о сборе дополнительной информации



Текст похож на то, как аргументировали разработчики включение функции «Сравнения» в Google Analytics, не находите? Однако никаких отчетов не будет. Можете ставить галочку, можете нет – как хотите.

● ввод названия контейнера и тип использования: веб-сайт, мобильные приложения на iOS или Android, AMP-страницы.

Добавление нового аккаунта

✓ Настройка аккаунта

Название аккаунта **watersports.ru**

✓ Передавать анонимные данные в Google и другие службы

2 Настройка контейнера

Название контейнера

watersports.ru

Где будет использоваться контейнер

Веб-сайт iOS Android AMP

создать ОТМЕНА

Рис. 77. Добавление нового аккаунта в GTM

За основу структуры аккаунт-контейнер вы можете взять традиционные аккаунт-ресурс из Google Analytics, где аккаунт – это верхний уровень доступа, а контейнер – ресурс из

Google Analytics, то есть домен сайта или мобильное приложение. Чаще всего используют два подхода при именовании:

1. **аккаунт** – название компании, сотрудник, ответственный (например, Яков), группа сайтов (общее название);
2. **контейнер** – конкретные сайты или мобильные приложения;

Вот как может выглядеть структура аккаунтов и контейнеров и двух разных проектов:

1. аккаунт = контейнер;
2. аккаунт – название компании, контейнеры – сайты по регионам;

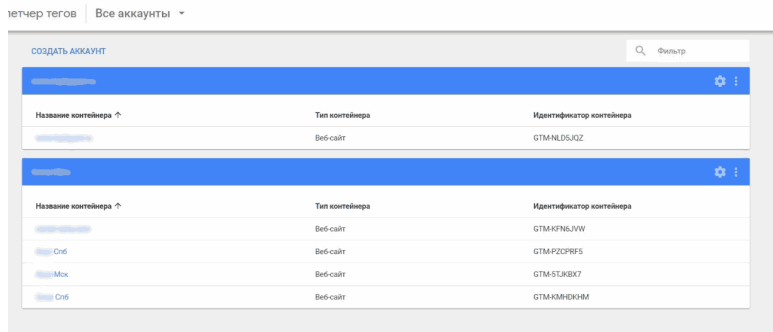


Рис. 78. Пример структуры в GTM

Вы в любой момент потом можете поменять как название аккаунта, так и название контейнера. После их ввода нажимаем «Создать». Далее внимательно читаем соглашение об

условиях использования диспетчера тегов Google, при желании переключив на русский язык. Если все устраивает (а другого и быть не может), то нажимаем «Да» в правом верхнем углу.

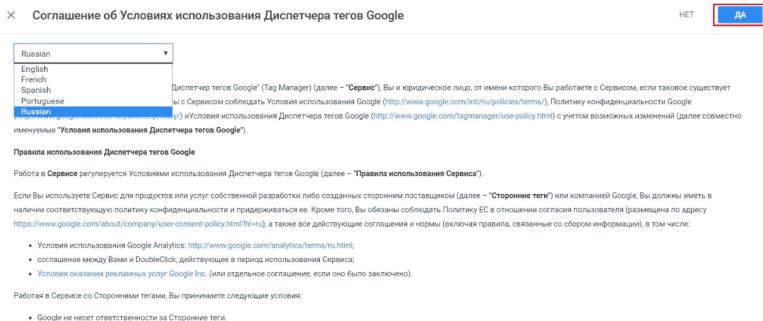


Рис. 79. Соглашение об Условиях использования Диспетчера тегов Google

## Установка контейнера

После перезагрузки страницы вам станет доступен основной интерфейс GTM. Но перед полноценным использованием всех его возможностей необходимо установить Tag Manager на каждую страницу сайта в раздел **<head>**, а часть кода после открывающего тега **<body>**. На экране появится соответствующий код:

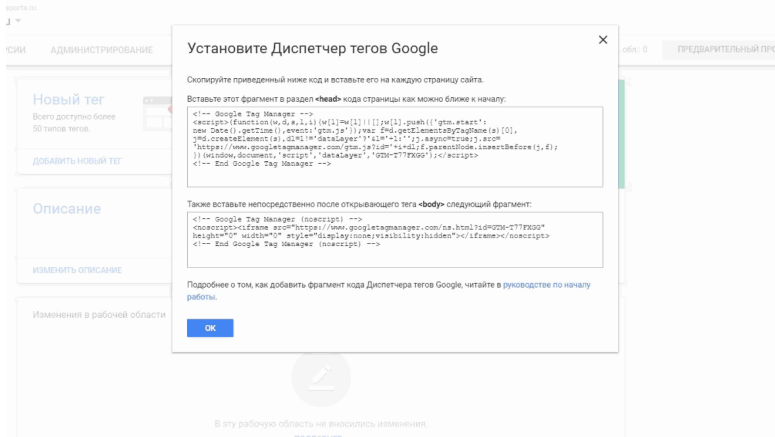


Рис. 80. Код установки Google Tag Manager

Если вы используете готовые CMS-движки, то в 99% случаев на рынке существует готовые решения, где с помощью установки дополнительного модуля или плагина можно исключить ручную вставку кода на сайт. Например, у движка интернет-магазина на cs-cart есть готовый модуль, благодаря которому весь процесс настройки GTM сводится к вставке ID контейнера в соответствующее поле:

## Настройки: [eCom Labs] Google Tag Manager

ID Контейнера ?

GTM-M2C2WP5

Для получения ID Контейнера создайте аккаунт на [tagmanager.google.com](https://tagmanager.google.com) и создайте там новый контейнер

Рис. 81. Вставка ID контейнера

После добавление кода GTM не забудьте опубликовать контейнер, нажав на кнопку «Отправить» – «Опубликовать».

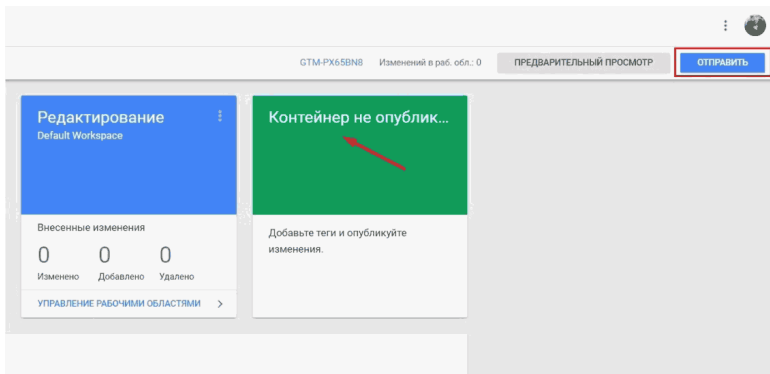


Рис. 82. Публикация контейнера GTM

До публикации ваш контейнер будет возвращать ответ

404. В **Google Tag Assistant** это будет выглядеть так:

# Google Tag Assistant



← Google Tag Manager GTM-PX65BN8

1/1 ^ v

Metadata

Code Snippet

Data Layer

URLs

Account ID

GTM-PX65BN8

Check Contained Tags

Where to optimize



HTTP response code indicates tag failed to fire: Status 404. This could be due to an empty or un-published container

Disable

Record

VIEW RECORDINGS

## Рис. 83. Google Tag Assistant. Ошибка 404

*HTTP response code indicates tag failed to fire: Status 404. This could be due to an empty or un-published container*

Если все сделаете правильно, увидите опубликованную версию и зеленую иконку в GTA:

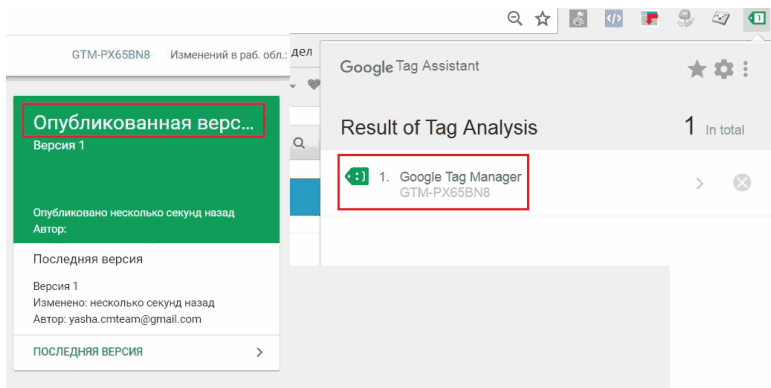


Рис. 84. Корректная установка кода GTM на сайт



# Установка Google Analytics и Яндекс.Метрика через GTM

После регистрации и установки контейнера Google Tag Manager на сайт произведем установку счетчиков веб-аналитики Google Analytics и Яндекс.Метрика.

## Установка кода Google Analytics через GTM

Получив идентификатор отслеживания в Google Analytics (UA-XXXXXXXXX-X), создадим переменную для экономии времени и избежание ошибок в дальнейшем.

**Переменная** – это объект, которому дано имя и который может принимать различные значения (вида пара “ключ-значение”). Переменные используются для хранения данных, а также если приходится несколько раз использовать одно и то же значение в различных процессах.

В Google Tag Manager используются переменные двух типов: *встроенные* и *пользовательские*. Подробнее о переменных будет разобрано дальше. А пока разберем создание пользовательской переменной. Для этого переходим в «**Переменные – Создать**».

Текущая рабочая область

Default Workspace >

Поиск

- Обзор
- Теги
- Триггеры
- Переменные**
- Папки

### Встроенные переменные ?

**НАСТРОИТЬ**

Имя ↑	Тип
Page Hostname	URL
Page Path	URL
Page URL	URL
Referrer	Источник ссылки HTTP

Пользовательские переменные

**СОЗДАТЬ**

Этот контейнер не содержит пользовательских переменных. Чтобы создать п

Рис. 85. Создание пользовательской переменной

Выбираем тип переменной – **«Настройки Google Analytics»**. Эта переменная позволяет задавать настройки Google Analytics для использования с разными тегами.

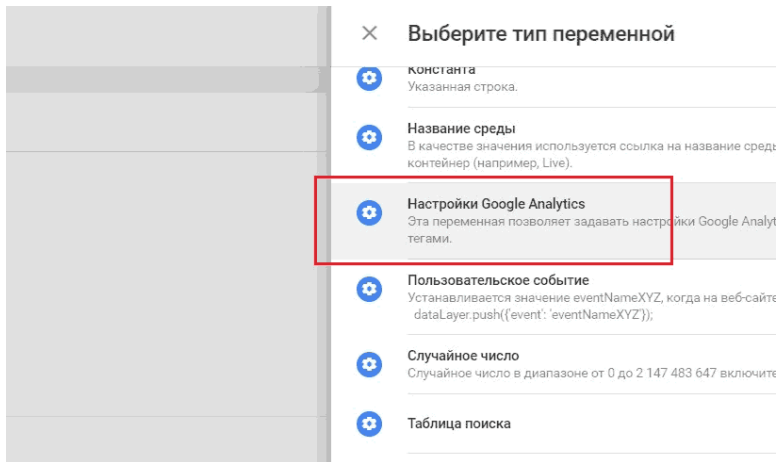


Рис. 86. Тип переменной – Настройки Google Analytics

Вводим название переменной, идентификатор отслеживания (код Google Analytics) и нажимаем «Сохранить».

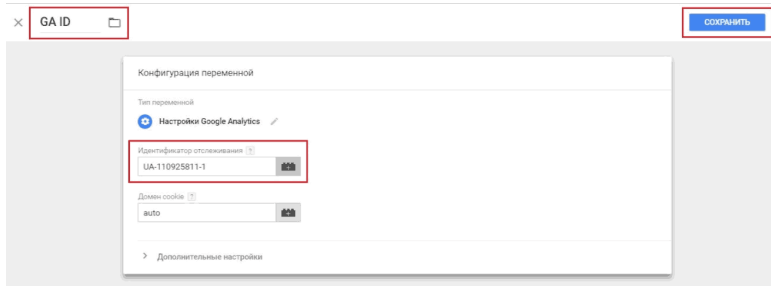
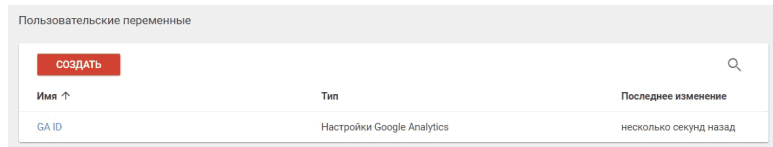


Рис. 87. Идентификатор отслеживания и Название пере-

менной

Теперь нет необходимости постоянно копировать полный код Analytics. С этого момента у нас есть пользовательская переменная, которую можно использовать при внедрении тега Google Analytics.



Пользовательские переменные

СОЗДАТЬ 🔍

Имя ↑	Тип	Последнее изменение
GA ID	Настройки Google Analytics	несколько секунд назад

Рис. 88. Пользовательская переменная GA ID

Переходим в меню «**Теги**», создаем новый тег.

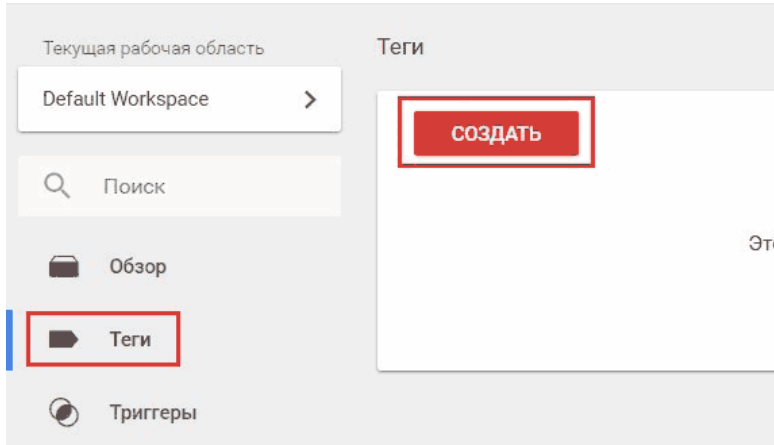


Рис. 89. Создание нового тега Google Analytics

Конфигурация тега – Universal Analytics, поскольку библиотека классического Google Analytics (*ga.js*) сильно устарела и больше не развивается. Однако есть достаточное количество сайтов, на которых по-прежнему установлен старый код. На смену ему уже пришли *analytics.js* и *gtag.js*.

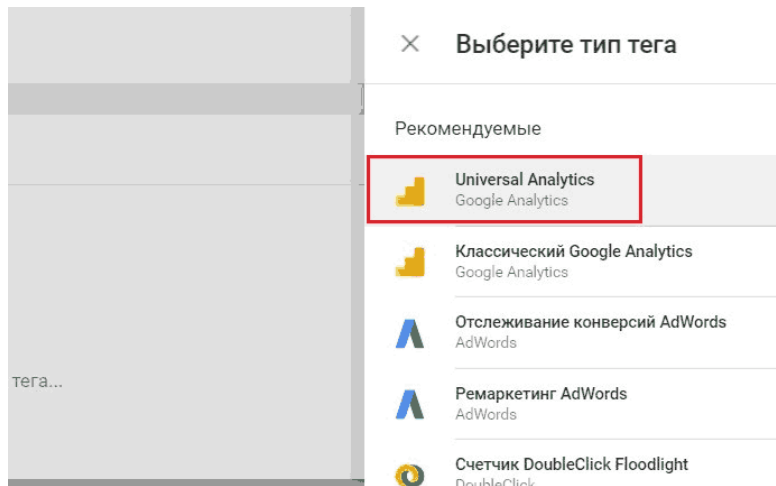


Рис. 90. Рекомендуемый тип тега – Universal Analytics

Настройки:

- **Тип отслеживания** – Просмотр страницы;
- **Настройки Google Analytics** – Выбираем нашу переменную GA ID (автоматически заключается в двойные фигурные скобки, как шаблон);

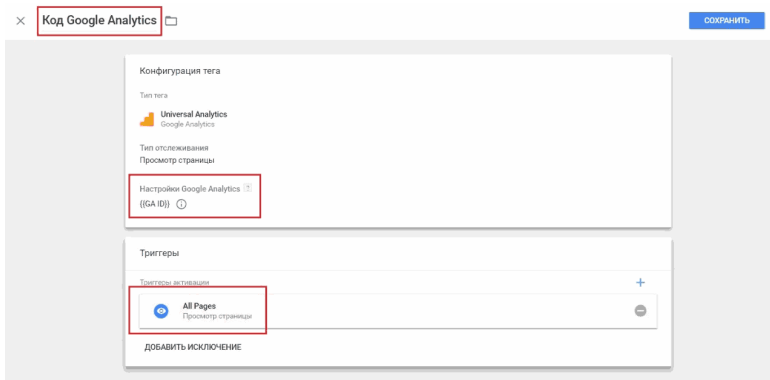


Рис. 91. Конфигурация тега

● **Триггер / Условие активации тега – All Pages** (на всех страницах).

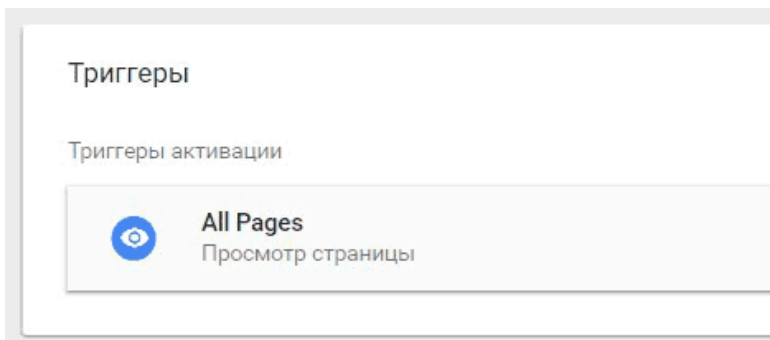


Рис. 92. Триггер активации – All Pages (на всех страницах сайта)

Вводим название тега и сохраняем его. На этом настройка кода отслеживания Google Analytics через Google Tag Manager на сайт завершена.

Проверим корректность настройки кода отслеживания GA традиционным способом через сам Google Analytics без использования предварительного просмотра в отладчике. Для этого в правом верхнем углу нажимаем «**Отправить**» – «**Опубликовать**» и «**Далее**». Новая версия контейнера опубликована!

Теперь просто перейдем в интерфейс Google Analytics и в отчетах «**В режиме реального времени**» увидим наш заход. Для наглядности пометим свой переход с помощью utm-меток. Можно перейти на различные разделы и посмотреть, фиксируются ли переходы по всем страницам сайта или нет.



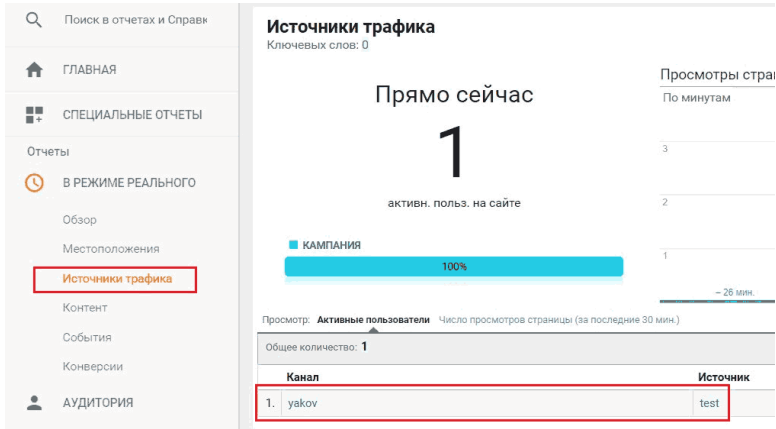


Рис. 93. Проверка установки кода Google Analytics в режиме реального времени

Данные передаются, тег настроен корректно. Проверить фиксацию данных также можно с помощью встроенного в GTM инструмента, который называется **«Предварительный просмотр»** (или режим отладки). Ему будет посвящена отдельная статья в этом руководстве, а пока установим код счетчика Яндекс.Метрика.

## Установка кода Яндекс.Метрика через GTM

Перед установкой кода ЯМ необходимо создать счетчик в аккаунте Яндекса и получить код отслеживания. Аналогично Google Analytics создаем новый тег. Только вместо реко-

мендуемого типа тега выбираем специальный – «**Пользовательский HTML**».

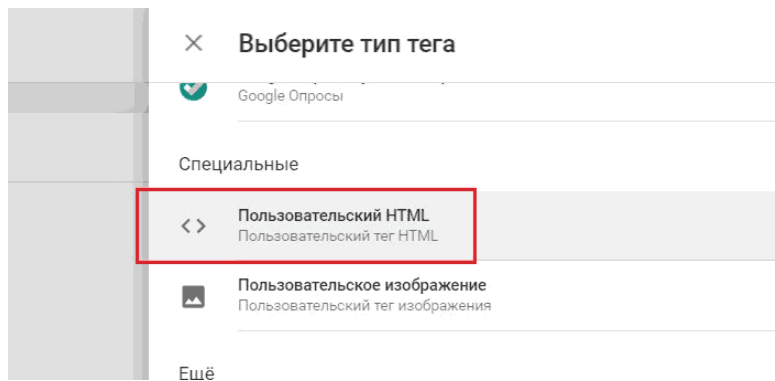


Рис. 94. Тип тега – Пользовательский HTML

Копируем код отслеживания из интерфейса Яндекс.Метрика и вставляем его в поле HTML. Не забываем про триггер – условие активации тега – All Pages (на всех страницах). Нажимаем «Сохранить».

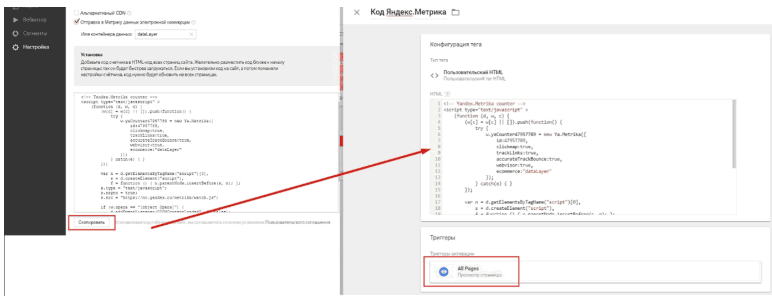


Рис. 95. Копирование кода из Я.Метрика в Пользовательский HTML GTM

В результате мы с вами получили два тега и два кода счетчика веб-аналитики.

Теги

[СОЗДАТЬ](#) 🔍

Имя ↑	Тип	Триггеры активации	Последнее изм...
Код Google Analytics	Universal Analytics	All Pages	10 часов назад
Код Яндекс.Метрика	Пользовательский HTML	All Pages	несколько секунд назад

Рис. 96. Теги двух кодов счетчиков аналитики

Чтобы данные передавались в Яндекс.Метрика, осталось только опубликовать новую версию контейнера на сайт. Действия прежние: в правом верхнем углу нажимаем **«Отправить»** – **«Опубликовать»** и **«Далее»**. Новая версия контейнера опубликована!

Самый простой способ проверить корректность настройки счетчика ЯМ – это обновить статус самого счетчика в списке всех счетчиков учетной записи.

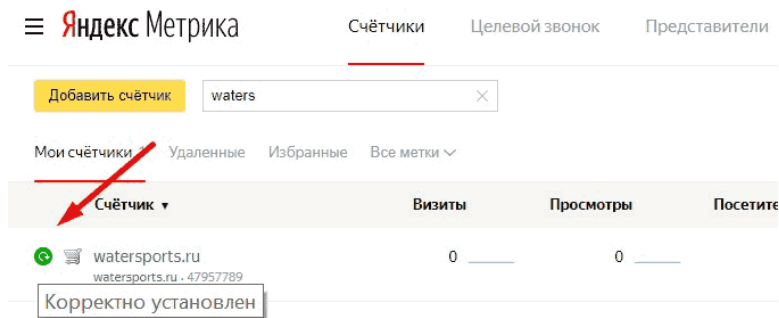


Рис. 97. Счетчик Яндекс.Метрика корректно установлен

Если вместо желтого или красного значка появился зеленый, значит он корректно установлен.

Есть еще один способ проверки установки кода счетчика – это добавление к адресу страницы параметра `_ym_debug` со значением `1`. В моем случае это `https://watersports.ru/?_ym_debug=1`

Далее необходимо вызвать консоль разработчика (в браузере Google Chrome кнопка «F12») и перейти на вкладку **Console**. Если код установлен правильно, в консоли вы увидите номер вашего счетчика и данные, которые отправ-

ЛЯЕТ КОД.

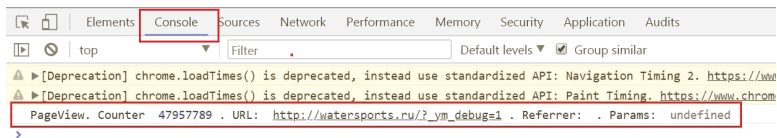


Рис. 98. Данные о просмотре страниц передаются

# Плагины и расширения для браузеров

Для упрощения работы с Google Tag Manager, Google Analytics и улучшения производительности интернет-маркетолог/веб-аналитик в своей практике может использовать дополнительные инструменты – плагины и различные расширения для браузеров. Давайте разберем несколько самых популярных помощников.

## Google Analytics Debugger

Расширение для браузера. Является надстройкой для консоли разработчика, в которую Debugger отправляет без изменения кода отслеживаемую информацию о том, что передается в Google Analytics, а также данные обо всех ошибках и сбоях. С помощью расширения можно не только проверить собственные страницы, но и узнать, как Analytics работает на других сайтах.

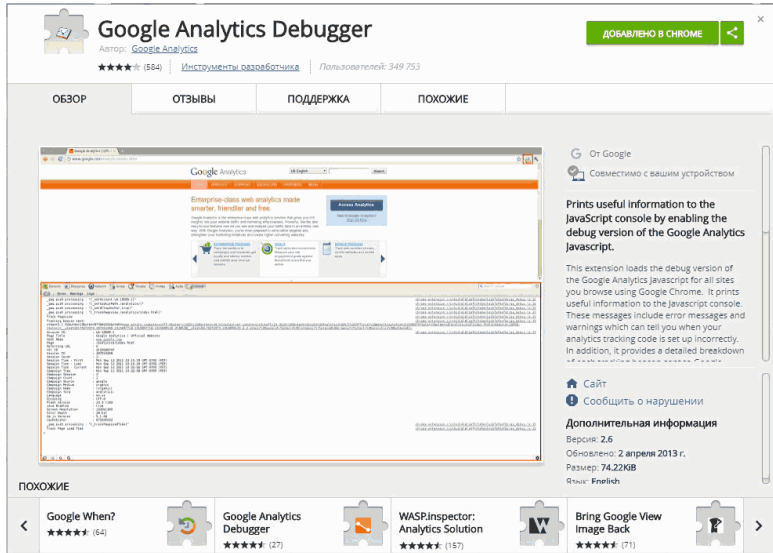


Рис. 99. Google Analytics Debugger

GA Debugger часто используют при решении различного рода задач. Например, при настройке целей и электронной торговли, когда необходимо фиксировать передаваемые данные.

Скачать расширение для браузера Google Chrome можно по ссылке.

## Google Tag Assistant

Универсальный помощник. Расширение для браузера

Google Chrome, которое позволяет записывать последовательность действий, выполняемых пользователями, поддерживает анализ тегов во многих продуктах Google – AdWords, DoubleClick, Tag Manager. Он также помогает отслеживать правильность установки кода, находить и исправлять проблемы, которые могут привести к искажению данных в ресурсах и представлениях Google Analytics.

Обладает функцией записи (**Google Tag Assistant Recordings, GTAR**), которая может регистрировать теги, события и взаимодействия для любой последовательности посещенных страниц или сайтов, и тех страниц, которые открывал пользователь, даже если часть из них находится за пределами основного домена.

The screenshot displays the Google Tag Assistant interface. At the top, it says "Tag Assistant (by Google)" with a star rating of 4.5/5 (913) and a link to "Инструменты разработчика". Below this is a navigation bar with tabs: "ОБЗОР", "ОТЗЫВЫ", "ПОДДЕРЖКА", and "ПОХОЖИЕ". The main content area is titled "Recording summary" and shows details for a recording on July 17, 2015, at 12:27:15 PM. It lists the recording origin as "none specified", total page loads as 8, and tracking codes as 2 (1 authorized for [redacted]). Below this is an "Alerts" section with three items: "Page had a GTID parameter, but it was dropped or masked by the Javascript on the page", "The site starts an additional session", and "Page in most likely missing a tag". A "Views summary" section follows, indicating that the recorded hits would appear in the selected Google Analytics views. On the right side, there is a sidebar with a "УСТАНОВИТЬ" button and a "G+" icon. Below the main content, there are links for "Сайт" and "Сообщить о нарушении", and a section for "Дополнительная информация" with details: "Версия: 17.285.0", "Обновлено: 12 октября 2017 г.", "Размер: 808KiB", and "Язык: English".



В моей книге «**Google Analytics для googлят**» и блоге **osipenkov.ru** есть целая статья на эту тему. Скачать расширение для браузера Google Chrome можно по ссылке.

## **Web Analytics Solution Profiler (WASP)**

Еще одно расширение для браузера, которое позволяет обнаружить теги любого типа, их взаимосвязь между собой и визуализировать полученные данные.

WASP запоминает состояние `dataLayer` после того, как вы покинули страницу и перешли на следующую. Этот функционал является большим плюсом, поскольку в режиме отладки GTM не позволяет видеть, что произошло на предыдущей странице.

Другие особенности:

- обнаружение всех типов тегов, включая те, которые отправляются через POST;
- SEO информация о странице (URL, title, description, keywords, cookie и др.);
- возможность скрытия и блокировки тега;
- декодирование информации для Google Analytics и других сервисов;
- удобная визуализация, которая показывает отношения между всеми тегами с помощью связей в виде «стрелочек».

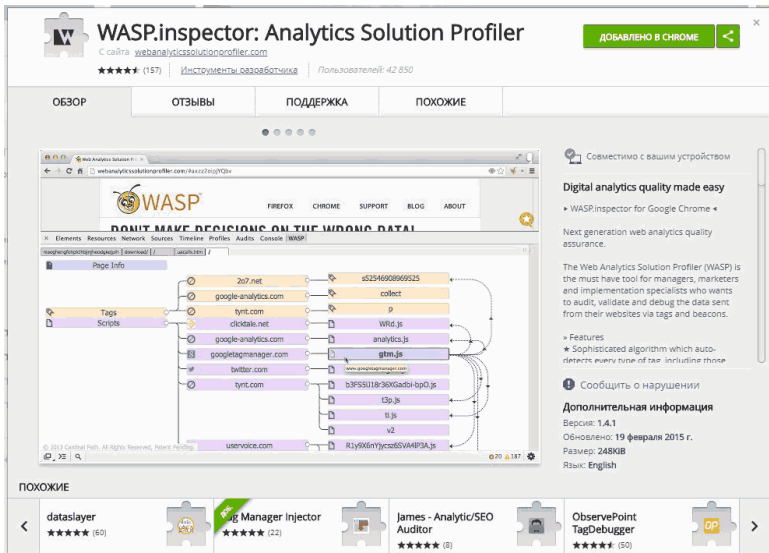


Рис. 101. WASP.inspector

Расширение предназначено для «продвинутых» пользователей. Скачать расширение для браузера Google Chrome можно по ссылке.

## Code Editor for GTM

Расширение подсвечивает синтаксис JavaScript кода в интерфейсе Google Tag Manager, а также позволяет работать в полноэкранном режиме. Настройка очень простая – скачи-

вание, установка, и активация на странице GTM.

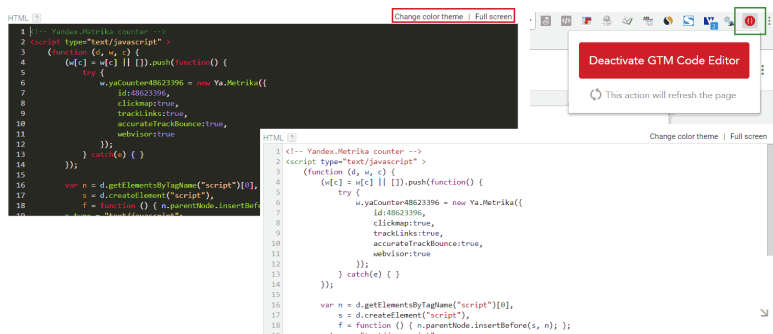


Рис. 102. Смена цветовой темы GTM

Будет особенно актуально для веб-разработчиков, которые привыкли использовать различные темы с подсветкой кода в других редакторах, например, в Notepad++, Sublime Text, Atom, Visual Studio Code и т.д.

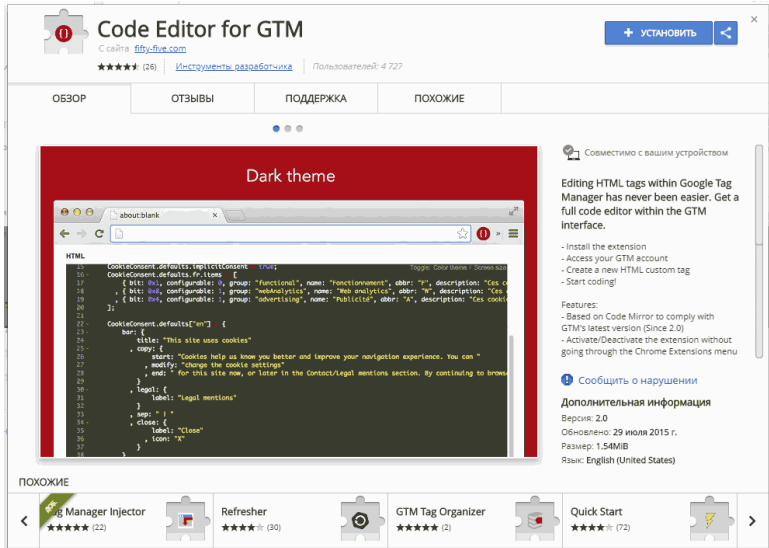


Рис. 103. Code Editor for GTM

Скачать расширение для браузера Google Chrome можно по ссылке.

## Tag Manager Injector

Расширение позволяет начать работу с Google Tag Manager до того, как код контейнера будет размещен на страницах сайта. Это бывает особенно полезным, когда требуется начать работу над установкой GTM, но доступы по какой-либо причине заказчик или его команда оперативно

предоставить не могут.

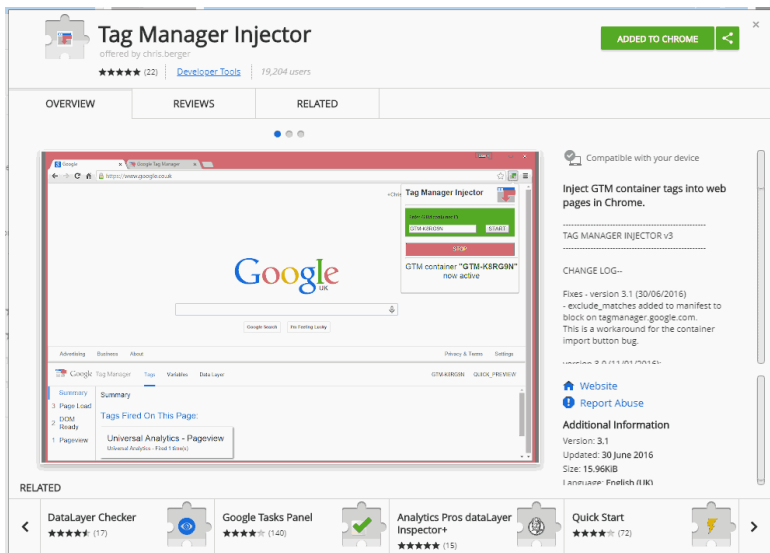


Рис. 104. Tag Manager Injector

Скачать расширение для браузера Google Chrome можно по ссылке. После этого в правой верхней части появится новая иконка, при клике на которую открывается окно с настройками Tag Manager Injector:

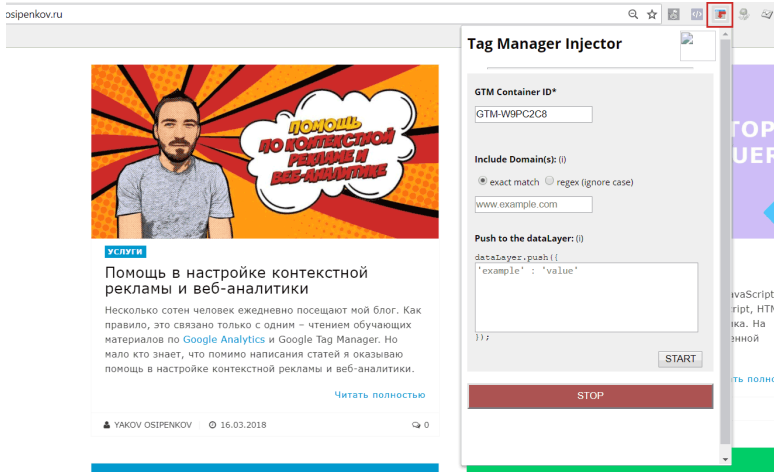


Рис. 105. Настройки Tag Manager Injector

Затем необходимо прописать идентификатор контейнера GTM без каких-либо пробелов (поле «**GTM Container ID**»), а также домен (поле «**Include Domain(s)**»), на котором данный контейнер будет использоваться и нажать на кнопку «**START**»:

# Tag Manager Injector



## GTM Container ID\*

## Include Domain(s): (i)

exact match  regex (ignore case)

## Push to the dataLayer: (i)

```
dataLayer.push({  
  'example' : 'value'  
});
```

## Рис. 106. Активация Tag Manager Injector

Настройки ТМІ позволяют добавлять фрагмент GTM только на определенные страницы сайта. Для этого нужно переключить поле «**exact match**» на «**regex (ignore case)**» и указать те страницы, к которым вы хотели бы применить изменения. Без помощи регулярных выражений здесь не обойтись.

В результате произойдет перезагрузка текущей страницы, Tag Manager Injector напишет, что GTM контейнер активирован, а внизу страницы появится панель отладки:



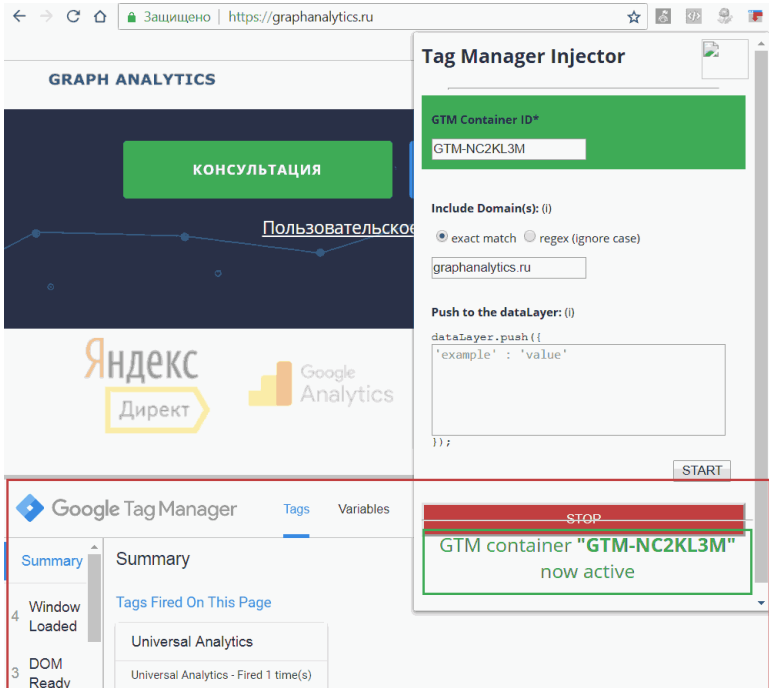


Рис. 107. Tag Manager Injector активирован

Теперь можете выполнять настройки в контейнере, запускать предварительный просмотр и все ваши изменения будут доступны на сайте, доменное имя которого указано в настройках плагина.

Tag Manager Injector позволяет имитировать выполнение кода, передающего информацию в уровень данных. Для этого в соответствующее поле введите информацию в виде пары

“ключ”:”значение”.

```
Push to the dataLayer: (i)  
dataLayer.push(  
  'example' : 'value'  
  );
```

Рис. 108. dataLayer.push ({});

После завершения всех настроек отключите расширение с помощью кнопки «**STOP**».



Рис. 109. Остановка работы Tag Manager Injector

**Важно:** плагин искусственно устанавливает GTM – но только для вас, в вашем браузере. Поделиться информацией с другим человеком вы не сможете. Также описанные выше действия не заменяют собой процесс внедрения Google Tag

Manager путем размещения его кода на всех страницах сайта. Если вы хотите, чтобы GTM корректно работал, необходимо добавить два фрагмента кода на отслеживаемые страницы сайта согласно руководству Google.

Tag Manager Injector может быть использовать в режиме «ИНКОГНИТО».

## Google Tag Manager Blocker

Очень простое расширение для браузера Mozilla Firefox, которое блокирует передачу данных посетителя сайта в GTM, следовательно, и в Google Analytics и Яндекс.Метрика. Включается/выключается всего одним кликом.



### Google Tag Manager Blocker

от [heubergen1](#)

Google Tag Manager Blocker allows you to block the GTag Manager

+ Добавить в Firefox

### Рис. 110. Google Tag Manager Blocker

Рекомендуется использовать при тестировании, каких-либо изменениях/правках на сайте и тогда, когда вы не хотите, чтобы ваш заход был зафиксирован системами веб-аналитики. Скачать расширение можно по ссылке.

# GTM Sonar

Расширение для браузера разработал эксперт по Google Tag Manager и Google Analytics, автор самого популярного блога в мире на эти темы, **Симо Ахава (Simo Ahava)**.

Его творение, GTM Sonar (прошрое название **GTM Auto-Event Listener Debugger**) позволяет блокировать переход на другую страницу, но при этом формирует работу тегов таким образом, как будто вы совершили переход. Данные по действиям (кликам по элементам, по ссылкам или по формам) сохраняются в массив с именем **debugDL**, содержимое которого можно в консоли разработчика.

Скачать расширение для браузера Google Chrome можно по ссылке.

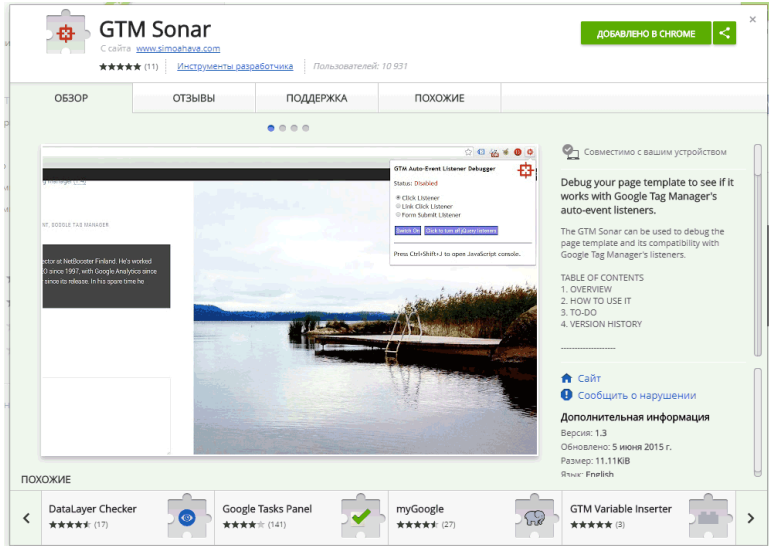


Рис. 111. GTM Sonar

В правом верхнем углу на панели вы увидите иконку в виде красного прицела:



Рис. 112. Иконка и настройки GTM Sonar

По умолчанию иконка красная и статус «**Выключен (Disabled)**». Чтобы включить GTM Sonar, необходимо выбрать тип отслеживания: **Click Listener**, **Link Click Listener** или **Form Submit Listener** и нажать «**Switch On**».

Если сопоставлять настройки с триггерами GTM, то это будет выглядеть так:

## Клик



Все элементы

Click Listener

---



Только ссылки

Link Click Listener

---

## Взаимодействия пользователей



Отправка формы

Form Submit Listener

---

Рис. 113. Сопоставление настроек GTM и GTM Sonar

Разберем пример взаимодействия пользователей с отправкой формы. Поскольку мы будем отслеживать форму, то в GTM Sonar выбираем **Form Submit Listener** и нажимаем **Switch On**. Чтобы исключить отслеживание кликов с помощью jQuery, нажмите **«Click to turn off jQuery listeners»**.

В результате мы получим зеленый прицел и статус **«Включен (Enabled)»**:

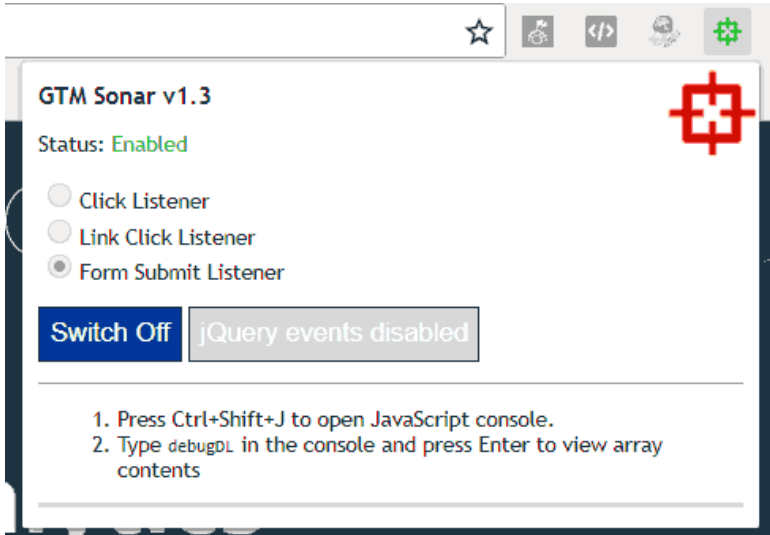


Рис. 114. GTM Sonar активирован

Когда мы включаем GTM Sonar, происходит следующее:

1. все действия по умолчанию, которые были осуществлены кликами мыши по странице, приостанавливаются;
2. когда вы кликаете на элемент, информация о данном действии сохраняется в массиве **debugDL** в том же формате, что и в GTM.

Чтобы продемонстрировать это на конкретном примере, я сделаю тестовую заявку:



# Бесплатный аудит рекламных кампаний

Ответим в течение 24 часов. Если Вам интересно,  
заполните, пожалуйста, форму ниже. Пример аудита  
доступен по ссылке

test

test.ru

test@yandex.ru

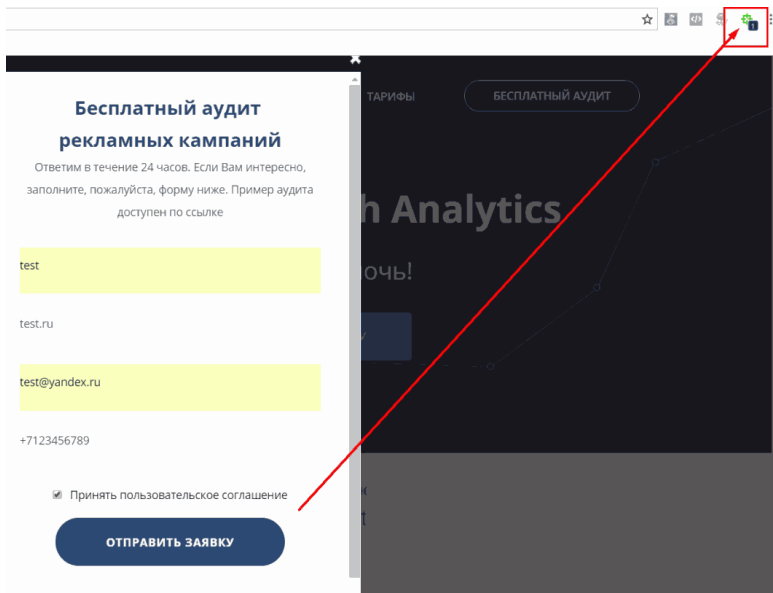
+7123456789

Принять пользовательское соглашение

**ОТПРАВИТЬ ЗАЯВКУ**

## Рис. 115. Пример тестовой заявки на сайте

Нажав на кнопку отправить, я получу уведомление о совершенном событии. Когда элемент добавлен в debugDL, счетчик в иконке браузера начинает свою работу, и данный счетчик показывает количество объектов в массиве **debugDL**. Появилась циферка «1»:



## Рис. 116. Количество объектов в массиве

Хотя в действительности я должен был увидеть вот это

## Спасибо за заявку!

Мы свяжемся с Вами в ближайшее время

Рис. 117. Это страница «Спасибо» должна была быть после отправки заявки

Для чего мы избегаем клики на странице? Дело в том, что при отладке было бы лучше, если бы вы находились на той же самой странице, не переключаясь на другую. Остановка действий по умолчанию предотвращает работу ссылок. По причине того, что `debugDL` – это объект в текущем документе, он будет работать только на той страничке, на которой вы находитесь.

Когда вы выключаете `debugger`, действия по умолчанию становятся активными. `DebugDL` массив не будет очищен, и вы можете наблюдать за содержимым массива в течение всего времени, пока будете находиться на данной странице.

Теперь чтобы перейти к событию, откроем консоль разработчика и перейдем на вкладку **Console**. Введя команду **debugDL** и нажав **Enter**, мы увидим наш массив с объектом:

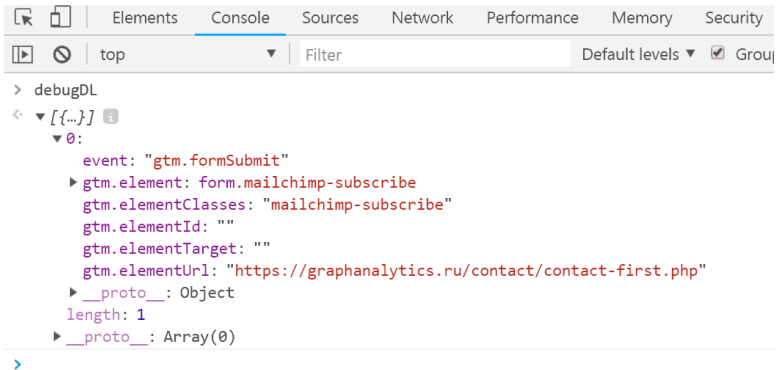


Рис. 118. Содержимое debugDL после отправки формы

Теперь эту информацию можно использовать для отслеживания формы и передачи данных в GTM и Google Analytics.

Подробнее об инструменте Симо Ахавы можете ознакомиться в его блоге по ссылке.

## Другие инструменты

Безусловно, есть еще целый ряд других полезных плагинов, дополнений и расширений для браузеров, например,

dataslayer, Analytics Pros dataLayer Inspector+, Da Vinci Tools, GTM Variable Builder и т.д. Мы рассмотрели самые популярные расширения для браузеров, которые используют в своей повседневной работе веб-аналитиков со всего мира.

# Глава 2. Разбор Google Tag Manager

## Интерфейс

В процессе знакомства с любым новым для себя продуктом у пользователей часто возникает проблема привыкания к самому сервису, программе или ее отдельным функциям. Будь-то это совершенно новый интерфейс Google AdWords, обновленный Google Analytics , или Google Tag Manager версии 2.

Вот так выглядит страница контейнера GTM одного из аккаунтов:

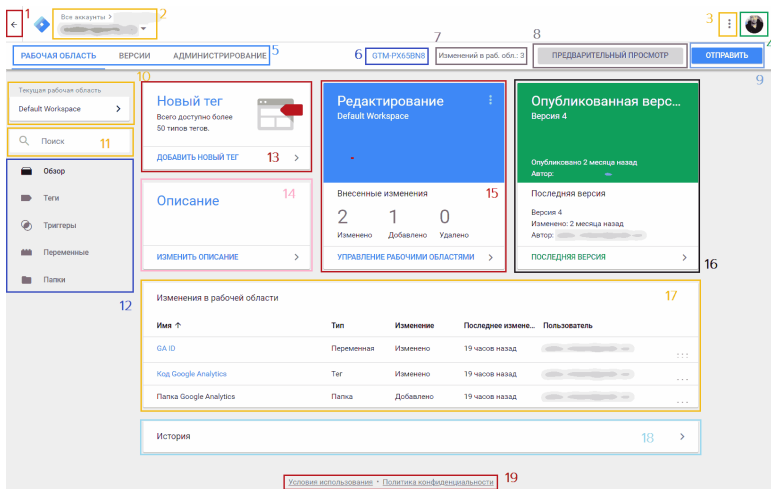
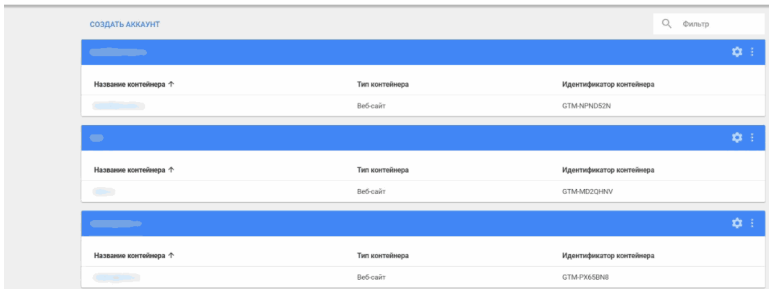


Рис. 119. Интерфейс Google Tag Manager

## 1. На главную страницу

При клике на данную стрелочку вы попадете на страницу [tagmanager.google.com/#/home](https://tagmanager.google.com/#/home), где будут отображены все доступные вам контейнеры GTM.



The screenshot shows the 'СОЗДАТЬ АККАУНТ' (Create Account) page in Google Tag Manager. At the top right, there is a search icon and a 'Фильтр' (Filter) button. Below this, there are three rows, each representing an account. Each row has a blue header bar with a gear icon and a vertical ellipsis. The first row is partially obscured by a blue bar. The second and third rows are fully visible and contain the following data:

Название контейнера ↑	Тип контейнера	Идентификатор контейнера
	Веб-сайт	GTM-NPMD52N
	Веб-сайт	GTM-MDQ2HNV
	Веб-сайт	GTM-PX65BN8

Рис. 120. Список аккаунтов и контейнеров GTM

На главной странице возможно создание нового аккаунта, фильтра по текущим, а также доступны общие настройки по каждому конкретному контейнеру. Фильтр доступен как по названию контейнера, так и по его идентификатору. То есть когда вы начнете вводить последовательность букв/цифр, Google Tag Manager сразу же выдаст соответствующий результат.



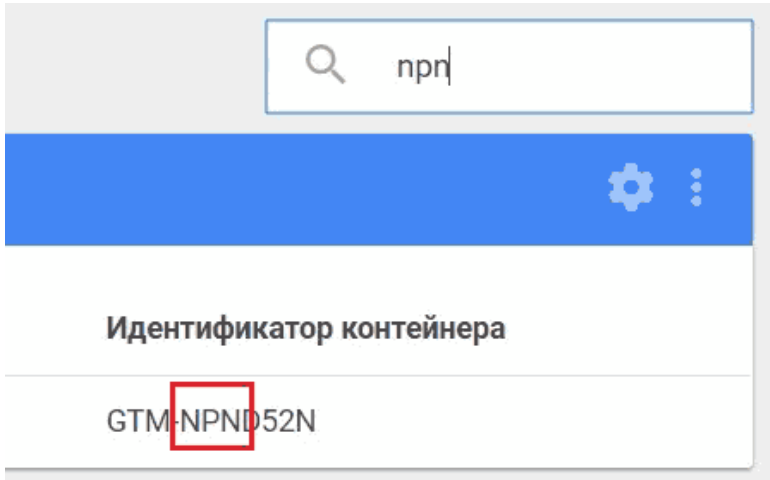


Рис. 121. Поиск по идентификатору контейнера

На главной странице в самом низу располагаются удаленные аккаунты и контейнеры. Чтобы удалить аккаунт или контейнер, необходимо обладать соответствующими правами.

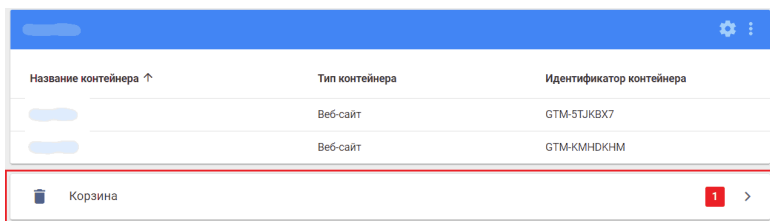
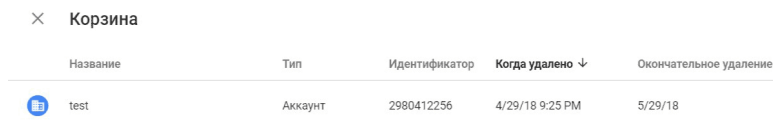


Рис. 122. Удаленные аккаунты и контейнеры в Google Tag Manager

При удалении аккаунта через «Администрирование» в GTM он будет перемещен в корзину и окончательно удален через 30 дней. Вместе ним будут удалены все контейнеры, включая все теги, триггеры и переменные. Теги из этого контейнера больше не будут активироваться при посещении вашего сайта пользователями. При удалении контейнера он также будет перемещен в корзину и окончательно удален через 30 дней.



Название	Тип	Идентификатор	Когда удалено ↓	Окончательное удаление
test	Аккаунт	2980412256	4/29/18 9:25 PM	5/29/18

Рис. 123. Корзина

По клику на контейнер мы можем узнать точную дату удаления. По прошествии 30 дней данного контейнера в корзине уже не будет, также, как и возможности его восстановления. Если же нам понадобится восстановить элемент из корзины, мы должны:

- выбрать его из списка;
- в открывшемся окне нажать кнопку «Восстановить».

Сведения об удаленном аккаунте

Идентификатор аккаунта  
2980412256

Кем удалено  
[REDACTED]

Когда удалено  
4/29/18 9:25 PM

Восстановление возможно до  
5/29/18

Контейнеры

Название	Тип контейнера	Идентификатор контейнера	
test.ru	Веб-сайт	GTM-MD2QNNV	ЭКСПОРТИРОВАТЬ

Пользователи

Адрес электронной почты	Разрешения ↓
[REDACTED]	Доступно для восстановления

Рис. 124. Сведения об удаленном аккаунте GTM

Еще нам будет доступна возможность экспорта контейнера и информация о пользователях, у которых есть разрешение на восстановление, а также электронная почта того человека, кто совершил операцию удаления (поле «Кем удалено»).

## 2. Портал Google Analytics Suite

В одном меню Google объединил все свои продукты, и те-

перь доступ к ним стал возможен через одну панель. Если вы захотите перейти в Google Analytics, Google Data Studio (Центр данных) или любой другой продукт, который привязан к данному аккаунту Google, просто нажмите на соответствующий значок на панели «**Все аккаунты**».

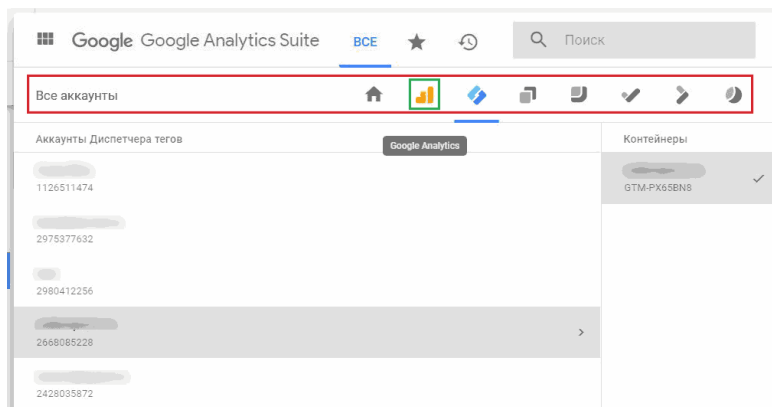


Рис. 125. Google Analytics Suite

### 3. Дополнительные настройки

Меню состоит из пользовательских настроек, отзывов и справки Google.

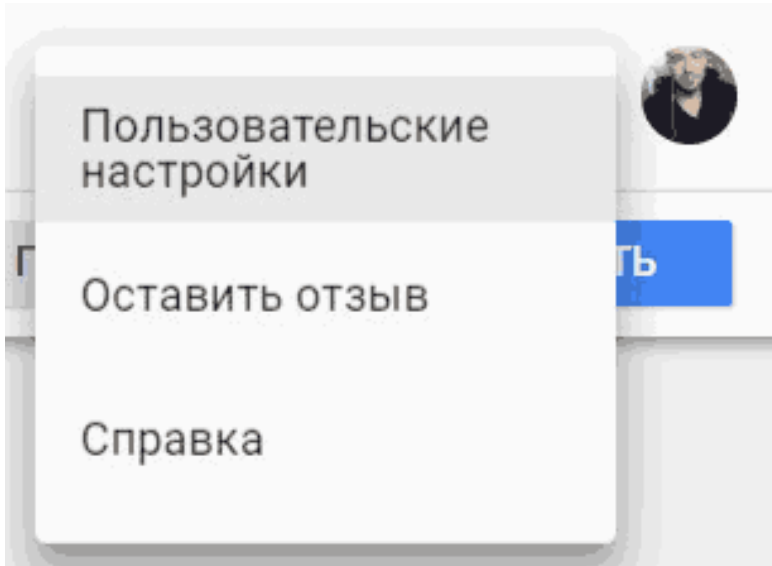


Рис. 126. Дополнительные настройки

В пользовательских настройках содержится информация по вашему роду деятельности (агентство, маркетолог, веб-разработчик), языку интерфейса Google Tag Manager и уведомлениях, связанных с новостями, анонсами новых функций, тестированию и различными предложениями от Google.

## Мои данные

Я обычно использую Диспетчер тегов Google как:

Маркетолог

На странице [настроек аккаунта Google](#) можно изменить адрес электронной почты или пароль, а также подробнее узнать о доступе к другим сервисам Google.

## Языковые настройки

Я буду использовать Диспетчер тегов Google на этом языке:

Русский

## Настройки сообщений электронной почты

Мы периодически отправляем нашим пользователям сообщения с последними новостями Диспетчера тегов Google. Вы всегда можете выбрать письма, которые хотите получать, указав их ниже.

Вне зависимости от выбранных вариантов вы иногда можете получать сообщения о важных обновлениях продукта, которые влияют на ваш аккаунт. Поскольку конфиденциальность пользовательских данных является для нас приоритетом, мы гарантируем, что ваша личная информация не будет предоставлена третьим лицам и нашим партнерам.

Новости и советы по повышению эффективности ?

Анонсы новых функций ?

Обратная связь и тестирование ?

Предложения от Google ?

СОХРАНИТЬ

ОТМЕНА

Рис. 127. Данные аккаунта

В разделе **«Оставить отзывы»** вы можете описать проблему работы с GTM или поделиться мнением с командой Google об этом инструменте. Справка позволяет получить ответы на интересующие вас вопросы. Просто введите необходимые ключевые слова и ознакомьтесь с результатами поиска.



## Справка



Поиск в Справке

### Популярные



Обзор Диспетчера тегов Google



Перед началом работы



Настройка и установка



Что лучше использовать: Диспетчер тегов Google или gtag.js?



Пользователи и разрешения

## **4. Аккаунты**

Можно быстро переключаться между аккаунтами при условии, что вы залогинены в несколько из них.





Yakov Osipenkov

[Политика конфиденциальности](#)



Blurred text



Blurred text



Blurred text



Blurred text



Blurred text

Рис. 129. Несколько аккаунтов Google

## 5. Горизонтальное навигационное меню



Рис. 130. Рабочая область – Версии – Администрирование

Позволяет быстро переключаться между рабочими областями, версиями и панелью администрирования аккаунта и контейнера.

По умолчанию активна первая вкладка **«Рабочая область»** (пространство для работы с изменениями в диспетчере тегов), но в любой момент можно перейти на **«Версии»** (сохраненная копия контейнера в данный момент времени) или **«Администрирование»**.

GTM-NPND52N

## Версия 5 опубликована

Опубликовано Apr 23, 2018, автор: [REDACTED]

3 Теги    1 Триггер    1 Переменная

Версии						
Идентификатор вер.	Статус	Название	Дата создания	Опубликовано	Автор	
5	Опубликовано, Последняя		23.04.2018	23.04.2018	[REDACTED]	Действия ▾
4			19.04.2018	19.04.2018	[REDACTED]	Действия ▾
3			19.04.2018	19.04.2018	[REDACTED]	Действия ▾
2			19.04.2018	19.04.2018	[REDACTED]	Действия ▾
1			19.04.2018	19.04.2018	[REDACTED]	Действия ▾

Рис. 131. Версии

На вкладке «**Администрирование**» можно управлять доступами, изменять настройки, на уровне аккаунта и контейнера, импортировать-экспортировать контейнеры, работать со средами и др.

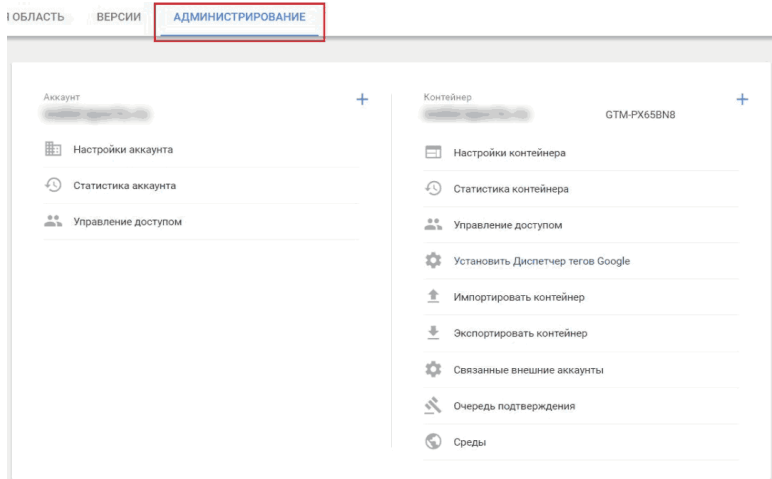


Рис. 132. Администрирование

## 6. Идентификатор контейнера GTM

При клике на ID контейнера Google Tag Manager откроется окно с кодом, часть которого необходимо вставить в раздел **<head>** кода страницы как можно ближе к началу, а часть после открывающего тега **<body>** при условии, что вы этого не сделали ранее. Тот же самый код вы видели на этапе регистрации в GTM и установки контейнера.



## Установите Диспетчер тегов Google

Скопируйте приведенный ниже код и вставьте его на каждую страницу сайта.

Вставьте этот фрагмент в раздел **<head>** кода страницы как можно ближе к началу:

```
<!-- Google Tag Manager -->  
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':  
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],  
j=d.createElement(s),dl=l||'dataLayer'?l:'';j.async=true;j.src=  
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);  
})(window,document,'script','dataLayer','GTM-PX65BN8');</script>  
<!-- End Google Tag Manager -->
```

Также вставьте непосредственно после открывающего тега **<body>** следующий фрагмент:

```
<!-- Google Tag Manager (noscript) -->  
<noscript><iframe src="https://www.googletagmanager.com/ns.html?id=GTM-PX65BN8"  
height="0" width="0" style="display:none;visibility:hidden"></iframe></noscript>  
<!-- End Google Tag Manager (noscript) -->
```

Подробнее о том, как добавить фрагмент кода Диспетчера тегов Google, читайте в [руководстве по началу работы](#).

OK

Рис. 133. Фрагменты кода Google Tag Manager

## 7. Изменения в рабочей области

Изменений в раб. обл.: 4

Рис. 134. Изменения в рабочей области

Счетчик отображает количество внесенных изменений в текущей рабочей области. Например, вы создали новую пе-

ременную. Изменений в рабочей области станет 1. Добавили новый триггер. Значение изменится на 2. Еще два триггера? Тогда изменений в рабочей области будет 4. Создали тег? Google Tag Manager зафиксирует еще одно изменение в рабочей области и т.д. Счетчик обнуляется после публикации версии.

## 8. Предварительный просмотр

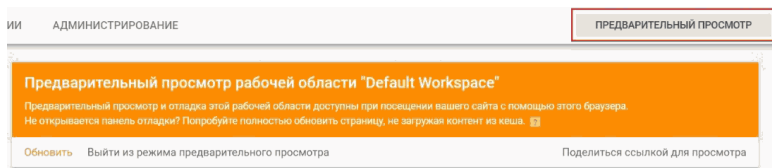


Рис. 135. Предварительный просмотр

Режим предварительного просмотра в GTM позволяет проверить работоспособность сайта с действующим контейнером. При каждом изменении конфигурации, корректировки настроек тегов, триггеров или переменных, рекомендуется проверять себя через отладку контейнера. Благодаря предварительному просмотру вы сможете избежать многих ошибок в процессе работы и узнать, какие теги активируются и в каком порядке.

Если включить режим предварительного просмотра, то на обзорной странице рабочей области вы увидите оранжевый

баннер с оповещением (на изображении выше), а на сайте станет доступна панель отладки. Она будет видна только в том браузере, в котором был включен этот режим, но вы можете предоставить доступ другим пользователям.

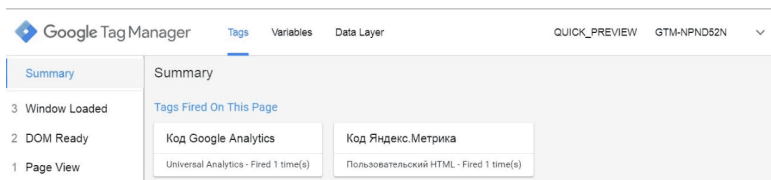


Рис. 136. Нижняя панель отладки

Сделать это можно с помощью опции **«Поделиться ссылкой для просмотра»**. Режим предварительного просмотра доступен пользователям с соответствующими правами.

## 9. Публикация контейнера

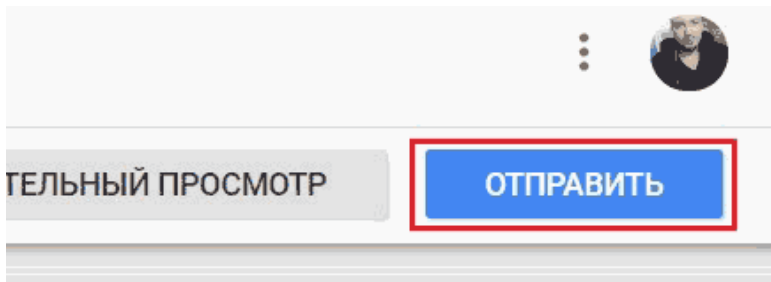


Рис. 137. Публикация контейнера

Чтобы добавить на сайт контейнер GTM впервые или отредактированную версию, его необходимо опубликовать. Сделать это можно через последовательность шагов по кнопке «Отправить» – «Публикация». Опубликовать контейнер могут пользователи с соответствующими правами доступа в одну из указанных сред.

## 10. Выбор рабочей области

Рабочая область в Google Tag Manager – это пространство для работы с изменениями в диспетчере тегов. Их можно сравнить с представлениями в Google Analytics.

По умолчанию, после создания контейнера GTM добав-



ляется рабочая область с названием «**Default Workspace**». В соответствующем меню можно выбрать одну из доступных рабочих областей, либо же создать новую через «+» и добавить (по желанию) описание.

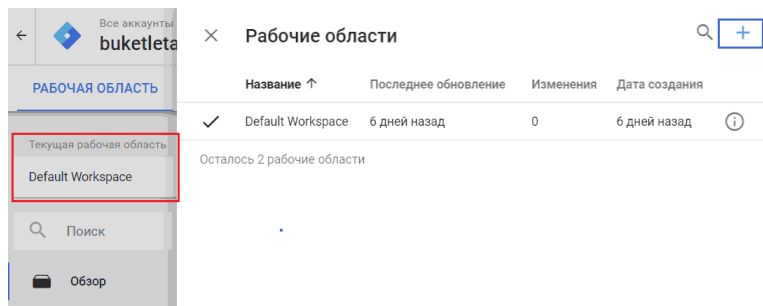


Рис. 138. Рабочие области

При выборе рабочей области доступен поиск по названию.

## 11. Поиск

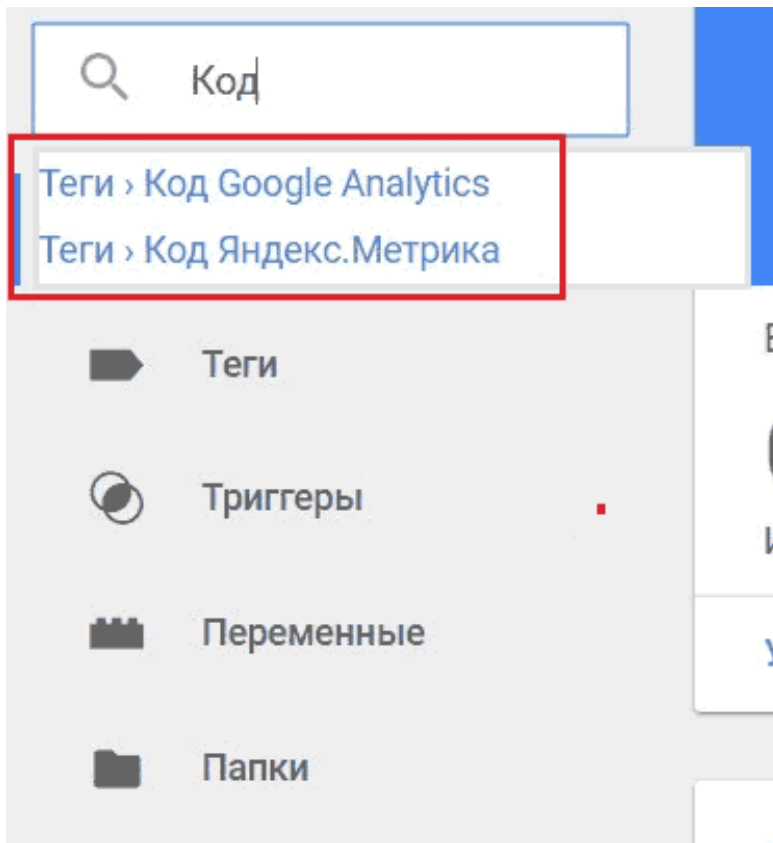


Рис. 139. Поиск в GTM

Функция поиска позволяет нам быстро искать информацию по тегам, триггерам и переменным в рамках выбранной рабочей области. Нажав на один из найденных элементов нас перенаправит на его настройки.

## **12. Вертикальное навигационное меню**

Сюда входят: обзор, теги, триггеры, переменные и папки.

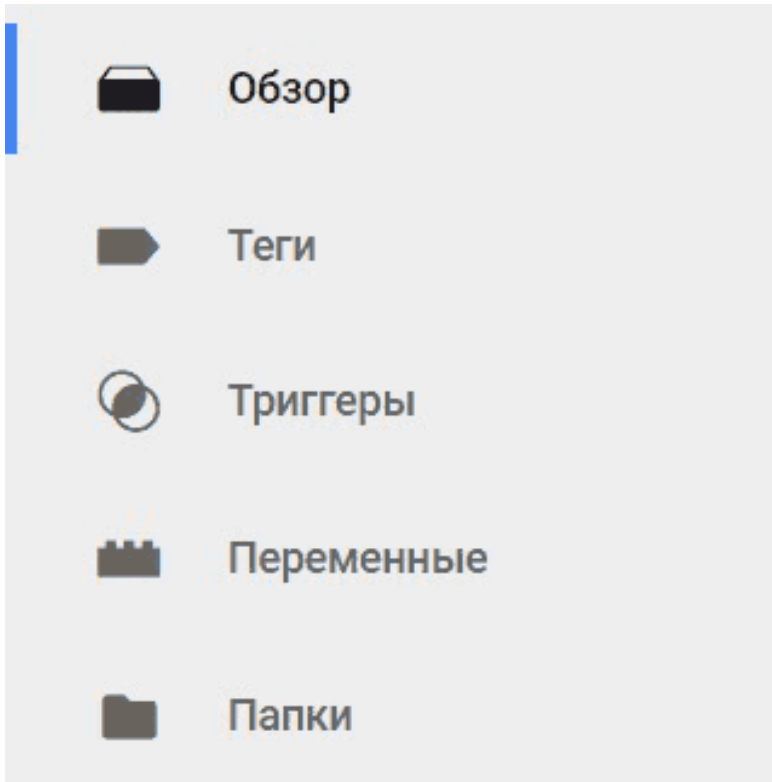


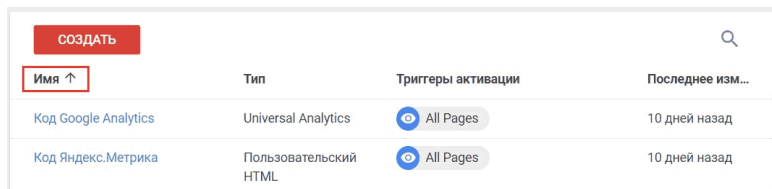
Рис. 140. Вертикальное навигационное меню

**Обзор.** В данный раздел входят пункты 13–18. Это общая страница, которая отображает основную информацию по текущему контейнеру GTM в виде 6 блоков (дашбордов).

**Теги, триггеры и переменные.** Читайте в статье

«Термины и определения, встречающиеся в работе с Tag Manager», а также ниже.

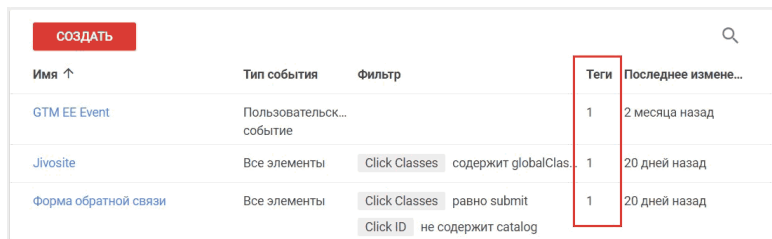
При переходе на каждый из этих разделов у вас есть возможность сортировки внутри таблиц по различным столбцам. Например, в тегах это можно делать по имени, типу объекта и последним изменениям.



Имя ↑	Тип	Триггеры активации	Последнее изм...
Код Google Analytics	Universal Analytics	All Pages	10 дней назад
Код Яндекс.Метрика	Пользовательский HTML	All Pages	10 дней назад

Рис. 141. Таблица тегов

В триггерах присутствует столбец, в котором выводится информация о количестве тегов, связанных с данным триггером.



Имя ↑	Тип события	Фильтр	Теги	Последнее изменение...
GTM EE Event	Пользовательское событие		1	2 месяца назад
Jivosite	Все элементы	Click Classes содержит globalClas...	1	20 дней назад
Форма обратной связи	Все элементы	Click Classes равно submit Click ID не содержит catalog	1	20 дней назад

Рис. 142. Столбец «Теги» в разделе «Триггеры»

**Папки.** Подробнее разберем чуть далее.

### 13. Блок «Новый тег»

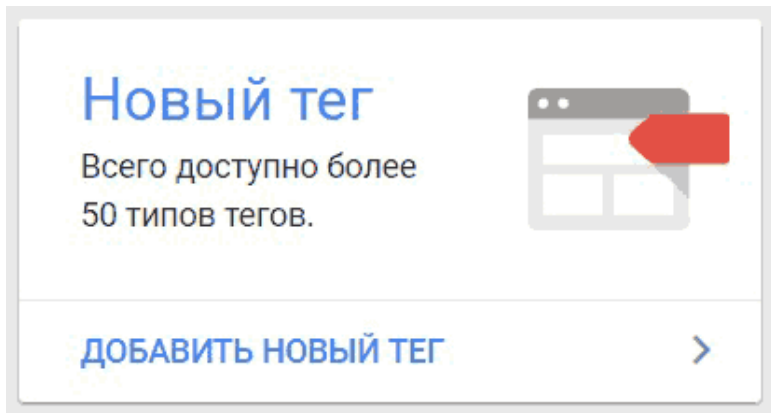


Рис. 143. Блок «Новый тег»

С помощью данного блока мы можем добавить новый тег в выбранную рабочую область.

## 14. Блок «Описание»

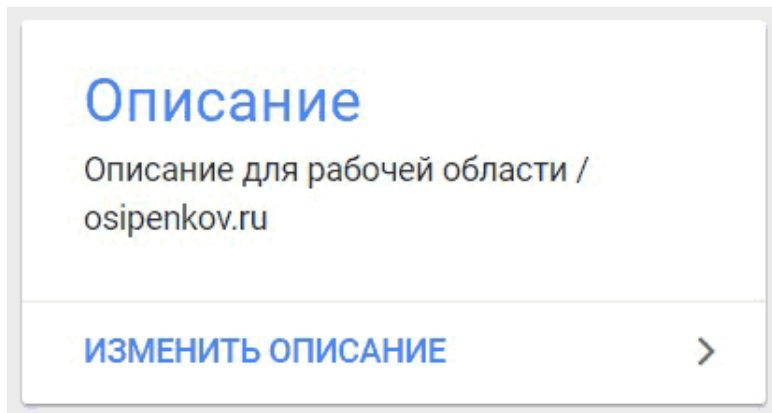


Рис. 144. Блок «Описание»

Предназначен для добавления описания к той области, в которой мы работаем. Чтобы вы, или человек, который будет работать с этой областью, понимали смысл всех внесенных изменений. Если вы используете различные рабочие области для сервисов Google и Яндекс, в описании к workspace можно написать: *«Используется для продуктов Google...или Яндекс»*.

## 15. Блок «Редактирование»

Редактирование

Новая рабочая область

Внесенные изменения

0	3	1
Изменено	Добавлено	Удалено

[УПРАВЛЕНИЕ РАБОЧИМИ ОБЛАСТЯМИ](#) >

Рис. 145. Блок «Редактирование»

В нем отображается информация о количестве тех изме-



нений, которые были выполнены в рабочей области перед непосредственной публикацией контейнера. Сумма изменений в этом блоке равна пункту 7 (Изменения в рабочей области).

Из раздела «**Редактирование**» можно перейти в управление рабочими областями (выделенное меню на изображении выше).

## 16. Блок «Опубликованная версия контейнера»

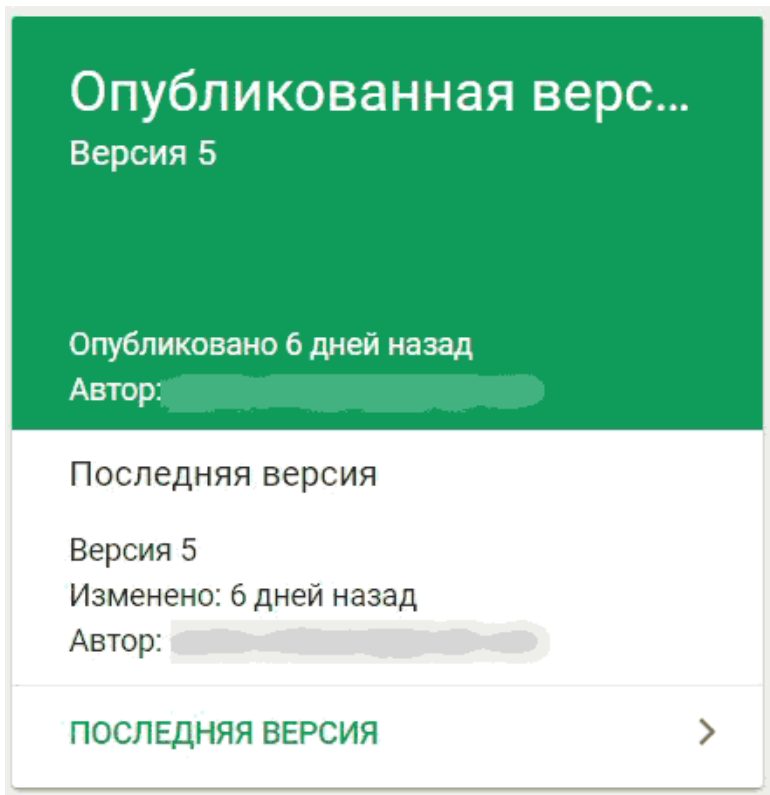


Рис. 146. Блок «Опубликованная версия контейнера»

Через этот блок мы можем перейти в раздел «Версии»,

а также получить информацию об актуальной версии, когда она была изменена и кем.

## 17. Блок «Изменения в рабочей области»



Изменения в рабочей области				
Имя ↑	Тип	Изменение	Последнее измене...	Пользователь
Event	Встроенная переменная	Отключено	10 дней назад	[Redacted]
GA ID	Переменная	Добавлено	10 дней назад	[Redacted]
Код Google Analytics	Тег	Добавлено	10 дней назад	[Redacted]
Код Яндекс.Метрика	Тег	Добавлено	10 дней назад	[Redacted]

Рис. 147. Изменения в рабочей области

Здесь отображаются изменения, которые были выполнены в выбранной рабочей области с момента последней публикации контейнера Google Tag Manager.

Можно использовать сортировку по имени, типу, пользователям и изменениям, а также отменить или просмотреть изменение. При выборе **«Просмотреть изменение»** у нас откроется дополнительное окно с настройками. Слева – это последняя версия (то, что в данный момент опубликовано), а текущая рабочая область – это черновик нашего контейнера.

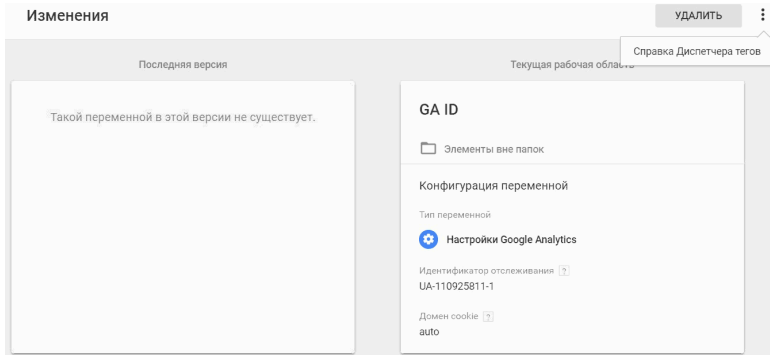


Рис. 148. Блок «Изменения в рабочей области»

В рассмотренном примере – это переменная GA ID с идентификатором отслеживания Google Analytics. В правом верхнем углу доступ справка Диспетчера тегов по рабочим областям.

## 18. Блок «История»



Рис. 149. Блок «История»

При клике на этот блок отобразится история всех изменений в выбранной рабочей области. А именно – кто сделал изменения, что было за действие (создание/изменение/удале-

ние), какого типа использовался элемент в GTM (тег, триггер или переменная), его название и дата изменения.

Пользователь	Действие	Тип	Название	Дата ↓
	Изменено	Триггер	Jivosite	20 дней назад
	Изменено	Тег	Обратная связь JiviSite	20 дней назад
	Создано	Тег	Обратная связь JiviSite	20 дней назад
	Создано	Триггер	Jivosite	20 дней назад
	Удалено	Триггер	Клик по кнопке отправки данных с форм	20 дней назад
	Удалено	Переменная	Type	20 дней назад
	Изменено	Тег	Обратная связь в шапке	20 дней назад
	Изменено	Триггер	Форма обратной связи	20 дней назад
	Удалено	Тег	Обратная связь - форма	20 дней назад
	Изменено	Триггер	Клик по кнопке отправки данных с форм	20 дней назад
	Изменено	Триггер	Клик по кнопке отправки данных с форм	20 дней назад

Рис. 150. История изменений в GTM

Если изменений было достаточно много, можно воспользоваться поиском в правом верхнем углу.

## 19. «Условия использования» и «Политика конфиденциальности»

[Условия использования](#) • [Политика конфиденциальности](#)

## Рис. 151. Условия использования и Политика конфиденциальности Google

Несмотря на то что Google является публичной много-миллиардной компанией и как никто другой заботится о защите персональных данных, не будет лишним хотя бы раз в своей жизни ознакомиться с материалами «**Политики конфиденциальности**» и «**Условиях использования**» их продуктов.

**<https://policies.google.com/privacy?hl=ru>**

По ссылке выше вы сможете узнать:

- какую информацию собирает Google;
- как они используют эту информацию;
- как изменить свои персональные данные;
- какую информацию Google передает третьим лицам;
- какие существуют варианты доступа к данным и их обновления;
- и т.д.

# Управление пользователями

В Google Tag Manager управлять пользователями и предоставлять им различные права доступа можно с помощью двух уровней.

Один из них – это **уровень аккаунта**, а другой – **уровень контейнера**. На первом уровне (аккаунта) доступы бывают двух типов:

Разрешения на уровне аккаунта

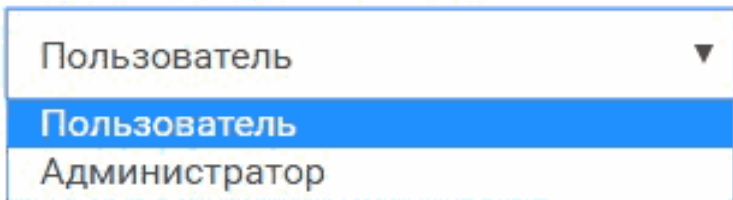


Рис. 152. Разрешения на уровне аккаунта

1. Пользователь;
2. Администратор.

Разрешение на уровне аккаунта типа **«Пользователь»**, к которому предоставлен данный уровень доступа, сможет **просматривать** статистику по аккаунту, список пользователей и настройки аккаунта. Разрешение на уровне аккаунта

та типа «Администратор» позволяет просматривать статистику, управлять списком пользователей (не только просматривать) и изменять настройки аккаунта.

На втором уровне (контейнера) возможны следующие варианты:

### Разрешения на уровне контейнера

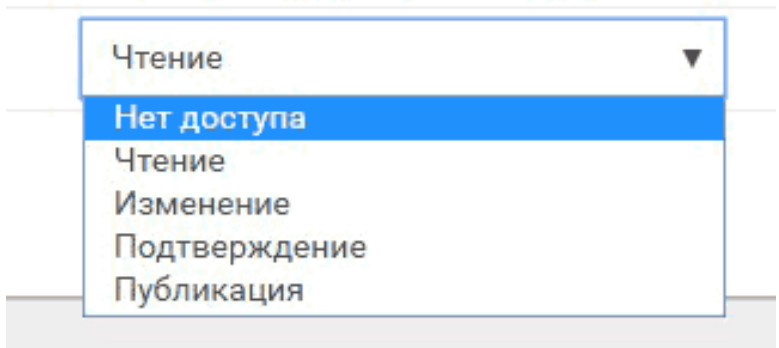


Рис. 153. Разрешения на уровне контейнера

● **Нет доступа.** Пользователь не будет видеть контейнер в аккаунте;



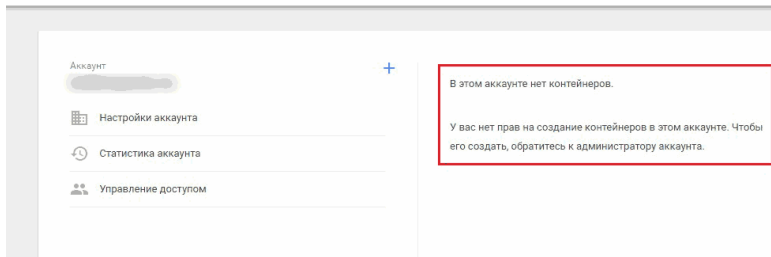


Рис. 154. Права доступа на уровне контейнера – Нет доступа

Разрешение на уровне аккаунта «**Пользователь**», а на уровне контейнера «**Нет доступа**».

● **Чтение.** Пользователь будет видеть контейнер в списке и сможет просматривать теги, триггеры и переменные в контейнере без возможности вносить изменения.

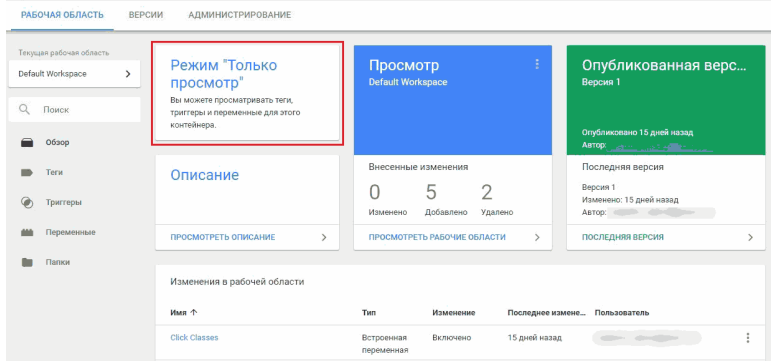


Рис. 155. Права доступа на уровне контейнера – Чтение

Кнопки «Предварительный просмотр» и «Отправить» в правом верхнем углу отсутствуют.

Однако на уровне такого доступа мы можем предварительно просматривать «Версии» и примечания к ним. Создавать версии и публиковать их возможности нет.

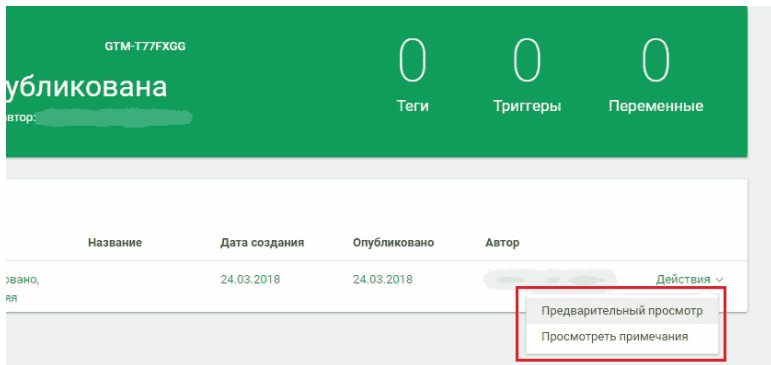


Рис. 156. Предварительный просмотр версий

Мы также можем экспортировать контейнер. Подробнее об импорт-экспорте контейнеров Google Tag Manager будет разобрано в отдельной статье.

Таким образом, разрешение на уровне контейнера «**Чтение**» позволяет просматривать рабочие области, настройки, теги, триггеры и переменные, статистику, пользователей, экспортировать контейнер, осуществлять предварительный просмотр версий и их примечаний.

● **Изменение.** Пользователь имеет право создавать рабочие области и вносить изменения, но не может создавать версии и осуществлять публикацию.

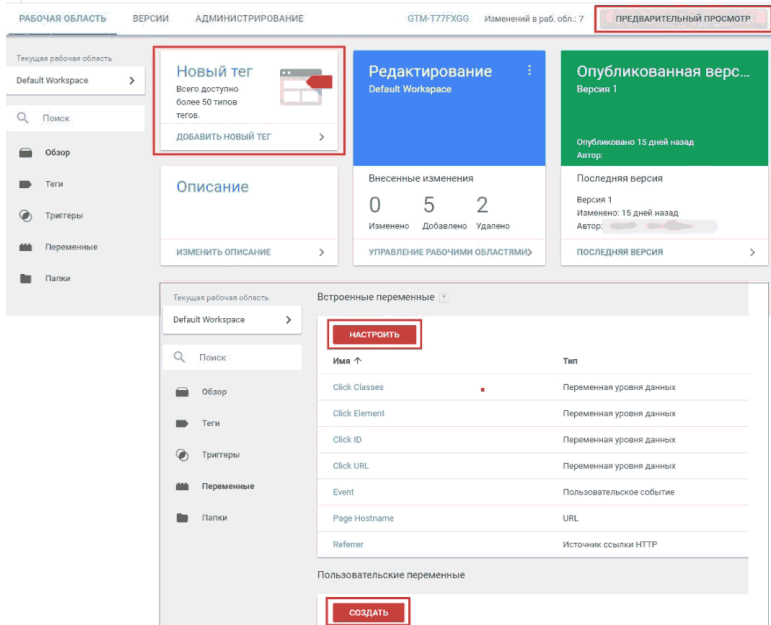


Рис. 157. Права доступа на уровне контейнера – Изменение

Доступны кнопки «Создать», «Настроить», «Предварительный просмотр».

Таким образом, разрешение на уровне контейнера «Изменение» позволяет создавать рабочие области, просматривать настройки, статистику и пользователей, импортировать-экспортировать контейнер, осуществлять предварительный просмотр версий и их примечаний, создавать теги,

триггеры и переменные.

● **Подтверждение.** Пользователь имеет право создавать версии и рабочие области и может вносить изменения, но не осуществлять публикацию.

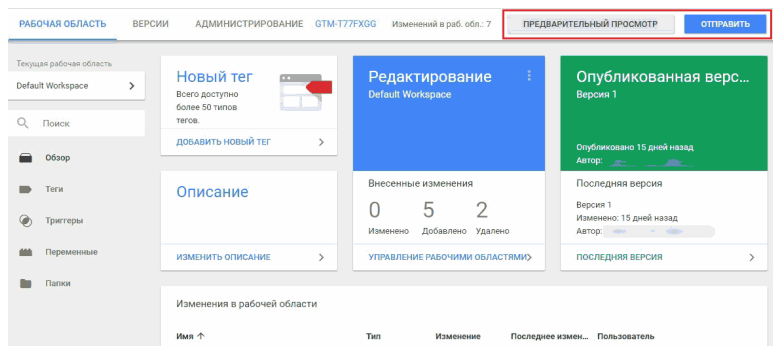


Рис. 158. Права доступа на уровне контейнера – Подтверждение

Доступны кнопки «Создать», «Настроить», «Предварительный просмотр» и «Отправить».

Отличие данного доступа от «Изменение» заключается в том, что разрешение на уровне контейнера «Подтверждение» позволяет еще редактировать настройки контейнера и создавать версии.

● **Публикация.** Пользователь имеет право создавать версии и рабочие области, вносить изменения и осуществлять публикацию.

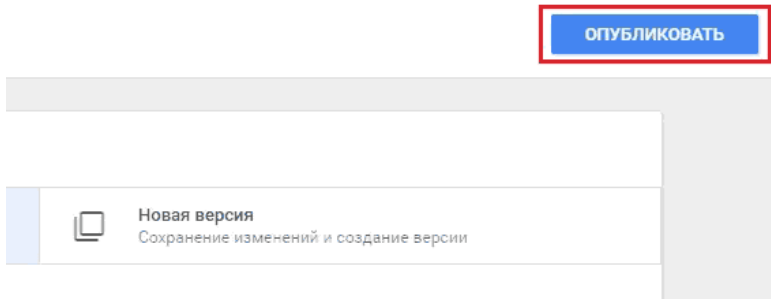


Рис. 159. Доступна публикация изменений

Доступны кнопки **«Создать»**, **«Настроить»**, **«Предварительный просмотр»**, **«Отправить»** и **«Публикация»**.

Наиболее полный доступ среди всех в Google Tag Manager с возможностью публикаций изменений в контейнере.

Чтобы предоставить или изменить права доступа на уровне аккаунта в GTM, необходимо:

- выбрать вкладку **«Администрирование»**;
- в разделе **«Аккаунт»** выберите **«Управление доступом»**;

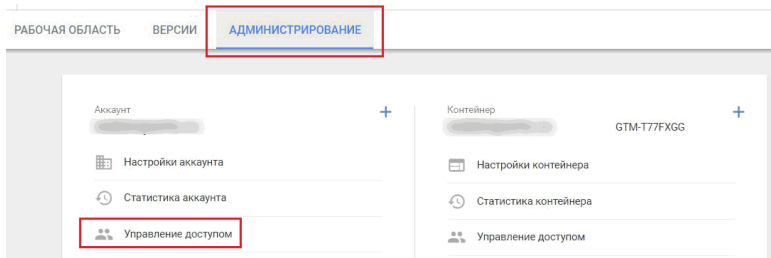


Рис. 160. Администрирование – Управление доступом

● нажмите кнопку **«Создать»**;

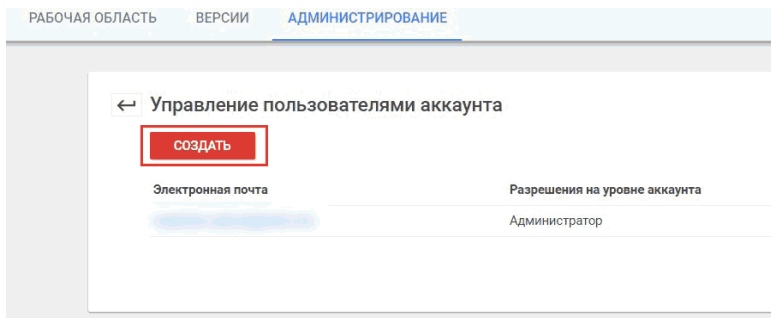


Рис. 161. Управление пользователями аккаунта – Создать

● указываем электронную почту, которая зарегистрирована в сервисах Google (как правило, Gmail.com), разрешение на уровне аккаунта и на уровне контейнера

● нажимаем кнопку **«Добавить»**.

← **Добавление нового пользователя**

Электронная почта

Разрешения на уровне аккаунта

Пользователь

Разрешения на уровне контейнера

Название контейнера

Разрешения на уровне контейнера

Нет доступа

**Нет доступа**

Чтение

Изменение

Подтверждение

Публикация

**ДОБАВИТЬ**    ОТМЕНА

Рис. 162. Добавление нового пользователя в GTM

Пользователь добавлен в аккаунт и ему присвоен соответствующий уровень доступа. Удалить из списка можно кликнув по его электронной почте и нажав на кнопку **«Удалить»**.

← **Изменение разрешений для пользователей**

Электронная почта

Разрешения на уровне аккаунта

Пользователь

Разрешения на уровне контейнера

Название контейнера

Разрешения на уровне контейнера

Публикация

**СОХРАНИТЬ**    ОТМЕНА    **УДАЛИТЬ**

Рис. 163. Изменение разрешений для пользователей – Удалить



Также в Google Tag Manager можно открыть доступ только на уровне контейнера.

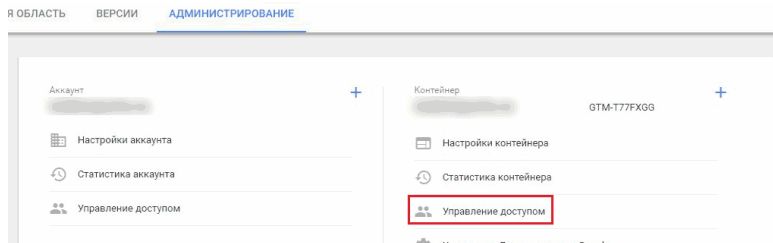


Рис. 164. Управление доступом контейнера

При желании можно также задать разрешения для каждого контейнера, к которому вы предоставляете доступ.

Разрешения на уровне контейнера

Название контейнера

Разрешения на уровне контейнера

Контейнер №1

Чтение

Контейнер №2

Изменение

ДОБАВИТЬ

ОТМЕНА

Рис. 165. Различные разрешения на уровне контейнера

Различные уровни доступа на уровне аккаунта и контейнера в Google Tag Manager предоставляются в зависимости

от решаемых задач. Например, для удаленной настройки уже имеющегося контейнера или для удаленного создания и настройки нового контейнера в аккаунте. Права с уровнем доступа без возможности редактирования можно выдать для поиска причины неработоспособности контейнера, отдельных тегов и т.д.

# Версии

Версия – это сохраненная копия контейнера в текущий момент времени. Делая изменения в рабочей области, вы всегда можете сохранить ее в виде версии. Благодаря этому при необходимости легко возвращаться к предыдущим изменениям. Создавать версии могут пользователи с правом на подтверждение или доступом более высокого уровня.

РАБОЧАЯ ОБЛАСТЬ    **ВЕРСИИ**    АДМИНИСТРИРОВАНИЕ

GTM-NC2KL3M

**Версия 12 опубликована**

Опубликовано Jun 21, 2018, автор: yasha.cmteam@gmail.com

5      4      5

Теги      Триггеры      Переменные

Версии						
Идентификатор в...	Статус	Название	Дата создания	Опубликовано	Автор	
12	Опубликовано, Последняя		21.06.2018	21.06.2018		Действия ▾
11			21.06.2018	21.06.2018		Действия ▾
10			14.05.2018	14.05.2018		Действия ▾
9			13.05.2018	13.05.2018		Действия ▾
8			13.05.2018	13.05.2018		Действия ▾

Рис. 166. Версии в GTM

Когда вы публикуете контейнер, автоматически записывается его версия. Таким образом, в вашем аккаунте будут находиться все опубликованные версии контейнера.

Вы можете сохранить текущий вариант рабочей области

в виде версии без публикации. Для этого во время публикации контейнера переключитесь с «**Публикация и создание версии**» на «**Новая версия**» и нажмите «**Создать**».

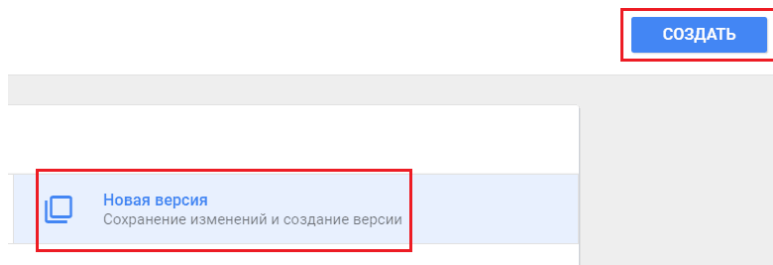


Рис. 167. Новая версия

С версиями в Google Tag Manager можно производить различные действия:

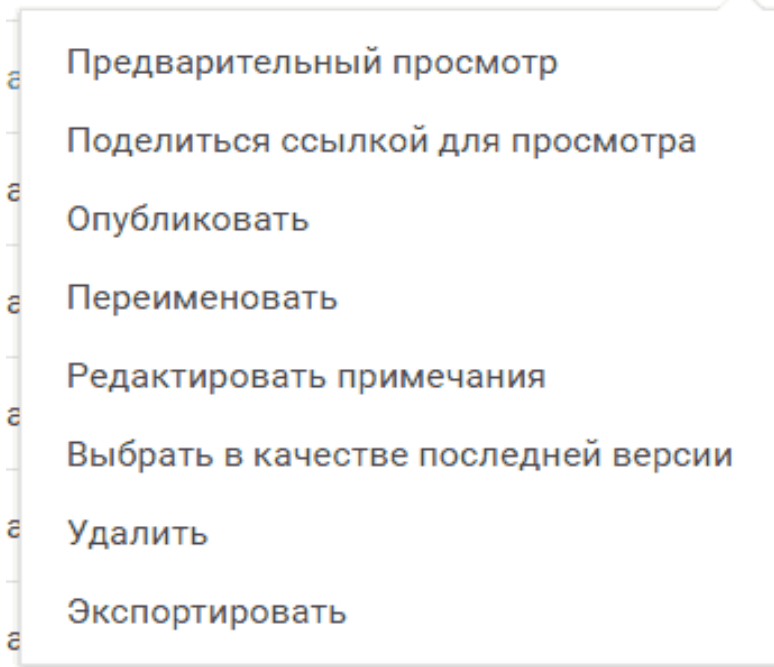


Рис. 168. Действия над версиями

- просматривать в режиме предварительного просмотра (режиме отладки);
- поделиться ссылкой для просмотра с другим человеком;
- опубликовать;
- переименовать;

- редактировать примечания (описание);
- выбрать в качестве последней версии (чтобы заменить текущую версию контейнера ранее сохраненной);
- удалить (будет удалена навсегда через 30 дней);
- экспортировать в другой контейнер (формат файла .json).

### Экспорт контейнера GTM-NC2KL3M версии 12

```
1 {  
2   "exportFormatVersion": 2,  
3   "exportTime": "2018-06-21 19:47:22",  
4   "containerVersion": {  
5     "path": "accounts/3154078312/containers/9017772/versions/12",  
6     "accountId": "3154078312",  
7     "containerId": "9017772",  
8     "containerVersionId": "12",  
9     "container": {  
10      "path": "accounts/3154078312/containers/9017772",  
11      "accountId": "3154078312",  
12      "containerId": "9017772",  
13      "name": "graphanalytics.ru",  
14      "publicId": "GTM-NC2KL3M",  
15      "usageContext": [  
16        "WEB"  
17      ],  
18      "fingerprint": "1529601351494",  
19      "trackingConsent": "1"
```

ЭКСПОРТИРОВАТЬ

ОТМЕНА

Рис. 169. Экспорт определенной версии контейнера

Диспетчер тегов сохраняет историю публикаций, так что вы всегда можете узнать, когда и кем была опубликована та или иная версия.

# Администрирование

Раздел «Администрирование» разделен на два уровня:

1. Аккаунт;
2. Контейнер.

На уровне аккаунта в GTM доступны настройки аккаунта, статистика аккаунта по действиям (создание, изменение, добавление тегов и т.д.) и опции управления доступами.

Администратор может разрешить вносить изменения в контейнеры только пользователям, прошедшим двухэтапную аутентификацию. Эта функция настраивается на уровне аккаунта и применяется в следующих случаях:

- создание или изменение переменных;
- создание или изменение пользовательских тегов HTML;
- изменение настроек пользователя.

## ← Настройки аккаунта

Идентификатор аккаунта

3154078312

Название аккаунта

osipenkov.ru

Страна

Россия

Передавать анонимные данные в Google и другие службы ?

Требовать двухэтапную аутентификацию для некоторых операций ?

Рис. 170. Двухэтапная аутентификация для некоторых операций

На уровне контейнера в GTM доступны следующие настройки:

- настройки контейнера (название и тип контейнера);
- статистика контейнера (создание, изменение, добавление тегов и т.д.);
- управление доступами пользователей;
- установить Диспетчер тегов Google (фрагменты кодов GTM);
- импорт-экспорт контейнеров (формат файла .json);
- связывание внешних аккаунтов (например, DoubleClick Bid Manager с контейнером GTM);
- очередь подтверждения (здесь принимаются запросы



на установление связи с аккаунтом DoubleClick Campaign Manager и подтверждение перемещение тегов);

● среды (подробнее в материалах ниже).



Настройки контейнера

---



Статистика контейнера

---



Управление доступом

---



Установить Диспетчер тегов Google

---



Импортировать контейнер

---



Экспортировать контейнер

---



Связанные внешние аккаунты

---



Очередь подтверждения

---



Среды

---

Рис. 171. Настройки на уровне контейнера

# Импорт-экспорт контейнеров

Для ускорения первичной настройки и переноса данных из одного контейнера в другой в Google Tag Manager можно использовать функции импорта-экспорта.

Например, вы имеете сайт с установленным на нем контейнером GTM. Долгое время вы настраивали все теги, триггеры и переменные. В ближайших планах у вас расширение и сделать еще один сайт, другой тематики, но с той же самой структурой. Чтобы заново вручную не настраивать всю конфигурацию, в Google Tag Manager предусмотрена автоматическая выгрузка контейнера как текстового файла в формате JSON. Всего в несколько кликов вы можете перенести все настройки из одного контейнера GTM в другой.

Несомненно, в новом контейнере все же придется внести некоторые изменения – у них могут быть разные идентификаторы счетчиков, конверсий или доменные имена. Экспортированный файл можно изменить, оставив в нем только нужные триггеры, теги и переменные. Это позволяет создавать стандартные шаблоны контейнеров и делиться ими с другими пользователями Google Tag Manager. Экспортированные контейнеры можно сравнивать между собой, изменять, сохранять в системе управления версиями и импортировать обратно в GTM.

Таким образом, основное практическое применение им-

порт-экспорта контейнеров в GTM – это сокращение времени на выполнение большого количества однотипных действий путем дублирования конфигурации тегов, триггеров и переменных.

В качестве примера я буду переносить контейнер с настроенными тегами Google Analytics и Яндекс.Метрика. Чтобы экспортировать (выгрузить) контейнер, перейдите в «Администрирование» и выберите «Экспортировать контейнер».

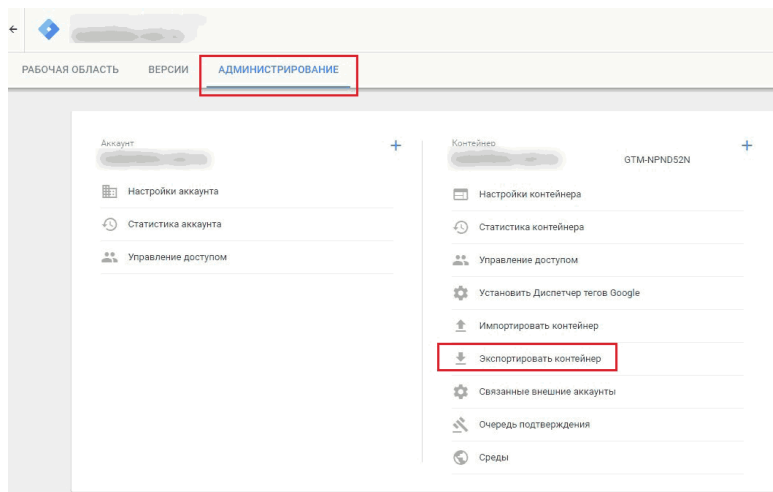


Рис. 172. Администрирование – Экспортировать контейнер GTM

Далее выбираем версию или рабочую область. В рабочей области может содержаться несколько версий. Например, одна для тегов аналитики, другая для целей и событий, в третьей версии присутствуют изменения по электронной торговле и т.д. При выборе рабочей области экспортируются настройки всех версий. Если же вы хотите скопировать настройки только из конкретной версии, выберите только ее. В нашем случае это последняя 4 версия, которая называется Google Analytics и Яндекс.Метрика.

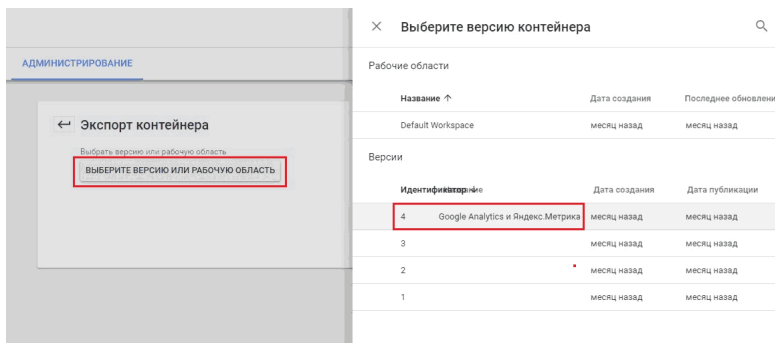


Рис. 173. Выбираем версию контейнера

В следующем окне нам доступен предварительный просмотр содержимого контейнера. Если все верно, нажимаем «Экспортировать».

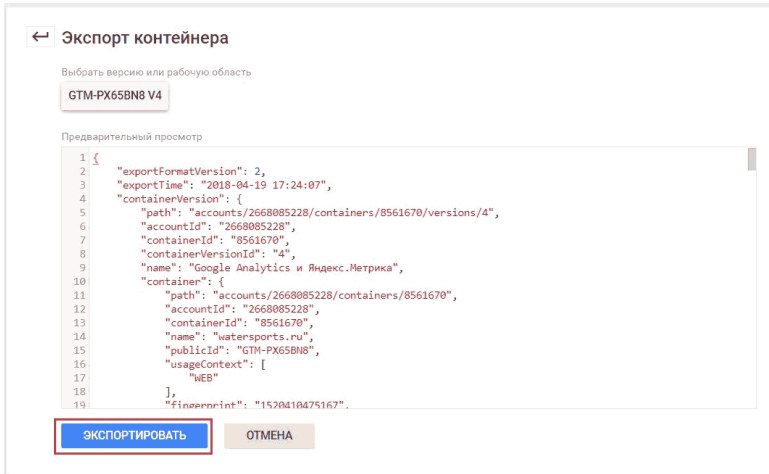


Рис. 174. Экспорт контейнера

На компьютер загрузится файл в формате JSON (формат обмена данными на JavaScript).

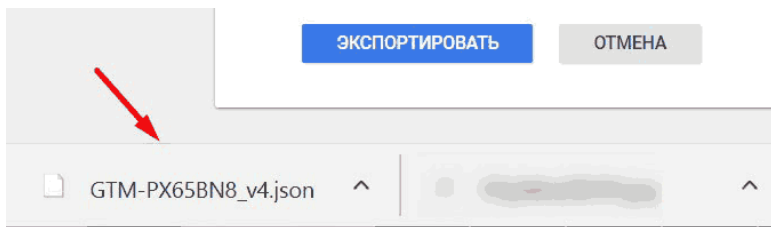


Рис. 175. Загрузка файла .JSON на компьютер

Что дальше делать с этим файлом? Есть два варианта:

1. сразу же импортировать в новый контейнер;
2. сравнить данные с новым контейнером.

Рассмотрим 2 вариант. Перед тем, как загрузить наш файл в другой контейнер, мы можем выгрузить данные из второго контейнера и с помощью утилиты сравнить оба файла (экспортируемого контейнера и текущего, который будем заменять). В этом нам может помочь утилита сравнения файлов `diff`, которая выводит разницу между двумя файлами. Эта программа выводит построчно изменения, сделанные в файле (для текстовых файлов). Если вы используете `diff`, вы можете увидеть, что именно изменилось. Это способ предотвратить проскальзывание нежелательных изменений в файле.

Один из наиболее распространенных сервисов сравнения файлов [jsondiff.com](https://www.jsondiff.com). Просто указываете два пути к файлам на компьютере и нажимаете «**Compare**». Все остальное программа сделает за вас, и где надо, подсветит код.

# JSON Diff

The semantic JSON compare tool

Validate, format, and compare two JSON documents. See the differences between the objects instead of just the new lines and mixed up properties.  
Created by **Zack Grossbart**. Get the [source code](#).  
Big thanks owed to the team behind [JSONLint](#).

Found 26 differences

Perform a new diff

Show:  4 missing properties  22 unequal values

1, {	1, {	<	1 of 26	>
2, "containerVersion": {	2, "containerVersion": {			
3, "accountId": "2668085228",	3, "accountId": "2975377632",			
4, "builtInVariable": {	4, "builtInVariable": {			
5, {	5, {			
6, "accountId": "2668085228",	6, "accountId": "2975377632",			
7, "containerId": "8561670",	7, "containerId": "8862974",			
8, "name": "Page URL",	8, "name": "Page URL",			
9, "type": "PAGE_URL",	9, "type": "PAGE_URL",			
10, },	10, },			
11, {	11, {			
12, "accountId": "2668085228",	12, "accountId": "2975377632",			
13, "containerId": "8561670",	13, "containerId": "8862974",			
14, "name": "Page Hostname",	14, "name": "Page Hostname",			
15, "type": "PAGE_HOSTNAME",	15, "type": "PAGE_HOSTNAME",			
16, },	16, },			
17, {	17, {			
18, "accountId": "2668085228",	18, "accountId": "2975377632",			
19, "containerId": "8561670",	19, "containerId": "8862974",			

Both sides should be equal strings

Рис. 176. JSON Diff

Утилита полезна не только при импорте-экспорте контейнеров в Google Tag Manager, но и тогда, когда требуется сравнить изменения в одном файле относительно другого. На изображении выше было найдено 26 различий, которые JSON Diff подсветил, а выделенная область синим указывает лишь на то, что два контейнера имеют разные **accountID**.

Познакомившись немного с инструментом diff, переходим к импорту нашего контейнера. Для этого переходим в контейнер, в который хотим загрузить новые настройки. Раздел «Администрирование» – «Импортировать контейнер».



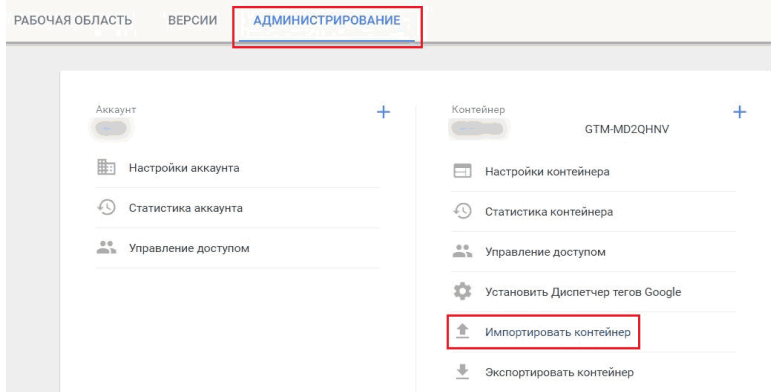


Рис. 177. Администрирование – Импортировать контейнер GTM

Загружаем файл, который мы экспортировали, и выбираем рабочую область. Рекомендация: если у вас уже есть настройки в текущем контейнере, лучше создавать новую рабочую область. Если контейнер пустой, но можно в **«Существующая»**.

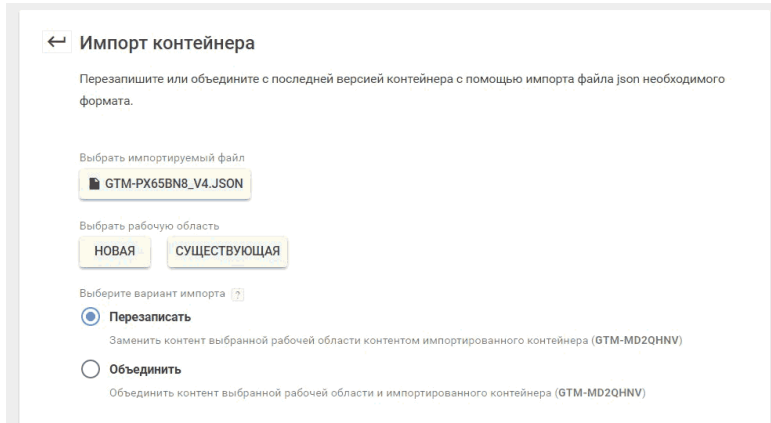


Рис. 178. Импорт контейнера

Выбираем вариант импорта:

- *перезаписать* – все старые настройки заменятся на новые;

- *объединить* – данные двух контейнеров будут объединены. При этом могут возникнуть ошибки. Чтобы это исправить, перезапишите, либо переименуйте конфликтующие теги, триггеры или переменные.

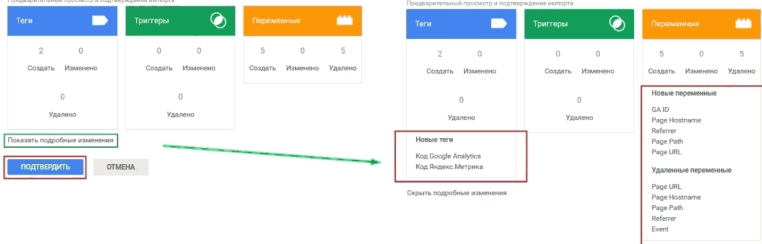


Рис. 179. Подробные изменения в контейнере

В предварительном просмотре Google Tag Manager визуально сообщит нам о том, какие теги, триггеры и переменные будут созданы, изменены или удалены. Можно посмотреть подробных список изменений, нажав на «**Показать подробные изменения**». Если все сделали правильно, останется только «**Подтвердить**».

На этом импорт данных в контейнер завершен. Не забудьте опубликовать контейнер, чтобы текущая конфигурация стала доступна пользователям. Если вы импортируете большое количество данных, перед публикацией обязательно проверяйте все изменения через отладчик GTM.

P. S. В интернете есть большое количество готовых JSON-файлов для Google Tag Manager. Например, компания LunaMetrics на своем сайте [lunametrics.com/labs/recipes/](https://lunametrics.com/labs/recipes/) публикует различные решения с подробным описанием. Analytics Mania в своем блоге также собрала «рецепты» от разных команд и веб-аналитиков, в числе которых: **Simo**

**Ahava, David Vallejo, Pat Grady, Daniel Carlbon и другие.**

# Среды

В Google Tag Manager есть возможность создания так называемых сред, которые позволяют протестировать изменения, внесенные в контейнер, перед их публикацией на сайте. Прежде чем разобраться в этом функционале, постараемся понять, как работает контроль версий в диспетчере тегов.



Рис. 180. Пример версий

При создании тега, триггера и переменной до публикации все изменения хранятся в черновой версии. Чтобы новые данные стали доступны пользователю, версию необходимо опубликовать. Все версии контейнера хранятся на вкладке **«Версии»**.

GTМ-PX65BN8

## Версия 4 опубликована

Опубликовано Мар 7, 2018, автор: [REDACTED]

2 Теги    0 Триггеры    1 Переменная

Версии						
Идентификатор ...	Статус	Название	Дата создания	Опубликовано	Автор	
4	Опубликовано, Последняя	Google Analytics и Яндекс.Метрика	07.03.2018	07.03.2018	[REDACTED]	Действия ▾
3			07.03.2018	07.03.2018	[REDACTED]	Действия ▾
2			07.03.2018	07.03.2018	[REDACTED]	Действия ▾
1			06.03.2018	06.03.2018	[REDACTED]	Действия ▾

Рис. 181. Версии

Во время внесения изменений в конфигурацию отправки версии у нас есть возможность задать среду для публикации.

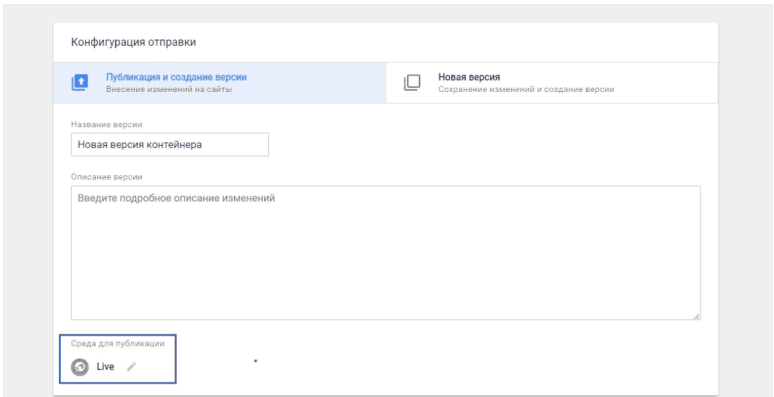


Рис. 182. Задание среды для публикации

Несмотря на то, что обычно публикуется последняя версия контейнера, Google Tag Manager также дает нам возможность переиздавать более старые версии. Эта функция как раз необходима для сред диспетчера тегов.

Когда-то GTM разрешал публиковать только одну версию контейнера. Это означало, что для каждого отдельного веб-сайта у вас был единственный фрагмент кода Google Tag Manager.

Но ситуация кардинально изменилась, когда пришло осознание того, что веб-сайты могут находиться на разных этапах разработки – от предварительной подготовки, производства, до бета-тестирования и окончательного запуска. При таком подходе все равно необходимо устанавливать теги, от-

слеживать статистические данные и обеспечивать должное качество работы.

Рассмотрим пример, в котором у нас есть две версии сайта:

1. тестовый вариант, над которым разработчики ведут работы (недоступен широкой публике);
2. конечный вариант, который в данный момент доступен для пользователей в сети.



Рис. 183. Пример версий сайтов

Обе версии являются копиями одного и того же сайта, и оба используют один и тот же контейнер Google Tag Manager. Но вы не будете публиковать версию тестового варианта, поскольку она сырая и над ней ведутся работы по устранению ошибок.

Благодаря средам в GTM стала доступна одновременная публикация нескольких версий контейнеров, что значительно упрощает тестирование и отладку тегов.





Рис. 184. Две среды разработки: конечная и промежуточная

В приведенном выше примере у нас есть две среды: *конечная* (*live*, 3) и *тестовая* (*промежуточная*, 4). Конечная среда будет содержать теги, которые мы хотим запустить на сайте, а промежуточная среда будет включать в себя дополнительные теги, которые либо проходят тестирование, либо не готовы к публикации в финальной версии.

Чтобы получить доступ к средам, перейдите в раздел «Администрирование» и откройте «Среды» на уровне контейнера.

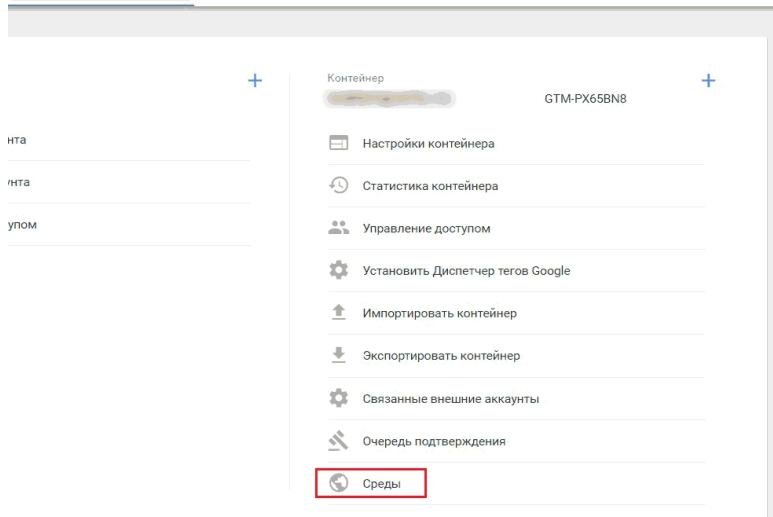


Рис. 185. Администрирование – Среды

По умолчанию в контейнере создается две среды – **Live** (реальная) и **Latest** (последняя).

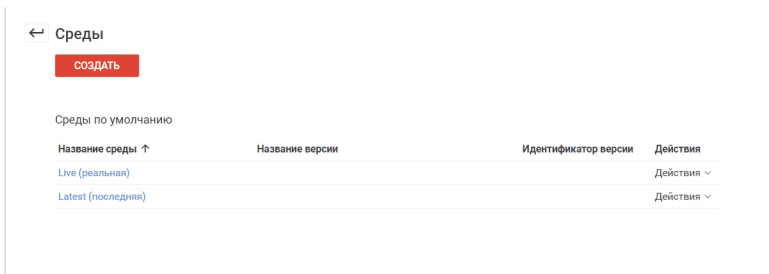


Рис. 186. Среды в GTM по умолчанию

**Live (реальная)** среда всегда указывает на версию контейнера, которая была опубликована, а **Latest (последняя)** среда указывает на последнюю версию контейнера, которая была создана. Однако это не обязательно та же версия, что и Live.

В Google Tag Manager можно создавать собственные среды. Для этого нажмите кнопку **«Создать»**. В конфигурации будет доступно 4 опции:

1. название среды;
2. описание (чтобы другие пользователи вашего контейнера GTM могли быстро определить, для чего используется среда);
3. включение отладки;
4. поле с указанием целевого URL.

Заполнив все необходимые данные, нажмите **«Создать среду»**.

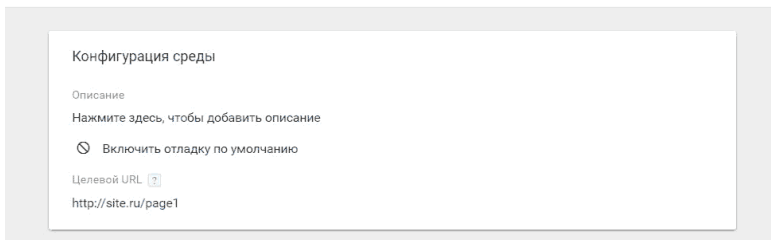


Рис. 187. Создание новой среды

Далее сообщение от Google о начале работы с новой средой.

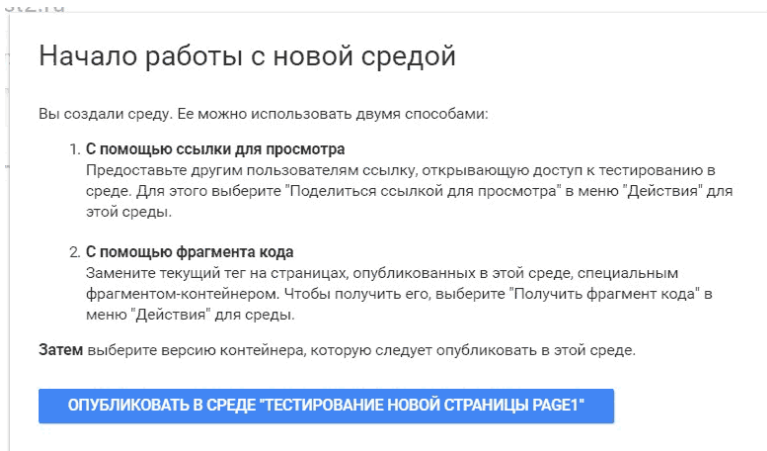


Рис. 188. Использование среды двумя способами

Вы создали среду. Ее можно использовать двумя способами:

### **1. С помощью ссылки для просмотра**

Предоставьте другим пользователям ссылку, открывающую доступ к тестированию в среде. Для этого выберите «Поделиться ссылкой для просмотра» в меню «Действия» для этой среды.

### **2. С помощью фрагмента кода**

Замените текущий тег на страницах, опубликованных в этой среде, специальным фрагментом-контейнером. Чтобы получить его, выберите «Получить фрагмент кода» в меню «Действия» для среды.

**Примечание:** не устанавливайте фрагмент контейнера среды рядом со стандартным фрагментом контейнера Google Tag Manager на том же сайте. Это может привести к дальнейшим ошибкам.

Также можно загружать среды на сайт с помощью функции предварительного просмотра. При этом браузер автоматически перейдет в режим предварительного просмотра, и вы сможете протестировать любую версию среды. Таким образом вы получаете доступ ко всем функциям тестирования без установки и внедрения фрагментов кода.

Если вы уже работали с контейнером и имеете различные версии, то система предупредит о том, что при публикации эти изменения появятся на сайте. Вы можете сначала выпол-

нить предварительный просмотр и отладку.

Публикация в среде "Тестирование новой"

При публикации эти изменения появятся на сайте. Вы можете сначала выполнить предварительный просмотр и отладку.

GTМ-PX65BN8

3	2	0	1
Неопубликованные изменения	Измененные теги	Измененные триггеры	Измененная переменная

Версия

Версия 3 ▼

**ОПУБЛИКОВАТЬ СЕЙЧАС**

ОТМЕНА

Рис. 189. Выбор версии и публикация в среде

Если же контейнер новый и не имеет различных версий, то при публикации выбранная версия будет перемещена в только что созданную среду. Нажимаем **«Опубликовать в среде»** и возвращаемся в меню, где видим только что созданную специальную среду.

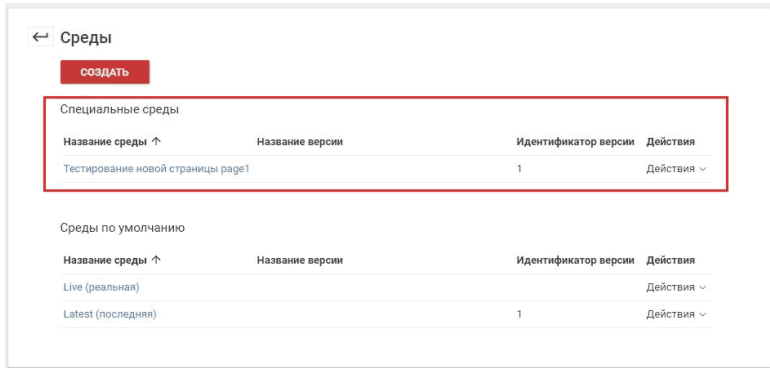
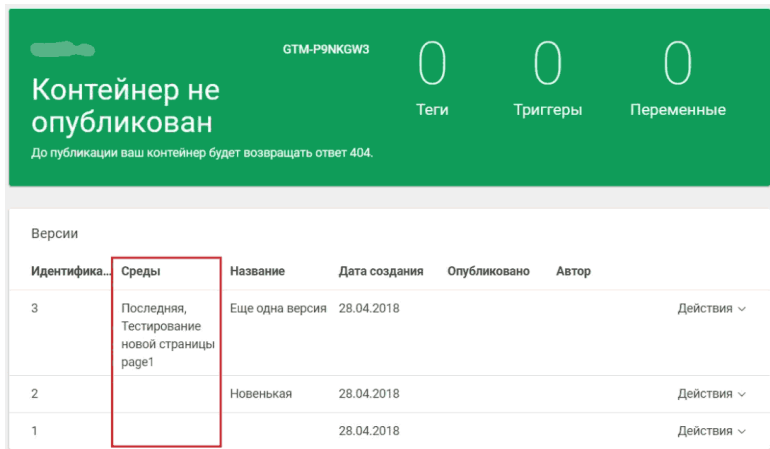


Рис. 190. Специальные среды

В разделе «Версии» появится столбец «Среды», в котором для каждой версии будет указана среда публикации.



## Рис. 191. Новый столбец «Среды»

В отличие от сред по умолчанию, которые динамически изменяются в зависимости от того, какая версия вашего контейнера GTM активна, пользовательские среды всегда будут загружать одну и ту же версию, если вы явно не измените опубликованную версию.

Вы можете задать неограниченное количество сред и опубликовать любые версии контейнеров в любой из них. Действия, которые можно совершать над средами:

Идентификатор версии	Действия
3	Действия ▾ <ul style="list-style-type: none"><li>Изменить настройки</li><li>Опубликовать в среде...</li><li>Поделиться ссылкой</li><li>Получить фрагмент кода</li><li>Сбросить среду</li><li>Удалить</li></ul>
Идентификатор	
3	



## Рис. 192. Действия со средами

- изменить настройки (описание, целевой URL, режим отладки);
- опубликовать в среде (возможность смены версии контейнера, можно задать любую версию контейнера, откатиться на более раннюю);
- поделиться ссылкой (открытие доступа к просмотру и тестированию в среде);

Поделитесь ссылкой для просмотра ×

Ссылка на предварительный просмотр **Тестирование новой страницы page1** – открывает **версию 3**.

Целевой URL ?

http://site.ru/page1

Разрешить отладку при просмотре

Отправьте ссылку тем, кому вы хотите разрешить предварительный просмотр этой среды контейнера:

```
https://www.googletagmanager.com/start_preview/gtm?uiiv2&id=GTM-P9NKGW3&gtm_auth=Lq867uNCr-dJmvsW4RoxpQ&gtm_preview=env-8&gtm_debug=x&url=http://site.ru/page1
```

**ЗАКРЫТЬ**

## Рис. 193. Поделиться ссылкой для просмотра

- получить фрагмент кода (от основного контейнера GTM отличаются параметры **gtm\_auth** и **gtm\_preview**);

Скопируйте приведенный ниже код и вставьте его на каждую страницу тестовой среды.

Вставьте этот фрагмент в раздел **<head>** кода страницы как можно ближе к началу:

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl+ '&gtm_auth=Lq867uNCr-
dJmvsW4RoxpQ&gtm_preview=env-8&gtm_cookies_win=x';f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-P9NKGW3');</script>
<!-- End Google Tag Manager -->
```

Также вставьте непосредственно после открывающего тега **<body>** следующий фрагмент:

```
<!-- Google Tag Manager (noscript) -->
<noscript><iframe src="https://www.googletagmanager.com/ns.html?id=GTM-
P9NKGW3&gtm_auth=Lq867uNCr-dJmvsW4RoxpQ&gtm_preview=env-8&gtm_cookies_win=x"
height="0" width="0" style="display:none;visibility:hidden"></iframe></noscript>
<!-- End Google Tag Manager (noscript) -->
```

Рис. 194. Фрагмент кода для тестовой среды

## ● сбросить среду;

Если вы не хотите, чтобы ссылка, которой вы поделились с другим человеком, была доступна, вы можете сбросить код авторизации (параметр **gtm\_auth**).

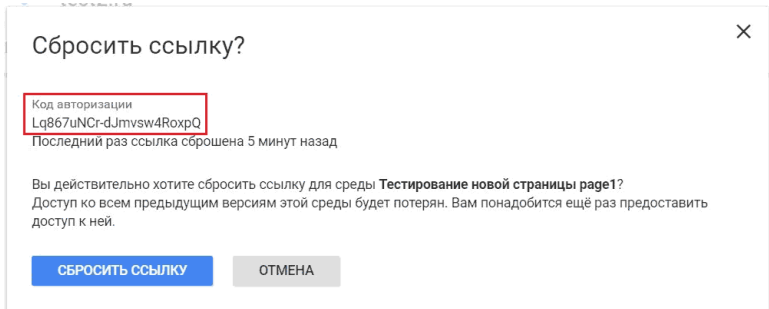


Рис. 195. Сбросить ссылку в GTM

**Примечание:** все установленные фрагменты кода также станут недействительны. Поэтому после сброса ссылки необходимо установить на сервер среды новый фрагмент кода контейнера Google Tag Manager.

● удалить среду;

Удаление среды приведет к потере доступа навсегда.

Как правило, среды в Google Tag Manager предназначены для работы с промежуточными версиями сайта. Если же сайт готов к полноценному запуску и никаких изменений в ближайшее время не предвидится, рекомендуется устанавливать стандартный фрагмент кода контейнера GTM.

# Папки

Еще один способ организации элементов внутри контейнера Google Tag Manager – это папки.

Со временем в Google Tag Manager накапливается большое количество тегов, триггеров и переменных. У нас есть возможность каждый объект в контейнере отнести к какой-то своей логической сущности. Иными словами, сгруппировать их таким образом, чтобы в дальнейшем было проще работать. А помогут нам в этой систематизации папки.

Группировать создаваемые внутри контейнера GTM элементы можно по-разному. Например:

- по инструментам (отдельно завести папку для Google AdWords, отдельно для Google Analytics и т.д.);
- по типу решаемой задачи (отслеживание определенного события или группы событий);
- по типу выполняемой работы (отдельно завести папку для специалиста по контекстной рекламе, отдельно для человека, который занимается SEO, отдельно для веб-аналитика и т.д.);
- по проектам (для каждого микросайта или рекламной кампании);
- по подрядчикам (отдельно для агентства такого-то, организации такой-то и т.д.);
- как-то иначе.

Чтобы создать новую папку, перейдите в раздел «Папки» и выберите «Новая папка».

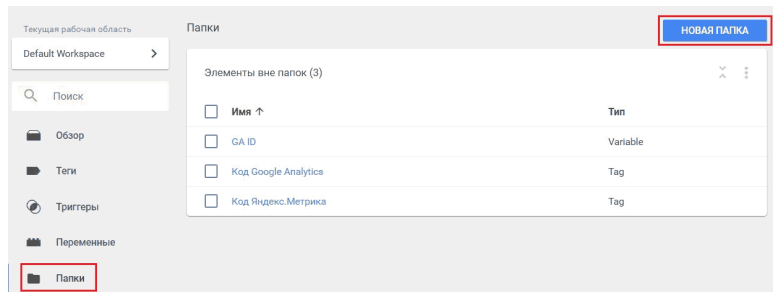


Рис. 196. Создание новой папки

Укажите ее название (например, «Папка Google Analytics») и нажмите «Создать». Имена папок следует задавать таким образом, чтобы из названия сразу было понятно, что находится в ней или к какой сущности принадлежат внутренние элементы.

# Новая папка

Укажите название папки:

Папка Google Analytics

СОЗДАТЬ

ОТМЕНА

Рис. 197. Название папки

Папка создана и теперь может быть использована для группировки различных объектов контейнера GTM. Перенесем в нее элементы, относящиеся к Google Analytics. В нашем примере это переменная GA ID (идентификатор счетчика Google Analytics) и сам тег Universal Analytics.

Ставим галочки напротив этих объектов и вверху выбираем **«Переместить – Папка Google Analytics»**.

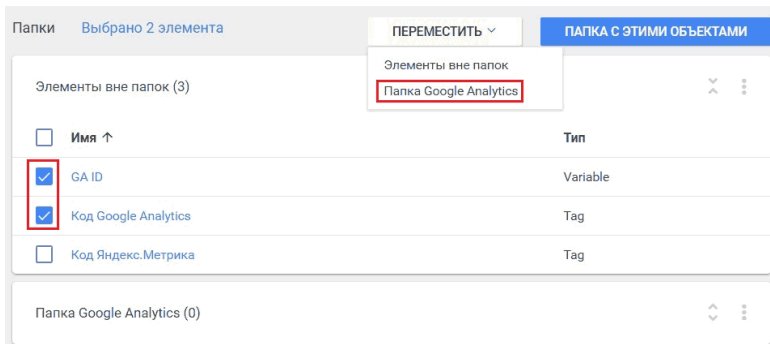


Рис. 198. Перемещение элементов в папку

После загрузки GTM выбранные элементы будут добавлены в папку.

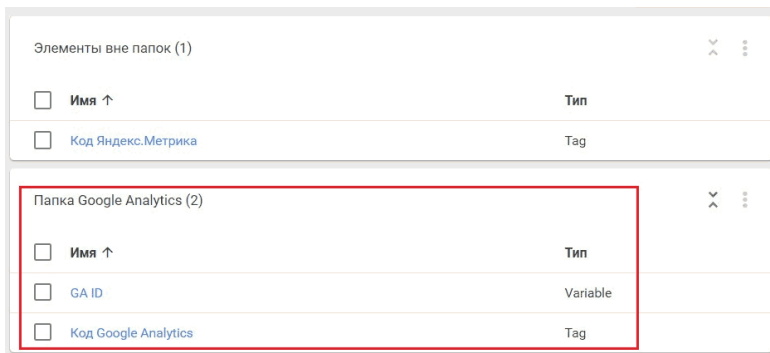


Рис. 199. Элементы в папке

Если нам необходимо исключить какой-либо объект из

папки, то выбираем данный элемент и вверху таблицы выбираем **«Переместить – Элементы вне папок»**.

Чтобы добавить элементы в папку, не обязательно было сначала создавать новую папку, а затем перемещать их в нее. Можно было сначала выбрать необходимые объекты, а далее нажать на **«Папка с этими объектами»**. После чего Google Tag Manager предложит создать папку.

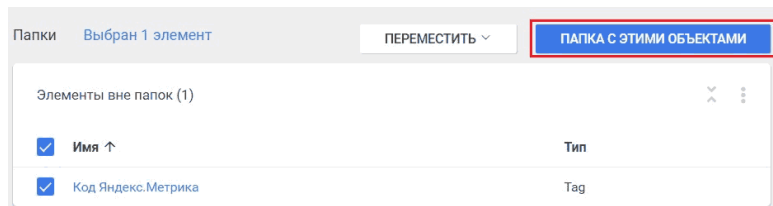
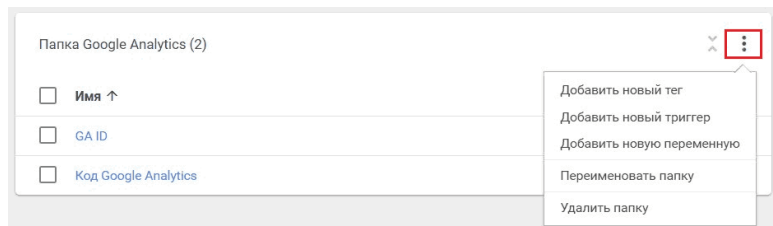


Рис. 200. Папка с этими объектами

Можно использовать оба варианта при группировке элементов в папки. Управлять папками и их содержимым можно с помощью меню.





## Рис. 201. Управление папками

Доступны:

- добавить новый тег;
- добавить новый триггер;
- добавить новую переменную;
- переименовать папку;
- удалить папку.

Если вы заходите удалить папку, в которой присутствуют теги, триггеры или переменные, то вы сначала должны очистить ее содержимое, сделать ее пустой, иначе GTM выдаст вам предупреждение:

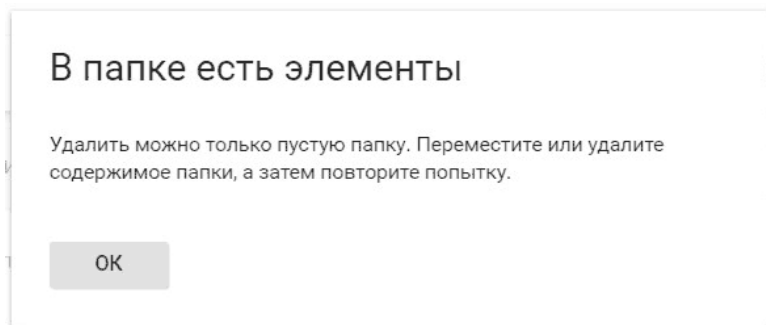


Рис. 202. Предупреждение Google Tag Manager

При создании новых объектов в GTM вы можете сразу указать конечную папку с помощью иконки:

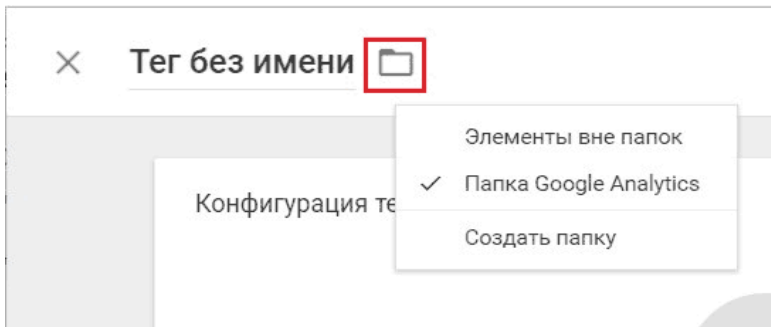


Рис. 203. Иконка папки GTM

И последнее. После создания папок в таблицах тегов, триггеров и переменных появляется дополнительный столбец «Папка», по которому можно сортировать элементы.

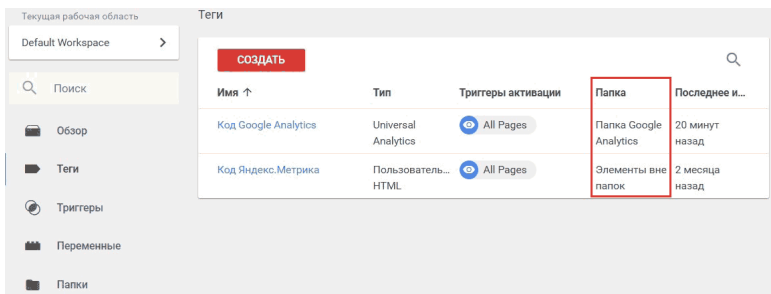


Рис. 204. Дополнительный столбец «Папка»

# Переменные

**Переменная в Google Tag Manager** – это объект, которому дано имя и который может принимать различные значения (вида пара «ключ-значение»). Имя переменной постоянно, а ее значение меняется в зависимости от того, какие действия происходят на сайте. Устаревшее название переменной в предыдущей версии диспетчера тегов Google – *макрос*.

Переменные в GTM практически ничем не отличаются от переменных в JavaScript. У каждого события на сайте есть свои переменные, в которых находится информация об этом событии. Например, когда в интернете-магазине мы нажимаем на кнопку *«Добавить в корзину»*, в переменных этого события будут лежать значения CSS-класса этого элемента или сам текст элемента (кнопки). При клике на ссылку в переменной будет передаваться еще и URL-адрес.

Переменные могут применяться как в триггерах, так и в тегах. В триггерах они используются в качестве фильтра активации, то есть когда следует запускать тот или иной тег.

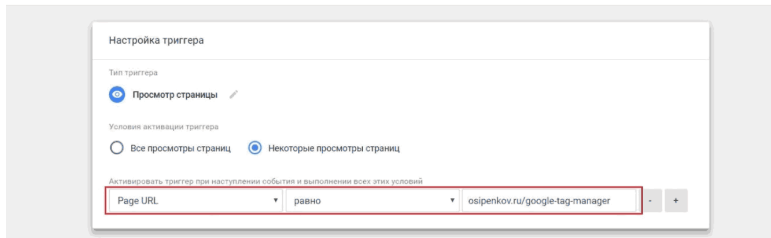


Рис. 205. Переменные в триггерах

Например, запускать триггер **«Просмотр страницы»** только тогда, когда переменная **«Page URL»** равна значению *osipenkov.ru/google-tag-manager*. Или выбрав тип триггера **«Клик – Все элементы»**, вы можете с помощью условий (переменная + фильтр + значение) активировать триггер по нажатию на определенные элементы, и т.д.

Переменные в тегах используются для получения динамических значений. Например, в Google Analytics о сумме транзакции, ID заказа, идентификаторе продукта и т.д.

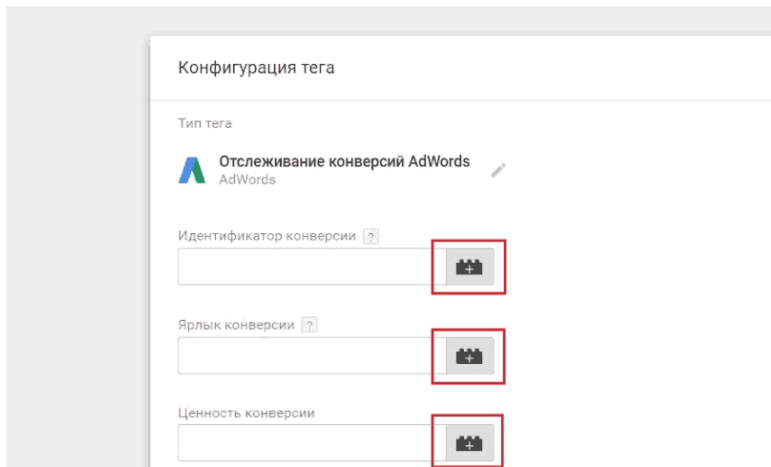


Рис. 206. Поле шаблона «Переменная»

В примере выше с помощью иконки (поле шаблона) «**Переменная**» для типа тега «**Отслеживание конверсий AdWords**» задаются значения для различных полей – идентификатор отслеживания, ярлык конверсии, ценность конверсии и т.д.

Также с переменными можно работать и в пользовательских HTML-тегах. Для этого нам необходимо использовать специальную конструкцию, заключенную в теги `<script></script>`. Тег `<script>` предназначен для описания скриптов и является контейнером для сценария, выполняющегося на стороне клиента, то есть в браузере пользователя. Как пра-

вило, тегом `<script>` подключают JavaScript.

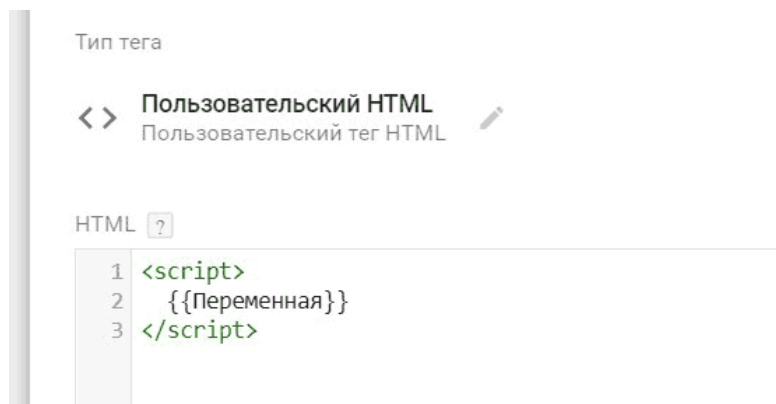


Рис. 207. Пользовательский HTML с переменной

Данный код работать не будет, поскольку он является лишь демонстрацией того, как в пользовательских HTML-тегах можно обращаться к переменным.

В GTM вызов функции переменной выполняется с помощью специального синтаксиса:

**{{имя переменной}}**

Переменные шаблона в Google Tag Manager, как и выражения, заключаются в двойные фигурные скобки `{{ }}` с обеих сторон. Данные в шаблон передаются в виде обычного объекта, переменным шаблона соответствуют свойства этого объекта.

С такой конструкцией мы с вами познакомились несколько ранее, когда устанавливали счетчики Google Analytics и Яндекс.Метрика через GTM. Там в качестве переменной выступал идентификатор отслеживания Google Analytics. Чтобы не вводить ID вручную при создании каждого тега, переменная для Universal Analytics создавалась как пользовательская. В дальнейшем, если идентификатор отслеживания GA изменится, достаточно в одном месте указать новый ID счетчика.

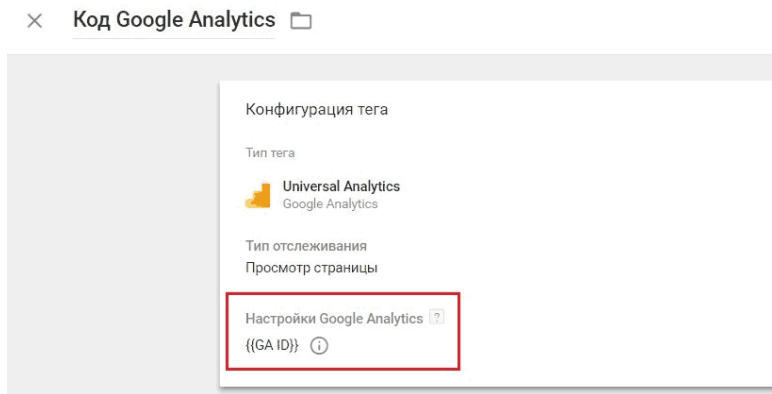


Рис. 208. Пользовательская переменная с ID счетчиком Google Analytics

**Примечание:** как только вы начнете использовать шаблон переменной и напишите {{, Google Tag Manager предло-

жит вам выбрать вариант из доступных переменных.

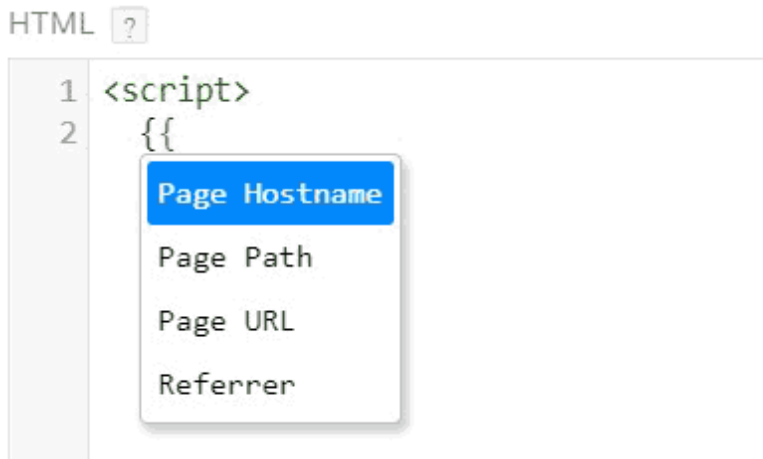


Рис. 209. Подсказка при написании

Любопытный момент: при изменении имени переменной все ссылки на переменную автоматически обновляются. Даже те, которые вы ввели вручную в пользовательских HTML-тегах и пользовательских переменных JavaScript. Это сильно экономит время, поскольку не нужно проходить каждую ссылку и переписывать код в соответствии с новым именем переменной.

Источником информации могут быть:

- слои данных;
- переменные JavaScript;



- cookies сайта;
- Document Object Model.

В Google Tag Manager переменные делятся на две категории:

- 1. встроенные переменные** – набор готовых, предоставленных в системе переменных;
- 2. пользовательские переменные** – самостоятельное создание с использованием существующих типов переменных.

## Встроенные переменные

Встроенные переменные создаются автоматически и их нельзя изменить. Часть из них в GTM не активна по умолчанию.

Встроенные переменные <span>?</span>	
<a href="#">НАСТРОИТЬ</a>	
Имя ↑	Тип
Event	Пользовательское событие
Page Hostname	URL
Page Path	URL
Page URL	URL
Referrer	Источник ссылки HTTP

## Рис. 210. Встроенные переменные GTM

Они включаются и отключаются путем простановки галочек (чекбоксов) напротив каждой из них.

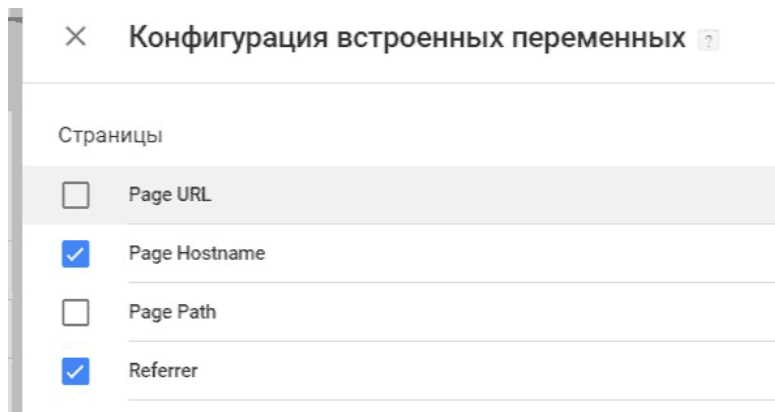


Рис. 211. Включение/Отключение встроенных переменных

Использование встроенных переменных не требует глубоких знаний в программировании и иногда позволяет заменить большое количество строк кода всего парой кликов мыши.

В Google Tag Manager на данный момент существует 9 категорий встроенных переменных:

- Страницы

- Утилиты
- Ошибки
- Клики
- Формы
- История
- Видео
- Прокрутка
- Видимость

# Страницы

## Страницы

<input checked="" type="checkbox"/>	Page URL
<input checked="" type="checkbox"/>	Page Hostname
<input checked="" type="checkbox"/>	Page Path
<input checked="" type="checkbox"/>	Referrer

Рис. 212. Встроенные переменные «Страницы»

Наиболее часто используемые переменные. Они позволяют определить где пользователь находится в данный момент относительно URL-структуры сайта. Страница может быть как текущая, на которой в данный момент находится посетитель сайта, так и та страница, с которой он перешел на текущую страницу (Referrer).

**Page URL** – переменная возвращает полный URL текущей страницы, но без хэша # (все, что после решетки). Например, если пользователь загрузил страницу `https://osipenkov.ru/analytics?parameter=true#vars`, то переменная вернет значение `https://osipenkov.ru/analytics?parameter=true`

**Page Hostname** – переменная возвращает имя хоста (доменное имя) в URL текущей страницы. Например, для страницы `https://osipenkov.ru/analytics?parameter=true#vars` дан-

ная переменная вернет значение `osipenkov.ru`.

**Page Path** – переменная возвращает путь к странице в текущем URL без учета GET-параметров, то есть всего того, что идет в URL после знака вопроса «?» и «&» между параметрами.

Например, на странице `https://osipenkov.ru/analytics?parameter=true#vars` в GTM данная переменная возвращает значение `/analytics`, а в Google Analytics Page Path вернет значение `/analytics?parameter=true`

**Referrer** – полный URL перехода к текущей странице, то есть путь (ссылка) откуда пришел пользователь на текущую страницу.

Например, если пользователь со страницы `https://osipenkov.ru/google-analytics-book/` перешел на страницу `https://osipenkov.ru/google-tag-manager/`, то значение данной переменной будет `https://osipenkov.ru/google-analytics-book/`

Первые три переменные предназначены для текущей страницы, а последняя Referrer необходима для работы с той URL-страницы, с которой пользователь перешел на наш сайт. По умолчанию в Google Tag Manager все 4 встроенные переменные из категории «**Страницы**» активированы.

В качестве примера разберем простой переход из поисковой системы Google на сайт `graphanalytics.ru`. В режиме отладки GTM это будет выглядеть так:

Page Hostname	URL	string	'graphanalytics.ru'
Page Path	URL	string	'/'
Page URL	URL	string	'https://graphanalytics.ru/'
Referrer	Источник ссылки HTTP	string	'https://www.google.ru/'

Рис. 213. Пример встроенных переменных «Страницы»

Переменная **Page Hostname** вернула имя домена «*graphanalytics.ru*», **Page Path** содержит путь к странице в текущем URL, что соответствует «/» (главная директория), **Page URL** – полный URL текущей страницы «*https://graphanalytics.ru/*», а переход на сайт был выполнен из поиска Google, поэтому переменная **Referrer** вернула значение «*https://www.google.ru/*».

# Утилиты

## Утилиты

<input checked="" type="checkbox"/>	Event
<input type="checkbox"/>	Environment Name
<input type="checkbox"/>	Container ID
<input type="checkbox"/>	Container Version
<input type="checkbox"/>	Random Number
<input type="checkbox"/>	HTML ID

Рис. 214. Встроенные переменные «Утилиты»

Набор из 6 встроенных переменных в GTM, которые в большей степени носят служебные функции.

**Event** – возвращает подстроку, которая содержит тип события, произошедшее на сайте. Например, при нажатии на любой элемент возвращает **gtm.click**, на ссылку – возвращает **gtm.linkClick**, при заполнении формы – **gtm.formSubmit**, при возникновении ошибки – **gtm.pageError**.

**Environment Name** – возвращает название текущей среды («Реальная», «Последняя» или «Редактирование»), если запрос контейнера выполнен из функции «Поделиться ссыл-

кой для просмотра» или из фрагмента кода среды.

**Container ID** – возвращает номер контейнера GTM. Например, GTM-NC2LK3M.

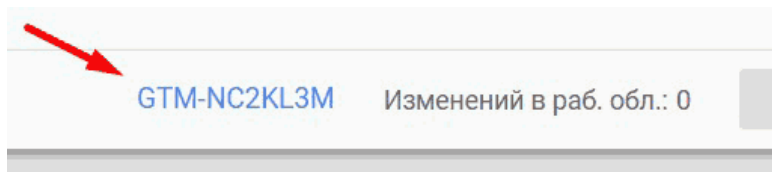


Рис. 215. Container ID

**Container Version** – возвращает номер версии контейнера в виде строкового значения. Например, 5.

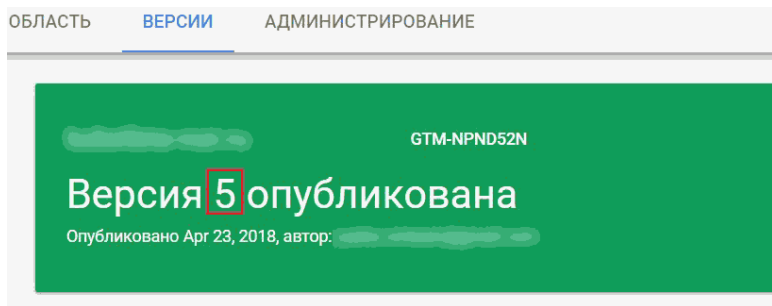


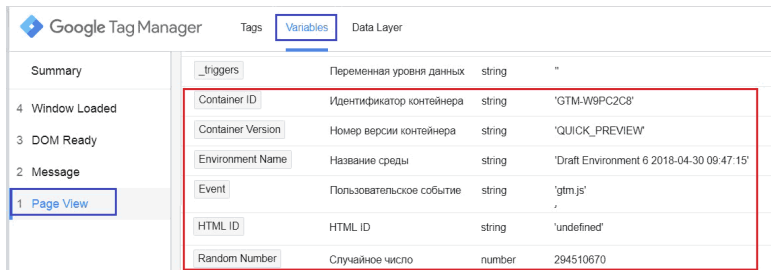
Рис. 216. Container Version

**Random Number** – возвращает случайное число от 0 до 2147483647. Например, 666.



**HTML ID** – возвращает идентификатор пользовательского HTML-тега. Используется с секвенированием (порядком активации) тегов.

Если мы с вами перейдем в отладчик GTM, то увидим такие значения переменных из категории «Утилиты»:



Summary	Variables	Data Layer
4 Window Loaded	Container ID	Идентификатор контейнера string 'GTM-W9PC2C8'
3 DOM Ready	Container Version	Номер версии контейнера string 'QUICK_PREVIEW'
2 Message	Environment Name	Название среды string 'Draft Environment 6 2018-04-30 09:47:15'
1 Page View	Event	Пользовательское событие string 'gtm.js'
	HTML ID	HTML ID string 'undefined'
	Random Number	Случайное число number 294510670

Рис. 217. Пример встроенных переменных «Утилиты»

В качестве события был выбран Page View (Просмотр страницы):

- **Container ID** – GTM-W9PC2C8 (идентификатор контейнера);

- **Container Version** – QUICK\_PREVIEW (режим предварительного просмотра);

- **Environment Name** – Draft Environment 6 2018–04–30 09:47:15 (название среды);

- **Event** – gtm.js (событие просмотра страницы);

- **HTML ID** – значение не присвоено. Значение ID переменной HTML отображается только в том случае, если мы

работаем с тегами типа пользовательский HTML;

● **Random Number** – 294510670 (генератор случайных чисел). При обновлении страницы и фиксации нового события данное число изменится.

# Ошибки

## Ошибки

Error Message

---

Error URL

---

Error Line

---

Debug Mode

---

Рис. 218. Встроенные переменные «Ошибки»

Переменные из этой категории позволяют разработчикам анализировать ошибки, происходящие на сайте (анализ страниц, типы устройств, версий браузеров, разрешений экранов и т.д.). Для использования этих переменных необходимо их активировать и создать хотя бы один триггер типа **«Ошибка JavaScript»**.

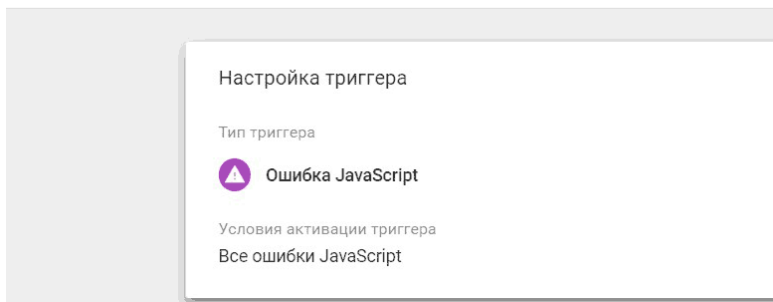


Рис. 219. Тип триггера – Ошибка JavaScript

**Error Message** – возвращает строку, содержащую сообщение об ошибке, отправленное с помощью триггера «Ошибка JavaScript».


**Error URL** – возвращает строку, содержащую URL-адрес скрипта, в котором была обнаружена ошибка.

**Error Line** – строка файла, в которой произошла ошибка.

**Debug Mode** – возвращает значение «true», если активирован режим отладки в Google Tag Manager.

В качестве наглядного примера создадим вынужденную ошибку в коде JS. Но перед тем, как запускать отладчик Google Tag Manager, не забываем создать новый триггер «**Ошибка JavaScript**» и обновить режим предварительного просмотра. Только после этого изменения вступят в силу.

## Предварительный просмотр рабочей области "Default Workspace"

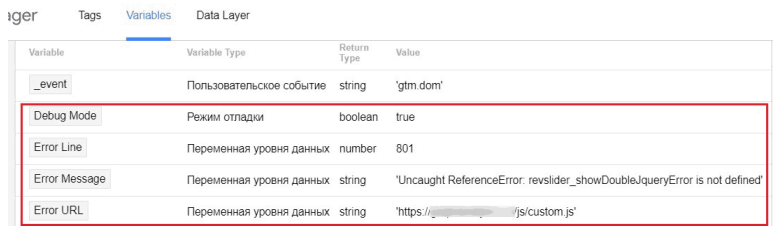
Предварительный просмотр и отладка этой рабочей области доступны при посещении вашего сайта с помощью Chrome DevTools. Не открывается панель отладки? Попробуйте полностью обновить страницу, не загружая контент из кеша. 

Обновить

Выйти из режима предварительного просмотра

Рис. 220. Обновление режима предварительного просмотра

В отладчике GTM будет доступна следующая информация:



Variable	Variable Type	Return Type	Value
_event	Пользовательское событие	string	'gtm.dom'
Debug Mode	Режим отладки	boolean	true
Error Line	Переменная уровня данных	number	801
Error Message	Переменная уровня данных	string	'Uncaught ReferenceError: revslider_showDoubleJqueryError is not defined'
Error URL	Переменная уровня данных	string	'https://[redacted]/js/custom.js'

Рис. 221. Пример встроенных переменных «Ошибки»

● **Debug Mode** – true, так как у нас активирован режим отладки;

● **Error Line** – 801 строка;

● **Error Message** – сообщение об ошибке;

● **Error URL** – URL-адрес скрипта, в котором обнаружена ошибка.

В случае, если мы в JavaScript используем конструкцию

"**try..catch**", то Google Tag Manager не зафиксировывает ошибку. Конструкция выглядит следующим образом:

```
1 try {  
2  
3     // код ...  
4  
5 } catch (err) {  
6  
7     // обработка ошибки  
8  
9 }
```

Рис. 222. Конструкция "try..catch"

Работает она так:

1. Выполняется код внутри блока **try**.
2. Если в нем ошибок нет, то блок **catch(err)** игнорируется, то есть выполнение доходит до конца **try** и потом прыгает через **catch**.
3. Если в нем возникнет ошибка, то выполнение **try** на ней прерывается, и управление прыгает в начало блока **catch(err)**.

При этом переменная **err** (можно выбрать и другое название) будет содержать объект ошибки с подробной информацией о произошедшем. Таким образом, при ошибке в **try**

скрипт не «падает», и мы получаем возможность обработать ошибку внутри **catch**.

Подробнее про `try..catch` читайте в справочниках по JavaScript или в этой статье.

# Клики

<input checked="" type="checkbox"/>	Click Element
<input checked="" type="checkbox"/>	Click Classes
<input checked="" type="checkbox"/>	Click ID
<input checked="" type="checkbox"/>	Click Target
<input checked="" type="checkbox"/>	Click URL
<input checked="" type="checkbox"/>	Click Text

Рис. 223. Встроенные переменные «Клики»

Переменные в этой категории используются для отслеживания кликов по элементам сайта. Разделить их можно на две составляющие:

- переменные, которые возвращают значения при клике по всем элементам (**gtm.click**);

- переменные, которые возвращают значения при клике только по ссылкам (**gtm.linkClick**).

**Click Element** – возвращает значение для того HTML-элемента, на который кликнул пользователь и который был объектом действия пользовательского события (Event – **gtm.click** или **gtm.linkClick**). Этот объект извлекается из ключа **gtm.element** в **dataLayer**.



**Click Classes** – возвращает строку, содержащуюся в ключе `gtm.elementClasses` значении атрибута `className` пользовательского события по которому был выполнен клик. Аналог `{{Click Element}}.className`

**Click ID** – возвращает строку, содержащуюся в ключе `gtm.elementId` значении атрибута `ID` пользовательского события по которому был выполнен клик. Аналог свойства `{{Click Element}}.id`

**Click Target** – возвращает строку, содержащуюся в ключе `gtm.elementTarget` значении атрибута `target` пользовательского события по которому был выполнен клик. Аналог свойства `{{Click Element}}.target`

**Click URL** – возвращает строку, содержащуюся в ключе `gtm.elementUrl` значении атрибута `href` пользовательского события по которому был выполнен клик. Аналог свойства `{{Click Element}}.href` для ссылок, и `{{Click Element}}.action` для форм.

**Click Text** – возвращает строку, содержащуюся в ключе `gtm.elementText` значении атрибута `textContent / innerText` пользовательского события по которому был выполнен клик. Аналог свойства `{{Click Element}}.innerText`

Для прослушивания кликов в Google Tag Manager используются встроенные триггеры из раздела «Клики».

# Клик



Все элементы



Только ссылки

Рис. 224. Клики – Все элементы или Только ссылки

При активации триггера, запуске отладчика в Google Tag Manager и клике по любому элементу своего сайта, вы можете наблюдать такую картину (на вкладке Variables):

The screenshot shows the Google Tag Manager interface with the 'Variables' tab selected. A table lists several variables related to a click event. The 'gtm.linkClick' variable is highlighted in the left sidebar. The table contains the following data:

Variable	Variable Type	Return Type	Value
_event	Пользовательское событие	string	'gtm.linkClick'
Click Classes	Переменная уровня данных	string	'button hp-button-1 slide-button w-button green'
Click Element	Переменная уровня данных	object	https://graphanalytics.ru/Google_Analytics_dlya_googlyat_2018.pdf
Click ID	Переменная уровня данных	string	'
Click Target	Переменная уровня данных	string	'
Click Text	Переменная автоматического	string	'СКАЧАТЬ'
Click URL	Переменная уровня данных	string	https://graphanalytics.ru/Google_Analytics_dlya_googlyat_2018.pdf

Рис. 225. Пример встроенных переменных «Клики»

Данный клик был осуществлен по кнопке «СКАЧАТЬ», которая имеет ссылку для скачивания электронной книги на



Яков Осипенков

GRAPH ANALYTICS

Google Analytics

для GOOGLЯТ

Практическое руководство по веб-аналитике

Версия v0.1.2018

Google Analytics

## GOOGLE ANALYTICS ДЛЯ GOOGLЯТ

Эта электронная книга отлично подойдет как для начинающих интернет-маркетологов и аналитиков, так и для опытных пользователей Google Analytics, имеющих глубокое понимание об инструментах веб-анализа. В ней подробно рассматриваются все тонкости работы с продуктом Google, описываются пошаговые алгоритмы настройки, разбираются основные функции и обновленный в 2017 году интерфейс Analytics, даются рекомендации по построению отчетов с практическими примерами их использования.

СКАЧАТЬ

Рис. 226. Клик по кнопке «Скачать»

Открыв консоль разработчика (в Google Chrome F12) и подсветив нужный фрагмент кода, мы увидим, что у данного элемента присутствуют всего два атрибута – *href* и *class*.

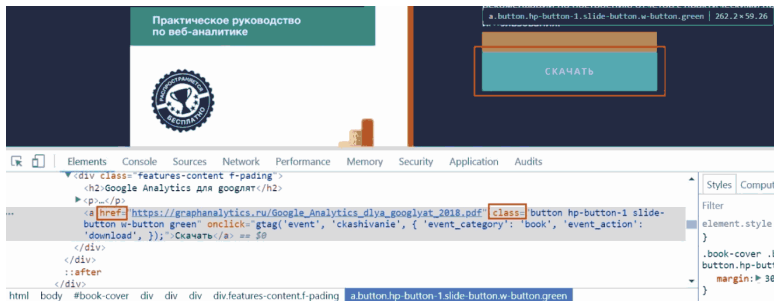


Рис. 227. Атрибуты href и class

Переменные в данном конкретном примере вернули значения:

- **\_event (пользовательское событие)** – gtm.linkClick, так как клик был осуществлен по ссылке;

- **Click Element** – [https://graphanalytics.ru/Google\\_Analytics\\_dlya\\_googlyat\\_2018.pdf](https://graphanalytics.ru/Google_Analytics_dlya_googlyat_2018.pdf)

- **Click Classes** – button hp-button-1 slide-button w-button green (класс кнопки)

- **Click ID** – пустое значение, нет данных по атрибуту id у этого элемента

- **Click Target** – пустое значение, нет данных по атрибуту target у этого элемента

- **Click URL** – [https://graphanalytics.ru/Google\\_Analytics\\_dlya\\_googlyat\\_2018.pdf](https://graphanalytics.ru/Google_Analytics_dlya_googlyat_2018.pdf)

- **Click Text** – СКАЧАТЬ

На вкладке Data Layer можно увидеть, как переменные

получают доступ к уровню данных и считывают ключи, которые задаются триггерами «**Все элементы**» и «**Только ссылки**».

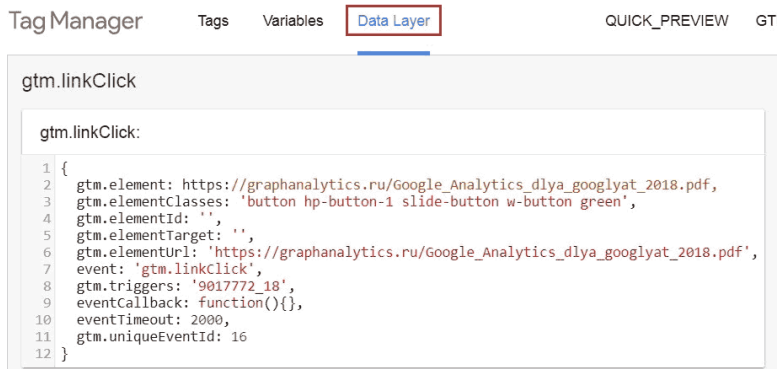


Рис. 228. Вкладка Data Layer в режиме отладки GTM

Аналогичным образом можно посмотреть значения передаваемых переменных по пользовательскому событию gtm.click.

## Формы

Схожа с категорией «**Клики**», только в качестве отслеживаемых объектов используются формы.

## Формы

Form Element

---

Form Classes

---

Form ID

---

Form Target

---

Form URL

---

Form Text

---

Рис. 229. Встроенные переменные «Формы»

Те же самые переменные, только вместо Click -> Form, а в качестве триггера прослушивания используется «**Отправка формы**», а пользовательское событие называется **gtm.formSubmit**.



Видео YouTube



Глубина прокрутки



Доступность элемента



Отправка формы

Рис. 230. Тип триггера – Отправка формы

**Form Element** – возвращает JS-объект для формы, которая была отправлена пользователем. Например, атрибуты class, ID, данные о родительских и дочерних элементах. Этот объект также извлекается из ключа gtm.element в dataLayer.

**Form Classes** – возвращает набор атрибутов class для отправленной формы. Содержится в ключе gtm.elementClasses.

**Form ID** – возвращает значение атрибута id для отправленной формы. Содержится в ключе gtm.elementId.

**Form Target** – возвращает значение атрибута target. Содержится в ключе gtm.elementTarget

**Form URL** – возвращает значение атрибута action для отправленной формы. Содержится в ключе gtm.elementUrl.

Данная переменная полезна, когда формам не присвоены

атрибуты `id` и `class`, и они отсылаются разным обработчикам на сервере. Например, форма обратной связи может отсылать данные на `/contacts.php`, а форма новостной подписки на `/subscribe.php`.

**Form Text** – переменная возвращает текст, содержащийся в отправленной форме и ее потомках. Содержится в ключе `gtm.elementText`.

В качестве примера отправим форму на тестовом сайте `graphanalytics.ru`

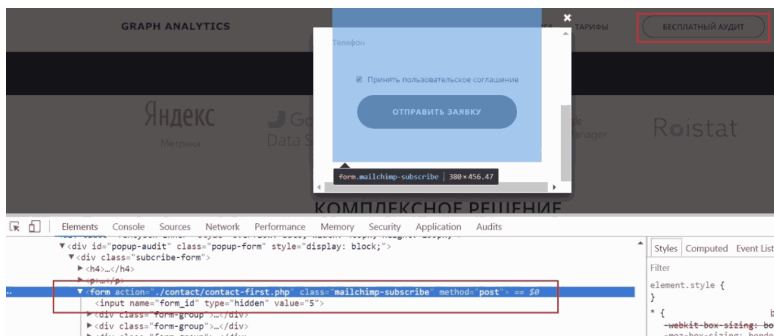


Рис. 231. Пример отправки формы

После отправки формы отладчик Google Tag Manager зафиксировал событие **gtm.formSubmit**, при переходе в которое (вкладка `Variables`) нам станут доступны значения переменных из этой категории:



The screenshot shows the Google Tag Manager interface with the 'Variables' tab selected. A table lists several built-in form variables. The table has four columns: Name, Type, Data Type, and Value. The variables listed are Form Classes, Form Element, Form ID, Form Target, Form Text, and Form URL. The 'Form Element' variable is highlighted with a blue selection box, and its value is '[object HTMLFormElement]'. The 'Form Text' variable has a value of 'Принять пользовательское соглашение ОТПРАВИТЬ ЗАЯВКУ'. The 'Form URL' variable has a value of 'https://graphanalytics.ru/contact/contact-first.php'.

Variable Name	Type	Data Type	Value
Form Classes	Переменная уровня данных	string	'mailchimp-subscribe'
Form Element	Переменная уровня данных	object	[object HTMLFormElement]
Form ID	Переменная уровня данных	string	''
Form Target	Переменная уровня данных	string	''
Form Text	Переменная автоматического уровня данных	string	'Принять пользовательское соглашение ОТПРАВИТЬ ЗАЯВКУ'
Form URL	Переменная уровня данных	string	'https://graphanalytics.ru/contact/contact-first.php'

Рис. 232. Пример встроенных переменных «Формы»

● **Form Element** – [object HTMLFormElement],

Форму представляет объект HTMLFormElement, созданный на основе HTMLElement и наследующий его свойства, методы и события.

● **Form Classes** – mailchimp-subscribe

● **Form ID** – пустое значение, нет данных по атрибуту id у этого элемента

● **Form Target** – пустое значение, нет данных по атрибуту target у этого элемента

● **Form Text** – Принять пользовательское соглашение ОТПРАВИТЬ ЗАЯВКУ (весь собранный текст из данной формы)

● **Form URL** – https://graphanalytics.ru/contact/contact-first.php (обработчик, на который отправляются данные из формы)

На вкладке Data Layer можно увидеть, как переменные получают доступ к уровню данных и считывает ключи, которые задаются триггерами «Отправка формы».

gtm.formSubmit:

```
1 {
2   gtm.element: [object HTMLFormElement],
3   gtm.elementClasses: 'mailchimp-subscribe',
4   gtm.elementId: '',
5   gtm.elementTarget: '',
6   gtm.elementUrl: 'https://graphanalytics.ru/contact/contact-first.php',
7   event: 'gtm.formSubmit',
8   gtm.triggers: '9017772_19',
9   eventCallback: function(){},
10  eventTimeout: 2000,
11  gtm.uniqueEventId: 21
12 }
```

Рис. 233. gtm.formSubmit в Data Layer

# История

## История

New History Fragment

---

Old History Fragment

---

New History State

---

Old History State

---

History Source

---

Рис. 234. Встроенные переменные «История»

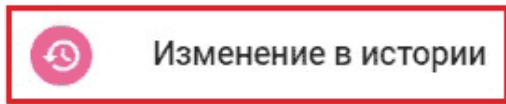
История – это место, где хранится информация о том, по каким ссылкам в открытой вкладке браузера переходил посетитель. Они фиксируются, начиная с первой и заканчивая текущей. С помощью JavaScript мы можем перемещаться по данной истории и возвращаться к определенному элементу, добавлять новый элемент и изменять содержимое текущего.

Подробную информацию о добавлении и изменении записей истории можно получить в документации разработчика на сайте [developer.mozilla.org](http://developer.mozilla.org).

Данная категория позволяет организовать навигацию или смоделировать поведение на сайтах, на которых контент подгружается динамически, без перезагрузки страниц (как правило, с помощью AJAX). При этом в адресной строке может меняться хэш (/#contacts, /#price, /#otzivi и т.д.).

Встроенный триггер в Google Tag Manager, который используется для прослушивания изменений в истории, называется **«Изменение в истории»** (событие `gtm.HistoryChange`).

Другое



Ошибка JavaScript

Рис. 235. Триггер – Изменение в истории

Категория **«История»** в GTM состоит из 5 встроенных переменных:

**New History Fragment** – возвращает переменную, которая содержит новое значение хэша (#) URL-сайта после совершения пользовательского события «Изменение в исто-

рии». Содержится в ключе `gtm.newUrlFragment`.

**Old History Fragment** – обратное действие, возвращает переменную, которая содержит предыдущее значение хэша URL-сайта до совершения пользовательского события. Содержится в ключе `gtm.oldUrlFragment`.

**New History State** – возвращает объект, содержащий новое состояние истории после того, как произошло событие и метод `history.pushState()` был выполнен. Содержится в ключе `gtm.newHistoryState`.

**Old History State** – возвращает объект, содержащий старое состояние истории перед тем, как произошло событие и метод `history.pushState()` был выполнен. Содержится в ключе `gtm.oldHistoryState`.

**History Source** – возвращает строку-событие, которое привело к изменению объекта истории. Например, `pushState` или `popstate`.

Подробнее о методе `pushState` читайте здесь.

Разберем на примере. Для демонстрации функциональности будем использовать меню тестового сайта.



Оставить заявку    Преимущества    Преимущества интернет-рекламы    Стоимость    Контакты

Рис. 236. Пример встроенных переменных «История»

Предположим, мой начальный путь был с ссылки раздела «Стоимость». Ссылка имеет вид `site.ru/#4`, где #4 – хэш, так называемый «якорь», разметка блоков на веб-страницах.

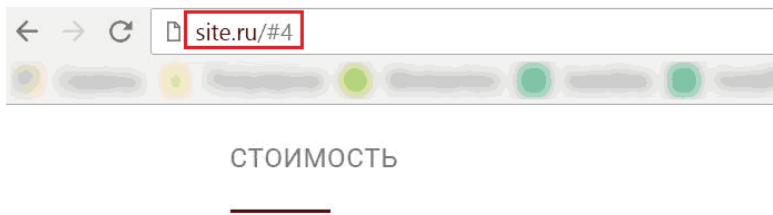


Рис. 237. Ссылка с якорем (хэшем)

Далее я перехожу на раздел «Оставить заявку», ссылка которого имеет вид `site.ru/#1`. Как видим, произошла подмена значения в ссылке (адресной строке браузера) без перезагрузки страницы.

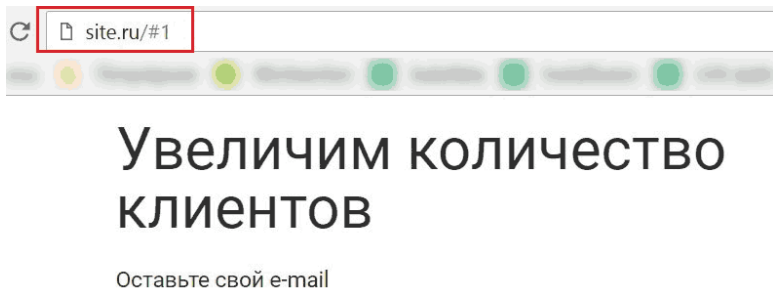
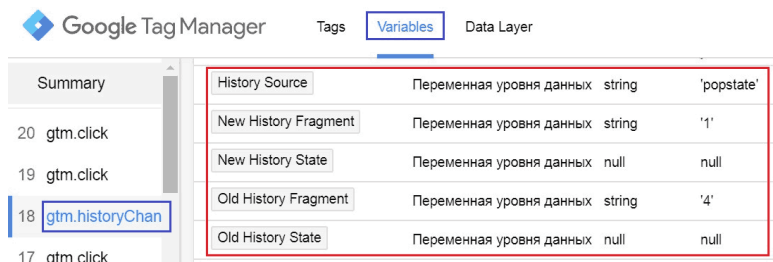


Рис. 238. Подмена значения в ссылке без перезагрузки

Google Tag Manager зафиксировал событие, связанное с изменением истории.



Summary	Tags	Variables	Data Layer
20 gtm.click		History Source	Переменная уровня данных string 'popstate'
19 gtm.click		New History Fragment	Переменная уровня данных string '1'
18 gtm.historyChan		New History State	Переменная уровня данных null null
17 atm click		Old History Fragment	Переменная уровня данных string '4'
		Old History State	Переменная уровня данных null null

Рис. 239. Пример встроенных переменных «История»

Переменные принимают соответствующие значения:

- **History Source** – popstate, поскольку всякий раз, когда пользователь переходит к новому состоянию, происходит событие popstate, и свойство события state содержит копию объекта записи истории.

Данное значение будет и тогда, когда мы используем переходы по кнопкам «Вперед» – «Назад» в браузере.



## Рис. 240. Кнопки «Вперед-Назад» в браузере

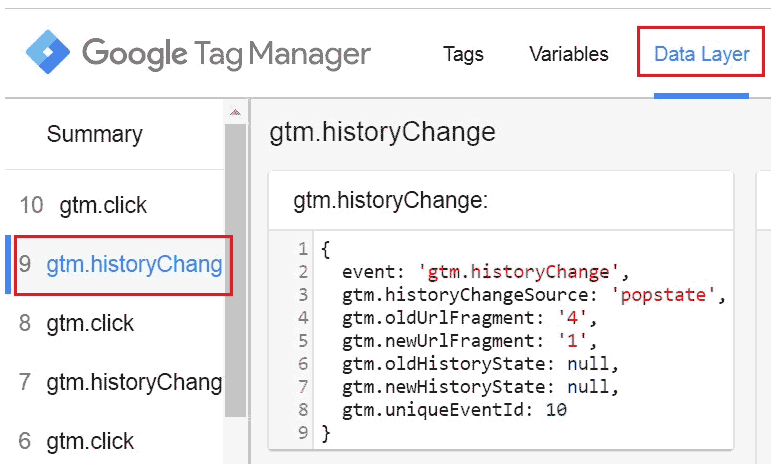
К тому же, браузеры работают с событием `popstate` по-разному. Chrome и Safari всегда вызывают `popstate` по окончании загрузки страницы, а Firefox не делает этого.

- **New History Fragment** – 1, так как мы осуществили переход `site.ru/#4` на `site.ru/#1`

- **Old History Fragment** – 4

- **New History State** и **Old History State** – null, пустое значение.

На вкладке **Data Layer** можно увидеть, как переменные получают доступ к уровню данных и считывает ключи, которые задаются триггерами «**Изменение в истории**».



The screenshot shows the Google Tag Manager interface. At the top, there are navigation tabs: "Tags", "Variables", and "Data Layer" (which is highlighted with a red box). Below the tabs, there is a list of tags on the left and a configuration editor on the right. The tag list includes "gtm.click" and "gtm.historyChange", with the latter highlighted by a red box. The configuration editor shows the following JSON code for the `gtm.historyChange` event:

```
gtm.historyChange:
1 {
2   event: 'gtm.historyChange',
3   gtm.historyChangeSource: 'popstate',
4   gtm.oldUrlFragment: '4',
5   gtm.newUrlFragment: '1',
6   gtm.oldHistoryState: null,
7   gtm.newHistoryState: null,
8   gtm.uniqueEventId: 10
9 }
```



Рис. 241. gtm.historyCHang в Data Layer

# Видео

## Видео

Video Provider

---

Video Status

---

Video URL

---

Video Title

---

Video Duration

---

Video Current Time

---

Video Percent

---

Video Visible

---

Рис. 242. Встроенные переменные «Видео»

Категория переменных, появившаяся в GTM в сентябре 2017 года. Они необходимы для отслеживания встроенных на сайт видео. Триггер в Google Tag Manager, который используется для прослушивания, называется «**Видео YouTube**» (событие **gtm.video**).



Видео YouTube

Рис. 243. Триггер – Видео YouTube

Google Tag Manager добавил 8 переменных в эту категорию:

**Video Provider** – возвращает значение поставщика видео. Сейчас доступен только YouTube. Содержится в ключе `gtm.videoProvider`.

**Video Status** – возвращает состояние видео в момент регистрации пользовательского события. Например, «Start» или «Pause» видео. Содержится в ключе `gtm.videoStatus`.

**Video URL** – URL-адрес, ссылка видео на YouTube. Содержится в ключе `gtm.videoUrl`.

**Video Title** – название видео. Содержится в ключе `gtm.videoTitle`.

**Video Duration** – общая продолжительность видео в секундах, выраженная целым числом. Содержится в ключе `gtm.videoDuration`.



**Video Current Time** – возвращает текущее время видео в секундах, в которое произошло пользовательское событие. Содержится в ключе `gtm.videoCurrentTime`.

**Video Percent** – возвращает значение воспроизведенного видео, выраженное в процентах целым числом (от 0 до 100), на момент, когда состоялось пользовательское событие. Содержится в ключе `gtm.VideoPercent`.


**Video Visible** – возвращает значение видимости видео в окне браузера. Если видео отображается в области просмотра, результатом будет значение `true`, если же в другой области (например, в нижней части страницы, на фоновой вкладке) – `false`. Содержится в ключе `gtm.videoVisible`.

Создадим триггер «**Видео YouTube**» и зададим ему соответствующие настройки:

Тип триггера

 Видео YouTube 

Что регистрируется

- Начало
- Завершение
- Приостановка, перемотка, буферизация
- Ход просмотра 

Процентные пороги

10, 20, 30, 40, 50, 60, 70, 80, 90



процент

Временные пороги

Дополнительно

- Включить поддержку JavaScript API для всех видео YouTube 

Условия активации триггера

- Все видео
- Некоторые видео

Рис. 244. Настройки триггера Видео YouTube

- **Начало (Start)** – пользователь начинает просмотр видео;
- **Завершение (Complete)** – пользователь достигает

конца видео;

● **Приостановка (пауза), перемотка, буферизация (Pause, Seeking, Buffering)** – пользователь останавливает, перематывает видео или когда происходит буферизация;

● **Ход просмотра (Progress)** – пользователь проходит процентный или временный порог (время измеряется в секундах). Целые положительные числа указываются через запятую.

● **«Включить поддержку JavaScript API для всех видео YouTube»**

Установив этот флажок, вы включите YouTube iFrame Player API. В результате ко всем URL видеопроигрывателя YouTube будет добавлен параметр *enablejsapi = 1* для управления проигрывателем через iframe или JavaScript.

**Примечание:** если при загрузке Google Tag Manager воспроизведение ролика уже началось, оно начнется сначала.

Сохраняем изменения. Обновляем режим предварительного просмотра и переходим в отладчик Google Tag Manager. На сайте перематываем видео чуть вперед и приостановим его на кадре:



Рис. 245. Пример видео с YouTube

В отладчике будет доступна информация по всем 8 переменным:

Summary	Variable Name	Description	Type	Value
20 gtm.video	Video Current Time	Переменная уровня данных	number	715
19 gtm.video	Video Duration	Переменная уровня данных	number	3606
18 gtm.video	Video Percent	Переменная уровня данных	number	20
17 gtm.video	Video Provider	Переменная уровня данных	string	'youtube'
16 gtm.video	Video Status	Переменная уровня данных	string	'pause'
15 gtm.video	Video Title	Переменная уровня данных	string	'Спектакль "Рикки"'
	Video URL	Переменная уровня данных	string	'https://www.youtube.com/watch?v=D80nY3FyE'
	Video Visible	Переменная уровня данных	boolean	true

Рис. 246. Пример встроенных переменных «Видео»

● **Video Current Time** – 715 (в секундах), текущее время,

в которое произошло пользовательское событие gtm.video;

● **Video Duration** – 3606 (в секундах), общая продолжи-

тельность видео;

● **Video Percent** – 20 (%), процент, процент воспроиз-

веденного видео до момента совершения пользовательского события gtm.video;

● **Video Provider** – youtube, поставщик видео (пока един-

ственный);

● **Video Status** – pause, так как оно приостановлено;

● **Video Title** – Спектакль Рикки, название видеоролика;

● **Video URL** – <https://www.youtube.com/watch?v=D80nY3FyE>

t=1118&v=u-wkJTzEdns, ссылка на видео на YouTube;

● **Video Visible** – значение true, так как видео отобража-

лось в видимой области экрана браузера.

На вкладке Data Layer можно увидеть, как переменные по-



лучают доступ к уровню данных и считывает ключи, которые задаются триггерами «**Видео YouTube**».

anager    Tags    Variables    **Data Layer**

```
gtm.video:
1 {
2   gtm.element: [object HTMLIFrameElement],
3   gtm.elementClasses: '',
4   gtm.elementId: '295582718',
5   gtm.elementTarget: '',
6   gtm.elementUrl: 'https://www.youtube.com/embed/D80nlY3fFyE?rel=0&showinfo=0&enablejsapi=1&origin=h
7
8   event: 'gtm.video',
9   gtm.videoProvider: 'youtube',
10  gtm.videoStatus: 'pause',
11  gtm.videoUrl: 'https://www.youtube.com/watch?v=D80nlY3fFyE',
12  gtm.videoTitle: 'Снектакль "Рикки"',
13  gtm.videoDuration: 3606,
14  gtm.videoCurrentTime: 715,
15  gtm.videoElapsedTime: 3,
16  gtm.videoPercent: 20,
17  gtm.videoVisible: true,
18  gtm.triggers: '8934827_12',
19  gtm.uniqueEventId: 20
20 }
```

Рис. 247. gtm.video в Data Layer

# Прокрутка

## Прокрутка

Scroll Depth Threshold

---

Scroll Depth Units

---

Scroll Direction

---

Рис. 248. Встроенные переменные «Прокрутка»

В октябре 2017 года в Google Tag Manager появился новый встроенный триггер – **Scroll Depth**, а с ним и три встроенных переменных. Триггер «**Глубина прокрутки**» включает базовые опции и позволяет отслеживать как вертикальную глубину скроллингу, так и горизонтальную.



Видео YouTube



Глубина прокрутки

Рис. 249. Триггер – Глубина прокрутки

Три переменных в GTM:


**Scroll Depth Threshold** – возвращает значение порога прокрутки, который был пересечен в результате пользовательского события `gtm.scrollDepth`. Содержится в ключе `gtm.scrollThreshold`.

**Scroll Depth Units** – переменная возвращает значение в пикселях (`pixels`) или процентах (`percent`), в зависимости от настроек триггера. Содержится в ключе `gtm.scrollUnits`.

**Scroll Direction** – переменная возвращает значение направления прокрутки (`vertical` или `horizontal`), в зависимости от настроек. Содержится в ключе `gtm.scrollDirection`.

Создадим триггер «**Глубина прокрутки**» и зададим ему соответствующие настройки (пример):

Тип триггера

 Глубина прокрутки 

Глубина вертикальной прокрутки 

Проценты


 %

Пиксели

Глубина горизонтальной прокрутки 

Проценты

Пиксели

 пикс.

Условия активации триггера

Все страницы  Некоторые страницы

Рис. 250. Настройки триггера «Глубина прокрутки»

● Глубина вертикальной прокрутки в процентах –

10, 25, 50, 75, 90;

- **Глубина горизонтальной прокрутки в пикселях** – 100;

- **Условия активации триггера** – Все страницы.

Сохраняем изменения. Обновляем режим предварительного просмотра и переходим в отладчик Google Tag Manager. На сайте проскроллим чуть вниз страницы и увидим пользовательское событие `gtm.scrollDepth`:

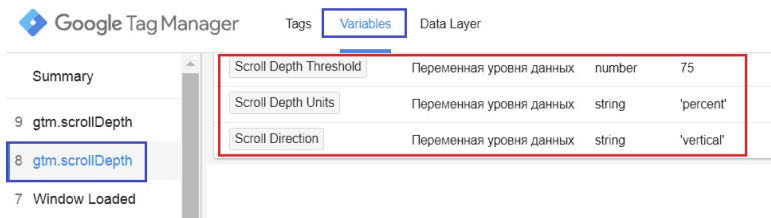


Рис. 251. Пример встроенных переменных «Прокрутка»

- **Scroll Depth Threshold** – 75% (процент прокрутки), поскольку вертикальная прокрутка у нас была задана в %;

- **Scroll Depth Units** – percent, здесь значение может быть «вертикальный» или «горизонтальный», в зависимости от типа прокрутки, который пересек порог;

- **Scroll Direction** – vertical, тип/направление прокрутки.

А вот так выглядит горизонтальный скроллинг в пикселях (в нашем примере):

Scroll Depth Threshold	Переменная уровня данных	number	100
Scroll Depth Units	Переменная уровня данных	string	'pixels'
Scroll Direction	Переменная уровня данных	string	'horizontal'

Рис. 252. Пример горизонтальной прокрутки

- **Scroll Depth Threshold** – 100;
- **Scroll Depth Units** – pixels;
- **Scroll Direction** – horizontal.

На вкладке Data Layer можно увидеть, как переменные получают доступ к уровню данных и считывает ключи, которые задаются триггерами «Глубина прокрутки».

The image shows a screenshot of the Data Layer interface. On the left, there is a 'Summary' list with several entries, including 'gtm.scrollDepth' at index 8, which is highlighted in blue. On the right, the configuration for 'gtm.scrollDepth' is displayed in a code editor. The configuration is as follows:

```
gtm.scrollDepth:
1 {
2   event: 'gtm.scrollDepth',
3   gtm.scrollThreshold: 75,
4   gtm.scrollUnits: 'percent',
5   gtm.scrollDirection: 'vertical',
6   gtm.triggers: '8934827_13',
7   gtm.uniqueEventId: 8
8 }
```

Рис. 253. gtm.scrollDepth в Data Layer

Отслеживать прокрутку без привязки к элементам сложно. 50% глубины прокрутки для статьи длиной 1000 сим-

волов и 50% прокрутки для статьи 20000 символов не дадут никаких данных для анализа. Также триггер не позволяет отслеживать прокрутку до конкретных HTML-элементов. В таких случаях используют другой триггер, который называется **«Видимость элемента»**.

# Видимость

## Видимость



Percent Visible

---



On-Screen Duration

Рис. 254. Встроенные переменные «Видимость»

Переменные из этой категории помогают отслеживать данные о том, где посетитель остановился и сколько времени потратил на просмотр элементов на сайте – заголовков, текстовых абзацев, изображений и т.д. Функция появилась не так давно, но является очень полезной, так как теперь есть возможность точнее получить представление о том, как пользователи взаимодействуют с контентом сайта и какие элементы остаются без внимания.

Встроенный триггер – «Доступность элемента», пользовательское событие **gtm.elementVisibility**.



## Взаимодействия пользователей

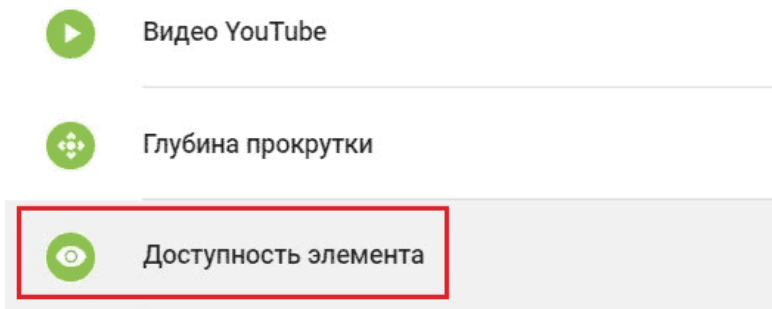


Рис. 255. Триггер – Доступность элемента

В Google Tag Manager в категории «Видимость» две переменных:

**Percent Visible** – возвращает числовое значение от 0 до 100, которое показывает, какой процент выбранного элемента был виден при срабатывании триггера. Содержится в ключе `gtm.visibleRatio`.

**On-Screen Duration** – возвращает числовое значение, которое показывает, как долго выбранный элемент был виден при срабатывании триггера. Содержится в ключе `gtm.visibleTime`.

Создадим триггер «Доступность элемента» и зададим ему соответствующие настройки (пример):

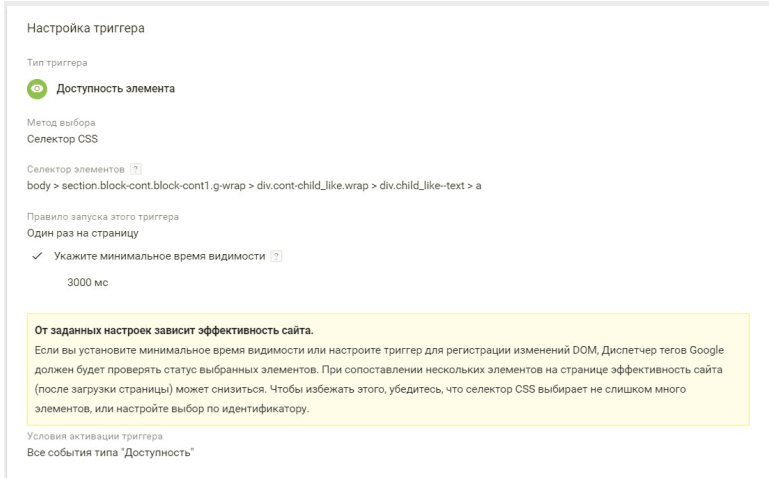


Рис. 256. Пример настройки триггера «Доступность элемента»

Какие-либо настройки триггеров в этой статье целенаправленно опускаются, поскольку более подробно о триггерах будет говориться в далее. Стоит отметить, что в данном примере был выбран элемент (кнопка «*Записаться на бесплатное занятие*»), при прокрутке которого активировалось пользовательское событие `gtm.elementVisibility`.

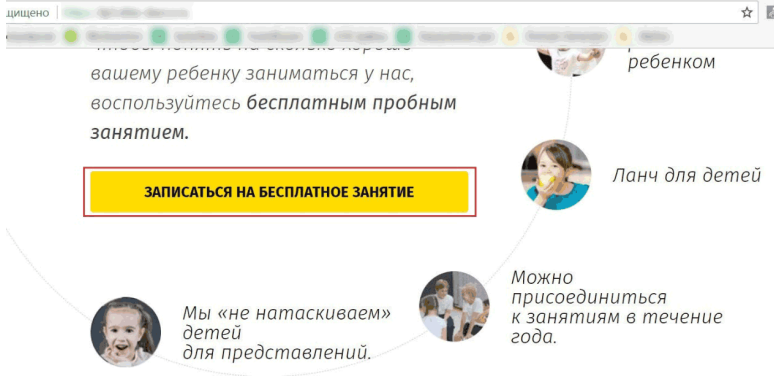


Рис. 257. Кнопка «Записаться на бесплатное занятие» как условие активации события

- **Метод выбора** – Селектор CSS;

- **Селектор элементов** – `body > section.block-cont.block-cont1.g-wrap > div.cont-child_like.wrap > div.child_like – text > a ;`

- **Правило запуска этого триггера** – Один раз на страницу;

- **Минимальное время видимости (в миллисекундах)** – 3000 (как долго выбранный элемент должен быть виден на экране для срабатывания триггера);

- **Условия активации триггера** – Все события типа «Доступность».

Сохраняем изменения. Обновляем режим предварительного просмотра и переходим в отладчик Google Tag Manager.

На сайте мы проскроллили до данной кнопки и через минимальное время видимости элемента произошла фиксация события `gtm.elementVisibility`.

Summary				
7	gtm.scrollDepth			
6	gtm.elementVisibi			

On-Screen Duration	Переменная уровня данных	number	3000
Percent Visible	Переменная уровня данных	number	100

Рис. 258. Пример встроенных переменных «Видимость»

- **On-Screen Duration** – 3000 (в миллисекундах), как долго выбранный элемент был виден при срабатывании триггера;

- **Percent Visible** – 100 (в процентах), видимость элемента. В нашем примере он был виден на все 100%.

На вкладке `Data Layer` можно увидеть, как переменные получают доступ к уровню данных и считывает ключи, которые задаются триггерами «Доступность элемента».



# Пользовательские переменные

Для создания переменной перейдем в «Переменные» и нажмем кнопку «Создать».

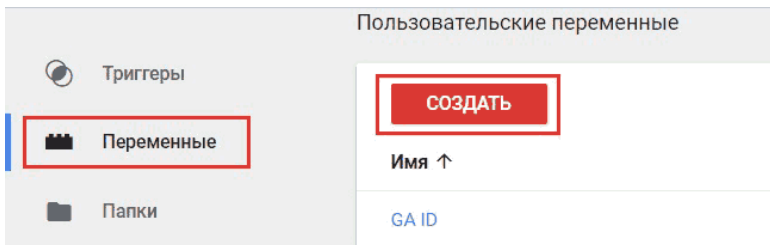


Рис. 260. Создание пользовательской переменной

При создании переменной необходимо указать ее имя. Далее нам доступно на выбор 5 категорий пользовательских переменных:

1. Навигация
2. Переменные страницы
3. Элементы страницы
4. Утилиты
5. Данные контейнера

# Навигация

## Навигация



**Источник ссылки HTTP**

Источник ссылки HTTP.

---



**URL**

---

Рис. 261. Пользовательские переменные «Навигация»

С помощью переменных этой категории можно отслеживать с какой страницы к вам пришел пользователь. Или если он перемещается внутри сайта, от страницы к странице, на каждой следующей странице доступна информация о предыдущей странице пользователя.

Доступно две пользовательских переменных:

**1. Источник ссылки HTTP** – значение извлекается из **document.referrer**. Аналог встроенной переменной (Referrer);

У данной переменной есть несколько типов компонентов: полный URL, протокол, имя хоста, порт, путь, запрос и фрагмент.

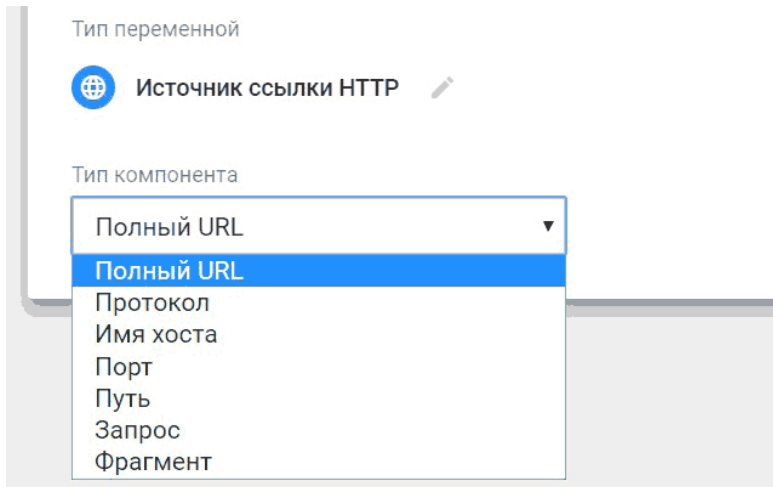


Рис. 262. Типы компонента «Источник ссылки HTTP»

Разберем все типы на примере:



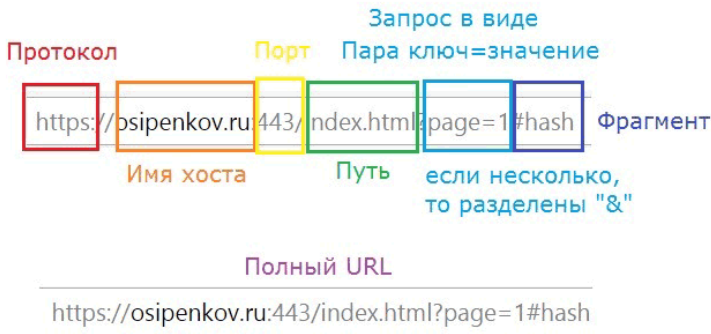


Рис. 263. Типы компонента на примере

**Полный URL-адрес** – возвращает полный URL-адрес без фрагмента хеширования (#). Например, `https://osipenkov.ru/index.html?page=1`

**Протокол** – возвращает протокол URL. Например, `https`

**Имя хоста** – возвращает имя хоста URL-адреса без номера порта. Например, `osipenkov.ru` или с `www`

**Порт** – возвращает номер порта, используемый в URL-адресе, или «80» для HTTP / или «443» для HTTPS, если URL-адрес не имеет номера порта. На примере выше число «443» вставлено лишь с целью демонстрации

**Путь** – возвращает только путь в URL. Например, `index.html`

**Запрос** – возвращает всю строку параметров запроса (без

указания «?») при условии, что вы не укажете ключ запроса. Если вы укажете ключ запроса, возвращается только значение этого ключа или не определено, если в URL-адресе такого ключа нет. Если помните настройку представления в Google Analytics под названием «**Отслеживание поиска по сайту**», то разобраться в типе компонента «запрос» не составит проблем

**Фрагмент** – возвращает значение фрагмента URL-адреса без ведущего «#». Например, hash

Вы можете создать триггер, который запускается, когда «Источник ссылки HTTP» не содержит ваш собственный домен. Таким образом можно отслеживать пользователей, перешедших на сайт извне, а не переходящих по страницам на сайте.

**2. URL** – универсальная переменная, которая может быть использована для доступа к компонентам текущей URL-страницы (по умолчанию) или любого URL-адреса, возвращаемой переменной. Аналог встроенных переменных Page URL, Page Hostname, Page Path.

Тип переменной



Тип компонента

Полный URL ▾

▼ Дополнительные настройки

Источник URL

URL страницы/по умолчанию ▾

Рис. 264. Пользовательская переменная URL

Доступны те же типы компонентов, что и у «**Источник ссылки HTTP**».

# Переменные страницы

## Переменные страницы



### Основной файл cookie

Первое значение из основного файла cookie с соответствующим названием.

---



### Переменная уровня данных

---



### Переменная JavaScript

Используется значение указанной глобальной переменной Javascript.

---



### Собственный код JavaScript

Значение этой переменной вычисляется в браузере с помощью заданной функции JavaScript. Функция вызывается при каждом использовании переменной, возвращая ее значение.

---

Рис. 265. Пользовательские переменные «Переменные страницы»

4 переменных:

● **основной файл cookie (1st Party Cookie)** – возвращает значение cookie, которое доступно для текущего сайта;

Тип переменной



Основной файл cookie



Название cookie



URI-декодирование файла cookie



Рис. 266. Основной файл cookie

В качестве названия cookie можно задать **\_ga** для различия пользователей. Таким образом при просмотре страницы сайта в отладчике Google Tag Manager будет доступно значение cookie пользователя:

Google Tag Manager    Tags    Variables    Data Layer

Summary	Variable Name	Value	Type	Value
	Old History State	Переменная уровня данных	undefined	undefined
	Page Hostname	URL	string	
	Page Path	URL	string	/'
	Page URL	URL	string	
	Random Number	Случайное число	number	1462667398
	Referrer	Источник ссылки HTTP	string	"
1 Page View	Основной файл cookie	Основной файл cookie	string	'GA1.2.1355654258.1526038404'

Рис. 267. Пример пользовательской переменной «Основной файл cookie»

**URI-декодирование файла cookie.** Приятной особенностью файлов cookie является то, что почти каждый браузер, поддерживающий JavaScript, обеспечивает и доступ сценариев к строкам cookie. Строки cookie представляются свойством cookie объекта Document. Это свойство доступно как для чтения, так и для записи. Когда вы присваиваете строку свойству document.cookie, браузер анализирует ее как строку файла cookie и добавляет ее в список строк cookie.

```
document.cookie = "username=yakov; expires=Friday, 01-Dec-2017 08:00:00 GMT; path=/home";
```

Довольно часто значения полей cookie шифруются, что может оказаться проблемой при вставке их в строку cookie. Данная функция выполняет URL-кодирование и декодирование строк, которые передаются им в качестве аргументов,

возвращая соответствующий результат.

Если cookie возвращает значение **undefined**, то она не определена. Подробнее о куки, document.cookie читайте в этой статье.

## ● Переменная уровня данных

Один из наиболее часто используемых типов в GTM. Когда вы используете уровень данных (Data Layer), вы передадите пару *key:value* с помощью конструкции:

```
dataLayer.push({'var': 'value'});
```

Тип переменной



Переменная уровня данных



Имя переменной уровня данных ?

key



Версия уровня данных

Версия 2



Установить значение по умолчанию

## Рис. 268. Переменная уровня данных

Чтобы данные стали доступны в Google Tag Manager создается пользовательская переменная типа **«Переменная уровня данных»**, и в поле **«Имя переменной уровня данных»** указывается key.

**Примечание:** переменные уровня данных назначаются для страниц, а не для сеансов.

Для простых значений (строки, числа, булевы, функции) переменная будет возвращать все, что было недавно передано в ключ. Для объектов и массивов переменная возвращает результат рекурсивного слияния, где заменяются только общие ключи.

Вы можете использовать точечную нотацию для доступа к переменным ключам Data Layer, которые имеют точку в имени (например, **gtm.element**) или для доступа к свойствам объектов DOM (например, **gtm.element.dataset.name**).



Переменная уровня данных



Имя переменной уровня данных ?

gtm.start





## Рис. 269. Пример точечной нотации

Если вы не указали значение по умолчанию, переменная `Data Layer` вернет **undefined**. При задании настроек переменной существует выбор версии:

- **Версия 1** – разрешает использовать точки в названии. Например, `dataLayer.push('k.e.y': 'value')` ключ будет интерпретироваться как `k.e.y` (т. е. `{'k.e.y': 'value'}`).

- **Версия 2** – интерпретирует точки, как иерархию. Например, `dataLayer.push({'k.e.y': 'value'})` будет интерпретировано три уровня: `{k: {e: {y: 'value'}}}`.

## ● Переменная JavaScript

Тип переменной



Переменная JavaScript

Имя глобальной переменной

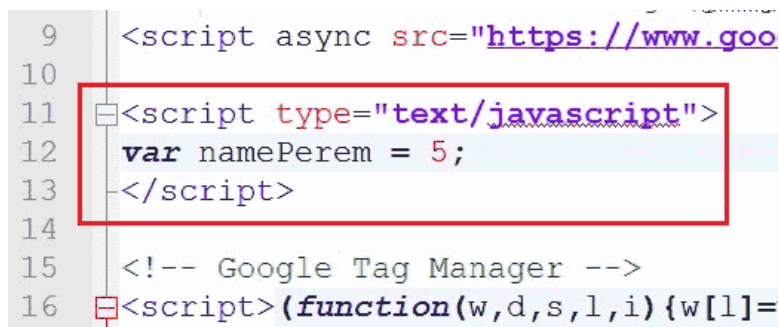
`namePeremen`

Рис. 270. Пользовательская переменная JavaScript

Принимает значение переменной JavaScript, имя которой указано в поле «Имя глобальной переменной». Если такой переменной не существует, вернется значение **undefined**.

Например, у вас на сайте в коде страницы объявлена глобальная переменная:

```
var namePeremen = 5
```

A screenshot of a code editor showing HTML code. Line 11 contains a script tag with type="text/javascript". Line 12 contains the JavaScript code var namePerem = 5;. Line 13 is the closing script tag. A red rectangular box highlights the entire script tag and its content. Other lines of code are visible but not highlighted.

```
9 <script async src="https://www.goo
10
11 <script type="text/javascript">
12   var namePerem = 5;
13 </script>
14
15 <!-- Google Tag Manager -->
16 <script>(function(w,d,s,l,i) {w[l]=
```

Рис. 271. Переменная namePeremen объявлена в коде сайта

В этом случае переменная JavaScript вернет значение глобальной переменной, что объявлена на странице, то есть 5 (и тип «number», «string» и т.д.):

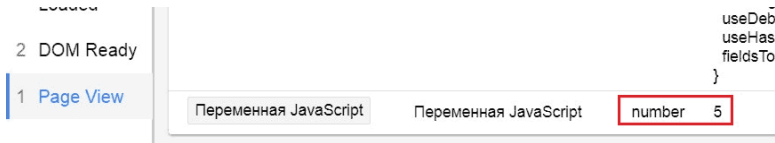


Рис. 272. Пример пользовательской переменной JavaScript

## ● Собственный код JavaScript

Пользовательские JavaScript переменная является наиболее универсальным переменной в наборе. Вы можете использовать ее для запуска произвольного JavaScript на странице. Он создает контекст сценария, то есть вы также можете вызывать другие переменные изнутри, используя соответствующий синтаксис.

Значение этой переменной вычисляется в браузере с помощью заданной функции JavaScript. Функция вызывается при каждом использовании переменной, возвращая ее значение.

Пользовательская переменная JavaScript должна следовать нескольким правилам:

1. скрипт должен быть размещен в функциональном блоке `function() { ... }`
2. функция должна иметь оператор **return**
3. функция должна возвращать только значение.

Если явного значения нет, функция возвращает неопределенное значение. Это может привести к нарушению нор-

мальной работы контейнера.

Тип переменной



## Собственный код JavaScript

Собственный код JavaScript ?

```
1 function() {  
2     var newPerem = 100;  
3     return newPerem;  
4 }  
5
```

Рис. 273. Пример собственного кода JavaScript

В отладчике Google Tag Manager:

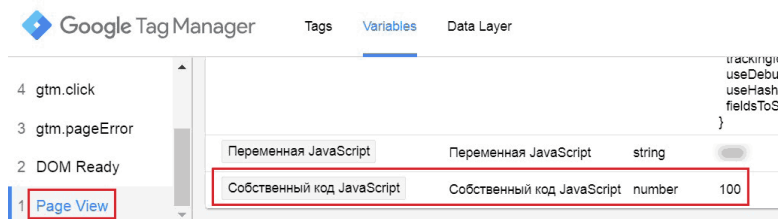
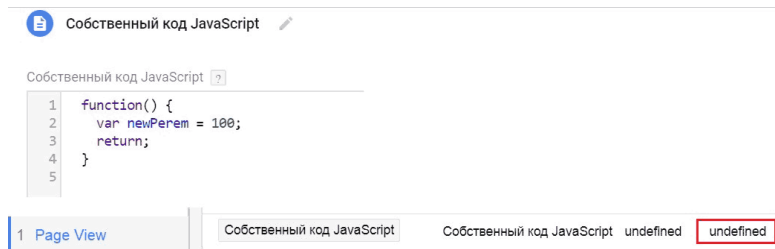


Рис. 274. Пример пользовательской переменной «Соб-

## Свой собственный код JavaScript»

Вы можете вернуть любую переменную или значение, даже другие функции, другие переменные GTM или ничего (return без возврата -> **undefined**, неопределенное значение).



The screenshot shows a code editor window titled "Свой собственный код JavaScript". The code inside is:

```
1 function() {  
2   var newPerem = 100;  
3   return;  
4 }  
5
```

Below the code editor, there are two tabs. The first tab is labeled "1 Page View". The second tab is labeled "Свой собственный код JavaScript" and shows the output "Свой собственный код JavaScript undefined". The word "undefined" is highlighted with a red rectangular border.

Рис. 275. Пример без возврата (неопределенное значение), return

# Элементы страницы

## Элементы страницы



### Видимость элемента

Значение зависит от видимости указанного элемента DOM.

---



### Переменная автоматического события

---



### Элемент DOM

Текст или заданный атрибут элемента DOM.

---

Рис. 276. Пользовательские переменные «Элементы страницы»

Следующая категория в Google Tag Manager состоит из 3 пользовательских переменных:

● **Видимость элемента** – позволяет вам определить какой конкретный элемент был виден в браузере пользователя. Используется с триггером «**Доступность элемента**».

Значение зависит от видимости указанного элемента DOM. В качестве примера зададим селектор элемента кнопки на сайте:

ЗАНЯТИЕМ.

a.btn.fancybox | 459.97 x 49.76

ЗАПИСАТЬСЯ НА БЕСПЛАТНОЕ ЗАНЯТИЕ

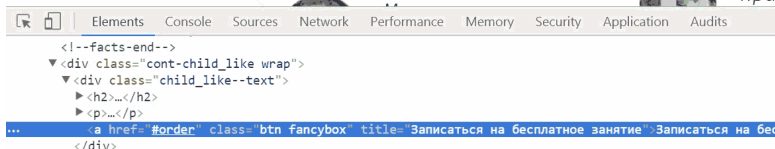


Рис. 277. Селектор элемента кнопки на сайте

Тип результата «истина/ложь» (true/false) и минимальный процент видимости:

Тип переменной



Видимость элемента



Метод выбора

Селектор CSS



Селектор элементов ?

body > section.block-cont.block-cc



Тип результата

True/False



Минимальный процент видимости ?

50



%

Рис. 278. Настройки в переменной «Видимость элемента»

В отладчике Google Tag Manager будет доступен следующую



щий результат:

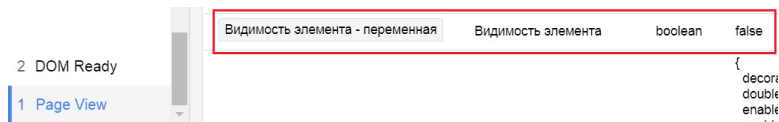



Рис. 279. Пример пользовательской переменной «Видимость элемента»

В случае видимости элемента в браузере при минимальном проценте видимости – *true*, в случае если элемент не был в поле зрения пользователя и не удовлетворяет проценту видимости – *false*.

### ● Переменная автоматического события (Auto-Event Variable)

Переменная автоматического события используется для доступа к целевому элементу действия пользовательского события (клики, ошибки, отправки формы и т.д.). Когда вы создаете новую переменную Auto-Event, вам нужно указать только тот компонент целевого элемента, к которому вы хотите получить доступ.

Тип переменной

<> Переменная автоматического события 

Тип переменной

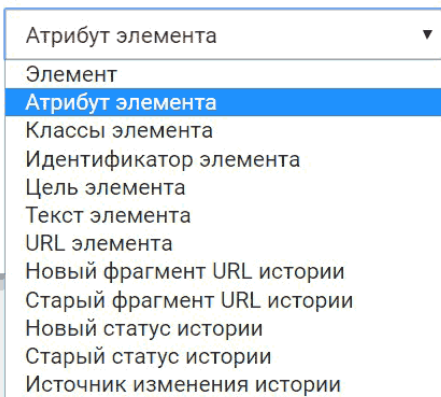


Рис. 280. Переменная автоматического события

Часть типов переменной совпадает со встроенными переменными (см. выше):

- **Элемент** = Click Element и Form Element;
- **Классы элемента** = Click Class и Form Class;
- **Идентификатор элемента** = Click ID и Form ID;
- **Цель элемента** = Click Target и Form Target;
- **Текст элемента** = Click Text и Form Text;
- **URL элемента** = Click URL и Form URL;
- **Новый фрагмент URL истории** = New History Fragment;

● **Старый фрагмент URL истории** = Old History Fragment;

● **Новый статус истории** = New History State;

● **Старый статус истории** = Old History State;

● **Источник изменения истории** = History Source.

Переменная автоматического события возвращает значение, соответствующее типу выбранного элемента. Если соответствующее автоматическое событие не было зарегистрировано, переменная возвращает значение по умолчанию (если установлено) или **undefined**.

## ● **Элемент DOM**

Вы можете использовать переменную для получения текстового содержимого любого элемента DOM или для извлечения значения любого атрибута элемента DOM.

Например, мы хотим получить значение текстового содержимого элемента кнопки «Заказать рекламу».

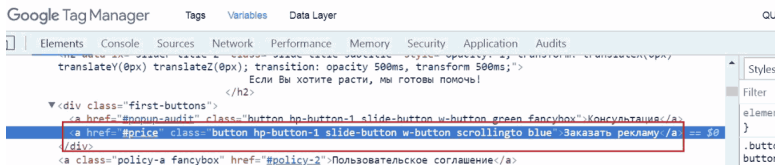
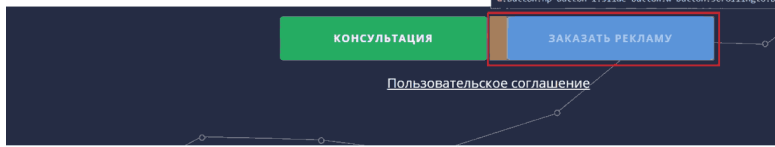


Рис. 281. Пример извлечения текстового содержимого элемента кнопки

Полный CSS-селектор у кнопки через Copy selector в браузере – `#home > div > div > div > div > div > a.button.hp-button-1.slide-button.w-button.scrollingto.blue`

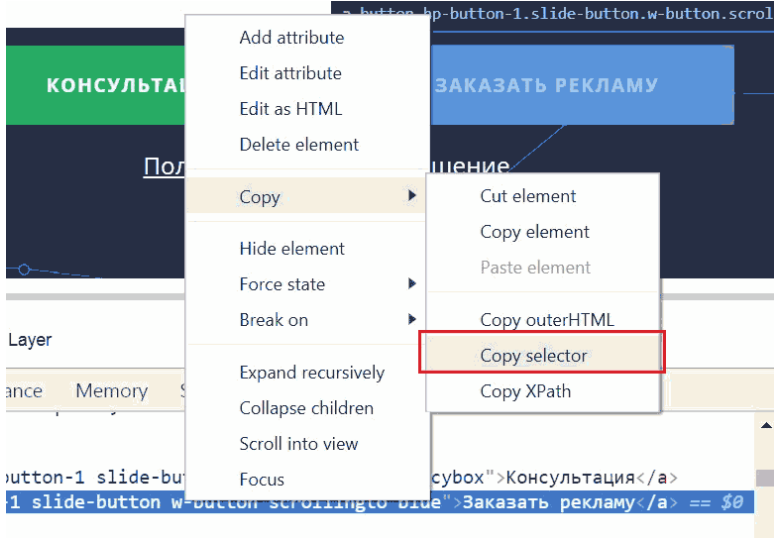


Рис. 282. Copy selector

Создадим пользовательскую переменную в Google Tag Manager:

Тип переменной



Элемент DOM

Метод выбора

Селектор CSS

Селектор элементов

#home > div > div > div > div > a.button.hp-button-1.slide-button.w-button.scrollingto.blue

Рис. 283. Пример пользовательской переменной «Эле-

**Примечание:** в данном примере CSS-селектор очень длинный и его можно сократить. Однако цель примера не в этом, так что оставим полный селектор, скопированный из консоли разработчика.

Если задано имя атрибута, переменной будет присвоено значение атрибута элемента DOM, в противном случае в качестве значения будет использоваться текст элемента DOM. В отладчике GTM:

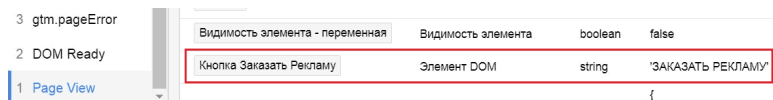


Рис. 284. Отображение значения кнопки в переменной «Элемент DOM»

Если элемент DOM с указанным идентификатором или CSS-селектором не найден, переменная возвращает нулевое значение (null).

# Утилиты

## Утилиты



### Константа

Указанная строка.



### Название среды

В качестве значения используется ссылка на название среды, через которую мог быть загружен контейнер (например, Live).



### Настройки Google Analytics

Эта переменная позволяет задавать настройки Google Analytics для использования с разными тегами.



### Пользовательское событие

Устанавливается значение eventNameXYZ, когда на веб-сайте выполняется следующий код:  
`dataLayer.push({'event': 'eventNameXYZ'});`



### Случайное число

Случайное число в диапазоне от 0 до 2 147 483 647 включительно.



### Таблица поиска



### Таблица регулярных выражений

Рис. 285. Пользовательские переменные «Утилиты»

**Константа.** Постоянная переменная является наглядным примером того, как переменные могут быть использованы повторно. Константа будет постоянно принимать значение из поля «Значение»:

Тип переменной



Константа



Значение

UA-113446186-1



Рис. 286. Переменная «Константа»

Чаще всего этот тип переменной используется для указания идентификатора отслеживания Google Analytics. Сохранив константу UA один раз, вам не нужно создавать его (идентификатор) каждый раз, когда вы создаете новый тег GA.

**Название среды.** Аналог встроенной переменной **Environment Name**, которая возвращает название пользовательской среды.



Тип переменной



Название среды



Конфигурация не требуется.

Рис. 287. Переменная «Название среды»

**Настройки Google Analytics.** Переменная возвращает набор параметров тега Universal Analytics. Используется **ТОЛЬКО** в теге Universal Analytics и для одновременной настройки нескольких тегов, например, для объединения своих пользовательских параметров (custom dimension) и полей.

Домен cookie ?


 

Рис. 288. cookieDomain “auto”

**Домен cookie.** По умолчанию стоит auto, также, как и при отслеживании в Google Analytics, для поля **cookieDomain** значение *auto*.

```
ga('create', 'UA-XXXXX-Y', 'auto');
```

Рис. 289. Код из Google Analytics

При автоматической *auto* конфигурации домена `cookieDomain` библиотека *analytics.js* самостоятельно подбирает домен для хранения файлов cookie.

Дополнительные настройки:

- Поля, которые необходимо задать – доступен раскрывающийся список с доступными полями;

▼ Поля, которые необходимо задать

Название поля	Значение
<input type="text"/>	<input type="text"/>
campaignContent	
campaignId	
campaignKeyword	
> campaignMedium	
campaignName	
> campaignSource	
clientId	
> cookieDomain	
cookieExpires	

Рис. 290. Поля, которые необходимо задать

- Специальные параметры;
- Специальные показатели;
- Группы контента;
- Реклама – Включить функции для контекстно-медий-

ной сети;

- Электронная торговля – Включить расширенные функции электронной торговли;
- Междоменное отслеживание;
- Расширенная конфигурация, в которой можно задать глобальную функцию, использовать отладочную версию и улучшенную атрибуцию ссылок.

## Пользовательское событие

Тип переменной



Пользовательское событие



Конфигурация не требуется.

Рис. 291. Пользовательское событие

Принимает значение, равное текущему значению переменной **\_event**. Аналог встроенной переменной Event (Утилиты).

Summary	gtm.click			
	Variable	Variable Type	Return Type	Value
7 gtm.click				
6 gtm.click	<code>_event</code>	Пользовательское событие	string	'gtm.click'
5 Window Loaded	Event	Пользовательское событие	string	'gtm.click'
4 gtm.pageError	Пользовательское событие	Пользовательское событие	string	'gtm.click'

Рис. 292. Пример пользовательского события gtm.click

## Случайное число

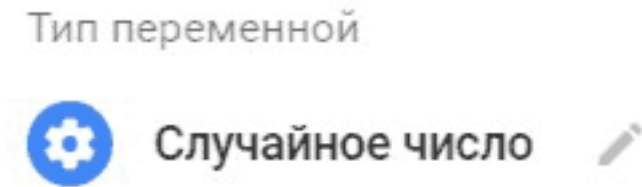


Рис. 293. Переменная «Случайное число»

Переменная возвращает случайное число в диапазоне от 0 до 2147483647. Аналог встроенной переменной **Random Number**.

**Таблица поиска.** Тип переменной, который позволяет избежать конструкции **if..else** и **switch-case**.





Входная переменная ?

{{Click Element}} ▾

Таблица поиска ?

Входные данные

Результат

1		element1		-
2		element2		-

+ ДОБАВИТЬ СТРОКУ

 Установить значение по умолчанию ?

Значение по умолчанию ?

Значение по умолчанию



## Рис. 294. Таблица поиска

В таблице поиска во входных данных задаются конкретные значения. Поиск значений всегда является точным совпадением и чувствительным к регистру.

### Как работает таблица поиска?



1. Задается входная переменная. Например, **{{Click Element}}**;
2. Далее идет проверка по таблице поиска и входным данными;
3. Если переменная **{{Click Element}}** принимает одно из значений из таблицы, то входная переменная **{{Click Element}}** примет значение из поля «Результат»;
4. Если значение переменной **{{Click Element}}** не найде-

но среди таблицы поиска, то будет использовано значение по умолчанию (если указано в настройках).

Если вы не укажете значение по умолчанию, переменная вернет неопределенное значение **undefined** в случае, если совпадение не будет выполнено.

## Таблица регулярных выражений

Тип переменной

 Таблица регулярных выражений 






Входная переменная   
{{Click URL}} 

Таблица регулярных выражений 

Шаблон	Результат
<input data-bbox="80 689 446 715" type="text" value="^mailto:"/>	<input data-bbox="508 689 874 715" type="text" value="email"/>
<input data-bbox="80 738 446 764" type="text" value="^tel:"/>	<input data-bbox="508 738 874 764" type="text" value="telephone"/>

Установить значение по умолчанию 

Значение по умолчанию 

Advanced Settings



- Игнорировать регистр
- Только полное соответствие 
- Включите группы для извлечения подстроки и функцию замены 

Рис. 295. Таблица регулярных выражений

Тип переменной, который похож на «Таблица поиска»,

только более гибкий, поскольку здесь можно использовать регулярные выражения.

## Данные контейнера

### Данные контейнера



#### Идентификатор контейнера

Общедоступный идентификатор контейнера (например, GTM-ABC42).

---



#### Номер версии контейнера

Номер предварительной версии, если контейнер находится в режиме предварительного просмотра. В противном случае – номер опубликованной версии.

---



#### Режим отладки

Значение true, если контейнер просматривается в режиме отладки.

Рис. 296. Пользовательские переменные «Данные контейнера»

Три переменные из этой категории:

● **Идентификатор контейнера** – возвращает номер контейнера GTM. Например, GTM-NC2LK3M. Аналог встроенной переменной **Container ID**.

Тип переменной



Идентификатор контейнера



Конфигурация не требуется.

Рис. 297. Переменная «Идентификатор контейнера»

● **Номер версии контейнера** – возвращает версию контейнера. Например, 5. Аналог встроенной переменной **Container Version**.

Тип переменной



Номер версии контейнера



Конфигурация не требуется.

Рис. 298. Переменная «Номер версии контейнера»

Значение «**QUICK\_PREVIEW**» возвращается в том случае, если вы просматриваете в режиме предварительного



просмотра.

Номер версии контейнера	Номер версии контейнера	string	'QUICK_PREVIEW'
-------------------------	-------------------------	--------	-----------------

Рис. 299. QUICK\_PREVIEW

**Режим отладки** – возвращает значение *true*, если пользователь просматривает контейнер в режиме предварительного просмотра, и *false* – если нет. Аналог встроенной переменной **Debug Mode**.

Режим отладки	Режим отладки	boolean	true
---------------	---------------	---------	------

Рис. 300. Режим отладки

С помощью этой переменной можно собрать все ваши хиты в отдельное тестовое свойство, а значение параметра «**Режим отладки**» в качестве пользовательского параметра, чтобы в Google Analytics создать под него отдельное представление и исключить режим отладки из статистики, отфильтровав его.

Встроенные и пользовательские переменные в GTM можно *копировать, удалять, просматривать изменения*, и у них можно *вывести примечания*. Для этого в правом верхнем углу нажмите на значок «три точки».

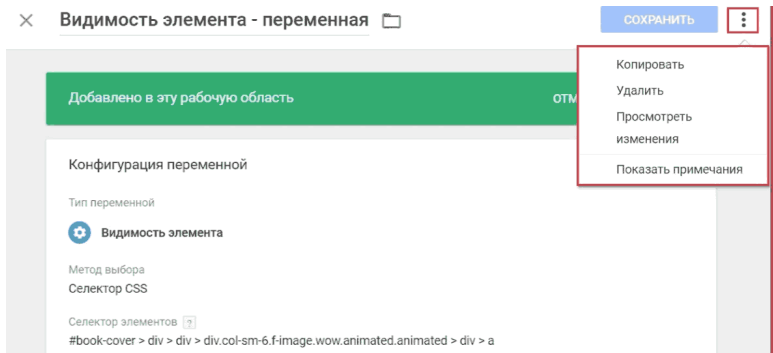


Рис. 301. Действия над переменными

В «**Просмотреть изменения**» можно узнать, что конкретно было изменено в переменной по сравнению с предыдущей версией, а в «Показать примечания» есть возможность заносить какие-то пометки, добавить описание по функциональности и т.д.

# Уровень данных

Уровень данных – это объект (массив объектов) или переменная JavaScript, которая хранит и передает информацию с вашего сайта в Google Tag Manager. Затем эти данные могут передаваться и другим сервисам, например, в Google Analytics, Facebook, Criteo OneTag и др.

Информацию с помощью уровня данных можно передавать самую разную – от специальных параметров и показателей, данных по транзакциям (настройка расширенной электронной торговли), User ID до каких-либо пользовательских событий.

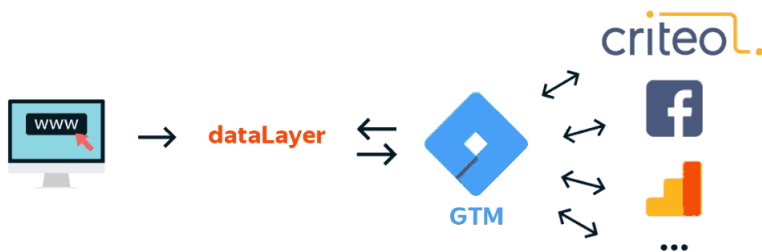


Рис. 302. Пример dataLayer

Очень часто уровень данных и **dataLayer** используют как синонимы. Это не так, поскольку уровень данных – это сам объект, с которым мы работаем и благодаря которому мы пе-

редаем данные в GTM, а **dataLayer** – всего лишь имя данного объекта. Оно в Google Tag Manager стандартное по умолчанию. Убедиться можно в этом перейдя в код контейнера:

## Установите Диспетчер тегов Google

Скопируйте приведенный ниже код и вставьте его на каждую страницу сайта.

Вставьте этот фрагмент в раздел **<head>** кода страницы как можно ближе к началу:

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-W9PC2C8');
<!-- End Google Tag Manager -->
```

## Рис. 303. Фрагмент кода контейнера GTM

Мы можем изменить данное имя и тогда будет использоваться не **dataLayer**, а то, которое захотим, например, такое:

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','urovenDannix','GTM-NC2KL3M');
<!-- End Google Tag Manager -->
```

## Рис. 304. Пример изменения уровня данных в коде сайта

Впоследствии все обращения к уровню данных мы должны выполнять, используя новое имя. Для того чтобы информация через уровень данных попадала в Google Tag Manager, необходимо использовать специальную конструкцию, кото-

рая задается по определенным правилам.

Все значения, которые используются в уровне данных GTM, состоят из пары **‘ключ’:’значение’**. Имя ключа и значения заключаются в одинарные кавычки, между собой они разделяются двоеточием. Например:

```
dataLayer = [{"ключ1": 'значение1'}];
```

Если мы используем несколько пар **‘ключ’:’значение’**, то такие пары между собой разделяются запятыми. Все пары заключаются в фигурные скобки. Например:

```
dataLayer = [{"ключ1": 'значение1', 'ключ2':  
'значение2'}];
```

Переменная *dataLayer* должна быть объявлена между тегами **<script></script>**. Существует несколько вариантов того, когда, каким образом и в каком месте должен быть описан уровень данных.

● **описание ДО кода контейнера GTM**. Конструкция ДО контейнера GTM выглядит так:

```
<script>  
dataLayer = [{"ключ": 'значение'}];  
</script>
```

К тому моменту, когда зафиксировается первое стандартное событие **«Просмотр страницы» (Page View)**, информация, которую мы передали с помощью данной конструкции, уже будет присутствовать в Google Tag Manager и мы с ней сможем работать.

В режиме предварительного просмотра это будет выглядеть так:

The screenshot displays the Google Tag Manager interface. At the top, the 'Data Layer' tab is selected. The main area is divided into two sections: 'DO GTM' and 'Data Layer values after this Message:'. Both sections show a single event object: `{event: 'DO GTM', gtm.uniqueEventId: 0}`. A red box highlights the '1 DO GTM' entry in the left-hand 'Summary' list. A red arrow points from this entry to the code block below. The code block, titled 'В коде страницы:', shows the following JavaScript code:

```
11 <!-- Начало данных DO контейнера GTM -->
12 <script>
13   dataLayer = [{event: 'DO GTM'}];
14 </script>
15
16 <!-- Google Tag Manager -->
17 <script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
18   new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
19   j=d.createElement(s),dl=l!='dataLayer'?'&l='<i>'' : '';j.async=true;j.src=
20   'https://www.googletagmanager.com/gtm.js?id=<i>id';j.parentNode.insertBefore(j,f);
21   })(window,document,'script','dataLayer','GTM-NC2KL3M');
```

Рис. 305. Пример описания ДО кода контейнера GTM

Для отслеживания различных действий пользователя в GTM предусмотрена специальная переменная `event`. Event можно использовать внутри обработчика того или иного события. Наличие ключа `event` сообщает GTM, что нужно что-то сделать. Далее он действует в зависимости от содержания `event`, в данном случае выводит значение «**ДО GTM**». Если `event` отсутствует или не задано, то в режиме отладки мы увидим слово «**Message**» и надпись:

Summary

Message

No tags were evaluated because there was no Event in the selected Message.

4 Window Loaded

3 DOM Ready

2 Page View

1 Message

```

11 <!-- Передача данных ДО контейнера GTM -->
12 <script>
13   dataLayer = [{key:'value'}];
14 </script>
15
16 <!-- Google Tag Manager -->
17 <script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({
18   new Date().getTime(),event:'gtm.js'});var f=d.getElemente
19   j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.as
20   'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.
21   })(window,document,'script','dataLayer','GTM-NC2KL3M')
```

Рис. 306. Отсутствие event, сообщение “Message»”

Например, при клике на кнопку, которая открывает форму с заказом услуги веб-аналитики, можно использовать конструкцию:

```

<a href="#analytics" data-tariff="Веб-аналитика" class="purchase-btn tariff-order fancybox"
onclick="window.dataLayer.push({'event':'button-click1',
'eventCategory' : 'click', 'eventAction' : 'кнопка'});">Заказать
услугу</a>

```

● **описание ПОСЛЕ кода контейнера GTM.** Конструкция ПОСЛЕ контейнера GTM выглядит так:

```
dataLayer.push ({'ключ':'значение'});
```

Отличие от описания ДО заключается в том, что здесь используется метод **push ()**, который добавляет один или более элементов в конец массива и возвращает новую длину массива. Помимо этого, квадратные скобки и равно заменились

# на круглые.

Google Tag Manager

Tags Variables **Data Layer**

Summary

ПОСЛЕ GTM

4 Window Loaded

3 DOM Ready

**2 ПОСЛЕ GTM**

1 Page View

ПОСЛЕ GTM:

```
1 [event: 'ПОСЛЕ GTM', gtm.uniqueEventId: 1]
```

Data Layer values after this Message:

```
1 [gtm: {start: 1527628254513, uniqueEventId: 1}, event: 'ПОСЛЕ GTM']
```

В коде страницы:

```
13 <!-- Google Tag Manager -->
14 <script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
15 new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
16 j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
17 'https://www.googletagmanager.com/gtm.js?id='+i+dl;j.parentNode.insertBefore(f,j)
18 })(window,document,'script','dataLayer','GTM-NC2KLB3M');</script>
19 <!-- End Google Tag Manager -->
20
21
22 <!-- Передача данных ПОСЛЕ контейнера GTM -->
23 <script>
24 dataLayer.push ({'event':'ПОСЛЕ GTM'});
25 </script>
26
```

Рис. 307. Пример описания ПОСЛЕ кода контейнера GTM

Подробнее о методе **push ()** читайте на javascript.ru. Описывая данные ДО кода GTM, уровень данных (dataLayer) еще не создан самим Google Tag Manager, а если ПОСЛЕ кода GTM, то уровень данных уже есть и нужно данные в нем дополнить теми, которые вам необходимы, а не создавать новый объект.

● **универсальное описание.** Для того, чтобы упростить работу в дальнейшем и не задаваться вопросом о том, когда вызывать уровень данных (ДО или ПОСЛЕ), используется конструкция следующего вида:

```
<script>
window.dataLayer = window.dataLayer || [];
window.dataLayer.push({'event':'value'});
```



</script>

В режиме предварительного просмотра Google Tag Manager будут зафиксированы сразу два события:

```
11 <!-- Передача данных ДО контейнера GTM -->
12 <script>
13 window.dataLayer = window.dataLayer || [];
14 window.dataLayer.push({'event':'ДО GTM'});
15 </script>
16
17 <!-- Google Tag Manager -->
18 <script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
19 new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
20 j=d.createElement(s),dl=l!='dataLayer'?'&l='+l:'';j.async=true;j.src=
21 'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
22 })(window,document,'script','dataLayer','GTM-NC2KL3M');</script>
23 <!-- End Google Tag Manager -->
24
25 <!-- Передача данных ПОСЛЕ контейнера GTM -->
26 <script>
27 window.dataLayer = window.dataLayer || [];
28 window.dataLayer.push({'event':'ПОСЛЕ GTM'});
29 </script>
```

Рис. 308. Универсальное описание

С помощью первой строки **window.dataLayer = window.dataLayer || []** мы проверяем, существует ли объект `dataLayer`. Если он существует, то используем его. В противном случае мы его создаем и он у нас пустой. Второй строчкой **window.dataLayer.push({'event':'value'})** мы дополняем с помощью метода `push()` тот объект (массив), который был создан в первой строчке.

Объект **window** сочетает два в одном: глобальный объект JavaScript и окно браузера. При добавлении объекта **window** эта переменная уровня данных также будет доступна из дру-

ного скрипта.

Важно отметить, что переопределение переменной уровня данных с тем же именем, что и существующая переменная, приведет к тому, что существующее значение будет перезаписано новым значением.

С помощью уровня данных можно передавать пользовательские события. Давайте разберем простой пример – передать данные в Google Analytics с помощью метода `push()` по клику на кнопке одной из форм.

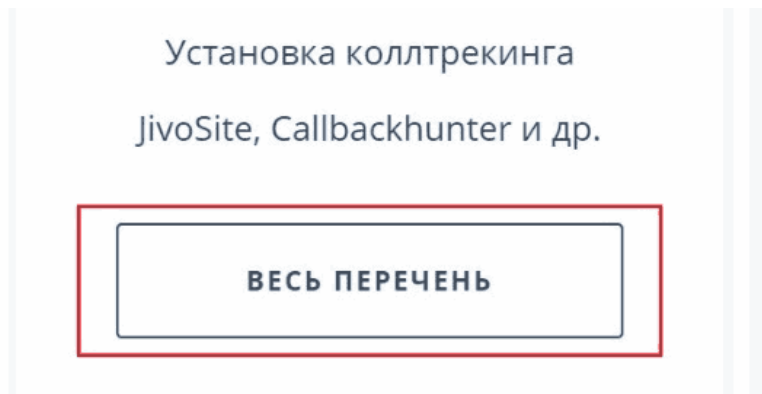


Рис. 309. Кнопка на одной из форм

Если бы вы не использовали GTM в своей работе, а только Google Analytics (код отслеживания Global Site Tag), и библиотеку `gtag.js`, то конструкция в коде страницы выглядела бы так:

```
</ul>
<a href="#web-analytics" data-tariff="Веб-аналитика" class="purchase-btn tariff-order fancybox"
onclick="gtag('event', 'analitika', { 'event category': 'click', 'event action': 'кнопка', });">
Весь перечень</a>
</div>
```

Рис. 310. Пример кода реализации конструкции из Google Analytics

Событие **onclick** возникает при щелчке левой кнопкой мыши на элементе, к которому добавлен атрибут **onclick**. Таким образом, когда пользователь нажимает на кнопку «Весь перечень», срабатывает событие, которое передает два обязательных условия (Категория *event\_category* и Действие *event\_action*) в Google Analytics, тем самым фиксируя цель.

Но когда Google Analytics установлен через GTM, так отслеживать события уже не получится. И поможет нам в этом уровень данных. Данные о событии мы можем передать сначала в *dataLayer*, а только оттуда – в *Analytics*.

Добавляем уровень данных к нашему элементу:


```
<a href="#web-analytics" data-tariff="Веб-аналитика" class="purchase-btn tariff-order fancybox "
onclick="window.dataLayer.push({'event':'UAevent', 'eventCategory':'click', 'eventAction':
'кнопка'});">Весь перечень</a>
</div>
```

Рис. 311. Добавление *dataLayer.push ()* к элементу

где *'event': 'UAevent'* – пользовательское событие, а *eventCategory* и *eventAction* – все те же обязательные условия.

**Крайне важно:** следите за символами кавычек. Копирование из разных мест (из чужих блогов и сайтов) и кодировок недопустимо.

```
<a href="#web-analytics" data-tariff="Веб-аналитика" class="purchase-btn tariff-order fancybox "
onclick="window.dataLayer.push({'event': 'UАevent', 'eventCategory': 'click', 'eventAction' :
'кнопка'})">Весь перечень</a>
```



Разные кавычки!

Рис. 312. Используйте одинаковые кавычки

В примере выше кавычки имеют разное написание, и событие фиксироваться не будет. Во всей конструкции используйте один тип символов!

Переходим к настройкам в интерфейсе Google Tag Manager. Нам необходимо создать:

1. две переменных уровня данных (**eventCategory** и **eventAction**);
2. триггер «**Пользовательское событие**» со значением **UАevent**;
3. тег для передачи данных в Google Analytics с типом отслеживания «**Событие**» и триггером активации **UАevent**;
4. цель в Google Analytics типа «**Событие**» со значениями категорий/действий = click/кнопка соответственно.

В GTM на основании передаваемых с помощью `dataLayer` данных можно создавать переменные и триггеры. Создадим их:

Переходим в «**Переменные**» – «**Создать пользовательскую переменную**». Выбираем из категории «**Пере-**

менные страницы» – «Переменная уровня данных». Задаем значение `eventCategory` и такое же имя. Сохраняем.

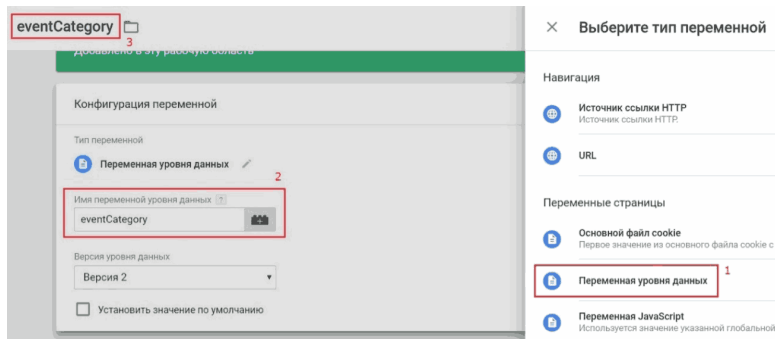


Рис. 313. Переменная уровня данных

Аналогично создаем вторую переменную уровня данных, только с именем **eventAction**. Далее создаем триггер типа «Пользовательское событие» с тем именем, которое указали в коде страницы у кнопки. В нашем случае – это **UAevent**. Вводим название и сохраняем триггер.

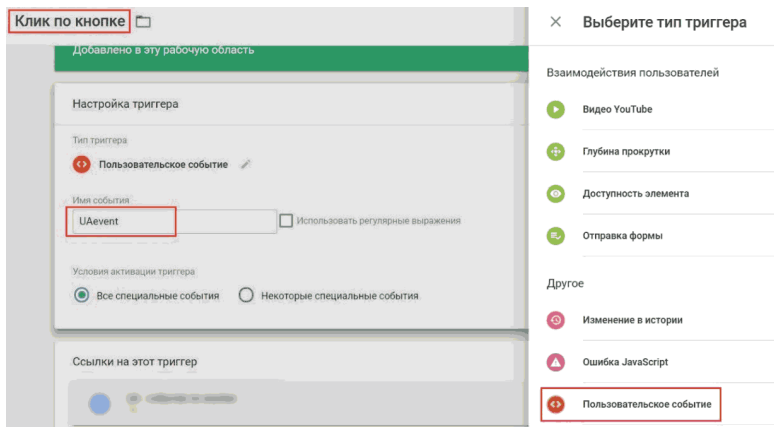


Рис. 314. Пользовательское событие

Теперь создаем тег типа «**Universal Analytics**» с настройками:

- Тип отслеживания – **Событие**
- Категория – **{{eventCategory}}**
- Действие – **{{eventAction}}**
- Идентификатор отслеживания – наш код отслеживания Google Analytics (можно создать переменную типа «**Константа**» и добавить туда идентификатор UA)
- Триггер активации – Пользовательское событие «**Клик по кнопке**».

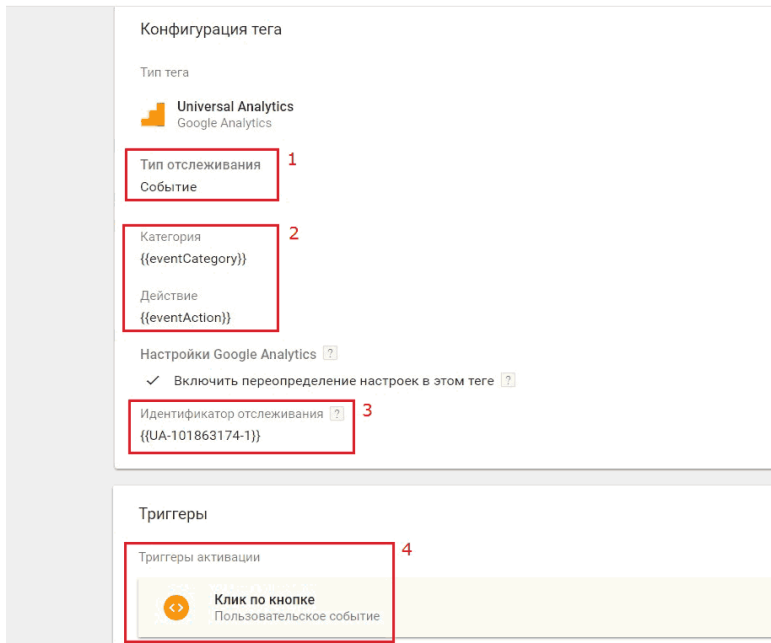


Рис. 315. Настройки тега

На последнем шаге переходим в Google Analytics и создаем цель «**Событие**» с условиями (те, которые указаны в коде страницы сайта):

- Категория – **click**
- Действие – **кнопка**

### 3 Подробные сведения о цели

#### Условия события

Задайте одно или несколько условий. Конверсия будет засчитываться, если произошло определенное событие. Для цели этого типа необходимо настроить по крайней мере одно условие.

[Подробнее...](#)

Категория	Равно ▾	click
Действие	Равно ▾	кнопка
Ярлык	Равно ▾	Ярлык
Ценность	Больше чем ▾	Ценность

Рис. 316. Настройка события в Google Analytics

Сохраняем цель. На этом настройка передачи события через уровень данных завершена. В режиме предварительного просмотра в GTM по клику на кнопку зафиксировано событие **UAevent** и сработал наш тег.

The screenshot shows the Google Tag Manager interface. At the top, there are tabs for 'Tags', 'Variables', and 'Data Layer'. The 'Summary' tab is active, showing a list of events on the left and a 'Summary' section on the right. The event '4 UAevent' is highlighted with a red box. The 'Summary' section shows 'Tags Fired On This Page' with a table listing 'UA - событие по кнопке' and 'Universal Analytics' tags, both of which have fired 1 time(s).

Event	Tags Fired On This Page
4 UAevent	UA - событие по кнопке Universal Analytics - Fired 1 time(s)
3 Window Loaded	Universal Analytics - Fired 1 time(s)
2 DOM Ready	
1 Page View	

Рис. 317. Событие UAevent в режиме отладки GTM



А с помощью отчетов «**В режиме реального времени**» в Google Analytics вы можете убедиться в корректности ее настройки и передачи данных.

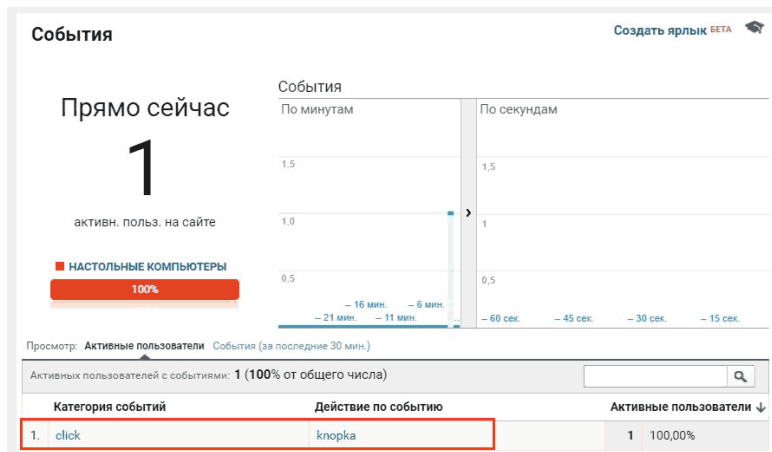


Рис. 318. Отладка события в «Режиме реального времени»

Чтобы проверить, что передается в dataLayer, можно воспользоваться консолью в панели инструментов разработчика (в Google Chrome – F12). Вводим dataLayer и нажимаем клавишу «**Enter**».



Рис. 319. Console – dataLayer – Enter

Как мы уже знаем, Google Tag Manager по умолчанию передает на уровень данных набор значений **gtm.js** (когда GTM готов к работе), **gtm.dom** (когда готова модель DOM) и **gtm.load** (когда окно полностью загружено). Последнее по счету событие **UAevent** – это наш клик по кнопке.

Использование уровня данных полезно при отслеживании отправки форм, данных по e-commerce, динамического ре-маркетинга, функции User ID и других событий на сайте.

Также с помощью уровня данных можно передавать сложные, вложенные структуры. Ярким примером служит передача значений о совершенных транзакциях расширенной электронной торговли. В примере ниже предполагается, что информация о продуктах, отображаемых на странице, известна в момент загрузки страницы:

```
1 <script>
2 window.dataLayer = window.dataLayer || [];
3 dataLayer.push({
4   'ecommerce': {
5     'currencyCode': 'RUB',
6     'impressions': [
7       {
8         'name': 'Продукт 1',
9         'id': 'id1',
10        'price': '666.6',
11        'brand': 'Бренд 1',
12        'category': 'Категория 1/Подкатегория 10',
13        'variant': 'Вариант 1',
14        'list': 'Список 1',
15        'position': 1
16      },
17      {
18        'name': 'Продукт 2',
19        'id': 'ID2',
20        'price': '1990.0',
21        'brand': 'Бренд 2',
22        'category': 'Категория 2/Подкатегория 20',
23        'variant': 'Вариант 2',
24        'list': 'Список 1',
25        'position': 2
26      }
27    ],
28    'event': 'gtm-ee-event',
29    'gtm-ee-event-category': 'Enhanced Ecommerce',
30    'gtm-ee-event-action': 'Product Impressions',
31    'gtm-ee-event-non-interaction': 'True',
32  });
33 </script>
```

Рис. 320. Фрагмент кода для передачи данных ecommerce

Подробнее об этом читайте в официальной справке разработчиков.

Понимание принципов работы и передачи данных с помощью dataLayer существенно упростит вам жизнь при работе с Google Tag Manager. Если вы до конца не разбираетесь в том, как устроен уровень данных, то реализацию dataLayer лучше всего доверить разработчику или специалисту с соответствующими техническими навыками.

# Триггеры

Триггер (от англ. слова *trigger* в значении спусковой крючок, или приводящий в действие элемент) в GTM – это условие, при котором активируется или блокируется тег. Для запуска тега должен быть хотя бы один триггер. Таким образом, нельзя создать тег без триггера. Триггер может принимать значение «*true*» (истина), либо «*false*» (ложь). Он выполняется только в том случае, когда его значение является истинным, и, если выполняются все условия триггера.

А чтобы это определить (*true* или *false*), значение переменной сравнивается с тем, которое задано в триггере. Все триггеры в Google Tag Manager связаны с событиями. Также они могут иметь дополнительные условия или фильтры. В старой версии диспетчера тегов Google триггер назывался «**правилом**».

Существует два способа создания триггеров:

1. перейти на вкладку «**Триггеры**» и нажать на кнопку «**Создать**»;

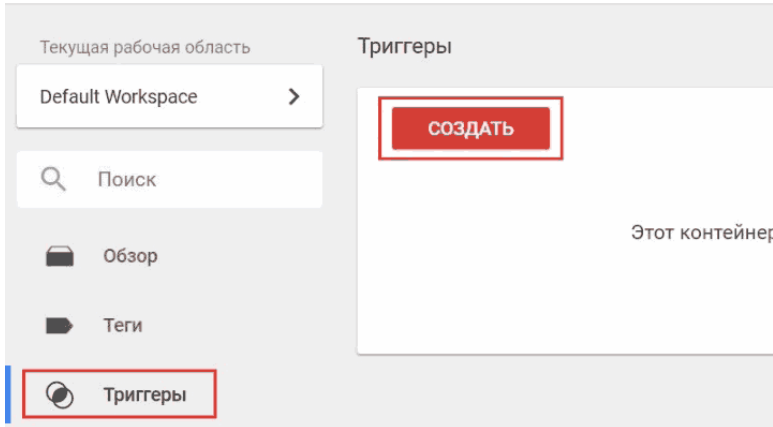


Рис. 321. Создание триггера

2. через сам тег в блоке «**Триггеры**» по нажатию на «+» если в нем уже есть триггеры;

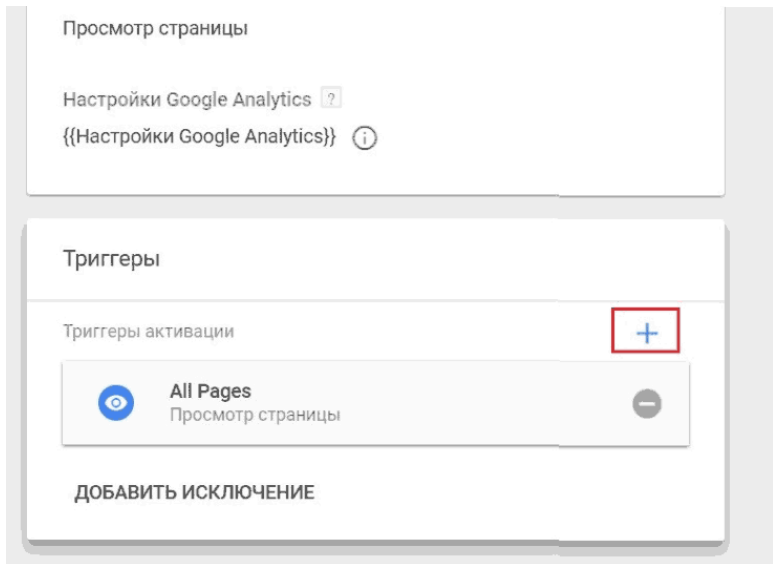


Рис. 322. Создание триггера

или же через пустую область блока, если триггеры к тегу еще не были добавлены.



Выберите триггер, который будет активировать этот тег...

[ПОДРОБНЕЕ...](#)

Рис. 323. Создание триггера

Тип триггера представляет собой комбинацию события GTM и типа взаимодействия (или события), которое вы хотите, чтобы GTM начал прослушивать. Они разбиты на несколько категорий:

- Просмотр страницы;
- Клик;
- Взаимодействие пользователей;
- Другое;

Разберем каждый из них последовательно. Но перед тем, как создавать новый триггер, вы должны четко осознать, какое событие GTM должно запустить ваш тег.



## Просмотр страницы

Просмотр страницы

Событие



Модель DOM готова

gtm.dom



Окно загружено

gtm.js



Просмотр страницы

gtm.load

Рис. 324. Триггеры «Просмотры страницы»

● **Модель DOM готова (DOM Ready)** – передается gtm.dom событие, как только браузер загрузил объектную модель документа;

● **Окно загружено (Window Loaded)** – передается gtm.load событие после завершения загрузки всей страницы, когда окно полностью загружено;

● **Просмотр страницы (Page View)** – передается gtm.js событие, когда контейнер GTM готов к работе.

Google Tag Manager передает эти три события на уровень

данных по умолчанию. Убедиться в этом можно перейдя в режим предварительного просмотра.

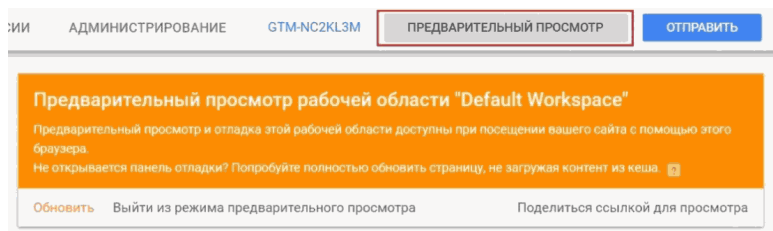


Рис. 325. Предварительный просмотр

Перейдя на сайт в режиме отладки, вы увидите:

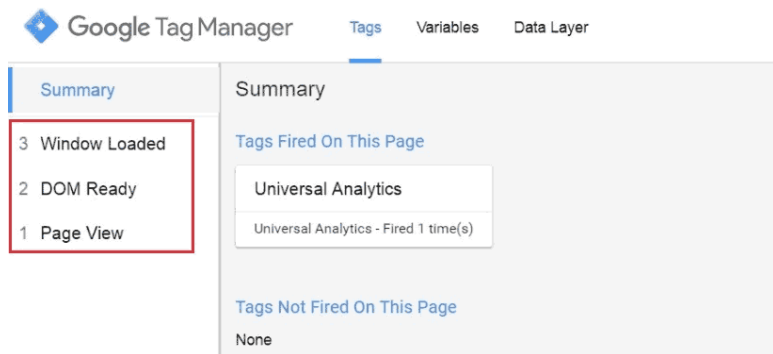


Рис. 326. Режим отладки – три события

Выбрав событие, например, **Window Loaded**, и перей-

дя на вкладку «**Variables**» (**Переменные**), можно посмотреть зафиксированное пользовательское событие (переменная **\_event**) и его значение (value – в нашем примере gtm.load):

Google Tag Manager Tags Variables Data Layer GTM-NC2KL3M

Summary Window Loaded

Variable	Variable Type	Return Type	Value
<b>_event</b>	Пользовательское событие	string	'gtm.load'
Click Classes	Переменная уровня данных	undefined	undefined
Click Element	Переменная уровня данных	undefined	undefined
Click ID	Переменная уровня данных	undefined	undefined

3 Window Loaded  
2 DOM Ready  
1 Page View

Рис. 327. Пользовательское событие

или у переменной «**Пользовательское событие**» :

Пользовательское событие	Пользовательское событие	string	'gtm.load'
--------------------------	--------------------------	--------	------------

Рис. 328. Переменная «Пользовательское событие»

## Клик

Тип триггера фиксирует клики по ссылкам или любым кликабельным элементам сайта.

Клик

Событие



Все элементы

gtm.click



Только ссылки

gtm.linkClick

Рис. 329. Триггеры «Клики»

● **Все элементы** – передается gtm.click событие когда на странице щелкнут любой элемент;

● **Только ссылки** – передается gtm.linkClick событие при нажатии на тег <a> элемента HTML.

Например, создадим триггер на «**Все элементы**» с условием активации триггера «**Все клики**».

## Тип триггера



Клик - Все элементы



## Условия активации триггера



Все клики



Некоторые клики

Рис. 330. Тип триггера – Клик – Все элементы

В таком случае при каждом клике по любой области на странице в режиме предварительного просмотра GTM будет фиксировать событие `gtm.click`:

Variable	Variable Type	Return Type	Value
<code>_event</code>	Пользовательское событие	string	'gtm.click'
Click Classes	Переменная уровня данных	string	"
Click Element	Переменная уровня данных	object	[object HTMLDivElement]
Click ID	Переменная уровня данных	string	"

Рис. 331. Фиксация кликов в режиме отладки

То есть вне зависимости от того, куда пользователь клик-

нул (на кнопку, на фон, в пустую область на странице) – все события будут зафиксированы. Если же мы хотим отслеживать события только по определенным элементам, то вместо условия активации триггера «**Все клики**» выбираем «**Некоторые клики**»:

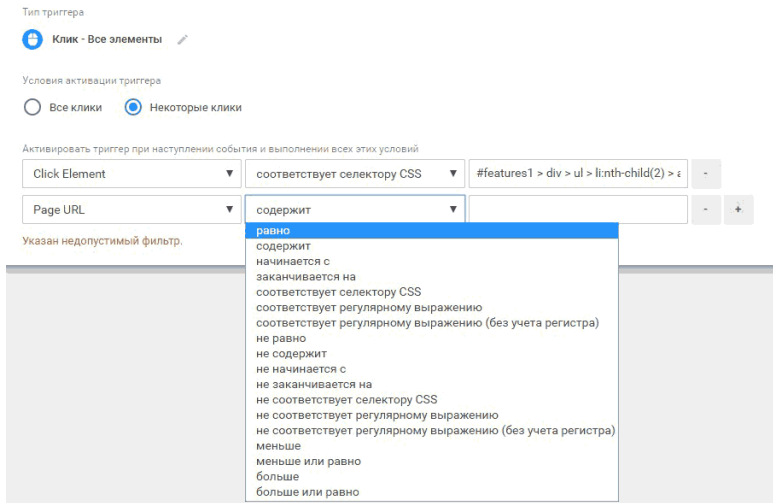


Рис. 332. Клики по определенным элементам

Нам станет доступна настройка активации триггера при наступлении события и выполнении всех условий, а именно:

1. сначала задается переменная (на примере выше – **Click Element**);
2. затем выбирается оператор из списка (равно, содер-

жит, соответствует селектору CSS, регулярному выражению и т.д.);

3. задается само значение (на изображении выше – CSS селектор одного элемента на сайте).

При необходимости можно задать несколько фильтров с помощью значка «+», либо же удалить ненужное с помощью «-».



Рис. 333. Добавление условий через +/-

Условия и фильтры связаны между собой логическим «И». То есть задав две строки, триггер будет срабатывать только тогда (событие будет считаться истинным и принимать значение true), когда одновременно соблюдаются первое **И** второе условия.

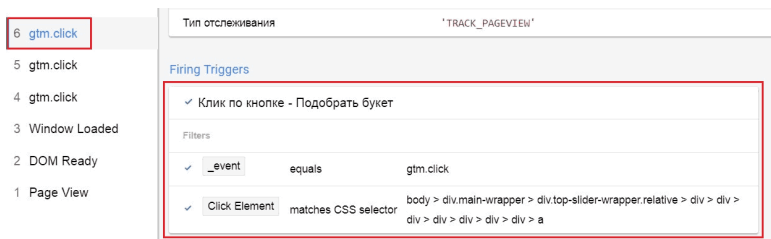
Активировать триггер при наступлении события и выполнении всех этих условий

Click Element	▼	соответствует селектору	▼	#features1 > div > ul > li:nth-child(2) > ε	-
Page URL	▼	содержит	▼	https://site.ru/catalog	- +

## Рис. 334. Логическое «И»

В этом примере триггер работает только тогда, когда пользователь кликнет по элементу с соответствующим селектором `#features1 > div > ul > li:nth...` **И** только на странице `https://site.ru/catalog`.

**Важно:** `gtm.click` – это предустановленные события GTM, они срабатывают при любом клике вне зависимости от того, был ли он осуществлен по некоторым элементам (при выборе такого) или же по всем элементам страницы. Главное, чтобы на нецелевых событиях не срабатывал тег, который активируется соответствующим триггером.



## Рис. 335. Соблюдение условий активации триггера

Аналогичным образом настраиваются триггеры для «Только ссылки», а событие в режиме отладки будет называться **gtm.linkClick**:



Google Tag Manager    Tags    **Variables**    Data Layer    QUICK\_PREVI

Summary    gtm.linkClick

Variable	Variable Type	Return Type	Value
<code>_event</code>	Пользовательское событие	string	'gtm.linkClick'
Click Element	Переменная уровня данных	object	https://graphanalytics.ru/#home

Рис. 336. Событие gtm.linkClick

Однако данный тип триггера имеет дополнительные настройки. Такие же опции появляются при создании триггера «Отправка формы».

Ждать теги ?

Макс. время ожидания ?

2000

мс

Проверка ошибок ?

Вы выбрали триггер "Ожидать теги" или "Проверка ошибок".

Рекомендуем включать его только на нужных страницах и проверять изменения перед публикацией.

[Подробнее...](#)

Рис. 337. Опции «Ждать теги» и «Проверка ошибок»

● **Ждать теги** – опция блокирует действие (задержка от открытия ссылок или отправки формы) до активации всех тегов в контейнере GTM, то есть пока все теги не загрузятся или пока не истечет указанное время ожидания (смотря что

произойдет раньше), событие зафиксировано не будет. Задается в миллисекундах (2000 мс = 2 секунды);

● **Проверка ошибок** – функция блокирует активацию тега, если для отслеживаемого элемента ссылки/формы не было выполнено действие по умолчанию («Клик по ссылке» или «Отправка формы»). Опция отвечает за то, на каких страницах нам необходимо использовать данный триггер.

Блок «**Проверка ошибок**» позволяет нам указать **ГДЕ** мы хотим использовать данный триггер, а блок с условиями активации определяет **КОГДА** этому быть. В качестве примера давайте разберем условие клика по ссылке (элемент – кнопка) со значением CSS-селектора и зададим «**Проверка ошибок**» на главной странице сайта.

Проверка ошибок ?

Вы выбрали триггер "Ожидать теги" или "Проверка ошибок".  
Рекомендуем включать его только на нужных страницах и проверять изменения перед публикацией. [Подробнее...](#)

Включить триггер при выполнении всех этих условий

Page URL	равно	http://rozi.ru/	+
----------	-------	-----------------	---

Условия активации триггера

Все клики по ссылкам  Некоторые клики по ссылкам

Активировать триггер при наступлении события и выполнении всех этих условий

Click Element	соответствует селектору CSS	body > div.main-wrapper > div.top-slide	-	+
---------------	-----------------------------	---	---	---

Рис. 338. КОГДА и ГДЕ

В режиме предварительного просмотра при клике на дан-

ную кнопку зафиксировать событие **gtm.linkClick**:



Рис. 339. Событие gtm.linkClick

Но если мы поменяем условие проверки ошибок, например, на такое:

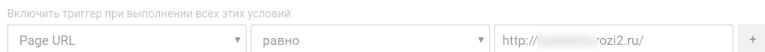


Рис. 340. Пример условия проверки ошибок

То события gtm.linkClick фиксироваться не будут, поскольку не соблюдено условие **ГДЕ**. До обновления Google Tag Manager «**Проверка ошибок**» и условие активации назывались «**Условия включения**» и «**Условия активации**». Рядовой пользователь мог легко запутаться в данной терминологии.

Для триггера типа «**Отправка формы**» распространенным условием активации функции «**Проверка ошибок**» является указание только тех страниц, на которых есть **формы**.

Триггер можно использовать со специально встроенными переменными: Click Element, Click Classes, Click ID, Click Target, Click URL, Click Text.

## Взаимодействие пользователей

Взаимодействия пользователей

Событие



Видео YouTube

gtm.video



Глубина прокрутки

gtm.scrollDepth



Доступность элемента

gtm.elementVisibility



Отправка формы

gtm.formSubmit

Рис. 341. Триггеры «Взаимодействие пользователей»

● **Видео YouTube** – срабатывает событие gtm.video когда пользователь просматривает видео на сайте (пока доступен только сервис-поставщик YouTube);

● **Глубина прокрутки** – срабатывает событие gtm.scrollDepth в том случае, когда пользователь прокручивает страницу (как горизонтально, так и вертикально);



● **Доступность элемента** – срабатывает событие gtm.elementVisibility когда элемент становится видимым на

странице;


● **Отправка формы** – срабатывает тег `gtm.formSubmit` в случае, если форма была отправлена.

## Видео YouTube

Тип триггера

 **Видео YouTube** 

Что регистрируется

- Начало
- Завершение
- Приостановка, перемотка, буферизация
- Ход просмотра 

Процентные пороги

10, 20, 30, 40, 50, 60, 70, 80, 90



процент

Временные пороги

Дополнительно

- Включить поддержку JavaScript API для всех видео YouTube 

Условия активации триггера

- Все видео
- Некоторые видео

## Рис. 342. Пример настройки триггера «Видео YouTube»

Тоже самое, что было описано в разделе «Переменные»:

- **Начало (Start)** – пользователь начинает просмотр видео;
- **Завершение (Complete)** – пользователь достигает конца видео;
- **Приостановка (пауза), перемотка, буферизация (Pause, Seeking, Buffering)** – пользователь останавливает, перематывает видео или когда происходит буферизация;
- **Ход просмотра (Progress)** – пользователь проходит процентный или временный порог (время измеряется в секундах). Целые положительные числа указываются через запятую.

### ● «Включить поддержку JavaScript API для всех видео YouTube»

Установив этот флажок, вы включите YouTube iFrame Player API. В результате ко всем URL видеопроигрывателя YouTube будет добавлен параметр *enablejsapi = 1* для управления проигрывателем через iframe или JavaScript.

Как правило, данный триггер используется в связке со встроенными переменными из блока «Видео»: Video Provider, Video Status, Video URL, Video Title, Video Duration, Video Current Time, Video Percent, Video Visible.

# Глубина прокрутки

Тип триггера



Глубина прокрутки 



Глубина вертикальной прокрутки 



Проценты

10, 25, 50, 75, 90



%



Пиксели



Глубина горизонтальной прокрутки 



Проценты



Пиксели

100



пикс.

Условия активации триггера



Все страницы



Некоторые страницы

## Рис. 343. Триггер «Глубина прокрутки»

- **Глубина вертикальной прокрутки в процентах** – 10, 25, 50, 75, 90 (пример);
- **Глубина горизонтальной прокрутки в пикселях** – 100 (пример);
- **Условия активации триггера** – Все страницы.

Как правило, данный триггер используется в связке со встроенными переменными из блока «**Прокрутка**»: Scroll Depth Threshold, Scroll Depth Units, Scroll Direction.



## Доступность элемента

Тип триггера



Доступность элемента



Метод выбора

Селектор CSS ▼

Идентификатор

Селектор CSS

Правило запуска этого триггера



Один раз на страницу



Один раз на элемент



При каждом появлении элемента на экране

Дополнительно

Минимальный процент видимости

## Рис. 344. Триггер «Доступность элемента»

Доступны следующие настройки:

Метод выбора:

### ● Идентификатор (id);

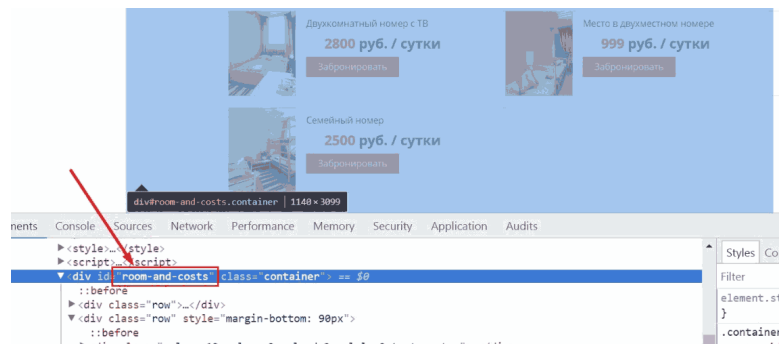


Рис. 345. Метод выбора – id

В большинстве случаев название атрибута id на странице уникальное.

### ● Селектор CSS

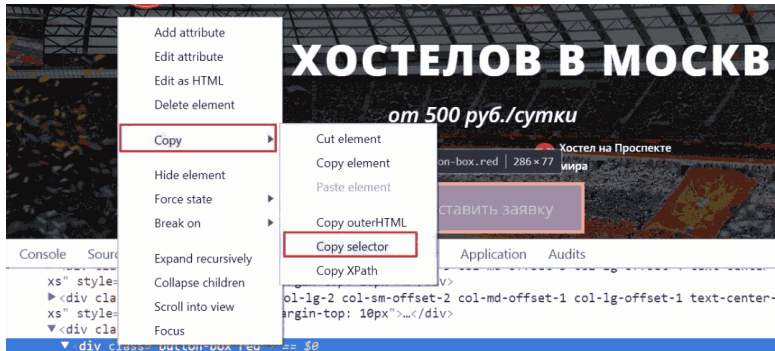


Рис. 346. Метод выбора – Селектор CSS

Применяется, если у отслеживаемого элемента нет атрибута **id**, а также если нам нужно отследить несколько элементов.

Например, у нас на сайте есть несколько форм отправки. У каждой из них есть свой уникальный **id**. Отслеживание всех форм через «Идентификатор» нам бы не подошло. А вот привязать метод выбора через «Селектор CSS» элементов вполне можно.

**Совет:** если не знаете, за что зацепиться, а нужно прослушивать конкретный элемент – цепляйтесь за его CSS-селектор. Получить его легко можно через консоль разработчика (в браузере Google Chrome вызывается через клавишу F12, см. скриншот выше).

## Правило запуска этого триггера

Правило запуска этого триггера

- Один раз на страницу ?
- Один раз на элемент ?
- При каждом появлении элемента на экране ?

Рис. 347. Правило запуска триггера

● **Один раз на страницу** – если несколько элементов на странице сопоставляются идентификатором или CSS-селектором, этот триггер запускается только один раз, когда первый из них отображается на этой странице.

При перезагрузке страницы триггер сработает повторно;

● **Один раз на элемент** – если несколько элементов на странице сопоставляются CSS-селектором, этот триггер будет запускаться когда каждый из них будет видимым на этой странице в первый раз. Если несколько элементов на странице имеют одинаковый идентификатор, только первый согласованный элемент будет запускать этот триггер;

Если на странице несколько форм, по которым по заполнения информации всплывает одно и то же сообщение, ис-

пользуйте эту настройку.

● **При каждом появлении элемента на экране** – в этом случае триггер срабатывает каждый раз, когда соответствующий элемент становится видимым.

**Минимальный процент видимости** – какой процент выбранного элемента должен быть виден на экране для срабатывания триггера. Можно использовать собственную переменные.

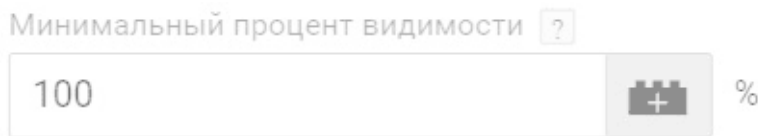


Рис. 348. Минимальный процент видимости

**Укажите минимальное время видимости** – как долго выбранный элемент должен быть виден на экране для срабатывания триггера. Время периодов видимости на одной странице суммируется. Например, если элемент был виден в течение 4000 мс (4 секунд), затем скрыт из видимости окна браузера и после этого снова виден в течение 2000 мс, то общее время видимости этого элемента составит 6 000 мс.



Укажите минимальное время видимости ?

4000



мс

Рис. 349. Минимальное время видимости

Если галочку не ставить, то время видимости для срабатывания триггера не будет играть роли.

**Регистрация изменений DOM** – настройка отслеживает элементы, которых изначально нет на странице, которые подгружены динамически и могут не входить в DOM (Объектная Модель Документа). Например, всплывающие формы. Если у вас есть на сайте есть такой контент, поставьте галочку, и такие элементы будут регистрироваться.



Регистрация изменений DOM ?

Рис. 350. Регистрация изменений DOM

## Отправка формы

Тип триггера




- Отправка формы 
- Ждать теги 
- Проверка ошибок 

Рис. 351. Триггер «Отправка формы»

Триггер аналогичен настройкам другого триггера **«Клик – Только ссылки»**. Можно использовать со специально встроенными переменными: Form Element, Form Classes, Form Target, Form URL, Form Text.

# Другое

Другое

Событие



Изменение в истории

`gtm.historyChange`



Ошибка JavaScript

`gtm.pageError`



Пользовательское событие



Таймер

`gtm.timer`

Рис. 352. Триггеры «Другое»

● **Изменение в истории** – срабатывает событие `gtm.historyChange` если изменился фрагмент URL (хэш, #). Как правило, используется на сайтах, контент которых подгружается динамически, без перезагрузки страниц;

В GTM можно использовать со встроенными переменными: `New History Fragment`, `Old History Fragment`, `New History State`, `Old History State`, `History Source`.

● **Ошибка JavaScript** – срабатывает событие `gtm.pageError` если происходит неперехваченное исключение JavaScript (`window.onError`);

В случае, если в JavaScript мы используем конструкцию



«try..catch», то GTM не зафиксирует ошибку. Также триггер не будет сигнализировать об ошибках, которые произошли до загрузки кода Диспетчера тегов. Можно использовать со встроенными переменными Google Tag Manager: Error Message, Error URL, Error Line, Debug Mode.

● **Пользовательское событие** – срабатывает событие event, которое отправили с помощью конструкции `dataLayer.push({'event': 'event_name'})`;

Пользовательские события позволяют отслеживать те взаимодействия, для которых не подходят стандартные методы. Например, отслеживание отправки формы с переопределением события submit.

Допустим мы хотим отправлять событие на уровень данных при нажатии кнопки «Отправить заявку». Для этого добавим код: `dataLayer.push({'event': 'button1-click'})`

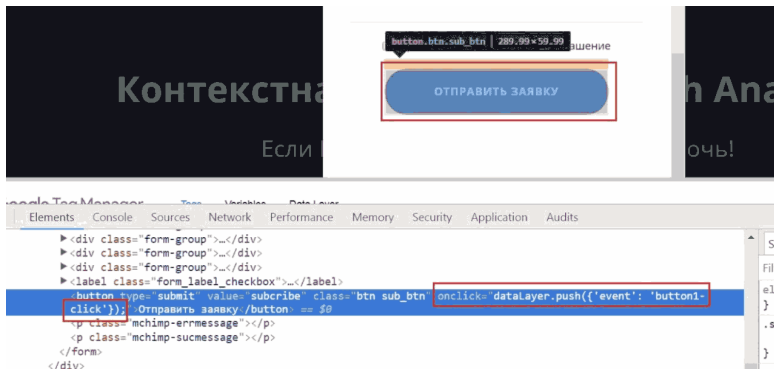


Рис. 353. `dataLayer.push({'event': 'button1-click'})`

где *button1-click* – название пользовательского события;

Далее необходимо создать тег, а в настройках триггера «**Пользовательское событие**» задать название *button1-click* и условие активации – «**Все специальные события**». При необходимости можно использовать регулярные выражения.

Тип триггера



Пользовательское событие 

Имя события



Использовать регулярные выражения

Условия активации триггера



Все специальные события



Некоторые специальные события

Рис. 354. Пользовательское событие *button1-click*

Когда пользователь нажмет кнопку, на уровень данных будет отправлено пользовательское событие со значением *button1-click*. Затем GTM определит *button1-click* как значение пользовательского события и активирует тег.



Рис. 355. Событие button1-click

В `dataLayer.push()` можно добавить не только одну переменную вида «ключ=значение», но и целый массив. А чтобы посмотреть порядок в котором события передаются на уровень данных, откройте консоль JavaScript в веб-браузере и введите `dataLayer`:



Рис. 356. Console – dataLayer

● **gtm.start** – запуск контейнера GTM;

- **gtm.dom** – модель DOM готова;
- **gtm.load** – окно загружено;
- **gtm.linkClick** – открытие формы;
- **button1-click** – клик по кнопке.

## Таймер


Событие **gtm.timer** срабатывает через заданный интервал времени (в миллисекундах).

Доступны следующие настройки:

Тип триггера

 Таймер 

Имя события



Интервал 



мс

Ограничение 



Включить триггер при выполнении всех этих условий


    

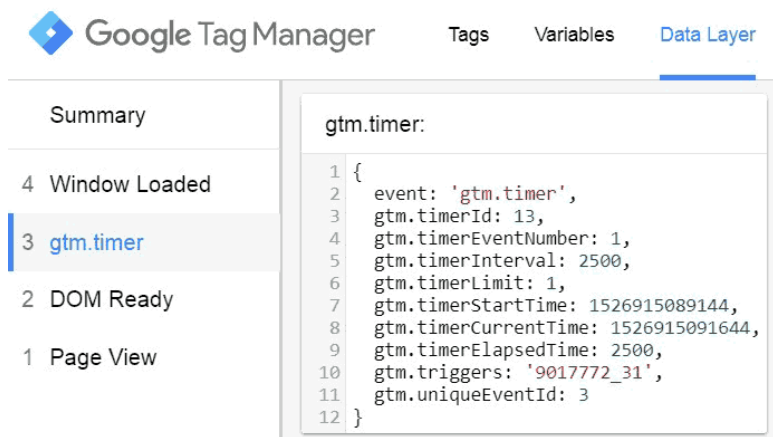
Рис. 357. Триггер «Таймер»

- **Имя события** (по умолчанию **gtm.timer**) – можем изменить на любое другое. Например, timer2018;
- **Интервал** – время, через которое должно активироваться событие. Например, через 2,5 секунды;

● **Ограничение** – максимальное число активаций события. Например, 1. Если оставить это поле пустым, событие будет активироваться до тех пор, пока пользователь не покинет страницу.

Если необходимо, чтобы таймер срабатывал на всех страницах сайта, воспользуйтесь конструкцией регулярного выражения.

Так выглядит уровень данных события **gtm.timer**:



The screenshot shows the Google Tag Manager interface. At the top, there are tabs for 'Tags', 'Variables', and 'Data Layer', with 'Data Layer' being the active tab. On the left, a 'Summary' sidebar lists several events: '4 Window Loaded', '3 gtm.timer' (highlighted in blue), '2 DOM Ready', and '1 Page View'. The main content area displays the JSON structure for the 'gtm.timer' event, with line numbers 1 through 12 on the left side of the code block.

```
gtm.timer:
1 {
2   event: 'gtm.timer',
3   gtm.timerId: 13,
4   gtm.timerEventNumber: 1,
5   gtm.timerInterval: 2500,
6   gtm.timerLimit: 1,
7   gtm.timerStartTime: 1526915089144,
8   gtm.timerCurrentTime: 1526915091644,
9   gtm.timerElapsedTime: 2500,
10  gtm.triggers: '9017772_31',
11  gtm.uniqueEventId: 3
12 }
```

Рис. 358. gtm.timer в Data Layer

Триггеры в Google Tag Manager делятся на триггеры активации тегов и триггеры блокировки тегов.

**Триггер активации** – это условие, при выполнении ко-

того срабатывает тег.

Например, вы хотите отслеживать клики по определенной кнопке на сайте. Зная значения переменной, вы можете настроить триггер активации, например, по Click ID:

Настройка триггера

Тип триггера

Клик - Все элементы

Условия активации триггера

Все клики  Некоторые клики

Активировать триггер при наступлении события и выполнении всех этих условий

Click ID ▾ равно ▾ menu-item-4306 - +

Рис. 359. Пример условия активации триггера

Таким образом, тег активируется только в том случае, когда идентификатор HTML-тега равен *menu-item-4306*.

Триггеры активации делятся на:

- *встроенные триггеры* – доступны к использованию готовые шаблоны;

- *пользовательские триггеры* – ручная самостоятельная настройка.

**Триггер блокировки** – это условие, при выполнении которого активация тега блокируется.

Например, вы хотите активировать тег на всех страницах сайта, кроме одной конкретной. Тогда необходимо создать триггер с условием активации на нужных страницах, а далее

в соответствующем теге добавить этот триггер в исключение.

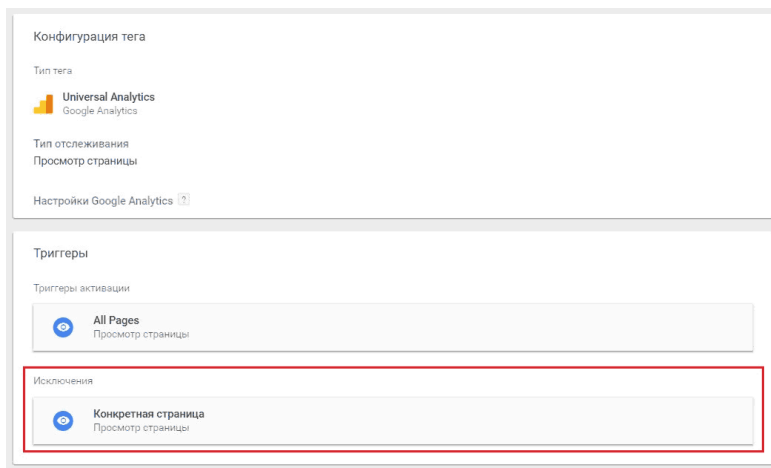


Рис. 360. Триггер блокировки или исключения

При конфликте триггеров активации и блокировки, приоритет отдается блокиратору. Добавить триггер блокировки можно с помощью кнопки «**Добавить исключение**» в настройках тега.

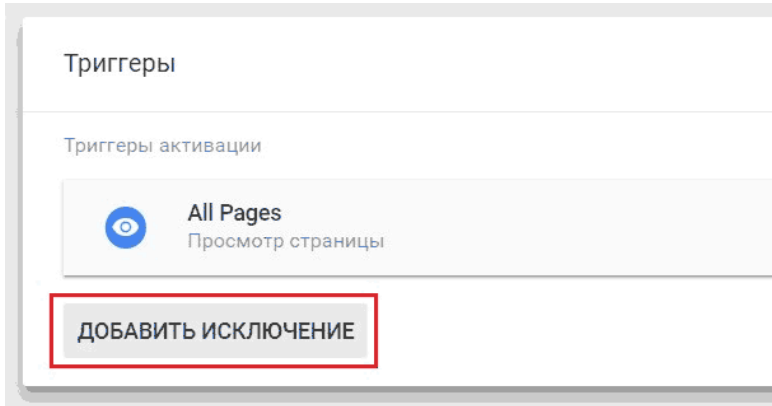


Рис. 361. Добавить исключение

В Google Tag Manager существует триггер, который добавлен по умолчанию и который нельзя удалить. Это **All Pages (Просмотр страницы)**.

Также как и переменные, триггеры в GTM можно *копировать, удалять, просматривать изменения*, и у них можно *вывести примечания*. Для этого в правом верхнем углу нажмите на значок «три точки».



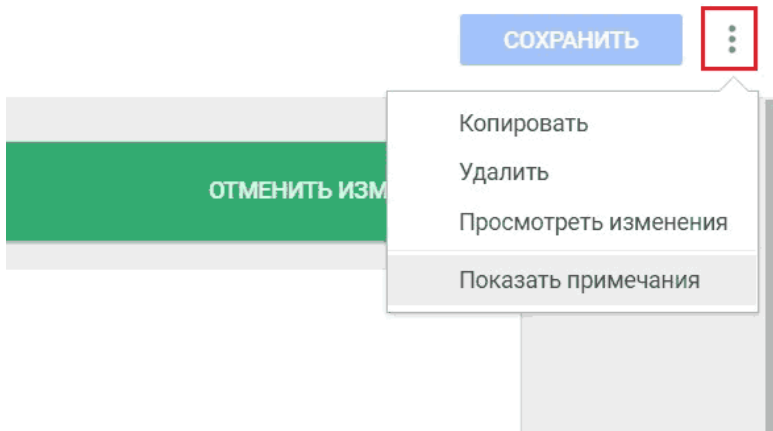


Рис. 362. Действия над триггерами

В «**Просмотреть изменения**» можно узнать, что конкретно было изменено в переменной по сравнению с предыдущей версией, а в «**Показать примечания**» есть возможность заносить какие-то пометки, добавить описание по функциональности и т.д.

После создания триггеров их следует добавить к тегам.

**Примечание:** для каждого тега не нужно создавать отдельный триггер. Вы можете использовать один триггер в разных тегах.

# Теги

Тег в Google Tag Manager – это фрагмент JavaScript кода, который собирает данные о посетителях на сайте и в приложении, а затем пересылает их на сторонние сервисы – Google Analytics, Google AdWords, Facebook, Яндекс.Метрика и т.д.

Если вы не используете Google Tag Manager, то все фрагменты кода других сервисов вам приходится вставлять вручную в исходный код страницы сайта. Это не очень удобно и отнимает много времени. Благодаря GTM все становится куда проще – исходный код менять не требуется, достаточно лишь указать в интерфейсе, какие теги вы хотите использовать и когда их нужно активировать.

В Google Tag Manager есть n-ое количество шаблонов тегов, которые облегчают их установку на сайте. Они разделены на категории:

● **рекомендуемые** – Universal Analytics, классический Google Analytics, отслеживание конверсий AdWords, ремаркетинг AdWords, счетчик DoubleClick Floodlight, тег продаж DoubleClick Floodlight, Google Optimize, Google Опросы;

## Рекомендуемые



### Universal Analytics

Google Analytics

---



### Классический Google Analytics

Google Analytics

---



### Отслеживание конверсий AdWords

AdWords

---



### Ретаркетинг AdWords

AdWords

---



### Счетчик DoubleClick Floodlight

DoubleClick

---



### Тег продаж DoubleClick Floodlight

DoubleClick

---



### Связывание конверсий

Google

---



### Google Optimize

Google Optimize

---



### Google Опросы: удовлетворенность посетителей сайтом

Google Опросы

Рис. 363. Рекомендуемые теги

● **специальные** – пользовательский HTML и пользовательское изображение;

## Специальные



Пользовательский HTML

Пользовательский тег HTML

---



Пользовательское изображение

Пользовательский тег изображения

---

Рис. 364. Специальные теги

● **еще** – сторонние сервисы, Twitter Universal Website Tag, Adometry, Crazy Egg, comScore Unified Digital Measurement, K50 tracking tag, Foxmetrics, Hotjar Tracking Code и другие.



## Выберите тип тега



**Стандартное отслеживание ClickTale**  
ClickTale



**comScore Unified Digital Measurement**  
comScore



**Crazy Egg**  
Crazy Egg



**Criteo OneTag**  
Criteo



**Ter DistroScale**  
DistroScale



**Универсальный пиксель Distillery**  
Distillery



**Eulerian Analytics**  
Eulerian Analytics



**FoxMetrics**  
FoxMetrics



**Google Trusted Stores**  
Google Trusted Stores



**Hotjar Tracking Code**  
Hotjar



**Тег отслеживания звонков Infinity**  
Infinity Tracking

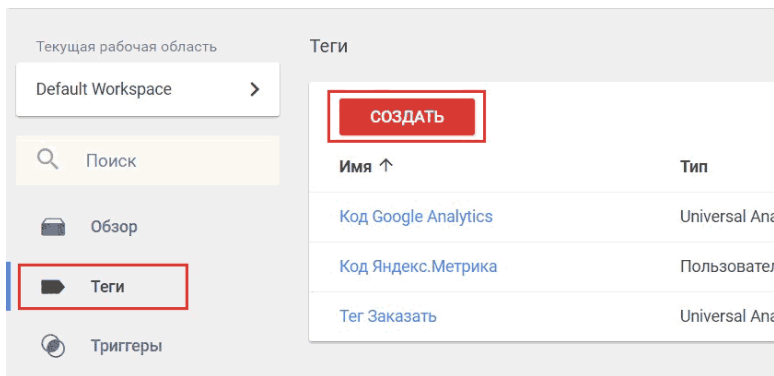


**Intent Media - Search Compare Ads**  
Intent Media

## Рис. 365. Теги сторонних сервисов

Разберем предметно первые две категории. Шаблоны, поддерживаемые в диспетчере тегов Google (последняя категория «еще»), можно посмотреть в официальной справке Google.

Чтобы создать тег в Google Tag Manager, перейдите на вкладку меню «Теги» и нажмите кнопку «Создать».



## Рис. 366. Создание тега

В появившемся окне нажмите на пустое пространство конфигурации тега и выберите его тип.

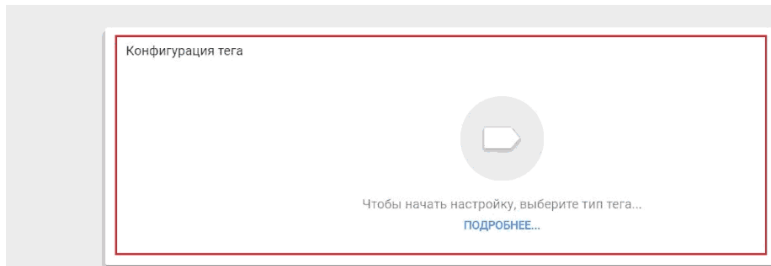


Рис. 367. Выбор конфигурации тега

## Рекомендуемые теги

Как правило, большинство маркетологов и веб-аналитиков работает с первыми четырьмя рекомендуемыми тегами: **Universal Analytics**, **Классический Google Analytics**, **Отслеживание конверсий Google AdWords** и **Ремаркетинг AdWords**. В зависимости от решаемых задач могут использоваться и другие.

Классический Google Analytics является устаревшей версией Universal Analytics. Однако он по-прежнему установлен на различных сайтах в интернете и официально поддерживается Google. На смену классическому пришел UA (Universal Analytics, 2012 год), в который привнесли ряд изменений:

- процесс сбора данных (у классического GA было 5 разных cookies, у UA – 1, ClientID);
- в UA добавили новый протокол передачи статистиче-

ских данных, который называется Measurement Protocol;

- пользовательские определения (параметры и показатели) пришли на смену пользовательским переменным (раньше для создания было доступно 5 метрик, то теперь их количество увеличилось до 20);

- в UA добавили расширенную электронную торговлю;

- в UA добавили функцию UserID, которая позволяет связывать разные устройства и объединять различные сеансы и действия во время этих сеансов с уникальным идентификатором. Это отслеживание еще называют кросс-девайсным.

- и т.д.

Создадим тег «**Universal Analytics**» и разберем его настройки:



Тип тега



Тип отслеживания

Просмотр страницы ▼

Настройки Google Analytics ?

{{GA ID}} ▼



Включить переопределение настроек в этом теге ?

▼ Расширенные настройки

Приоритет активации тегов ?

Включить специальное расписание активации тега

Активировать этот тег только в опубликованных контейнерах. ?

Настройки активации тега

Один раз на событие ▼

> Порядок активации тегов ?

## Рис. 368. Тип тега – Universal Analytics

В зависимости от выбранного в теге типа отслеживания дополнительные поля настроек будут меняться:

Тип отслеживания	Поля, которые необходимо задать	Специальные параметры	Специальные показатели	Группы контента	Электронная торговля	Реклама	Междоменное отслеживание	Расширенная конфигурация
Просмотр страницы	+	+	+	+	+	+	+	+
Событие	+	+	+	+	+	+	-	+
Транзакция	+	+	+	+	-	+	-	+
Социальные сети	+	+	+	+	-	+	-	+
Время	+	+	+	+	-	+	-	+
Внешний вид ссылки	+	+	+	+	-	+	-	+
Изменить внешний вид формы	+	+	+	+	-	+	-	+

Рис. 369. Сравнение полей настроек в зависимости от типа отслеживания

Для простого отслеживания Universal Analytics необходимо добавить один тег и выбрать тип отслеживания **«Просмотр страницы»**. Если же вас интересуют события или транзакции электронной торговли, потребуется добавить и другие теги.

## Тип отслеживания – «Просмотр страницы» (Page View)

Выбирается в том случае, когда вас интересует, какие страницы просматривают посетители на вашем сайте.

Для того, чтобы информация из GTM передавалась в Google Analytics, необходимо в тег добавить идентификатор отслеживания GA. Сделать это можно несколькими способами:

1. указать переменную в опцию **«Настройки Google Analytics»**;

При такой настройке следует добавить идентификатор отслеживания Google Analytics (например, UA-113446186-1)

в пользовательскую переменную типа «Константа», предварительно создав ее.

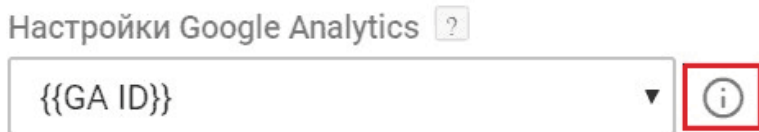


Рис. 370. Настройки Google Analytics

При клике на значок «i» откроется окно с заданной переменной, в которой можно оперативно поменять настройки. поставить галочку «**Включить переопределение настроек в этом теге**» и вручную добавить идентификатор отслеживания Google Analytics.

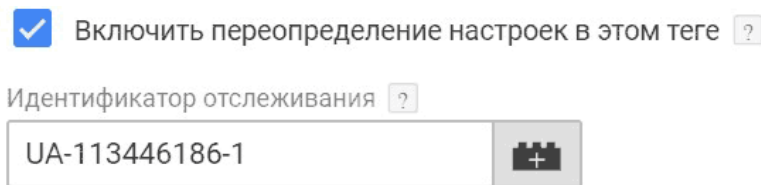


Рис. 371. Включить переопределение настроек в этом теге

При активации данного способа у тега появляются дополнительные поля переопределения настроек, в которых мы можем изменять настройки счетчика Google Analytics через интерфейс Google Tag Manager. Она служит для на-

стройки конфигураций Google Analytics, общих для нескольких тегов. Когда вы создаете тег Google Analytics с помощью Universal Analytics, Диспетчер предложит вам выбрать или создать переменную настроек Google Analytics. Большинство пользователей начнут работу с переменной настроек, содержащей только идентификатор отслеживания Google Analytics. Эту переменную можно будет использовать в любых дополнительных конфигурациях тегов с тем же идентификатором отслеживания Google Analytics (при этом вам не понадобится повторно его указывать).

● **Поля, которые необходимо задать** – доступен раскрывающийся список с полями, в которых задаются пара «ключ=значение»;

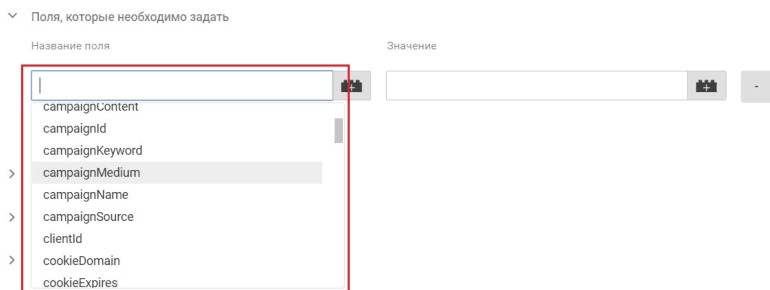


Рис. 372. Поля, которые необходимо задать

Например, вы можете добавить поле для кроссплатформенного отслеживания (функция User ID) или для от-

слеживания поддоменов. При настройке электронной коммерции в **Fields to Set** также можно передавать значения по товарам – идентификатор (ecommm\_prodid), тип страницы (ecommm\_pagetype) или общую ценность товаров (ecommm\_totalvalue), а также многое другое.

- Специальные параметры;
- Специальные показатели;
- Группы контента;
- Реклама – «Включить функции для контекстно-медийной сети»;

Если вас интересуют отчеты по демографии и интересам, ремаркетинг Google Analytics и интеграция с DoubleClick Campaign Manager (DCM), задайте значение «**True**». Включить эту функцию можно непосредственно через сам Google Analytics (на уровне ресурса «**Отслеживание – Сбор данных – Ремаркетинг «Вкл.»**)

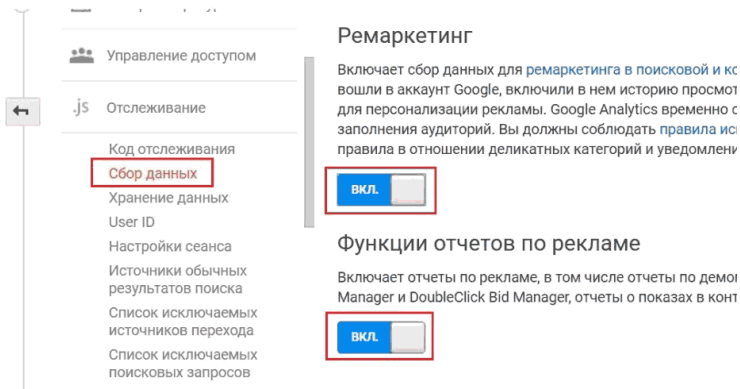


Рис. 373. Включение ремаркетинга и функций отчетов по рекламе в GA

● Электронная торговля – Включить расширенные функции электронной торговли;

При значении «**True**» вы сможете узнавать, когда пользователи добавляли товары в корзину, переходили к оформлению покупки и завершали ее, а также какие сегменты покупателей не совершают заказ. В Google Analytics настройка задается на уровне представления в разделе «**Настройки электронной торговли**».

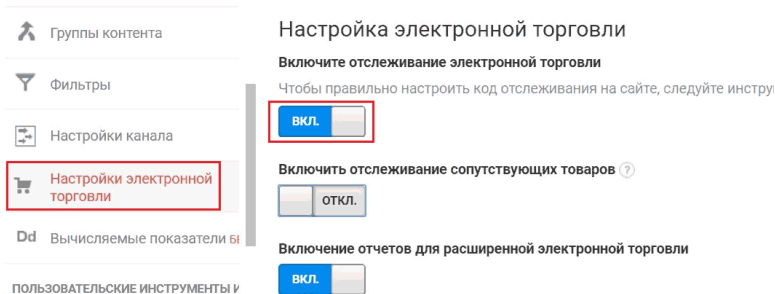


Рис. 374. Включение отслеживание электронной торговли

**Примечание:** просто включив эту настройку, данные по электронной коммерции передаваться не будут. Настройка Enhanced Ecommerce куда более сложный процесс и требует определенного времени и навыков.

### ● Междоменное отслеживание;

Для корректного сбора данных можно настроить междоменное отслеживание. Тогда сеанс с переходом между вашими разными сайтами, например, domain.com, domain.ru, domain.org, будет засчитан как один сеанс, а время сеанса будет определено как общее время пребывания на всех посещенных ресурсах.

## Междоменное отслеживание

Автоматическое связывание доменов ?

{{gaCrossDomains}}



Использовать решетку в качестве разделителя

False

Изменение внешнего вида форм

False

Рис. 375. Междоменное отслеживание

**Автоматическое связывание доменов.** Перечислите через запятую домены, для которых следует использовать общий идентификатор клиента, либо же создайте переменную типа «Константа» и задайте все домены там;

**Использовать в качестве разделителя решетку.** Позволяет использовать решетку (#) вместо знака вопроса (?) для добавления значений cookie в URL;

**Изменение внешнего вида форм.** Позволяет добавить информацию отслеживания к действию с формой.

● **Расширенная конфигурация,** в которой можно задать глобальную функцию, использовать отладочную версию и улучшенную атрибуцию ссылок. Настройки в этой катего-



рии используются крайне редко и предназначены для опытных пользователей.

▼ Расширенная конфигурация

Название глобальной функции  ?

Использовать отладочную версию

Включение улучшенной атрибуции ссылок

Задать имя трекера  ?

Рис. 376. Расширенная конфигурация

**Название глобальной функции.** Позволяет переименовать глобальную функцию, используемую тегом Universal Analytics.

В некоторых случаях при добавлении элементов библиотеки *analytics.js* на страницу переменная **ga** может быть уже занята. Эту проблему можно решить путем переименова-

ния глобального объекта **ga**. Например, чтобы переименовать объект **ga** в **analytics**, нужно изменить код отслеживания следующим образом:

```
<!-- Google Analytics -->
<script>
(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
(i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),
m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
})(window,document,'script','//www.google-analytics.com/analytics.js', 'analytics')
  Было ga
analytics('create', 'UA-XXXXX-Y', 'auto');
analytics('send', 'pageview');
</script>
<!-- End Google Analytics -->
```

Рис. 377. Изменение переменной **ga** в коде отслеживания Google Analytics

Подробнее о переименовании глобальной функции (объектов **ga**) читайте в официальной справке разработчиков Google.

**Использовать отладочную версию.** Этот вариант библиотеки *analytics.js* при выполнении создает подробные записи в консоли JavaScript.

Они содержат сведения об успешно выполненных командах, а также предупреждения и сообщения об ошибках в коде отслеживания. Кроме того, с их помощью можно получить подробную информацию о каждом обращении, отправленном в Google Analytics. Это позволит проверить, какие именно данные отслеживаются.

Чтобы включить отладочную версию библиотеки *analytics.js*, измените URL в коде отслеживания JavaScript с *https://www.google-analytics.com/analytics.js* на *https://www.google-analytics.com/analytics\_debug.js*

```
(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){(i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)})(window,document,'script','https://www.google-analytics.com/analytics_debug.js','ga');

ga('create', 'UA-XXXXX-Y', 'auto');
ga('send', 'pageview');
```

Рис. 378. Включение отладочной версии через код отслеживания GA

**Включение улучшенной атрибуции ссылок.** Позволяет повысить точность отчета «Статистика страницы». Когда страница содержит несколько ссылок на один URL, к каждой ссылке добавляется уникальный идентификатор элемента.

В Google Analytics эта функция находится на уровне ресурса в разделе «Настройки ресурса».

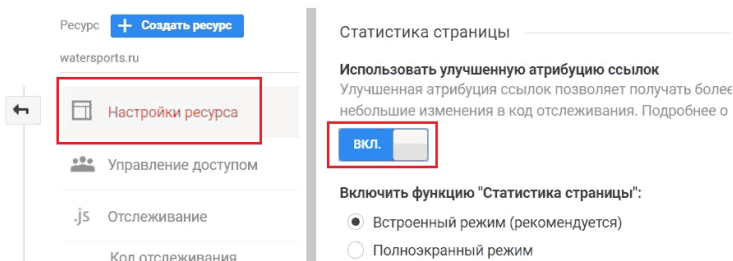


Рис. 379. Использование улучшенной атрибуции ссылок в Google Analytics

**Задать имя трекера.** Позволяет присвоить название объекту трекера. Не рекомендуется использовать в GTM, так как указав уже существующее имя трекера, это может привести к искажению данных. Хотя присвоение имени позволяет согласовать работу трекеров, созданных в Google Tag Manager с уже существующим кодом Google Analytics, вместо этого лучше перейти на *dataLayer*.

С помощью функции переопределения можно сделать так, чтобы тег Google Analytics использовал все те же настройки, которые были заданы в переменной настроек Google Analytics, но данные передавались на другой идентификатор отслеживания.

Подробнее о том, какие теги Google Analytics и дополнительные настройки поддерживает GTM, читайте в официальной справке Google.

## Расширенные настройки

**Приоритет активации тегов** – целое число (положительное или отрицательное), определяющее порядок запуска тегов. Чем больше данное число, тем раньше будет активирован тег при истинности одного из триггеров, связанных с данным тегом. По умолчанию значение «0», и все теги начинают выполняться одновременно независимо от того, закончилось ли выполнение предыдущего тега. Например, тег с приоритетом «3» будет запускаться до тегов с приоритетами 1 и 2.

▼ Расширенные настройки

Приоритет активации тегов

Рис. 380. Приоритет активации тегов

### **Включить специальное расписание активации тега.**

Если вы знаете, что некоторый тег понадобится вам на сайте в определенный период времени (например, при проведении рекламной акции вы отслеживаете определенные действия пользователей на сайте) просто активируйте данную

настройку. Вы можете задать дату начала (время начала) и дату окончания (время окончания). Также дополнительно необходимо указать часовой пояс. Дата и время окончания не могут предшествовать дате и времени начала.

Включить специальное расписание активации тега

Дата начала:	<input type="text" value="05/27/2018"/>	Время начала:	<input type="text" value="12:00"/>
Дата окончания:	<input type="text" value="06/15/2018"/>	Время окончания:	<input type="text" value="23:59"/>
Часовой пояс:	<input type="text" value="GMT +03:00 (local time)"/>		

Рис. 381. Включение специального расписания активации тега

Активировать этот тег только в опубликованных контейнерах. Настройка позволяет исключать запуск тегов в определенных ситуациях, например, в режиме предварительного просмотра.

Активировать этот тег только в опубликованных контейнерах. [?](#)

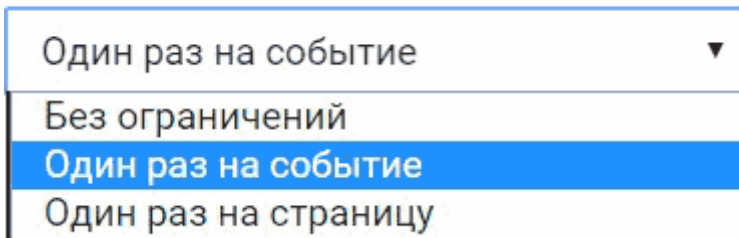
Рис. 382. Активировать тег только в опубликованных контейнерах

Когда вы находитесь в режиме отладки, как правило, вы контейнер не публикуйте до тех пор, пока не убедитесь в корректности работы всех настроек. При активации

данной настройки тег не будет активироваться, поскольку он не опубликован. Чтобы результаты тестирования были более точными, не включайте эту функцию.

## Настройки активации тега

Настройки активации тега



Один раз на событие ▼

Без ограничений

Один раз на событие

Один раз на страницу

Рис. 383. Настройки активации тега

**Без ограничений.** Тег активируется при каждом срабатывании триггера. Этот вариант используется, только если задан порядок активации тегов.

**Один раз на событие (указывается по умолчанию).** Тег активируется только один раз при выполнении определенного события. Этот вариант полезен для передачи информации на уровень данных. Информация передается один раз, поэтому и тег срабатывает один раз.

**Один раз на страницу.** Тег активируется один раз при загрузке страницы. Этот вариант применяется, когда необ-

ходимо активировать какой-то пользовательский скрипт JS, который мы подгрузили на страницу с помощью пользовательского HTML-тега.

Например, активации тега «**Один раз на страницу**» полезна, когда мы хотим отследить прокрутку страницы. Пользователь дошел до какого определенного элемента один раз, и мы это действие отследили. При последующих прокрутках туда-сюда нам уже это не так важно.

## Порядок активации тегов

▼ Порядок активации тегов ?

Активировать тег перед тегом **Ter Universal Analytics**

Ter setup

Выберите тег >

Активировать тег после тега **Ter Universal Analytics**

Ter cleanup

Выберите тег >

Рис. 384. Порядок активации тегов

В Google Tag Manager есть возможность задать тег, кото-



рый будет активирован перед текущим тегом и тег, который будет активирован после выполнения текущего тега.

После выбора тега у нас появляется настройка, которая позволяет указать параметр.

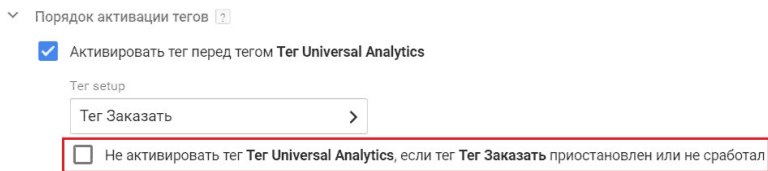


Рис. 385. Порядок активации тегов

Если вы выбираем активировать тег перед текущим, то мы можем указать, что данный тег, в котором мы сейчас находимся не должен быть активирован, если тег, который задан перед ним, приостановлен или не сработал.

Если мы выбираем тег, который должен быть активирован после данного тега, в котором мы сейчас находим и выполняем настройку, то поставив галочку мы указываем, что тег, который выбран из списка, не должен быть активирован, если тег, в котором мы сейчас находимся, приостановлен или не сработал.

В качестве простого примера разберем:

1. клик по кнопке «Подобрать букет» на сайте;

# ПОДОБРАТЬ БУКЕТ

Рис. 386. Пример отслеживания клика по кнопке

2. вывод с помощью пользовательского HTML-тега сообщения в консоли браузера «Hello, World». Конструкция имеет такой вид:

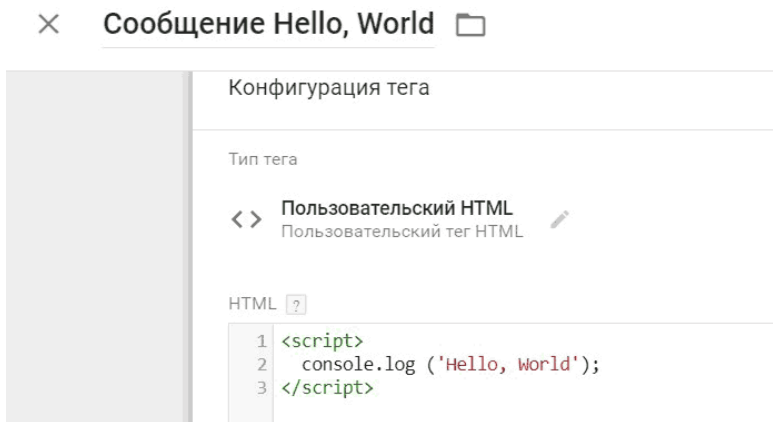


Рис. 387. Вывод сообщения с помощью пользовательского HTML-тега

В качестве порядка активации тегов зададим следующие значения:

● в теге – «Подобрать букет» активируем настройку запуска тега перед текущим, выбрав наш пользовательский HTML-тег «Сообщение Hello, World»

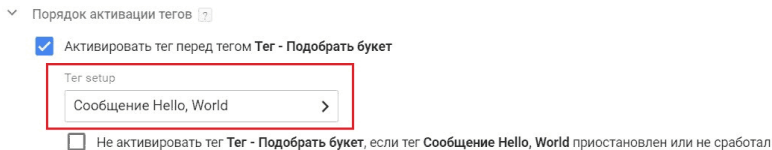


Рис. 388. Активация тега перед Тег – Подобрать букет

Таким образом, у нас сообщение в консоли «Hello, World» должно появиться до активации тега клика по кнопке «Подобрать букет».

У тега «**Сообщение Hello, World**» не заданы триггеры для активации, но есть информация о том, что данный тег будет активироваться непосредственно перед тегом «Подобрать букет».

## Триггеры

Триггеры активации

Не заданы триггеры для активации этого тега.

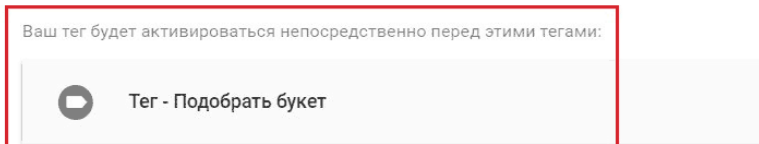
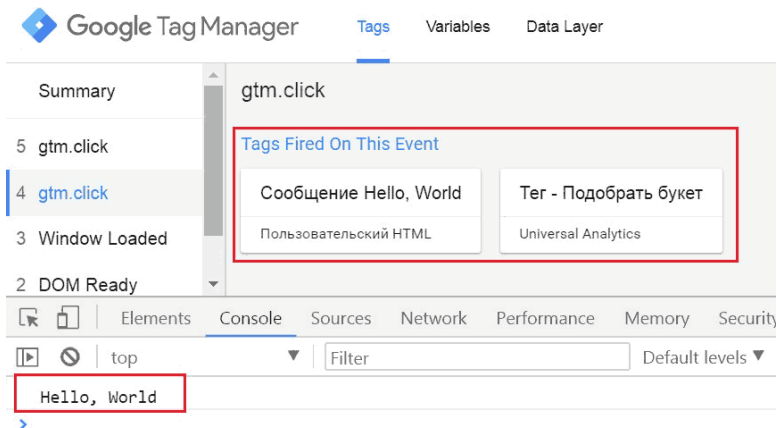


Рис. 389. Активация тега перед другими тегами

Перейдя в режим отладки Google Tag Manager и кликнув по кнопку, мы увидим, что по событию **gtm.click** активировалось два тега, несмотря на то, что у нас в настройках задан только один тег – это «**Подобрать букет**».



## Рис. 390. Пример активации тегов

Данные теги связаны между собой последовательностью активации. Аналогичным образом настраивается порядок активации «ПОСЛЕ».

Разберем другие типы отслеживания.

Тип отслеживания

Время	▼
Просмотр страницы	
Событие	
Транзакция	
Социальные сети	
<b>Время</b>	
Внешний вид ссылки	
Изменить внешний вид формы	

ого времени

Рис. 391. Типы отслеживания

### **Тип отслеживания – Событие**

В этом типе отслеживания задаются значения, которые использовались при настройке событий в Google Analytics.

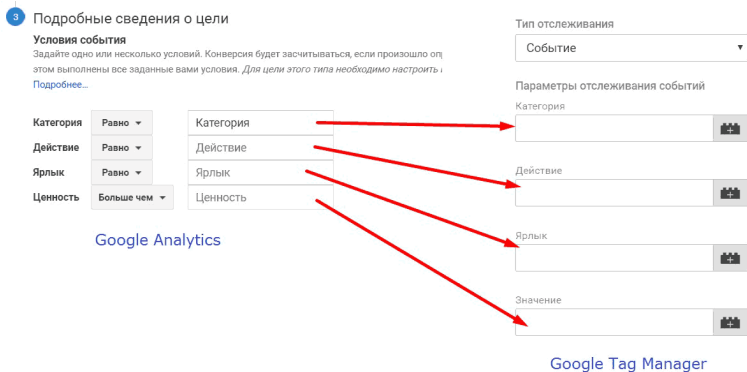


Рис. 392. Тип отслеживания – Событие

Опция **«Не взаимодействие»**. Если вы хотите, чтобы отправка события влияла на ваш показатель отказов, то необходимо установить значение *«False»* для этого параметра. В противном случае – устанавливайте значение *«True»*.

Не взаимодействие

Рис. 393. Опция «Не взаимодействие»

Все остальные настройки идентичны вышеописанным в типе отслеживания **«Просмотр страницы»**.

## Тип отслеживания – Транзакция

После того, как будет настроена передача данных о покупке, необходимо настроить тег, который будет передавать данные о транзакции в Google Analytics. Для этого и используется тип отслеживания – Транзакция.

**Важно:** включение отслеживания электронной торговли и создание тега с данным типом не позволит вам отслеживать данные о покупках пользователей. Для того, чтобы Google Tag Manager начал передавать сведения в Google Analytics обо всех транзакциях (идентификаторе товара, стоимости товара, наименовании и т.д.), необходимо использовать уровень данных или dataLayer, а также определенную конструкцию для передачи соответствующих параметров.

Подробнее читайте в официальной справке Google.

## Тип отслеживания – Социальные сети

С помощью данного типа можно отслеживать социальные взаимодействия пользователей с сайтом, например, лайки/репосты Facebook, Vk, Twitter и т.д.

## Тип отслеживания – Время

Отслеживание событий, срабатывающих по таймеру, может использоваться, если у вас есть страница (например, для просмотра видео), на которой пользователи могут долго оставаться, не запуская событий. Поскольку сеансы Google Analytics по умолчанию прекращаются через 30 минут, вам может понадобиться реализовать пользовательское событие, чтобы активность возобновлялась через какой-то другой промежуток времени.

Как правило, тип отслеживания **«Время»** в GTM используется в связке с триггером **«Таймер»**, а для передачи данных в Google Analytics задаются параметры отслеживания пользовательского времени: переменная, категория, значение, ярлык.

## Тип отслеживания – Внешний вид ссылки

Данный тип предназначен для отслеживания исходящих/внешних ссылок, то есть выходов (ухода) пользователей с сайта или загрузку файлов на странице (брошюр, прайс-листов, купонов и т.д.).

В качестве дополнительной настройки доступен параметр внешнего вида ссылки # (решетки) в качестве разделителя.



## Параметры внешнего вида ссылки

Использовать решетку в качестве разделителя

Рис. 394. Параметры внешнего вида ссылки

### **Тип отслеживания – Изменить внешний вид формы**

Этот тип отслеживания позволяет добавить информацию отслеживания к действию с формой. Например, чтобы отслеживать формы по доменам, информация об отслеживании должна быть добавлена к URL-адресу назначения формы при отправке формы. URL-адрес назначения формы – это URL-адрес в атрибуте действия формы:

```
<form action = "http://www.site.ru">
```

Отсюда URL-адрес назначения формы называют действием с формой, а добавление информации отслеживания к действию с формой называют изменением внешнего вида формы.

В качестве дополнительной настройки также доступен параметр внешнего вида ссылки # (решетки) в качестве разде-

лителя.

## Отслеживание конверсий AdWords

Рекомендуемый тег в Google Tag Manager предназначен для отслеживания действий пользователя на сайте и передачи информации в Google AdWords. Например, когда пользователь оформляет заказ на сайте, подписывается на рассылку, звонит в вашу компанию, скачивает приложение и т.д. Все эти действия называются конверсиями.

После создания действия конверсии в AdWords необходимо настроить тег.

Вы создали действие-конверсию. Теперь настройте тег и добавьте его на сайт

Настройка тега

Выберите способ добавления тега

Самостоятельно установить тег    Отправить тег по электронной почте    **Использовать Диспетчер тегов**

Инструкции

1. Установите тег отслеживания конверсий AdWords с помощью Диспетчера тегов Google. Вам потребуется предоставить следующую информацию о действии конверсии **Конверсия**  
Conversion ID  
858777881  
Ярлык конверсии  
poMhCMX1koMBEJnSv5kD
2. Убедитесь, что вы добавили тег связывания конверсий и настроили его так, чтобы он активировался на всех страницах сайта.

Тип тега

**Отслеживание конверсий AdWords**

Идентификатор конверсии

Ярлык конверсии

Ценность конверсии

Идентификатор заказа

Код валюты

Рис. 395. Отслеживание конверсий AdWords

Добавляем в тег «Отслеживание конверсий AdWords»:

● **Идентификатор конверсии (Conversion ID);**

## ● Ярлык конверсии.

Остальные значения заполнять не обязательно.

### ▼ Связывание конверсий

Включите связывание конверсий

Префикс файла cookie

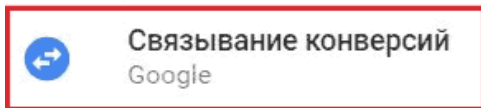
 

Рис. 396. Связывание конверсий

По умолчанию в опциях «**Связывание конверсий**» включено связывание конверсий (true) и задан префикс файла cookie **\_gcl**.

На втором этапе нужно добавить тег связывания конверсий и настроить его так, чтобы он активировался на всех страницах сайта.

## Тег «Связывание конверсий»



Google Optimize  
Google Optimize

Рис. 397. Тег «Связывание конверсий»

При нажатии на объявление URL целевой страницы обычно содержит информацию об этом клике. Когда посетитель выполняет нужное вам действие (например, активируя тег отслеживания конверсий AdWords), эта информация используется для связывания произошедшей конверсии с кликом, который привел пользователя на сайт.

Тег связывания конверсий автоматически считывает информацию о клике из URL целевой страницы и сохраняет эти данные в собственные файлы cookie в вашем домене.

Тип тега



Связывание конверсий

Google



No configuration required. [Learn more.](#)

Linker Options



Override cookie settings (advanced)



Рис. 398. Настройки тега «Связывание конверсий»

В качестве активации тега используется триггер **«Все страницы»**. В большинстве случаев для работы тега связывания конверсий будет достаточно базовой настройки. Однако вы можете переопределить значения полей Имя (Name prefix), Домен (Domain) и Путь (Path) через настройку **Linker Options**:

## Linker Options



Override cookie settings (advanced) [?](#)

Name prefix [?](#)

Example: `_gcl`



Domain [?](#)

Example: `example.com`



Path [?](#)

Example: `/`



Рис. 399. Linker Options

● **Имя.** Префикс, используемый как часть имен cookie, по умолчанию – `_gcl`. Например, если вы измените имя на `_gcl2`, тег связывания конверсий установит файлы cookie под названием `_gcl2_aw` и `_gcl2_dc`. При изменении префикса все теги, считывающие информацию о кликах из этих файлов cookie (например, теги отслеживания конверсий AdWords), нужно будет перенастроить на использование такого же префикса.

● **Домен.** Домен, в котором устанавливаются собственные файлы cookie. По умолчанию тег связывания конверсий использует домен самого высокого уровня. Например, если адрес вашего сайта – `blog.site.ru`, тег связывания конверсий будет использовать домен `site.ru`. Это поле следует задавать, только если вы хотите указать для файлов cookie домен более низкого уровня.

● **Путь.** Путь для файлов cookie. По умолчанию используется корневой уровень (`/`). Это поле следует задавать, только если вам нужно указать для файлов cookie подкаталог домена.

## Специальные теги Google Tag Manager

1. Пользовательский HTML
2. Пользовательское изображение

### Пользовательский HTML

У тегов, которых нет встроенных шаблонов в GTM, используется категория «Специальные теги». К ним относятся, например, коды Яндекс.Метрика, Facebook Pixel, Ретаргетинг ВК и другие.

Пользовательский тег HTML – это код стороннего сервиса, который должен быть заключен внутри тегов `<script></`

**script>**. Вот так выглядит пользовательский HTML тег счетчика Яндекс.Метрика:

Тип тега

<> Пользовательский HTML  
Пользовательский тег HTML

HTML 

```
1 <!-- Yandex.Metrika counter -->
2 <script type="text/javascript" >
3   (function (d, w, c) {
4     (w[c] = w[c] || []).push(function() {
5       try {
6         w.yaCounter48623396 = new Ya.Metrika({
7           id:48623396,
8           clickmap:true,
9           trackLinks:true,
10          accurateTrackBounce:true,
11          webvisor:true
12        });
13      } catch(e) { }
14    });
15
16    var n = d.getElementsByTagName("script")[0],
17        s = d.createElement("script"),
18        f = function () { n.parentNode.insertBefore(s, n); };
19    s.type = "text/javascript";
```

Рис. 400. Пользовательский HTML-тег, Яндекс.Метрика

Вы также можете включить вызовы функции **document.write()** в тегах, установив соответствующий флажок.

```
alt="" /></div></noscript>
29 <!-- /Yandex.Metrika counter -->
```

Поддержка функции **document.write** 

Рис. 401. Поддержка функции **document.write**



**document.write()** – метод добавления текста к документу. Он работает только пока HTML-страница находится в процессе загрузки и дописывает текст в текущее место HTML еще до того, как браузер построит из него DOM. Подробнее об этом методе читайте на [learn.javascript.ru](http://learn.javascript.ru).

Также в пользовательский HTML тег можно вставлять встроенные и пользовательские переменные. Для этого используют конструкцию двойных фигурных скобок `{{myVariable}}`.

HTML

```
1 <script>
2 var namePerem = {{myVariable}};
3 </script>
4
```

Рис. 402. Вставка переменных в HTML-тег

## Пользовательское изображение

Помогает отслеживать поведение пользователей там, где не работают остальные теги (например, в браузерах с отключенным выполнением скриптов). В конце ссылки добавляется невидимый пиксель изображения с URL-адресом определенного формата со специальными параметрами. По ним

происходит передача данных.

Universal Analytics, Пользовательский HTML и многие другие теги работают на основе JavaScript. Если они загружаются на странице, на которой отключена поддержка JavaScript, то теги не работают.

В связи с этим разработчики Google предусмотрели возможность загрузки через тег **<iframe>**. А он, в свою очередь, загружается частью контейнера **<noscript>** и показывает свое содержимое, если браузер не поддерживает работу со скриптами или их поддержка отключена пользователем. В остальных случаях браузер игнорирует этот тег и все, что располагается внутри него.

Именно для этих целей и нужна вторая часть кода контейнера Google Tag Manager, которую Google просит разместить после открывающего тега **<body>**:

Вставьте этот фрагмент в раздел **<head>** кода страницы как можно ближе к началу:

```
<!-- Google Tag Manager -->
<script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({'gtm.start':
new Date().getTime(),event:'gtm.js'});var f=d.getElementsByTagName(s)[0],
j=d.createElement(s),dl=l!='dataLayer'?s+l:'';j.async=true;j.src=
'https://www.googletagmanager.com/gtm.js?id='+i+dl;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-NPND52N');
<!-- End Google Tag Manager -->
```

Также вставьте непосредственно после открывающего тега **<body>** следующий фрагмент:

```
<!-- Google Tag Manager (noscript) -->
<noscript><iframe src="https://www.googletagmanager.com/ns.html?id=GTM-NPND52N"
height="0" width="0" style="display:none;visibility:hidden"></iframe></noscript>
<!-- End Google Tag Manager (noscript) -->
```

Рис. 403. Фрагменты кода контейнера GTM

Наиболее частое применение тега «**Пользовательское изображение**» в GTM – отслеживание посещений для пикселя Facebook. У этого типа тега всего две настройки:

Тип тега



**Пользовательское изображение**

Пользовательский тег изображения



URL изображения



**Включить блокировку кэша**

Параметр блокировки кэша



Рис. 404. Настройки пользовательского изображения

- **URL изображения** – поле, где содержится адрес изображения со всеми параметрами, которые нужно передать;
- **Включить блокировку кэша (параметр gtmcb по умолчанию)** – при включении обращения будут регистри-

роваться в том числе с браузеров с кэшированным изображением.

Знаете ли вы почему сайты загружаются так быстро? Одной из причин является *кэширование*. Первый раз, когда вы заходите на новый сайт, все статические элементы (шрифты, изображений, стили, скрипты и т.д.) загружаются во временную папку, которая будет хранить все эти файлы и в последующих заходах каждый раз показывать на соответствующих страницах. Периодически браузер проверяет, не обновились ли эти файлы на стороне сервера, и при необходимости загружает их заново.

С точки зрения пользователя – это удобно. Нет необходимости ждать очередной полной загрузки страницы. С точки зрения интернет-маркетологов и веб-аналитиков – это не очень хорошо, поскольку на странице данные не поменялись, файлы загрузились один раз во временную папку пользователя и больше отследить к ним обращения возможности нет.

Если отключить галочку параметра **«Включить блокировку кэша»**, то тег сработает только при первой загрузке сайта, так как изображение будет загружено во временное хранилище. А при включении опции Google Tag Manager добавит параметр `gtmcb` к URL изображения со случайным значением. Например, так:

**`site.ru/image.jpeg?gtmcb=1713862799`**

Поскольку значение в URL всегда разное, браузеру при-

дется каждый раз скачивать изображение. А раз это будет происходить каждый раз, то мы сможем отслеживать это событие постоянно. Поэтому этот параметр лучше всегда держать включенным.

В категории тегов «Еще» содержатся шаблоны к сервисам, которые не так популярны у нас, но которые часто используются в мире. Это и **comScore Unified Digital Measurement**, и тег **Adroll Smart Pixel**, и **Criteo One Tag**, и **Hotjar Tracking Code**, и другие. Руководства по настройке какого-либо тега из этой категории вы найдете на сайте поставщика.










	Twitter Universal Website Tag Твиттер
	Центр аудиторий 360 Центр аудиторий 360
	AB TASTY Generic Tag AB TASTY
	Adometry Adometry
	Ter AdRoll Smart Pixel AdRoll
	Конверсия AWIN AWIN
	AWIN Journey AWIN
	Bizrate Insights Buyer Survey Solution Bizrate Insights
	Bizrate Insights Site Abandonment Survey Solution Bizrate Insights

Рис. 405. Теги сторонних сервисов

Обычно теги активируются при загрузке страницы или в ответ на какое-либо действие на ней. В Google Tag Manager вы устанавливаете триггеры, определяющие, когда должны активироваться теги. Например, триггер **«Все страницы»** будет запускать тег на всех страницах при загрузке сайта.

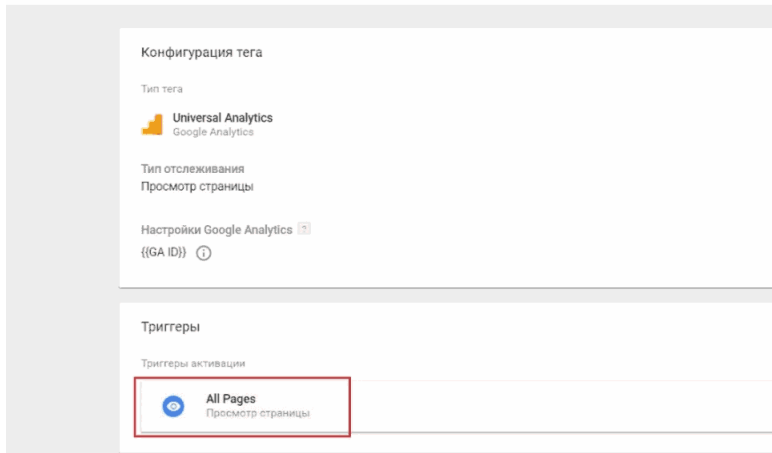


Рис. 406. Триггер активации – All Pages

Для тегов можно задавать как триггеры активации, так и триггеры блокировки (исключения).

# Отладчик

Режим отладки (он же режим предварительного просмотра) позволяет проверить корректность настроек тегов Google Tag Manager перед публикацией контейнера, выявить текущие проблемы на этапе тестирования и снизить риск их возникновения в будущем.

Режим предварительного просмотра в GTM еще называют режимом отладки. Проще говоря – это этап, на котором обнаруживают, локализуют и устраняют ошибки. Почему не активируется тег, не срабатывает событие, не определяется переменная? Ответить на эти и многие другие вопросы поможет именно отладка.

Предположим, что в своем контейнере вы настроили какие-то теги, добавили триггеры, переменные, и теперь хотите проверить корректность их выполнения. Чтобы включить режим предварительного просмотра, в интерфейсе Google Tag Manager нажмите на кнопку **«Предварительный просмотр»**.



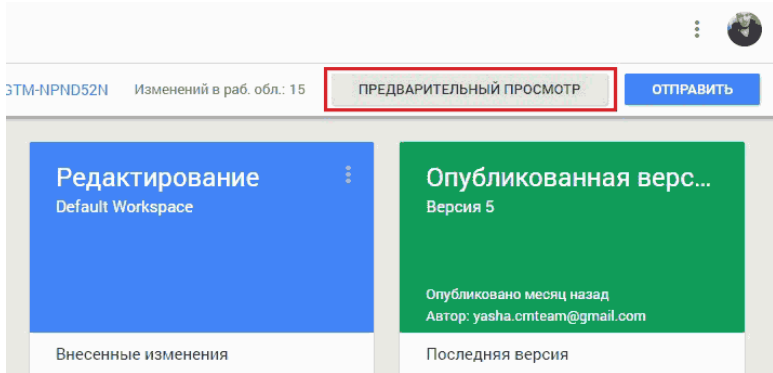


Рис. 407. Предварительный просмотр

Таким образом вы активируете режим предварительного просмотра и сверху в оранжевом прямоугольнике появится надпись:

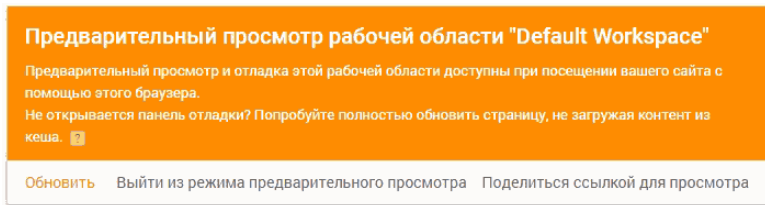


Рис. 408. Предварительный просмотр активирован

Доступны три функции:

- **Обновить** – если в процессе отладки, не выходя из

режима, вы внесли изменения в тегах, триггерах или переменных, то после каждого такого изменения следует обновлять предварительный просмотр, иначе Google Tag Manager не увидит правок и будет показывать предыдущую версию контейнера;

● **Выйти из режима предварительного просмотра** – выход из режима отладки;

Перед выходом GTM переспросит у вас:

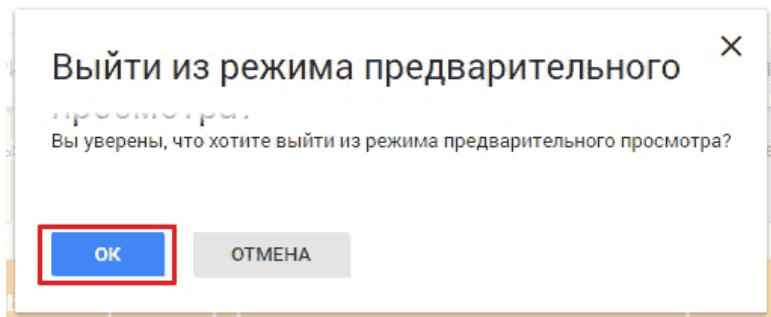


Рис. 409. Окно выхода

● **Поделиться ссылкой для просмотра** – отправить доступ другим людям;

Режим предварительного просмотра доступен только в том браузере, в котором был включен этот режим. Но вы можете предоставить доступ другим пользователям. Для этого необходимо нажать на кнопку «**Поделиться ссылкой для просмотра**». Далее вводите целевой URL, разрешаете от-

ладку при просмотре (gtm\_debug=x), копируете ссылку и отправляете ее другому пользователю.

X

## Поделитесь ссылкой для просмотра

Предоставьте ссылку на предварительный просмотр: **Default Workspace**.

Целевой URL ?

https://watersports.ru

Разрешить отладку при просмотре

Отправьте ссылку тем, кому вы хотите разрешить предварительный просмотр этой рабочей области контейнера.

https://www.googletagmanager.com/start\_preview/gtm?uiv2&id=GTM-NPND52N&gtm\_auth=6ByCux-9bIFtmMPHoonamA&gtm\_preview=env-7&gtm\_debug=x&url=https://watersports.ru

ЗАКРЫТЬ

Рис. 410. Поделитесь ссылкой для просмотра

Пользователь, в свою очередь, должен будет перейти по ссылке и кликнуть на целевой URL. После этого режим предварительного просмотра в его браузере будет активирован.

## Preview Container

Your browser is now in preview and debug mode for the **Draft Environment 6** **2018-04-19 01:46:44** environment of **Container GTM-NPND52N**. To start previewing, please visit the website <https://watersports.ru>

[Exit preview and debug mode](#)

Рис. 411. Preview Container

После активации предварительного просмотра в соседней вкладке откройте ваш сайт. В нижней части окна браузера появится консоль отладки с подробной информацией о запущенных тегах, порядке их активации и собираемых данных. Эту консоль будете видеть только вы – другим посетителям сайта она недоступна.

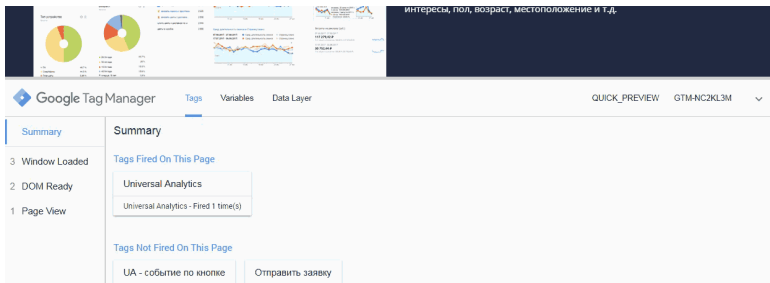


Рис. 412. Панель отладки

Если после сделанных работ вы не увидели нижнюю консоль, то это может быть связано с тем, что:

- вы используете какие-либо блокировщики, например, AdBlock или AdGuard (хотя в некоторых случаях GTM корректно работает и с ними);

- ваш антивирус блокирует всплывающие окна;

- сам браузер настроен таким образом, что воспрепятствует появлению дополнительных окон.

Если ничего из вышперечисленного не помогло, попробуйте зайти в режим отладки с другого браузера или компьютера.

Давайте разберем полностью всю навигацию консоли отладки.

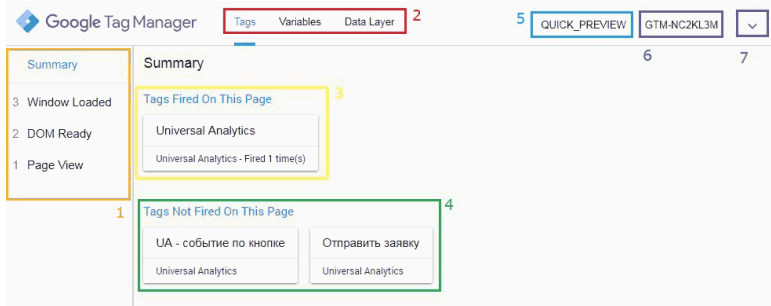


Рис. 413. Интерфейс панели отладки

## 1. Сводка, шкала временных событий

В этом блоке отображаются все события в хронологическом порядке, которые зафиксировал Google Tag Manager. Как мы уже знаем, по умолчанию диспетчер тегов Google передает на уровень данных 3 события вне зависимости от того, настроили вы что-то еще или нет. Это **gtm.js (Page View)**, **gtm.dom (DOM Ready)** и **gtm.load (Window Loaded)**. Цифра рядом с каждым событием означает порядок активации. В данном случае событие «**Page View**» (1) было зафиксировано раньше остальных.

При клике на любое событие из списка мы можем посмотреть дополнительную информацию о нем. Например, событие `gtm.click`:

The screenshot shows the Google Tag Manager interface. At the top, there are tabs for 'Tags', 'Variables', and 'Data Layer'. The 'Tags' tab is active. On the left, a list of tags is shown, with 'gtm.click' selected and highlighted by a red box. The main content area displays 'gtm.click' at the top. Below it, there are sections for 'Tags Fired On This Event' (showing 'None') and 'Tags Not Fired On This Event' (showing three buttons: 'UA - событие по кнопке', 'Universal Analytics', and 'Отправить заявку'). At the bottom, there are three more 'Universal Analytics' buttons.

Рис. 414. Дополнительная информация о событии gtm.click

Если наверху выбрана вкладка «**Tags (Теги)**», то мы можем увидеть какие теги были активированы по данному событию, а какие нет.

**Tags Fired On This Page** – теги, которые были активированы (цифра 3);

**Tags Not Fired On This Page** – теги, которые не были активированы (цифра 4);

В данном примере ни один из тегов не был активирован.

Если выбрана вкладка «**Variables (Переменные)**», то мы получаем подробные сведения о переменных в выбранном событии в тот момент, когда оно произошло. GTM будет передавать те значения переменных, которые были активированы в интерфейсе, даже если они не определены (undefined).





Google Tag Manager Tags Variables **Data Layer**

Summary	gtm.click	
5 gtm.click		
4 <b>gtm.click</b>		
3 Window Loaded		
2 DOM Ready		
1 Page View		

gtm.click:

```

1 {
2   gtm.element: [object HTMLDivElement],
3   gtm.elementClasses: '',
4   gtm.elementId: '',
5   gtm.elementTarget: '',
6   gtm.elementUrl: '',
7   event: 'gtm.click',
8   gtm.uniqueEventId: 4
9 }
```

1

Data Layer values after this Message:

```

1 {
2   gtm: {
3     start: 1528026060865,
4     uniqueEventId: 4,
5     element: [object HTMLDivElement],
6     elementClasses: '',
7     elementId: '',
8     elementTarget: '',
9     elementUrl: ''
10  },
11  event: 'gtm.click'
12 }
```

2

Рис. 416. Data Layer

Верхний пункт «**Summary**» показывает общую информацию о работе всех тегов GTM, которые сработали на данной странице за все время работы независимо от времени их загрузки.

Summary	Summary			
3 Window Loaded	<p>Tags Fired On This Page</p> <div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>Universal Analytics</p> <p>Universal Analytics - Fired 1 time(s)</p> </div>			
2 DOM Ready				
1 Page View	<p>Tags Not Fired On This Page</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;"> <div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>UA - событие по кнопке</p> <p>Universal Analytics</p> </div> </td> <td style="width: 50%; padding: 5px;"> <div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>Отправить заявку</p> <p>Universal Analytics</p> </div> </td> </tr> </table>		<div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>UA - событие по кнопке</p> <p>Universal Analytics</p> </div>	<div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>Отправить заявку</p> <p>Universal Analytics</p> </div>
<div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>UA - событие по кнопке</p> <p>Universal Analytics</p> </div>	<div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;"> <p>Отправить заявку</p> <p>Universal Analytics</p> </div>			

## Рис. 417. Summary

Если тег сработал несколько раз, то в Summary рядом с этим тегом будет изменено значение **Fired .. time (s)**

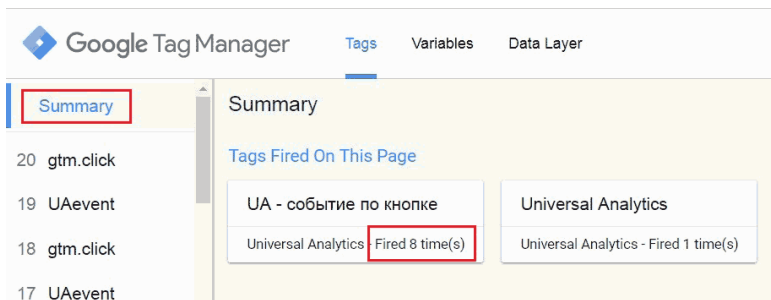


Рис. 418. Количество совершенных событий Fired X time(s)

В Summary информация по переменным недоступна, поскольку их значения меняются в зависимости от события.

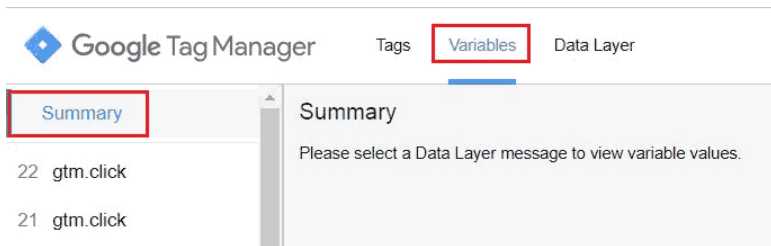


Рис. 419. Please select a Data Layer message to view variable

values

Ее необходимо просматривать только тогда, когда вы выбираете конкретное событие.

В Summary на вкладке «**Data Layer**» выводится вся информацию по уровню данных в хронологическом порядке – события с номерами, та информация, которая была передана по событию и общая информация об уровне данных на текущий момент, **Current values of the Data Layer** (блок выделен зеленым):

The screenshot displays the Google Tag Manager interface. At the top, the 'Data Layer' tab is selected. The main area is titled 'Summary' and shows a list of events on the left side, including 'gtm click' and 'UAevent'. A red arrow points from the 'gtm click' event at index 22 to its corresponding data object in the central pane. Another red arrow points from the 'UAevent' event at index 10 to its data object. At the bottom of the list, a 'Show More' button is highlighted with a red box. On the right side, a box titled 'Current values of the Data Layer:' contains a JSON object representing the current state of the data layer.

```
Current values of the Data Layer:
1 {
2   gtm: {
3     start: 1528027107251,
4     uniqueEventId: 30,
5     element: {object HTMLDivElement},
6     elementClasses: '',
7     elementId: '',
8     elementTarget: '',
9     elementUrl: ''
10  },
11  event: 'gtm.click',
12  eventCategory: 'click',
13  eventAction: 'knopka'
14 }
```

Рис. 420. Summary – Data Layer

Если событий было зафиксировано большое количество, то с помощью кнопки «**Show more**» можно просмотреть предыдущий список событий.

Возвращаемся назад к временной шкале событий и на

вкладку «Tags» (цифра 2).

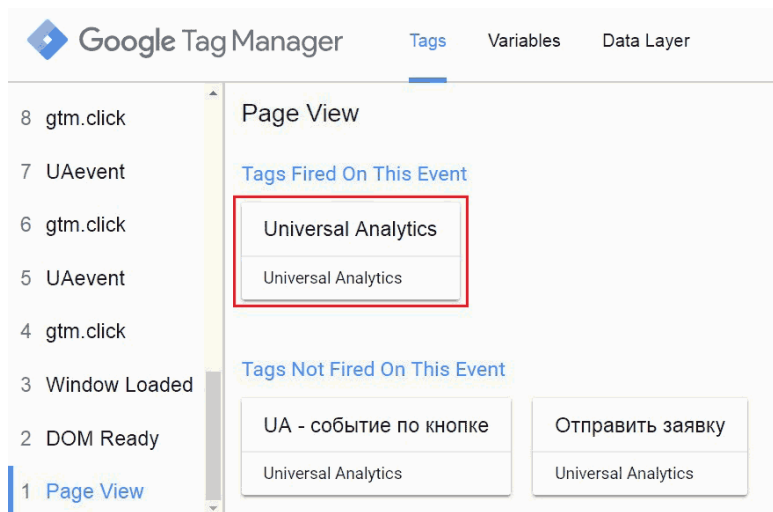


Рис. 421. Вкладка Tags

Нажав на тег, можно просмотреть его свойства и значения, а также триггеры активации (**Firing Triggers**) и блокировки (**Blocking Triggers**).

Google Tag Manager Tags Variables Data Layer

Page View > Display Variables as  Names  Values

✓ Universal Analytics **Название тега**

Properties **Свойства**

Name	Value
Type	Universal Analytics
Включить переопределение настроек в этом теге	true
Тип отслеживания	'TRACK_PAGEVIEW'
Идентификатор отслеживания	'UA-101863174-1'

Firing Triggers **Триггеры активации**

✓ All Pages

Filters

✓ `_event` equals `gtm.js`

Blocking Triggers **Триггеры блокировки**

No blocking triggers

14 gtm.click  
13 UAevent  
12 gtm.click  
11 UAevent  
10 gtm.click  
9 UAevent  
8 gtm.click  
7 UAevent  
6 gtm.click  
5 UAevent  
4 gtm.click  
3 Window Loaded  
2 DOM Ready  
1 Page View

Рис. 422. Свойства и значения

Триггер активации сработал и нет (отличия):

**Firing Triggers**

✓ Клик по кнопке

Filters

✓ `_event` equals `UAevent`

**Firing Triggers**

✗ Отправить заявку

Filters

✗ `_event` equals `button1-click`

Рис. 423. Триггер активации и триггер блокировки (отличия)

Активация триггера может состоять из нескольких условий. В этом случае какие-то отдельные условия могут работать, а какие-то нет:



Рис. 424. Несколько условий активации триггера

При несоблюдении хотя бы одного условия триггер не работает. В верхней части блока отладки на вкладке «Tags» есть настройка, которая называется **Display Variables as**.

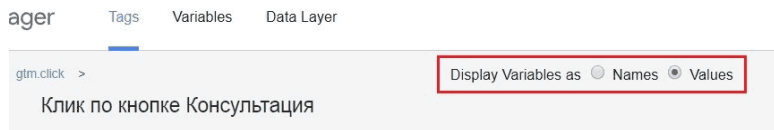


Рис. 425. Настройка Display Variables as

Она позволяет выбрать режим представления переменных: в виде имени (Names) или в виде значения (Values) пе-

ременной в момент активации.

The image shows two screenshots of the GTM filter configuration interface. The top screenshot is in the 'Names' tab, and the bottom is in the 'Values' tab. Both show a filter named 'Клик по кнопке Консультация' with three active filters. In the 'Names' tab, the filter names are 'Click Element', 'Page URL', and '\_event'. In the 'Values' tab, the filter values are 'https://graphanalytics.ru/#popup-audit', 'https://graphanalytics.ru/', and 'gtm.click'. Red boxes highlight the filter names in the top screenshot and the filter values in the bottom screenshot.

Filter Name	Operator	Value
Click Element	matches CSS selector	#home > div > div > div > div > a.button.hp-button-1.slide-button.w-button.green.fancybox
Page URL	contains	osipenkov.ru
_event	equals	gtm.click

Filter Value	Operator	Value
https://graphanalytics.ru/#popup-audit	matches CSS selector	#home > div > div > div > div > a.button.hp-button-1.slide-button.w-button.green.fancybox
https://graphanalytics.ru/	contains	osipenkov.ru
gtm.click	equals	gtm.click

Рис. 426. Различия в отображении

На примере выше условия активации триггера изменились с **Click Element**, **Page URL** и **\_event** (названия переменных) на значения переменных в момент активации.

Как мы уже знаем, для отслеживания различных действий пользователя в GTM предусмотрена специальная переменная event. Она можно использоваться внутри обработчика того или иного события. Если event отсутствует или не задано, то в режиме отладки мы увидим слово «**Message**» и надпись:

The screenshot shows the Google Tag Manager interface. On the left, a 'Summary' sidebar lists events: '4 Window Loaded', '3 DOM Ready', '2 Page View', and '1 Message'. The '1 Message' item is highlighted with a red box. To the right, a 'Message' box contains the text: 'No tags were evaluated because there was no Event in the selected Message.' Two red arrows originate from this message box: one points to the '1 Message' item in the sidebar, and the other points to a line of code in a code editor. The code editor shows the following code:

```
11 <!-- Передача данных ДО контейнера GTM -->
12 <script>
13   dataLayer = [{'key': 'value'}];
14 </script>
15
16 <!-- Google Tag Manager -->
17 <script>(function(w,d,s,l,i){w[l]=w[l]||[];w[l].push({
18   new Date().getTime(),event:'gtm.js'});var f=d.getElem
19   eById('script');j=d.createElement(s),dl=l!='dataLayer'?&l='+l:'';j.as
20   src='https://www.googletagmanager.com/gtm.js?id='+i+dl;f.p
21   rty=true;f.parentNode.insertBefore(j,f);
})(window,document,'script','dataLayer','GTM-NC2KL3M')
```

Рис. 427. «Message»

Если мы передаем через уровень данных пользовательское событие, то имя этого события выводится на временной шкале. Например, клик по кнопке зафиксировал пользовательское событие «**Яша Осипенков**».



The screenshot shows the Google Tag Manager interface. At the top, there are navigation tabs for 'Tags', 'Variables', and 'Data Layer'. On the left, a 'Summary' sidebar lists several events: '4 Яша Осипенков' (highlighted with a red box), '3 Window Loaded', '2 DOM Ready', and '1 Page View'. The main area displays the details for the selected event 'Яша Осипенков'. It is categorized under 'Tags Fired On This Event' and lists two tags: 'UA - событие по кнопке' and 'Universal Analytics'. Below this, the 'Tags Not Fired On This Event' section lists two tags: 'Universal Analytics' and 'Отправить заявку' (with 'Universal Analytics' listed below it).

Рис. 428. Событие с собственным именем

Название может быть, как на кириллице, так и на латинице. Однако устоявшимся традиционным написанием пользовательских событий все же является латинский алфавит. А информация на уровне данных выглядит следующим образом:

The screenshot shows the Google Tag Manager interface. At the top, there are tabs for 'Tags', 'Variables', and 'Data Layer' (which is selected and highlighted with a red box). On the right side, there are links for 'QUICK\_PREVIEW' and 'GTM'. The main content area is titled 'Яша Осипенков'. On the left, a sidebar shows a list of events: 'Яша Осипенков' (highlighted in blue), 'Window Loaded', 'DOM Ready', and 'Page View'. The main area displays the event configuration for 'Яша Осипенков' with the following code:

```
1 {event: 'Яша Осипенков', eventCategory: 'click', eventAction: 'knopka', gm.uniqueEventId: 4}
```

Below this, there is a section titled 'Data Layer values after this Message:' containing the following JSON object:

```
1 {
2   gtm: {start: 1528030843829, uniqueEventId: 4},
3   event: 'Яша Осипенков',
4   eventCategory: 'click',
5   eventAction: 'knopka'
6 }
```

Рис. 429. Data Layer

## 5. QUICK\_PREVIEW



Рис. 430. Предварительный просмотр

Эта надпись свидетельствует о том, что в данный момент мы находимся в режиме предварительного просмотра (отладки) контейнера Google Tag Manager.

## 6. Код контейнера

GTM-NC2KL3M

Рис. 431. Код контейнера

Идентификатор контейнера GTM.

## 7. Сворачивание панели отладки



Рис. 432. Сворачивание панели отладки

Иногда панель отладки Google Tag Manager загромождает нижнюю часть экрана. В этом случае мы на некоторое время можем свернуть его для внесения каких-либо правок с помощью данной опции.

В свернутом состоянии панель отладки выглядит так:



Рис. 433. Панель отладки в свернутом состоянии

При клике на элемент вернется исходное состояние GTM. Помимо этого, управлять высотой панели можно с помощью ее перетягивания левой кнопкой мыши вверх-вниз.

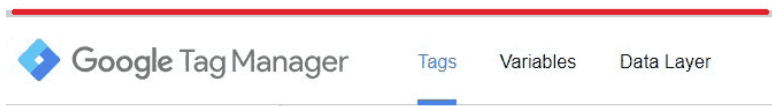


Рис. 434. Изменение высоты панели

После того, как вы проверили корректность всех настроек в режиме отладки и внесли соответствующие изменения, их можно опубликовать. Для этого необходимо нажать на кнопку **«Отправить»** в правом верхнем углу интерфейса Google Tag Manager.

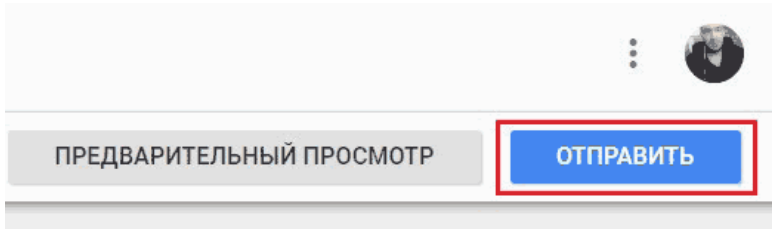


Рис. 435. Отправить на публикацию

Указав название версии или создав новую, а также выбрав среду, опубликуйте контейнер.

**Рекомендации:** если вы проверяете корректность срабатывания событий Google Analytics, то используйте для этого группу отчетов «В режиме реального времени».

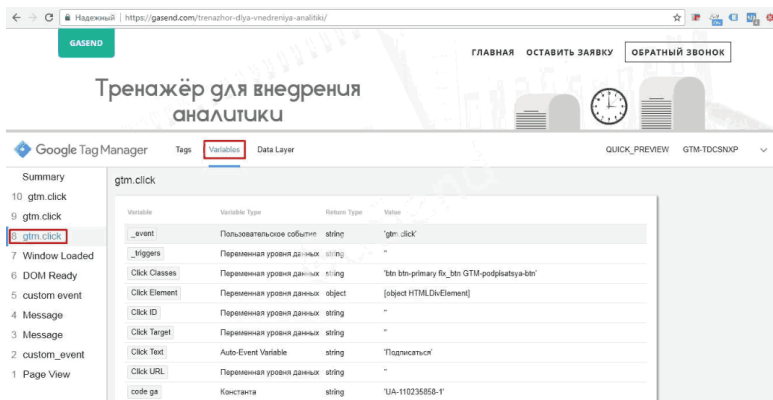
# Глава 3. Google Tag Manager на примерах

В первую версию книги «**Google Tag Manager для goog-лят**» попали самые распространенные примеры настроек GTM. Если вы нашли ошибку или хотите дополнить этот раздел собственной публикацией, присылайте материал мне на почту [ya.osipenkov@icloud.com](mailto:ya.osipenkov@icloud.com)

Данная глава подготовлена командой [GaSend.com](https://gасom.com).

# Прослушивание кликов в GTM

Используя отладчик (режим отладки) GTM, можно определить те переменные, с помощью которых будет осуществляться отслеживание действий пользователей на сайте.



The screenshot shows the Google Tag Manager interface in debug mode. The browser address bar displays the URL: [https://gasesend.com/trenazhor-diya-vnedreniya-analitiki/](https://gasesend.com/trenazhor-dlya-vnedreniya-analitiki/). The page title is "Тренажёр для внедрения аналитики". The GTM interface shows the "Variables" tab selected. In the left sidebar, the event "8 gtm.click" is highlighted with a red box. The main content area displays the details for the "gtm.click" event in a table format.

Variable	Variable Type	Return Type	Value
._event	Пользовательское событие	string	'gtm.click'
._higgers	Переменная уровня данных	string	''
Click Classes	Переменная уровня данных	string	'btn btn-primary fix_btn GTM-podpisatya-btn'
Click Element	Переменная уровня данных	object	{object HTMLClickElement}
Click ID	Переменная уровня данных	string	''
Click Target	Переменная уровня данных	string	''
Click Text	Auto-Event Variable	string	'Подписаться'
Click URL	Переменная уровня данных	string	''
code ga	Константа	string	'UA-110235858-1'

Рис. 436. Режим отладки – gtm.click

Очень часто такие события как клик по ссылке (**gtm.linkClick**), клик по кнопке (**gtm.click**) и отправка формы (**gtm.formSubmit**) не фиксируются в режиме отладки.

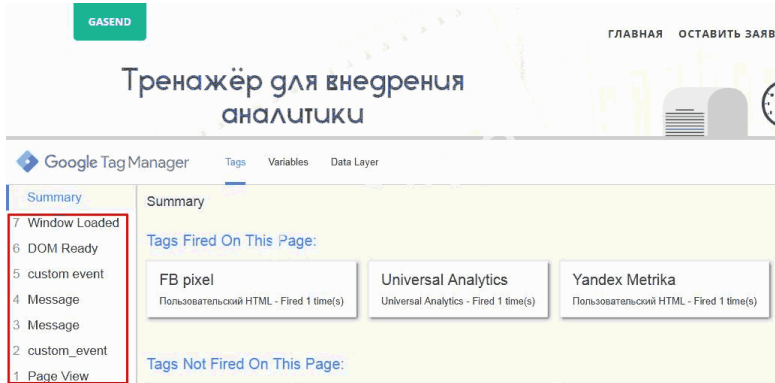


Рис. 437. События в режиме отладки

Как видно из скриншота, Google Tag Manager не зафиксировал ни одного из выше перечисленных событий. Для того чтобы события начали фиксироваться, необходимо создать три дополнительных триггера:

1. Начнем с настройки триггера для прослушивания кликов (**gtm.click**), который будет фиксировать события, связанные с кликами на странице.

Для этого создаем новый триггер и называем его **«Все клики»**.

- Тип триггера – **«Клик – Все элементы»**;

- Триггер активации – **«Все клики»**.

Сохраняем триггер.



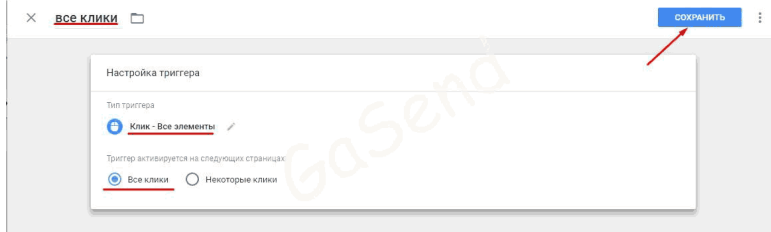


Рис. 438. Создание триггера типа «Все клики»

1. Триггер для прослушивания кликов по ссылкам (**gtm.linkClick**). Называем его «**Все клики по ссылкам**».

● Тип триггера – «**Клик – Только ссылки**»;

● Триггер: активации – «**Все клики по ссылкам**».

Сохраняем триггер.

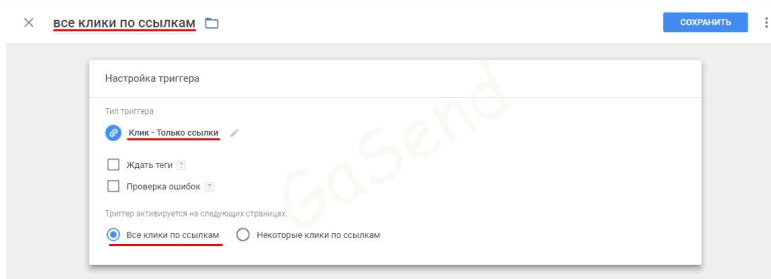


Рис. 439. Создание триггера типа «Клик – Только ссылки»

2. Создаем триггер для прослушивания отправок форм (**gtm.formSubmit**). Называем его «**Отправка всех форм**».

- Тип триггера – «Отправка формы»;
  - Триггер активации – «Все формы».
- Сохраняем триггер.

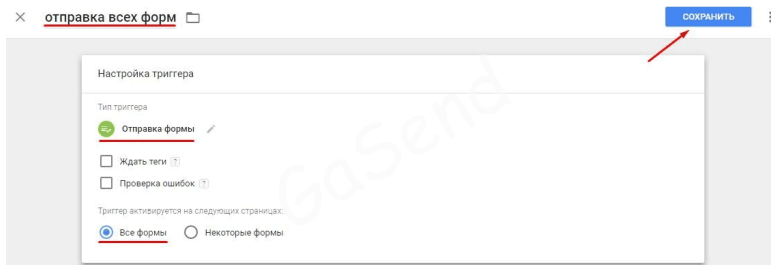


Рис. 440. Создание триггера типа «Отправка формы»

После настройки триггеров переходим в режим отладки. И видим, что теперь нам доступны наши события **gtm.click**, **gtm.linkClick**, **gtm.formSubmit**.

← → C Надежный | https://gasend.com/trenazhor-dlya-vnedreniya-analitiki/ ☆

**GASEND**

ГЛАВНАЯ ОСТАВИТЬ ЗАЯВКУ ОБРАТНЫЙ

# Тренажёр для внедрения аналитики

Google Tag Manager Tags Variables Data Layer QUICK\_PREVIEW

28 gtm.click  
27 gtm.linkClick  
26 gtm.click  
25 gtm.click  
24 gtm.click  
23 gtm.click  
22 gtm.click  
21 gtm.click  
20 gtm.formSubmit  
19 gtm.click  
18 gtm.click

## gtm.formSubmit

Variable	Variable Type	Return Type	Value
<u>_event</u>	Пользовательские события	string	'gtm.formSubmit'
<u>_triggers</u>	Переменная уровня данных	string	'7860511_24.7860611_29'
<u>Click Classes</u>	Переменная уровня данных	string	'form-validate'
<u>Click Element</u>	Переменная уровня данных	object	[object HTMLFormElement]
<u>Click ID</u>	Переменная уровня данных	string	'GTM_vopros'
<u>Click Target</u>	Переменная уровня данных	string	'-'
<u>Click Text</u>	Auto-Event Variable	string	'Введите имя Введите телефон * Введите email * Введите сайт Введите Ваш вопрос акая на кнопку, вы даете согласие на обработку своих персональных данных и лщается с политикой конфиденциальности Оставить заявку'
<u>Click URL</u>	Переменная уровня данных	string	'-'

Рис. 441. События gtm.click, gtm.linkClick, gtm.formSubmit

С этого момента мы можем настраивать отслеживание нужных нам событий, привязываясь к соответствующим значениям переменных.

# Отслеживание кликов с помощью GTM

Рассмотрим настройки на примере клика по кнопке «**Обратный звонок**».

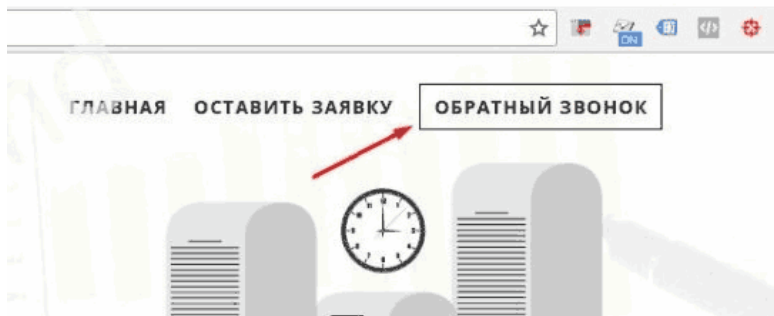


Рис. 442. Отслеживание кнопки «Обратный звонок»

Последовательность действий:

1. Настраиваем прослушивание кликов;
2. Переходим в пункт меню «**Переменные**», нажимаем на кнопку «**Настроить**» и активируем соответствующие переменные:

## × Конфигурация встроенных переменных ?

Error URL

Error Line

Debug Mode

### Клики

Click Element

Click Classes

Click ID

Click Target

Click URL

Click Text

Рис. 443. Активация встроенных переменных типа «Клики»

3. Активируем режим предварительного просмотра:

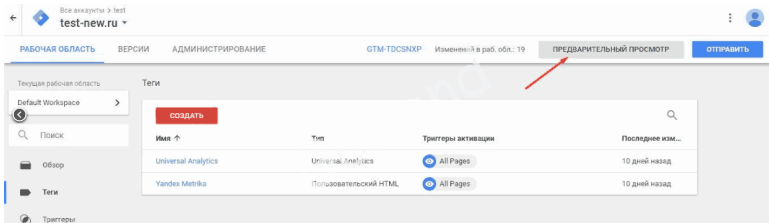


Рис. 444. Активация режима предварительного просмотра

4. Переходим на необходимую страницу сайта и обновляем ее для активации режима отладки

5. Далее кликаем по элементу, для которого хотим настроить отслеживание целей. В нашем случае это будет кнопка «Обратный звонок». Видим, что зафиксировалось соответствующее событие **gtm.click**:

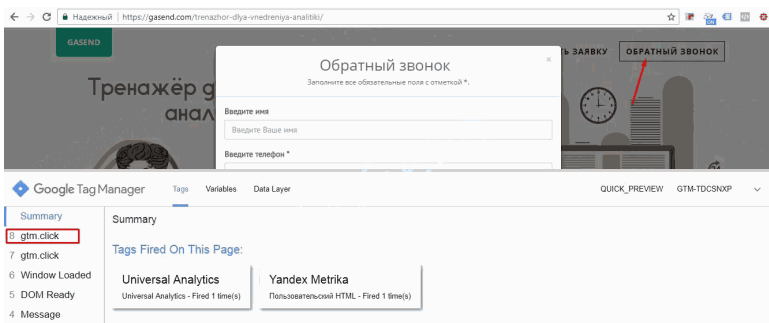
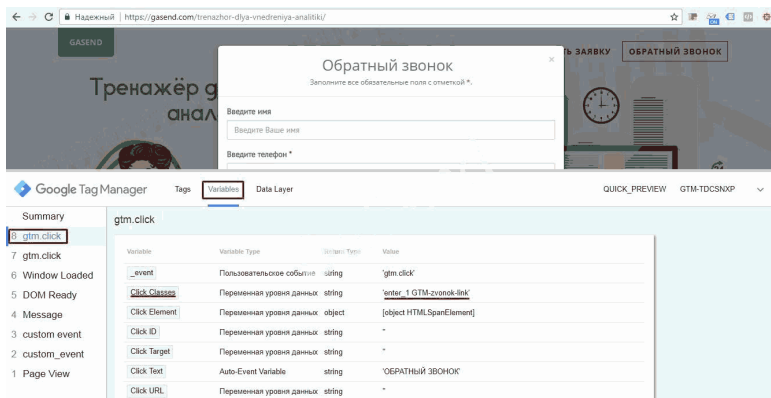


Рис. 445. Зафиксированное событие gtm.click

6. Выбираем событие **gtm.click** и переходим на вкладку «**Variables**». Данному событию соответствуют несколько переменных. В нашем случае, для дальнейших настроек копируем значение переменной **Click Classes**.

**Примечание:** если по каким-то причинам у вас нет необходимой переменной при клике по кнопке, то обратитесь к вашему программисту, чтобы он ее добавил, или же используйте CSS-селекторы.



The screenshot shows the Google Tag Manager interface. At the top, there is a browser window displaying a form titled "Обратный звонок" (Callback) with fields for name, phone number, and a "ОБРАТНЫЙ ЗВОНОК" button. Below the browser, the Google Tag Manager interface is visible. The "Tags" tab is selected, and the "gtm.click" event is chosen. The "Variables" sub-tab is active, showing a list of variables. The "Click Classes" variable is highlighted, and its value, "enter\_1 GTM-zvonok-link", is copied. The interface also shows a "Summary" panel on the left with a list of events, including "gtm.click", "Window Loaded", "DOM Ready", "Message", "custom\_event", "custom\_event", and "Page View".

Variable	Variable Type	Output Type	Value
._event	Пользовательское событие	string	'gtm.click'
<b>Click Classes</b>	Переменная уровня данных	string	'enter_1 GTM-zvonok-link'
Click Element	Переменная уровня данных	object	{object HTMLSpanElement}
Click ID	Переменная уровня данных	string	-
Click Target	Переменная уровня данных	string	-
Click Text	Auto-Event Variable	string	'ОБРАТНЫЙ ЗВОНОК'
Click URL	Переменная уровня данных	string	-

Рис. 446. Копируем Click Classes элемента

7. Переходим к настройкам GTM. В первую очередь создаем соответствующий триггер, который будет принимать истинное значение, когда будет происходить клик по кнопке «**Обратный звонок**». В качестве условия активации

триггера задаем ранее скопированное значение переменной **Click Classes**. Для этого необходимо задать следующие настройки:

- Тип триггера – **«Клик – Все элементы»**
- Триггер: активации – **«Некоторые клики»**
- Click Classes равно **enter\_1 GTM-zvonok-link**

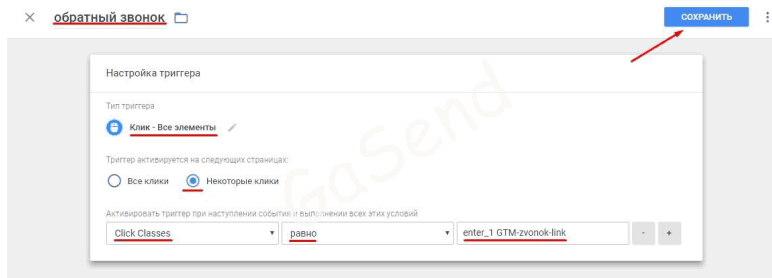


Рис. 447. Условие активации триггера по некоторым кликам

8. Создаем тег, который будет передавать события в Google Analytics. Называем тег таким образом, чтобы было понятно за что он отвечает. В качестве типа тега указываем **«Universal Analytics»**.

- Тип тега – **«Universal Analytics»**
- Тип отслеживания – **«Событие»**

Задаем значения переменных, которые будут передаваться в Google Analytics при нажатии на кнопку. Не обязательно





заполнять все поля. Они нужны для более детального анализа в отчетах Google Analytics.

× **клик по кнопке обратный звонок** □

---

Тип тега


 **Universal Analytics**  
Google Analytics 

Тип отслеживания


Событие ▾

Параметры отслеживания событий


Категория

клик 


Действие

обратный звонок 

Ярлык

\_\_\_\_\_ 

Значение

\_\_\_\_\_ 

Не взаимодействие

False ▾

Рис. 448. Настройки тега Universal Analytics

9. Указываем идентификатор отслеживания. В данном примере мы его создали через переменную.

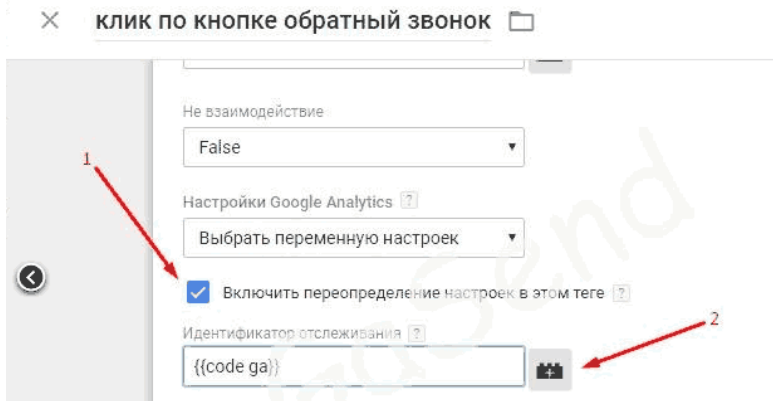


Рис. 449. Включить переопределение настроек в этом теге и добавить переменную отслеживания

## Создание переменной `{{code ua}}`

Значение кода счетчика (идентификатора отслеживания) берем с «**Google Analytics – Ресурс – Код отслеживания**».

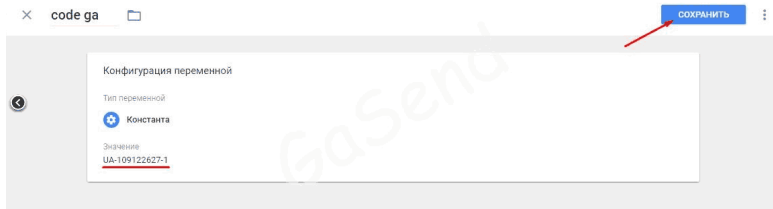


Рис. 450. Создание пользовательской переменной с кодом

GA

10. Возвращаемся к настройкам тега. В качестве триггера активации выбираем ранее созданный триггер «**обратный звонок**» и сохраняем.

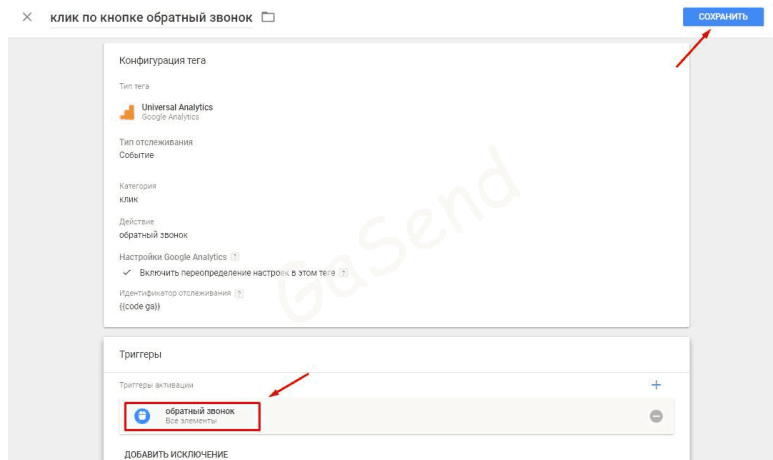


Рис. 451. Условие активации тега

11. Теперь переходим к созданию тега для отслеживания клика по кнопке в Яндекс.Метрика. Называем тег таким образом, что бы потом можно было его отличить от тега для GA. В качестве типа тега задаем «**Пользовательский HTML**».

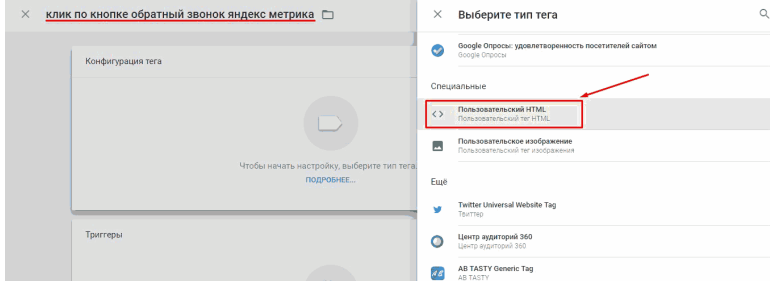


Рис. 452. Создание пользовательского HTML-тега

В поле HTML укажем следующий код:

```
<script>yaCounter46519032.reachGoal('obratni_zvonok',  
script>
```

где:

- 46519032 – номер вашего счетчика в Яндекс.Метрика;
- obratni\_zvonok – идентификатор цели, который в дальнейшем будет прописан в настройках цели в самой Яндекс.Метрика.

В качестве триггера указываем ранее созданный в п. 7 триггер «**обратный звонок**» и сохраняем. Один триггер будет активировать два тега. Поэтому нет необходимости создавать для каждого тега отдельный триггер.

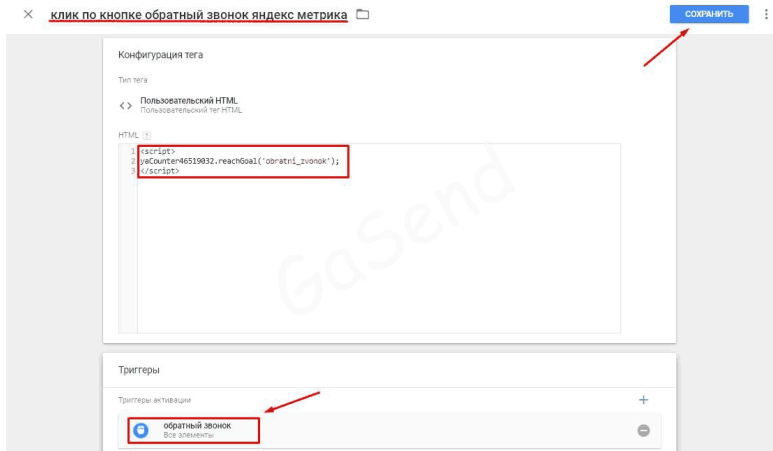


Рис. 453. Условие активации тега «обратный звонок»

12.Теперь перейдем к настройке целей. Начнем с Яндекс.Метрика. Переходим в «**Настройки – Цели**».

test-new.ru ▾ gasend.com · 46519032

## Настройка

Основное Код счётчика Вебвизор **Цели** Фильтры 1 Уведомления Загрузка данных

Вы можете определить 200 целей для каждого сайта, указав:

- какое количество просмотренных страниц является показателем успешности рекламной кампании;
- один или несколько URL, посещение которых является желаемым действием.

Название цели

Описание

Цели отсутствуют.

Рис. 454. Настройки – Цели

Указываем название цели. «Тип условия» выбираем «**JavaScript-событие**». В качестве идентификатора цели задаем то значение, которое указали в предварительно созданном теге GTM для Яндекс.Метрика в п. 11. Сохраняем цель, нажав на кнопку «**Добавить цель**».

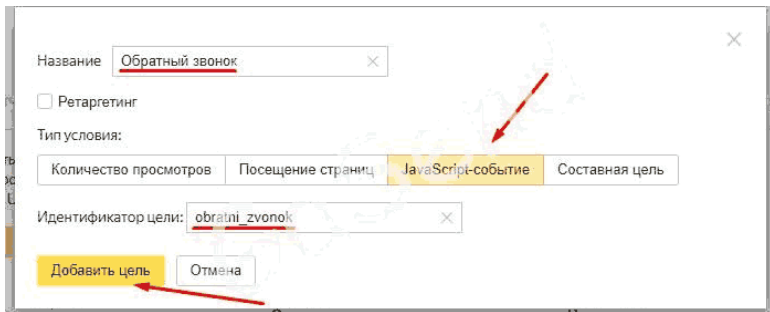


Рис. 455. JavaScript-событие

13. Создадим цель в Google Analytics:

- переходим в меню «Цели» на уровне «Представление»;
- нажимаем на кнопку «+Цель»;
- выбираем пункт «Собственная» и нажимаем «Далее»;
- указываем название цели, в качестве типа выбираем «Событие» и нажимаем «Далее»;
- в качестве условий берем те значения, которые указали в соответствующем теге (см. Рис. )

### 3. Подробные сведения о цели

#### Условия события

Задайте одно или несколько условий. Конверсия будет засчитываться, если произошло определенное событие и при этом выполнены все заданные вами условия. Для цели этого типа необходимо настроить по крайней мере одно событие.

[Подробнее...](#)

<u>Категория</u>	Равно ▾	<u>клик</u>
<u>Действие</u>	Равно ▾	<u>обратный звонок</u>
Ярлык	Равно ▾	Ярлык
Ценность	Больше чем ▾	Ценность

#### Использовать ценность события в качестве ценности цели конверсии

ДА

Если в условии выше не задано значение, соответствующее вашему коду отслеживания событий, ценность цели не будет указана.

Проверка цели последние 7 дней. Узнайте, сколько конверсий, соответствующих этой цели, можно получить, на основе данных за последние 7 дней.

Рис. 456. Задаем категорию и действие в Google Analytics

Сохраняем. Теперь цель создана и будет фиксировать клики по кнопке «**Обратный звонок**». После проверки целей (сделать это можно с помощью отчетов «В режиме реального времени») необходимо опубликовать контейнер в GTM, чтобы все изменения были активированы для посетителей сайта.



# Как задать ценность покупки для пикселя Facebook

Для отслеживания покупок на сайте (например, страница «Спасибо за заказ») в рекламной системе Facebook предусмотрено соответствующее стандартное событие:

```
<script>  
fbq('track', 'Purchase', {currency: 'USD', value: 0.00});  
</script>
```

Рис. 457. Код события «Покупка»

Необходимо, чтобы этот скрипт срабатывал после базового пикселя Facebook во время осуществления транзакции.

```
<!-- Facebook Pixel Code -->
<script>
!function(f,b,e,v,n,t,s){if(f.fbq)return;n=f.fbq=function(){n.callMethod?
n.callMethod.apply(n,arguments):n.queue.push(arguments)};if(!f._fbq)f._fbq=n;
n.push=n;n.loaded=!0;n.version='2.0';n.queue=[];t=b.createElement(e);t.async=!0;
t.src=v;s=b.getElementsByTagName(e)[0];s.parentNode.insertBefore(t,s)}(window,
document,'script','https://connect.facebook.net/en_US/fbevents.js');
// Insert Your Custom Audience Pixel ID below.
fbq('init', '<FB_PIXEL_ID>');
fbq('track', 'PageView');
</script>
<!-- Insert Your Custom Audience Pixel ID below. -->
<noscript></noscript>
<!-- End Facebook Pixel Code -->
```

Базовый пиксель Facebooke

```
<script>
// Assuming a single item is purchased
fbq('track', 'Purchase', {
  value: 199.50,
  currency: 'USD'
});
</script>
```

Пиксель транзакции Facebooke

Рис. 458. Срабатывание события «Покупка» после базового пикселя

Переменная «**currency**» отвечает за то, в какой валюте осуществляется транзакция. Просто указываем наименование валюты согласно специального классификатора.

А вот как быть с переменной «**value**»? Ведь сумма покупки в отличии от валюты очень часто меняется. Эта переменная динамическая и должна соответствовать сумме покупке.

В этом примере мы расскажем как задать «**value**» в том случае, если у вас на сайте настроена электронная торговля

Google Analytics и установлен код Google Tag Manager.

Допустим, у вас уже установлен на странице «*Спасибо за заказ*» код e-commerce, который передает данные в Google Analytics. В нашем случае он выглядит следующим образом:

```
1 <script>
2 window.dataLayer = window.dataLayer || [];
3 dataLayer.push({
4   'ecommerce': {
5     'currencyCode': 'USD',
6     'purchase': {
7       'actionField': {
8         'id': 'ER444',
9         'affiliation': 'Site',
10        'revenue': '120',
11        'tax': '12',
12        'shipping': '15'
13      },
14      'products': [{
15        'name': 'Product 1',
16        'id': 'YU112',
17        'price': '60',
18        'brand': 'Good',
19        'category': 'Category 1',
20        'variant': '44',
21        'quantity': 2,
22      }]
23    }
24  },
25  'event': 'gtm-e-commers',
26  'gtm-e-commers': 'Ecommerce',
27  'gtm-e-commers': 'Purchase',
28  'gtm-e-commers': false,
29 });
30 </script>
```

Рис. 459. Код e-commerce

Теперь создадим специальную переменную в Google Tag Manager, которая будет принимать значение, указанное в переменной «**revenue**», в которую передается сумма заказа.

Переходим в GTM в раздел «**Переменные**» и нажимаем кнопку «**Создать**».

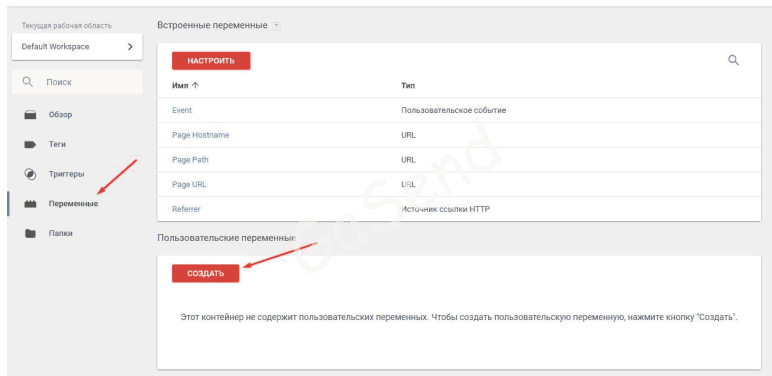


Рис. 460. Создание пользовательской переменной

Указываем название переменной и выбираем тип переменной «**Переменная уровня данных**». В качестве имени переменной указываем:

**ecommerce.purchase.actionField.revenue**

и сохраняем.

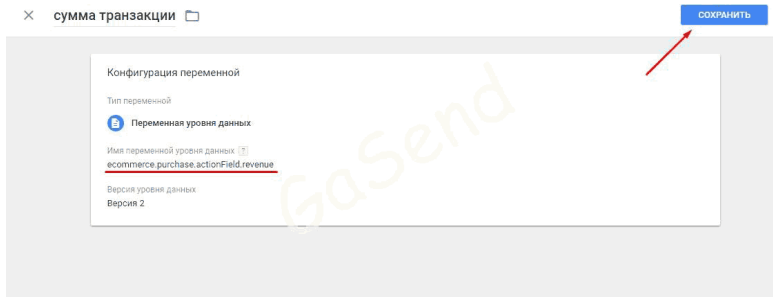


Рис. 461. Имя переменной уровня данных – `ecommerce.purchase.actionField.revenue`

Указанное имя переменной – это элемент массива, в котором находится сумма транзакции.

```
1 <script>
2 window.dataLayer = window.dataLayer || [];
3 dataLayer.push({
4   'ecommerce': {
5     'currencyCode': 'USD',
6     'purchase': {
7       'actionField': {
8         'id': 'ER444',
9         'affiliation': 'Site',
10        'revenue': '120',
11        'tax': '12',
12        'shipping': '15'
13      },
14      'products': [{
15        'name': 'Product 1',
16        'id': 'YU112',
17        'price': '60',
18        'brand': 'Good',
19        'category': 'Category 1',
20        'variant': '44',
21        'quantity': 2,
22      }]
23    }
24  },
25  'event': 'gtm-e-commers',
26  'gtm-e-commers': 'Ecommerce',
27  'gtm-e-commers': 'Purchase',
28  'gtm-e-commers': false,
29  });
30 </script>
```




Рис. 462. `ecommerce.purchase.actionField.revenue` – элемент массива

Создаем тег, который будет отправлять данные о продажах в ваш аккаунт Facebook. Называем тег, и выбираем соответствующий тип тега **«Пользовательский HTML»**. Вставляем наш стандартный пиксель. Но только еще дополнительно в качестве значения переменной указываем ранее созданную нами переменную **{{сумма транзакции}}**, вот в

таким виде:

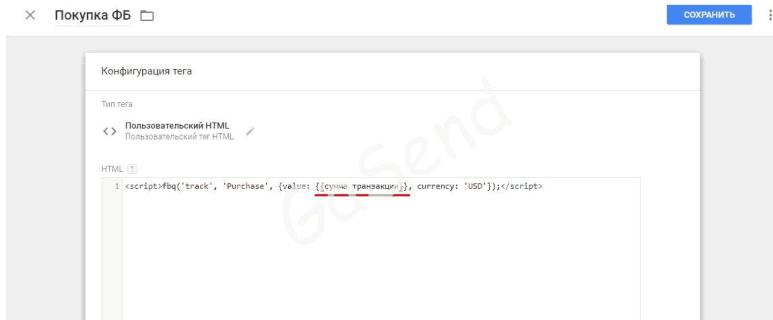


Рис. 463. Переменная `{{сумма транзакции}}` в событии транзакции Facebook

Так же задаем порядок активации тегов. Необходимо, чтобы пиксель транзакции срабатывал после базового пикселя. Ставим галочку «**Активировать тег после тега Покупка ФБ**».

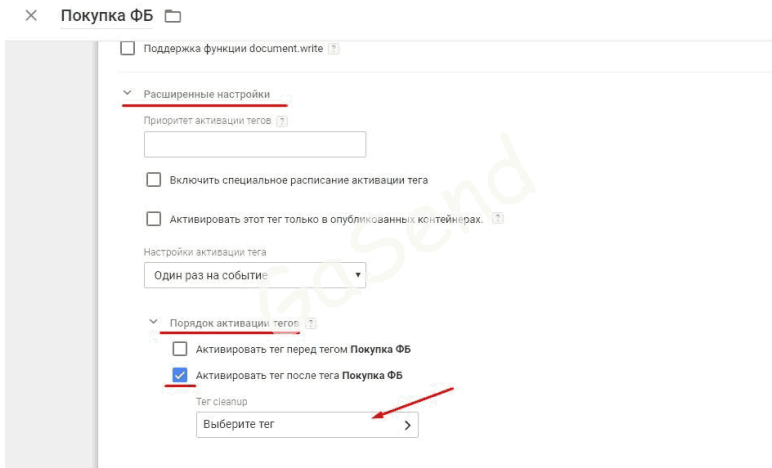


Рис. 464. Активировать тег после тега Покупка ФБ

Выбираем нужный нам тег (Базовый пиксель ФБ) и ставим еще одну галочку:

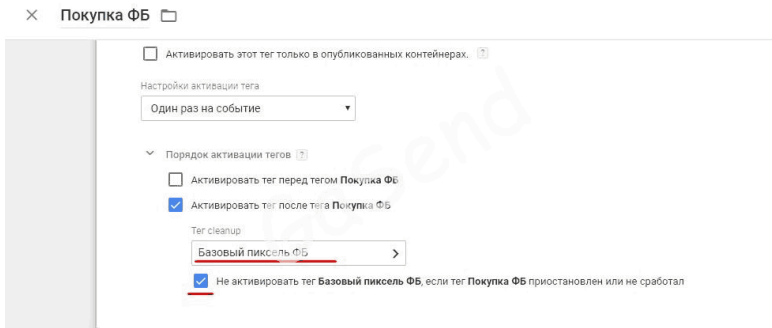




Рис. 465. Не активировать тег «Базовый пиксель ФБ», если тег «Покупка ФБ»

приостановлен или не сработал

Затем задаем триггер и сохраняем. Нужно указать тот триггер, который активируется при срабатывании кода электронной торговли.

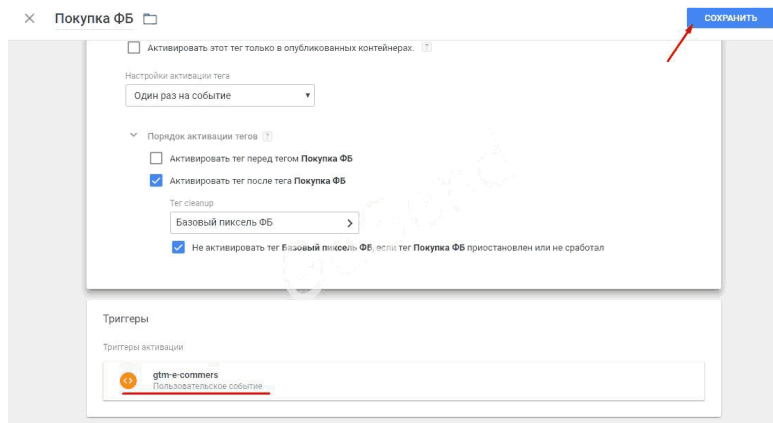


Рис. 466. Триггер активации gtm-e-commerz (событие электронной торговли)

**Примечание:** событие gtm-e-commerz было создано на этапе настройки электронной торговли и в данной статье его добавление в GTM опускается.

Теперь осталось опубликовать контейнер, чтобы данные о

транзакциях начали собираться в Facebook.

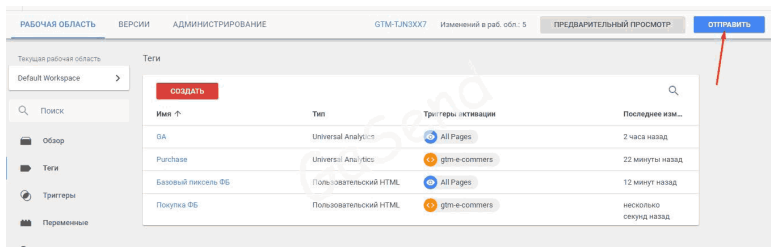


Рис. 467. Публикация контейнера

В скором времени вы сможете видеть статистику по достигнутым транзакциям, а также сумма транзакций будет отображаться в вашем аккаунте.

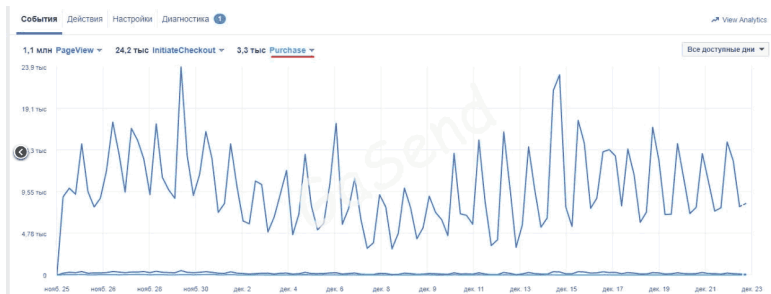
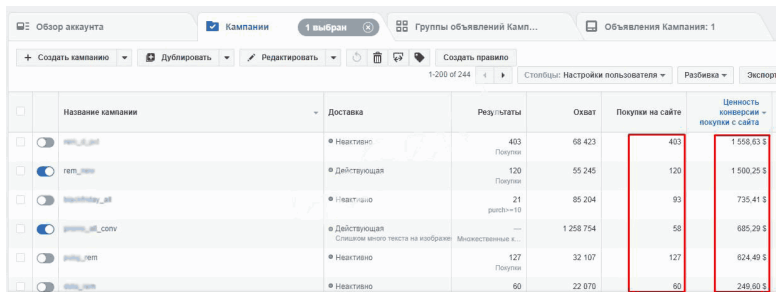


Рис. 468. Статистика по сумме транзакций в аккаунте Facebook

Данная информация позволит оптимизировать ваши ре-

кламные кампании. Вы будет знать насколько они эффективны в денежном эквиваленте.



	Название кампании	Доставка	Результаты	Охват	Покупки на сайте	Ценность конверсии - покупки с сайта
<input type="checkbox"/>	new_0_gpt	● Неактивно	403 Покупки	68 423	403	1 558,63 \$
<input type="checkbox"/>	get_new	● Действующая	120 Покупки	55 245	120	1 500,25 \$
<input type="checkbox"/>	weektoday_all	● Неактивно	21 reach=10	85 204	93	735,41 \$
<input type="checkbox"/>	promo_all_conv	● Действующая	Словом много текста на изображ... Множественные в...	1 258 754	58	685,29 \$
<input type="checkbox"/>	promo_get	● Неактивно	127 Покупки	32 107	127	624,49 \$
<input type="checkbox"/>	promo_get	● Неактивно	60	22 870	60	249,60 \$

Рис. 469. Покупки на сайте в аккаунте Facebook

# Передача Client ID в Google Analytics через Google Tag Manager

В книге «**Google Analytics для googлят: Практическое руководство по веб-аналитики. Издание 2**» вы научились передавать данные уникального идентификатора клиента (Client ID) в Google Analytics в виде пользовательского параметра без GTM. В этой статье научимся передавать Client ID в GA через Google Tag Manager.

Но перед тем как это делать, давайте разберемся для чего идентификатор клиента вообще отслеживать и передавать.

Применение Client ID, в первую очередь, служит для более точной идентификации пользователей. Также следует отметить, что этот параметр является обязательным при передаче данных в Google Analytics через Measurement Protocol, данных об оплатах из CRM и т.д.

Практическое применение в отчетах Google Analytics:

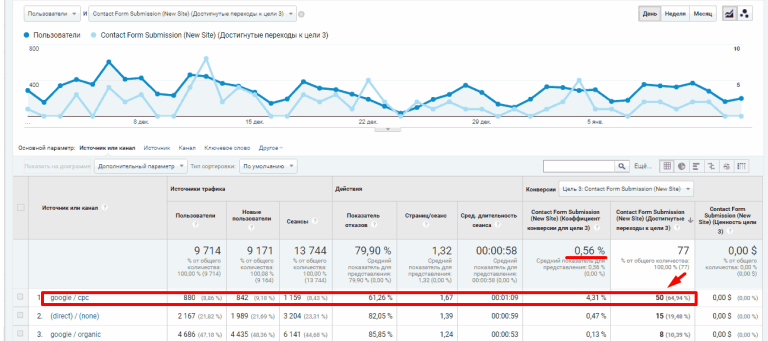


Рис. 470. Отчет «Источники или канал»

Обычный отчет «Источники или канал». Конверсия по google / src составляет 4,31% (50 конверсий), в то время как средний показатель по всем сессиям 0,56% (77 конверсий).

У нас есть данные по Client ID. Добавим его в качестве дополнительного параметра и посмотрим еще раз на отчет:

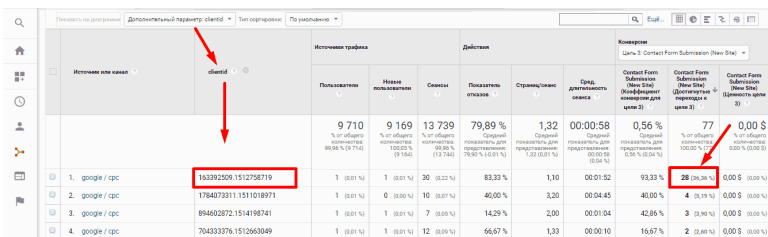


Рис. 471. Отчет «Источники или канал» с дополнительным параметром Client ID

Мы видим, что один пользователь в течение определенного периода постоянно оставляет заявки. Это могут быть продажи конкурентов, ботский трафик, нецелевые заявки или же Client ID вашего менеджера, который заполняет формы на сайте вручную. Учитывая данный фактор, конверсия по этому источнику будет совершенно другая.

Настройка будет состоять из двух частей – в Google Analytics и Google Tag Manager.

В Google Analytics нужно создать пользовательский параметр. Для этого заходим в «Администратор» и на уровне «Ресурс» выбираем «Пользовательские определения» – «Пользовательские параметры» – «+ Специальный Параметр».

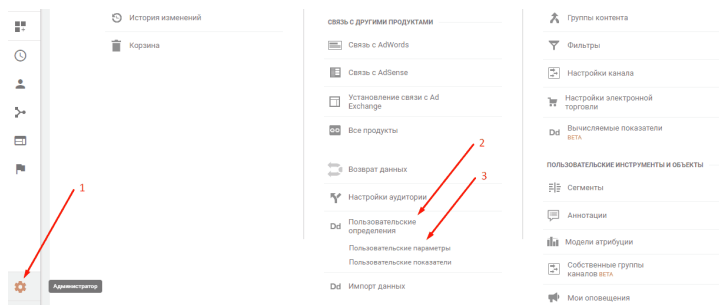


Рис. 472. Создание пользовательского параметра в Google Analytics

Введите:

- название **clientID**
- область действия «**Пользователь**»
- активная «**ОК**»

**Изменение пользовательского параметра**

Название: clientID

Область действия: Пользователь

Активная:

[Отправить](#) [Отмена](#)

**Примеры кода для этого параметра**

Скопируйте приведенный ниже фрагмент кода для своей платформы. ОБЯЗАТЕЛЬНО замените значение параметра dimensionValue на собственное.

**JavaScript (только для ресурсов, на которых используется Universal Analytics)**

```
var dimensionValue = 'SOME_DIMENSION_VALUE';
ga('set', 'dimension1', dimensionValue);
```

**SDK для Android**

```
String dimensionValue = "SOME_DIMENSION_VALUE";
tracker.set(Fields.customDimension(1), dimensionValue);
```

**SDK для iOS**

```
NSString *dimensionValue = @"SOME_DIMENSION_VALUE";
[tracker set:[GAIFields customDimensionForIndex:1] value:dimensionValue];
```

Рис. 473. Настройки пользовательского параметра

На этом настройка в Google Analytics закончилась. Переходим в Google Tag Manager. Добавьте код счетчика GA в GTM. Для этого создаем одну переменную и один тег.

- Заходим в «**Триггеры**» – «**Пользовательские переменные**» – «**Создать**»;

- Называем переменную (в примере **cod ua**)

- Тип переменной – «**Константа**»

В значение прописываем код счетчика Google Analytics и сохраняем настройки.

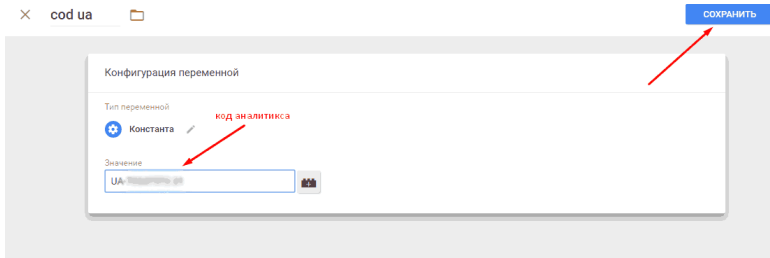


Рис. 474. Создание переменной-константы счетчика GA

Обычный отчет «**Источники или канал**». Конверсия по google / cpc составляет 4,31% (50 конверсий), в то время как средний показатель по всем сеансам 0,56% (77 конверсий).

У нас есть данные по Client ID. Добавим его в качестве дополнительного параметра и посмотрим еще раз на отчет:

Источники или канал	Источники трафика			Действия			Конверсии			
	Посетителей	Новые пользователи	Сеансы	Показатель оттока	Страниц/сеанс	Сред. длительность сеанса	Средний показатель конверсии для целевого действия	Средний показатель конверсии для целевого действия (Новый пользователь)	Средний показатель конверсии для целевого действия (Новый пользователь)	Средний показатель конверсии для целевого действия (Новый пользователь)
Источники или канал	9 710 % от общего количества: 80,06% (9 714)	9 169 % от общего количества: 102,02% (9 164)	13 739 % от общего количества: 99,88% (13 744)	79,89% Средний показатель оттока для представления: 79,91% (0,01%)	1,32 Средний показатель для представления: 1,32 (0,01%)	00:00:58 Средний показатель для представления: 00:00:58 (0,04%)	0,56% Средний показатель для представления: 0,56% (0,04%)	77 % от общего количества: 100,00% (77)	0,00%	0,00%
1. google / cpc	163392509.1512758719	1 (0,01%)	1 (0,01%)	30 (0,22%)	83,33%	1,10	00:01:52	93,33%	28 (36,28%)	0,00%
2. google / cpc	1784072011.1511018971	1 (0,01%)	0 (0,00%)	10 (0,07%)	40,00%	3,20	00:04:45	40,00%	4 (0,19%)	0,00%
3. google / cpc	894602872.1514198741	1 (0,01%)	1 (0,01%)	7 (0,05%)	14,29%	2,00	00:01:04	42,86%	3 (3,96%)	0,00%
4. google / cpc	704333376.1512663049	1 (0,01%)	1 (0,01%)	12 (0,09%)	66,67%	1,33	00:00:10	16,67%	2 (0,08%)	0,00%

Рис. 475. Отчет «Источники или канал» с дополнительным параметром Client ID

Мы видим, что один пользователь в течение определенного периода постоянно оставляет заявки. Это могут быть про-



делки конкурентов, ботский трафик, нецелевые заявки или же Client ID вашего менеджера, который заполняет формы на сайте вручную. Учитывая данный фактор, конверсия по этому источнику будет совершенно другая.

Настройка будет состоять из двух частей – в Google Analytics и Google Tag Manager.

В Google Analytics нужно создать пользовательский параметр. Для этого заходим в «Администратор» и на уровне «Ресурс» выбираем «Пользовательские определения» – «Пользовательские параметры» – «+ Специальный Параметр».

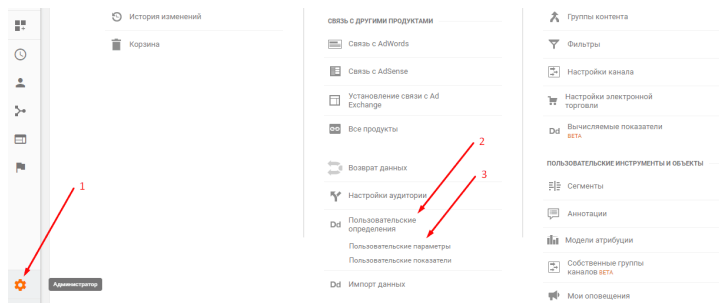


Рис. 476. Создание пользовательского параметра в Google Analytics

Введите:

- название **clientID**
- область действия «**Пользователь**»

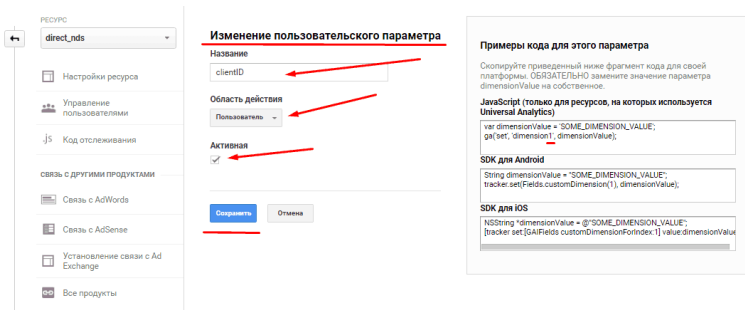


Рис. 477. Настройки пользовательского параметра

На этом настройка в Google Analytics закончена. Переходим в Google Tag Manager. Добавьте код счетчика GA в GTM. Для этого создаем одну переменную и один тег.

● Заходим в «Триггеры» – «Пользовательские переменные» – «Создать»;

● Называем переменную (в примере `cod ua`)

● Тип переменной – «Константа»

В значение прописываем код счетчика Google Analytics и сохраняем настройки.

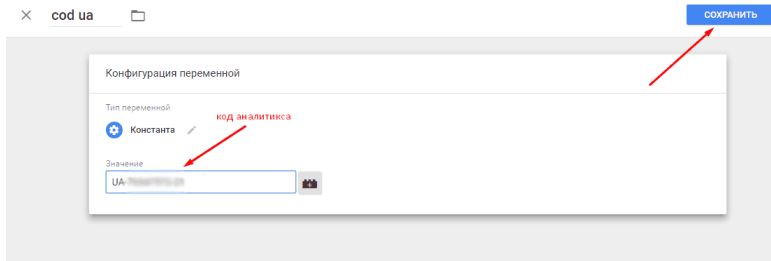


Рис. 478. Создание переменной-константы счетчика GA

Затем добавляем тег типа «Universal Analytics». Для этого пройдите путь: **Теги – Создать – Конфигурация тега (Редактировать) – Тип тега «Universal Analytics»**.

Настройки тега:

- Тип отслеживания (Просмотр страницы);
- Включить переопределение настроек в этом теге;
- В идентификатор отслеживания записываем переменную **cod ua** (создали выше);
- Триггер активации тега «**All pages (Все страницы)**».

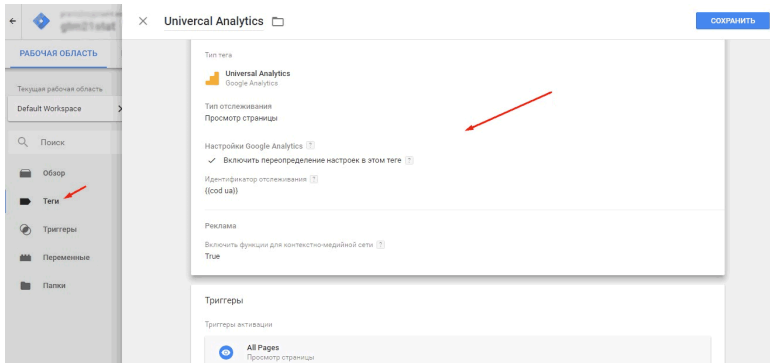


Рис. 479. Настройки тега Universal Analytics

На следующем этапе необходимо настроить передачу Client ID в GTM, создав 2 переменные, 1 триггер и 1 тег.

Создаем переменную «**GA Tracking Code**». Последовательность действий:

- Переменные – Пользовательские переменные – Создать;

- Название – **GA Tracking Code**;

- Конфигурация переменной – Редактировать;

- Тип переменной – Константа;

- Значение – Выбираем переменную **cod ua**.

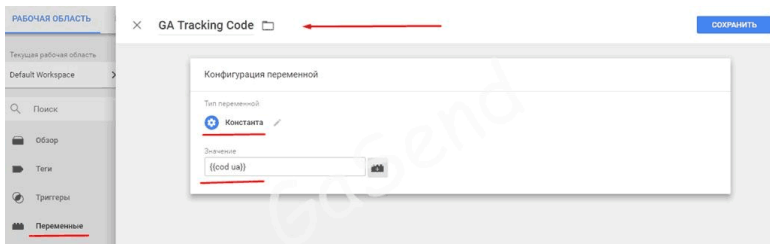


Рис. 480. Переменная типа «Константа»

Создаем переменную «**Get Client ID for current Tracker**». Последовательность действий:

- Переменные – Пользовательские переменные – Создать;
- Название – **Get Client ID for current Tracker**;
- Конфигурация переменной – Редактировать;
- Тип переменной – Собственный код JavaScript;
- Значение – Код.

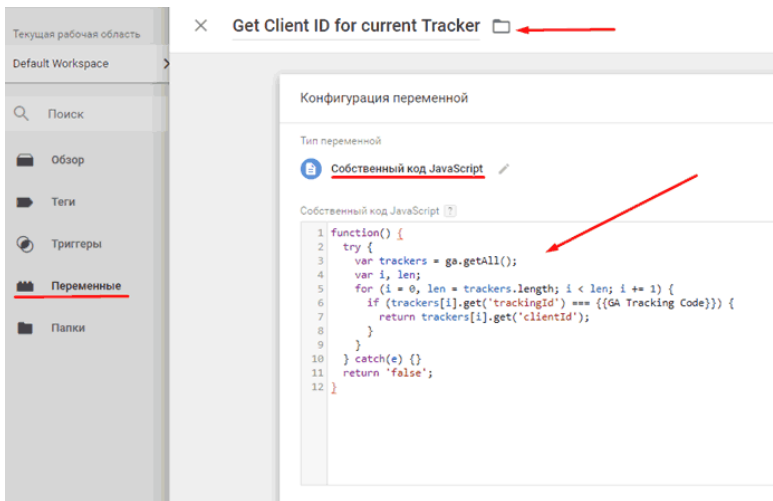


Рис. 481. Собственный код JavaScript

Пример кода JavaScript:

```
function() {  
  try {  
    var trackers = ga.getAll();  
    var i, len; for (i = 0, len = trackers.length; i < len; i +=  
1) {  
  if (trackers[i].get('trackingId') == {{GA Tracking Code}})  
{  
    return trackers[i].get('clientId'); }  
  } catch(e) {}  
  return 'false';  
}
```

В результате мы создали две пользовательские переменные – **GA Tracking Code** и **Get Client ID for current Tracker**. Переходим к созданию триггера.

- Триггеры – Создать;
- Название – **Window Loaded Trigger**;
- Тип триггера – Окно загружено;
- Триггер активируется на следующих страницах – Некоторые события «Окно загружено».

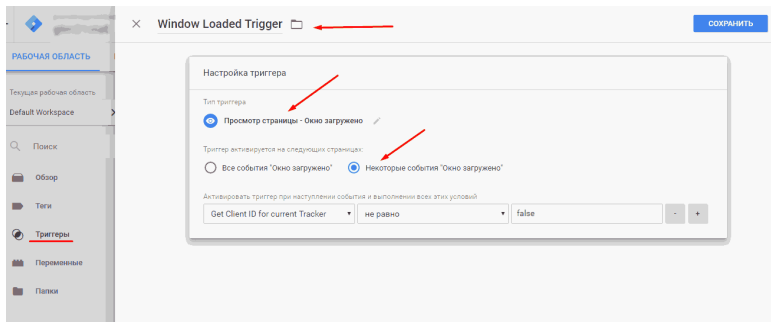


Рис. 482. Настройки триггера

Теперь создаем тег **Cid**. Последовательность действий:

- Теги – создать – Конфигурация (Редактировать);
- Тип тега – Universal Analytics;

Внимательно прописываем настройки:

- Тип отслеживания – Событие;
- Категория – **clientid**;
- Действие – **отправлен**;

## ● Не взаимодействие – True.

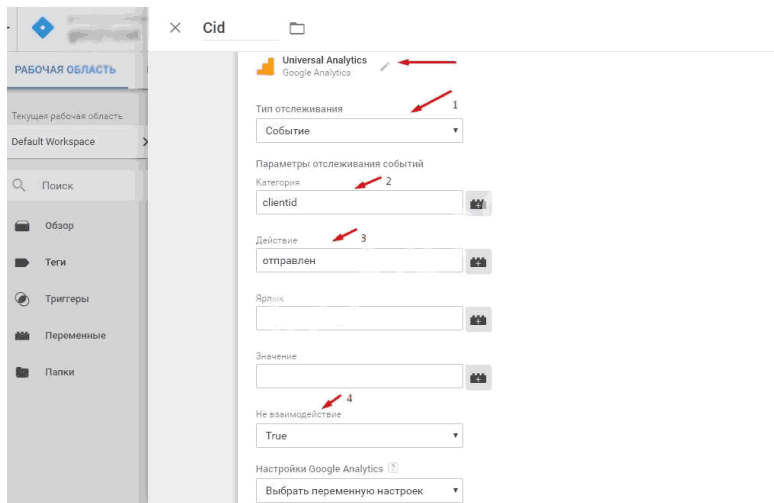


Рис. 483. Настройки тега

Еще настройки тега Universal Analytics:

- Выбираем «Включить переопределение настроек в этом теге»;
- Идентификатор отслеживания – **cod ua**;
- Дополнительные настройки – Специальные параметры – +Специальный параметр – Индекс: **1** и Значение: **Get Client ID for current Tracker** (переменная, которую мы создали выше).

**Важно:** индекс 1 – это индекс параметра, который мы со-



здали в Google Analytics.

The image shows a configuration interface for a Google Analytics tag. At the top, there is a dropdown menu set to 'True'. Below it, the text 'Настройки Google Analytics' is followed by a red arrow labeled '5' pointing to a 'Выбрать переменную настроек' dropdown. A checked checkbox 'Включить пересопределение настроек в этом теге' is followed by a red arrow labeled '6' pointing to an 'Идентификатор отслеживания' field containing '{{cod ua}}'. Under the 'Дополнительные настройки' section, the 'Специальные параметры' sub-section is expanded and underlined. A red arrow labeled '7' points to the 'Индекс' field containing '1'. Another red arrow labeled '8' points to the 'Значение параметра' field containing '{{Get Client ID for current Tracker}}'. A '+ СПЕЦИАЛЬНЫЙ ПАРАМЕТР' button is visible below the index field, and a 'Специальные показатели' section is partially visible at the bottom.

Рис. 484. Прописываем специальный параметр в теге Universal Analytics

В качестве активации тега выбираем триггер активации, ранее созданный «**Window Loaded Trigger**».

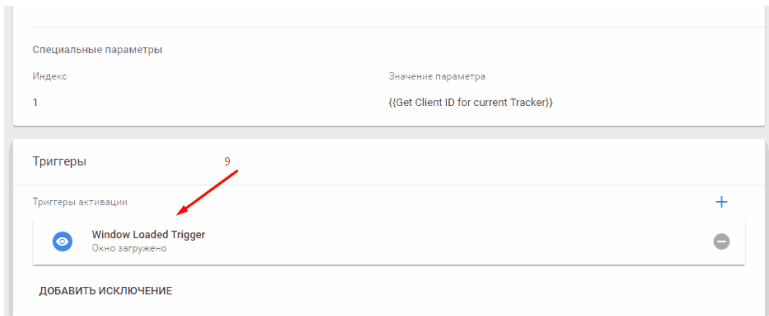


Рис. 485. Триггер активации Window Loaded Trigger

Перед тем, как опубликовать изменения, нужно проверить будет ли срабатывать тег. Переходим в режиме предварительного просмотра (режим отладки), заходим на сайт, где установлен GTM, обновляем страницу и проверяем тег:

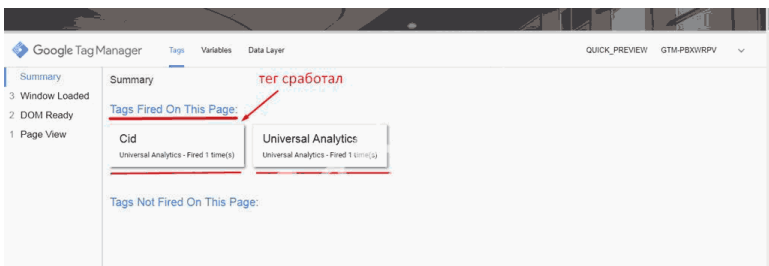


Рис. 486. Тег cid активировался

Переходим внутрь тега и видим:

	ИМЯ	VALUE
	Тип	Universal Analytics
5 gtm.timer	Не взаимодействие	true
4 gtm.click	Включить переопределение настроек в этом теге	true
3 Window Loaded	Категория	'clientId'
2 DOM Ready	Тип отслеживания	'TRACK_EVENT'
1 Page View	Действие	'отправлен'
	Специальные параметры	[[{index: '1', dimension: '1792684636.1529998546'}]]

Рис. 487. Проверка передачи данных специального параметра в режиме отладки

Также можно проверить в панели для разработчиков, установив расширение GA Debugger.

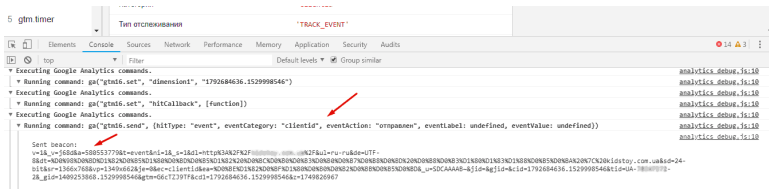


Рис. 488. Расширение GA Debugger

Если все корректно передается, публикуем контейнер GTM. Нажимаем «Отправить», вводим название и описание версии, затем «Опубликовать».

Теперь CID (Client ID) вы можете увидеть в дополнительных параметрах Google Analytics или построив специальный

(кастомный) отчет.

Источник или канал	clientid	Источники трафика			Действия			Конверсии		
		Семант.	Новые семант. %	Новые пользователи	Показатель отказов	Страниц/сеанс	Сред. длительность	Сайбэк (Коэффициент конверсии для цели S)	Сайбэк (Дополнительные переходы к цели S)	Сайбэк (Ценность цели S)
		26 037 % от общего количества: 99,88 % (26 007)	72,01 % Средний показатель для предоставления: 71,93 % (5,10 %)	18 749 % от общего количества: 99,99 % (18 751)	24,16 % Средний показатель для предоставления: 24,19 % (0,11 %)	1,67 Средний показатель для предоставления: 1,67 (0,00 %)	00:01:45 Средний показатель для предоставления: 00:01:45 (0,10 %)	0,08 % Средний показатель для предоставления: 0,08 % (0,12 %)	20 % от общего количества: 100,00 % (20)	0,00 P % от общего количества: 0,00 % (0,00 P)
1. facebook / display	1512042625.1471175723	1 (0,00 %)	0,00 %	0 (0,00 %)	0,00 %	1,00	00:07:48	100,00 %	1 (0,00 %)	0,00 P (0,00 %)
2. google / cpc	1696551259.1491809912	3 (0,01 %)	33,33 %	1 (0,01 %)	33,33 %	1,00	00:00:22	33,33 %	1 (0,00 %)	0,00 P (0,00 %)
3. google / cpc	292300728.1491200344	3 (0,01 %)	33,33 %	1 (0,01 %)	33,33 %	3,00	00:20:17	33,33 %	1 (0,00 %)	0,00 P (0,00 %)
4. google / cpc	463803033.1491731457	3 (0,01 %)	33,33 %	1 (0,01 %)	0,00 %	1,00	00:00:18	33,33 %	1 (0,00 %)	0,00 P (0,00 %)
5. google / cpc	828931775.1492011870	4 (0,02 %)	25,00 %	1 (0,01 %)	0,00 %	1,00	00:02:14	25,00 %	1 (0,00 %)	0,00 P (0,00 %)
6. google / cpc	94158056.1490347515	1 (0,00 %)	0,00 %	0 (0,00 %)	0,00 %	1,00	00:32:52	100,00 %	1 (0,00 %)	0,00 P (0,00 %)
7. Instagram / display	1016105785.1492860945	5 (0,02 %)	20,00 %	1 (0,01 %)	0,00 %	3,20	00:01:56	20,00 %	1 (0,00 %)	0,00 P (0,00 %)
8. Instagram / display	1102937262.1493377132	2 (0,01 %)	50,00 %	1 (0,01 %)	50,00 %	1,00	00:00:53	50,00 %	1 (0,00 %)	0,00 P (0,00 %)
9. Instagram / display	1851182575.1493382078	1 (0,00 %)	100,00 %	1 (0,01 %)	0,00 %	2,00	00:02:54	100,00 %	1 (0,00 %)	0,00 P (0,00 %)

Рис. 489. clientid как дополнительный параметр в Google Analytics

# Передача ошибок JavaScript сайта в Google Analytics через Google Tag Manager

Эта статья подойдет для тех, кто не сильно разбирается в JavaScript, но хочет, чтобы сайт работал на все 100% без ошибок. С помощью Google Tag Manager мы сможем настроить передачу данных об ошибках JS, которые есть на сайте, в Google Analytics. В результате вы получите пользовательский отчет с JavaScript ошибками, который можете переслать веб-разработчику для последующих исправлений.

**Примечание:** хотелось бы отметить, что GTM фиксирует не все ошибки JS на сайте. Но это лучше, чем ничего. При том, что настройка занимает около 10 минут.

В Google Tag Manager нам нужно активировать некоторые переменные, создать триггер и тег, а в Google Analytics настроить пользовательский отчет.

Переходим к настройкам в GTM. Активируем встроенные переменные:

- **Error Message** – Название ошибки JS;
- **Error URL** – url страницы, где была допущена ошибка;
- **Error Line** – номер строки, в которой была допущена ошибка.

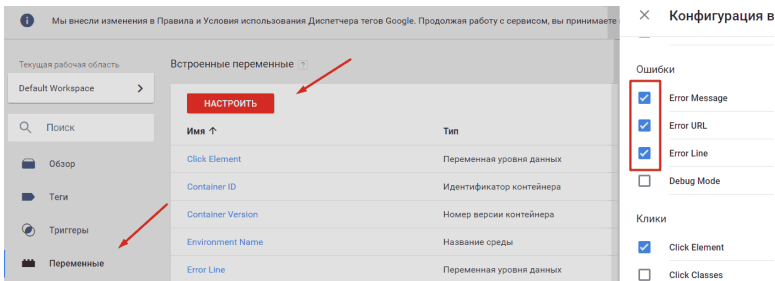


Рис. 490. Активируем встроенные переменные типа «Ошибки»

**Примечание:** не обращаем внимание на пропущенную в названиях триггера и тега букву «r» в слове «error».

Создаем триггер:

- Название – **erros js**;

- Тип триггера – Ошибка JavaScript;

- Триггер активируется на следующих страницах – Все ошибки JavaScript

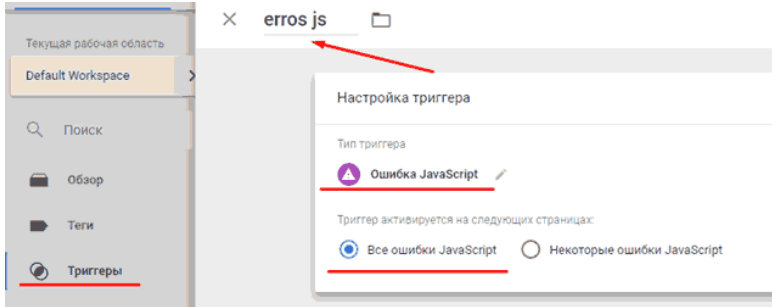


Рис. 491. Настройки триггера erros js

Создаем тег:

- Название – **Erros JS**;
- Тип тега – Universal Analytics;
- Тип отслеживания – Событие;
- Категория – **Errors js**;
- Действие – Выбираем встроенную переменную **{{Error Message}}**;

● Ярлык – Выбираем сразу две переменные: **{{Error Url}}** и в скобках сразу же прописываем **{{Error Line}}**. Именно через ярлык события в Google Analytics будет передаваться переменная url-страницы, а в скобках номер строки с ошибкой;

- Не взаимодействие – True;
- Включаем переопределение настроек в этом теге;
- Идентификатор отслеживания – **cod ua** (создаем пользовательскую переменную типа «Константа» с идентифика-

тором отслеживания Google Analytics).

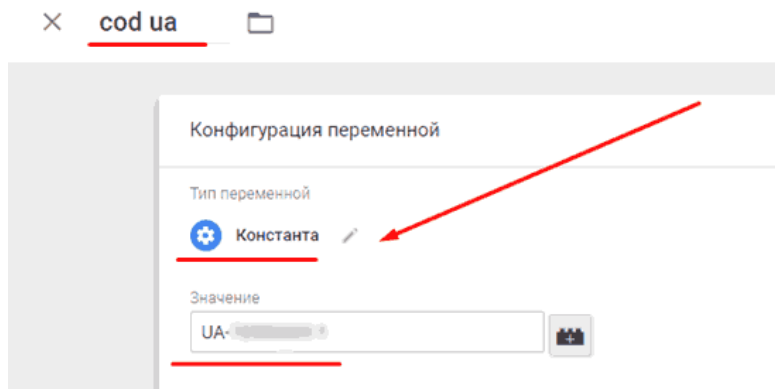


Рис. 492. Идентификатор отслеживания GA в переменной «Константа»

Триггер активации тега – **erros js** (который мы предварительно создали). Итоговые настройки тега выглядят следующим образом:



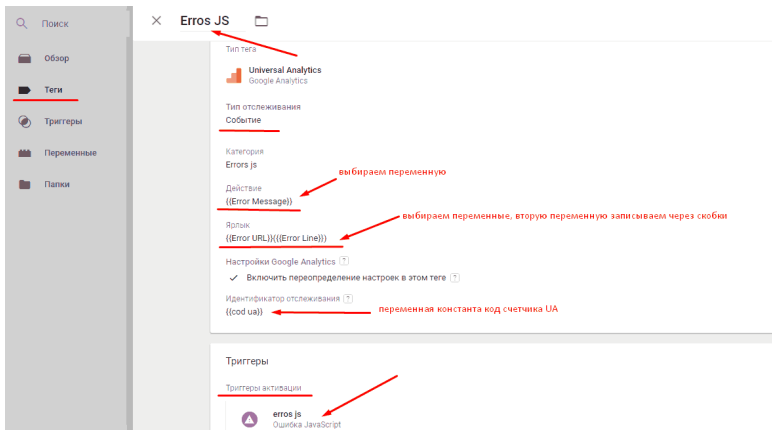


Рис. 493. Настройки тега Universal Analytics

Отправляем контейнер GTM на публикацию. Затем переходим в Google Analytics, чтобы настроить пользовательский отчет. Последовательность действий:

● Переходим в «**Специальные отчеты – Мои отчеты**»;

● Тип – Простая таблица;

● Параметры – Категория событий, Действие по событию, Ярлык события;

● Показатели – Всего событий, Уникальные события;

● Фильтр – Категория событий – точное соответствие – **Errors js**

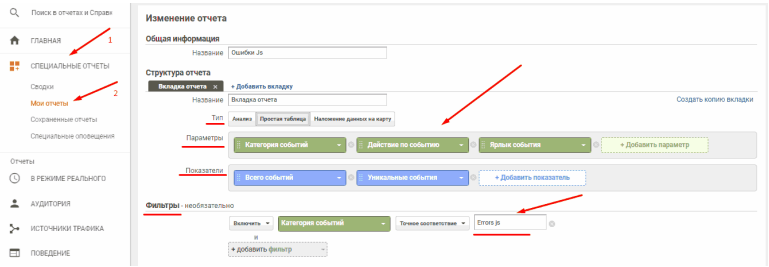


Рис. 494. Специальный отчет в Google Analytics

В результате возникновения ошибок на сайте данные в пользовательский отчет Google Analytics начнут поступать. В отчете мы видим: название ошибки, url-страницы с ошибкой и номер строки кода на странице (в скобках). Можно выгрузить этот файл в Excel и отправить на обработку.

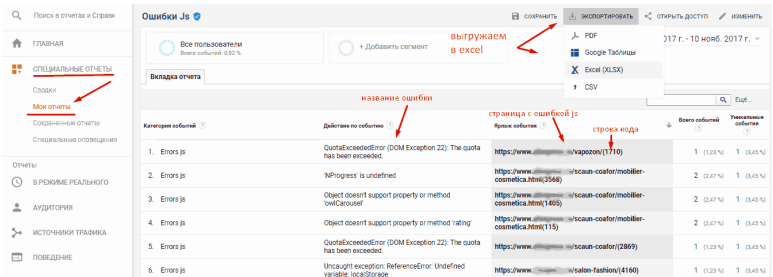


Рис. 495. Пример отчета Google Analytics с зафиксированными ошибками на сайте

# Настройка междоменного отслеживания через Google Tag Manager

Иногда бывает, что мы рекламируем один сайт (к примеру, домен *site.ru*), на который пользователи переходят по рекламе, а затем идут на второй наш сайт (к примеру, домен *anothersite.com*) и совершают конверсию.

В результате при стандартных настройках Google Analytics будет зафиксировано два сеанса:

- один на *site.ru*;
- второй на сайте *anothersite.com*.

Также конверсия будет закреплена по источнику *site.ru / referral*, и все основные показатели будут некорректными. По такой статистике нельзя сделать никаких выводов и предпринимать действий.

Чтобы данные отображались правильно, нам нужно настроить междоменное отслеживание.

**Междоменное отслеживание** – функция, позволяющая регистрировать посещение нескольких сайтов в Google Analytics как один сеанс. То есть при переходе с *siteA.ru* на *siteB.ru* будет регистрироваться один пользователь и одна сессия.

Переходим к настройке в GTM. Создаем пользователь-

скую переменную:

- Название – **Cross domain settings** (произвольное название);
- Тип переменной – Настройки Google Analytics;
- Идентификатор отслеживания: UA- XXXXXX-X (из Google Analytics);
- Домен cookie – auto;

Дополнительные настройки – Поля, которые необходимо задать:

- Название поля – **allowLinker**, Значение – **true**;
- Междоменное отслеживание:
- Автоматическое связывание доменов – *siteA.ru, siteB.ru* (вводим домены через запятую, которые хотим отслеживать).

Необязательно:

- Использовать решетку в качестве разделителя – Верно, если вы добавляете Client ID через параметр;
- Изменение внешнего вида форм – Верно, если есть форма, при отправке которой пользователя перебрасывает на второй сайт.

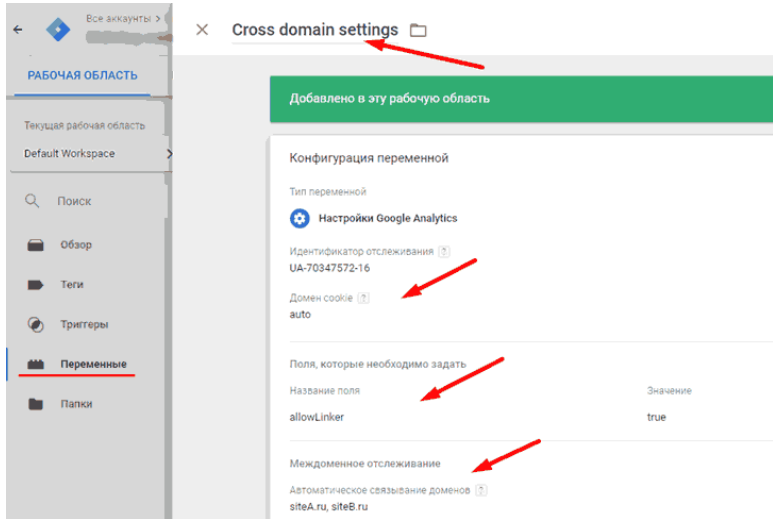


Рис. 496. Настройки переменной «Настройки Google Analytics»

Создаем тег. Если у вас настроен тег об отправке данных в Google Analytics, вам нужно изменить его конфигурацию. Если нет – создать новый.

- Название тега – Universal Analytics;
- Тип тега – Universal Analytics;
- Настройки Google Analytics – выбираем переменную **{{Cross domain settings}}**;
- Триггер активации – «All Pages (Все страницы)».

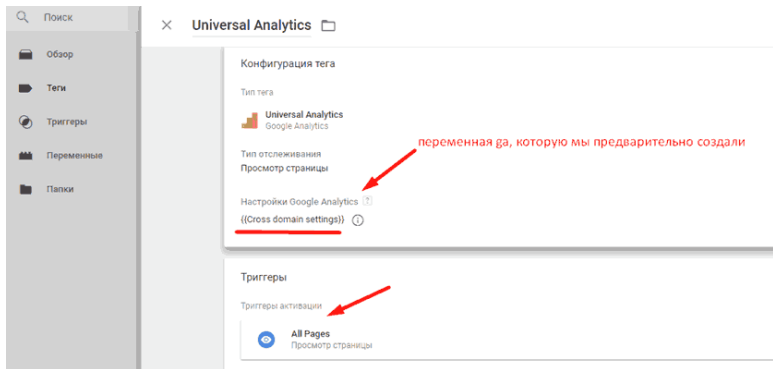


Рис. 497. Настройки тега Universal Analytics

На следующем шаге переходим в Google Analytics иключаем рефералов. Если этого не сделать, в отчетах GA 1 пользователь при переходе с сайта на сайт будет фиксироваться как два сеанса, а нам нужно 1 пользователь – 1 сеанс.

- «Ресурс – Отслеживание – Список исключаемых источников перехода»

- Исключаем два домена, между которыми мы будем настраивать междоменное отслеживание;

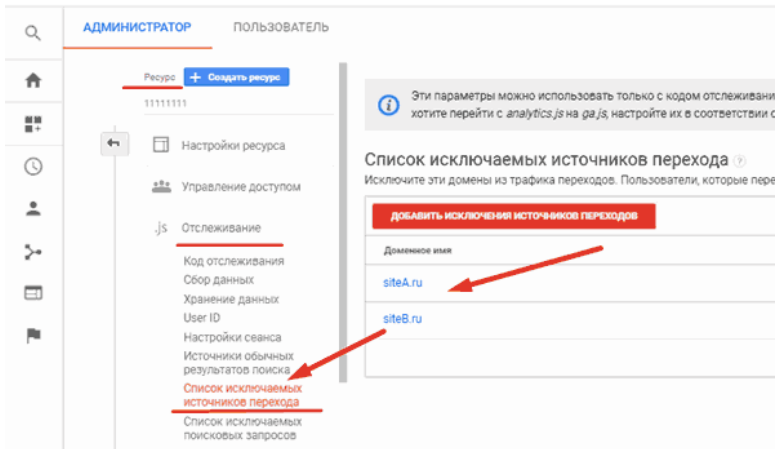


Рис. 498. Список исключаемых источников перехода

Осуществляем проверку. При переходе с одного домена на второй вы должны увидеть передачу Client ID.

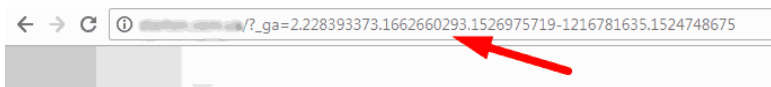


Рис. 499. Параметр связывания

Междоменное отслеживание выполняется путем передачи уникального идентификатора клиента между исходным и целевым доменом. Эта процедура выполняется в два этапа:

- В исходном домене проверяется наличие идентификатора клиента во всех URL, указывающих на целевой домен.

● В целевом домене проверяется наличие идентификатора клиента в URL, по которому перешел пользователь.

Плагин для связывания реализует этот механизм путем добавления параметра связывания в URL-адреса, указывающие на целевой домен. Этот параметр содержит идентификатор клиента, а также закодированные текущую временную метку и метаданные браузера, которые позволяют избежать проблем с передачей URL между пользователями.

Именно значение **true** в параметре **allowLinker** осуществляет проверку параметров связывания.

Для анализа данных в интерфейсе Google Analytics по страницам с названиями доменов нужно создать отдельное представление с фильтром.

Создаем представление. Добавляем фильтр через «**Фильтры – Добавить фильтр**».

- Название фильтра – Любое;
- Тип фильтра – Пользовательский и Расширенный;
- Поле A -> Извлечь A – **Имя хоста (\*.\*)**. Извлекаем название домена, имя хоста равно регулярному выражению);
- Поле B –> Извлечь B – **URI запроса (\*.\*)**. Извлекаем url-запроса, url-запроса равно регулярному выражению);
- Вывод –> Конструктор – **URI запроса (\$A1\$B1)**. Делаем склейку поля A и поля B, \$ – метасимвол регулярного выражения.

Ставим галочки напротив строк:

- Поле A обязательное для заполнения



## ● Перезаписать поле вывода Сохраняем фильтр.

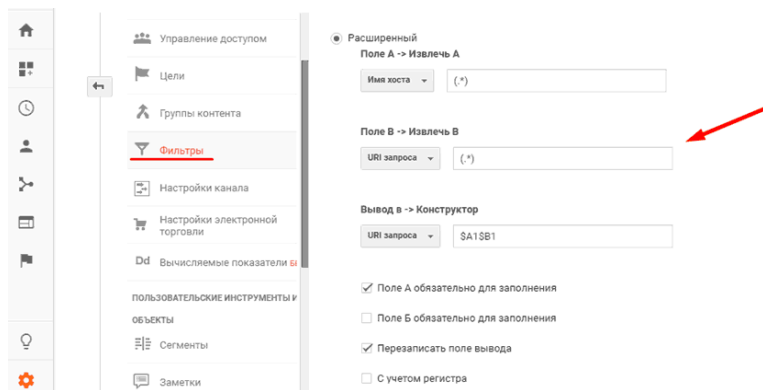


Рис. 500. Настройки фильтра Google Analytics

В результате мы сможем видеть статистику по страницам с названием домена.

Основной параметр: Страница		Название страницы		Другое ▾	
Показать на диаграмме		Дополнительный параметр: Источник или канал ▾		Расширен	
Тип сортировки: По умолчанию ▾					
<input type="checkbox"/>	Страница ?	Источник или канал ?	Просмотры страниц ?	Уникальн просмот страниц ?	
			4 % от общего количества: 16,00 % (25)	% от общ колличес: 15,79 %	
<input type="checkbox"/>	1. <a href="#">k...com.ua/testimonial/</a>	test11 / test123	2 (50,00 %)	1 (33,33 %)	
<input type="checkbox"/>	2. <a href="#">k...com.ua/</a>	test11 / test123	1 (25,00 %)	1 (33,33 %)	
<input type="checkbox"/>	3. <a href="#">s...com.ua/</a>	test11 / test123	1 (25,00 %)	1 (33,33 %)	

Рис. 501. Отчет Google Analytics по страницам

# Отслеживание прокрутки страниц

Настройка отслеживания прокрутки страниц даст вам информацию о том, как пользователи просматривают ваш контент. Настроить данное отслеживание через Google Tag Manager не составит трудностей.

Переходим к настройке GTM. Активируем встроенные переменные **Scroll Depth Threshold**, **Scroll Depth Units** и **Scroll Directions**.

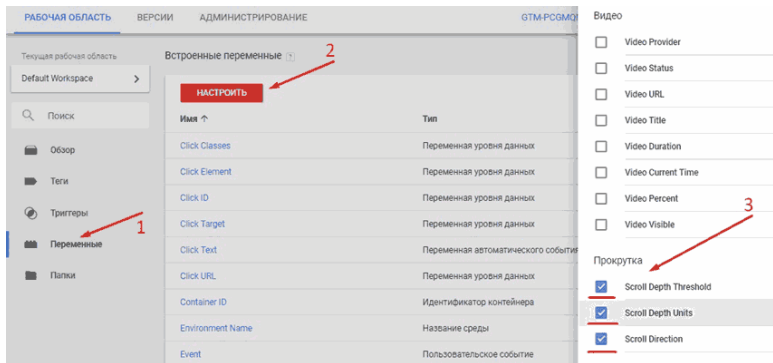


Рис. 502. Активация встроенных переменных типа «Прокрутка»

● **Scroll Depth Threshold** – значения порога, который был пересечен (в зависимости от настроек будет либо в про-

центах, либо в пикселях);

- **Scroll Depth Units** – переменная покажет пиксели (pixels) или процент (percent), в зависимости от настроек триггера;

- **Scroll Directions** – переменная покажет вертикально (vertical) или горизонтально (horizontal), в зависимости от настроек триггера. Вертикальный скролл действует только вниз, а горизонтальный только вправо.

На примере мы будем отслеживать вертикальную прокрутку по пикселям на страницах блога. Создаем триггер:

- Название – **scroll blog**;
- Тип триггера – Глубина прокрутки;
- Выбираем «Глубина вертикальной прокрутки»;
- Проценты 25,50,75,100 (числа через запятую);
- Условие активации триггера (будем активировать его на страницах блога согласно заданию выше).

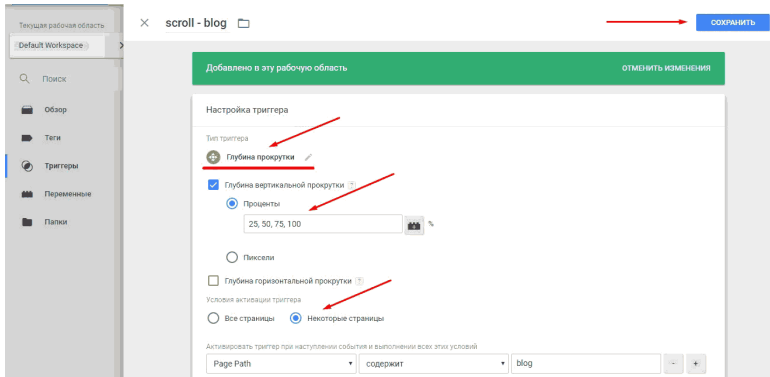


Рис. 503. Настройки триггера

Создаем тег:

- Название – **ga – scroll – blog**;

- Тип тега – Universal Analytics;

- Тип отслеживания – Событие;

- Категория – **scroll blog** (название может быть другим);

- Действие – Переменная **{{Page Path}}** (часть url-страницы);

- Ярлык – Переменная **{{Scroll Depth Threshold}}**, которую мы активировали и «%» (что бы в отчетах Google Analytics мы видели значение и %, например 25%);

- Не взаимодействие – True.

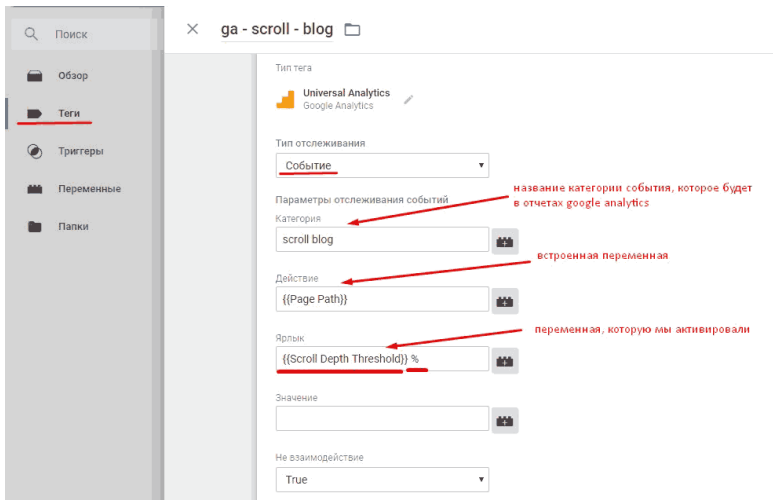


Рис. 504. Настройки тега

● Идентификатор отслеживания – Прописываем код счетчика Google Analytics;

В данном примере код счетчика записан через переменную **cod ua**. Пример настройки переменной идентификатора GA:

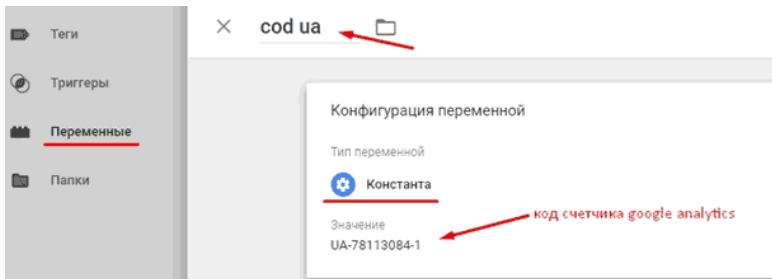


Рис. 505. Переменная GA типа «Константа»

● Триггер активации – **scroll – blog** (триггер, который мы создали выше)

Сохраняем тег.

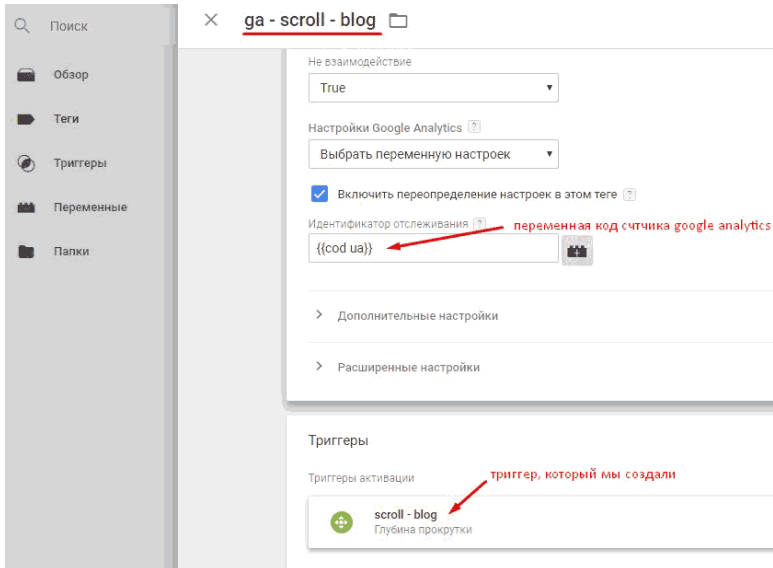


Рис. 506. Конечные настройки тега ga – scroll – blog

Проверяем данные через режим предварительного просмотра.



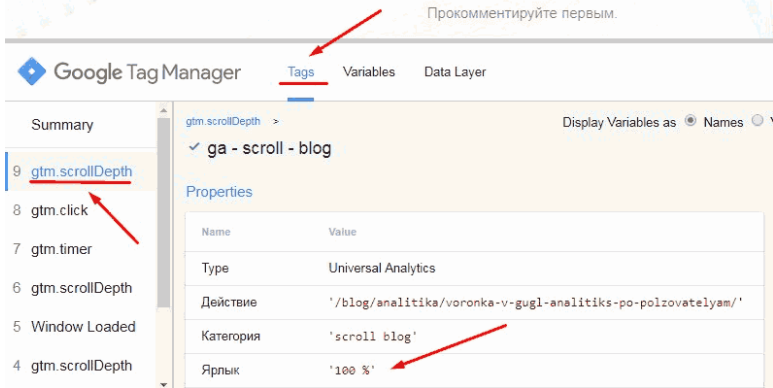


Рис. 507. Передача данных о глубине прокрутки в режиме отладки

## Просматриваем переменные:

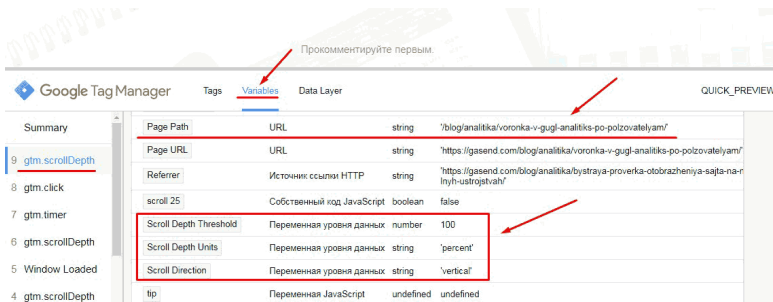


Рис. 508. Переменные

Если все передается корректно, не забудьте опубликовать

контейнер GTM с изменениями.

Следует отметить, что если вы находитесь в самом низу страницы (100%) и обновите страницу, то триггер сработает для всех значений, которые вы настроили (25%, 50%, 75%,100%). Аналогично Google Tag Manager будет работать с якорными ссылками.

Данные о прокрутке страниц в Google Analytics доступны в отчете «**Лучшие события**», который находится в разделе «**Поведение**».

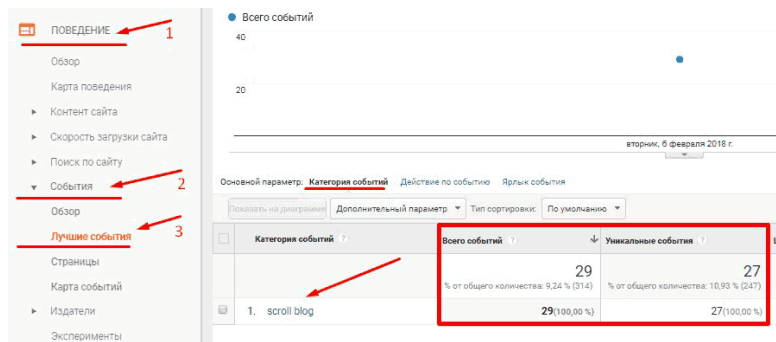


Рис. 509. Категория событий scroll blog

Категория **scroll blog** соответствует названию, которое мы прописали в теге.

Переходим на вкладку «**Действия событий**». В GTM мы прописывали часть url-страницы **Page Path**.

Основной параметр: Действие по событию Ярлык события Другое

Показать на диаграмме Дополнительный параметр Тип сортировки: По умолчанию

<input type="checkbox"/>	<u>Действие по событию</u> Page Path	Всего событий	Уникальные события
		29 % от общего количества: 9,15 % (317)	27 % от общего количества: 10,80 % (250)
<input type="checkbox"/>	1. /blog/analitika/bystrayz-proverka-otobrazheniya-sajta-na-mobilnyh-ustrojstvah/	8 (27,59 %)	6 (22,22 %)
<input type="checkbox"/>	2. /blog/analitika/kak-bystro-proverit-tranzaktsii-na-dubli-v-google-analytics/	6 (20,69 %)	6 (22,22 %)
<input type="checkbox"/>	3. /blog/analitika/kak-sekonomit-vremya-pri-rabote-v-google-tag-manager/	4 (13,79 %)	4 (14,81 %)
<input type="checkbox"/>	4. /blog/analitika/voronka-v-gugl-analitiks-po-polzovatelyam/	4 (13,79 %)	4 (14,81 %)
<input type="checkbox"/>	5. /blog/analitika/kak-svyazat-gugl-analitiks-i-edvords/	3 (10,34 %)	3 (11,11 %)

Рис. 510. Действия по событию – Page Path

Выбираем нужную нам страницу (кликаем по ней) и смотрим настроенный процент просмотров.

Основной параметр: Ярлык события Другое

Показать на диаграмме Дополнительный параметр Тип сортировки: По умолчанию

<input type="checkbox"/>	Ярлык события	Всего событий	Уникальные события
		4 % от общего количества: 1,25 % (320)	4 % от общего количества: 1,58 % (253)
<input type="checkbox"/>	1. 25%	2 (50,00 %)	2 (50,00 %)
<input type="checkbox"/>	2. 50%	1 (25,00 %)	1 (25,00 %)
<input type="checkbox"/>	3. 75%	1 (25,00 %)	1 (25,00 %)

Рис. 511. Ярлык события – Процент скроллинга

# Отслеживание глубины скроллинга в GTM с привязкой к автору статьи и категории в блоге

Немного усложним задачу. Теперь необходимо настроить передачу данных в Google Analytics и получить срез по каждой из статей в блоге по показателям (просмотры страниц, показатель отказов, средняя длительность просмотра страницы и % скроллинга), а также с учетом автора и категории публикации.

**% скроллинга (глубина прокрутки, глубина пролистывания) страницы** – это значение, которое показывает вовлеченность пользователя в контент. Если оно низкое, значит посетителю не интересен данный материал, и он просто покидает страницу, даже не пролистнув ее вниз.

Итоговый отчет должен был выглядеть примерно так:

Статья	Просмотры страниц	Показатель отказов, %	Средняя длительность просмотра страницы	Средний % скроллинга
link1.ru	1560	43,95%	4:18	84%
link2.ru	394	56,03%	3:32	47,50%
link3.ru	258	38,57%	1:35	50,30%
link4.ru	39	21,23%	0:21	62,80%
Категория	Просмотры страниц	Показатель отказов, %	Средняя длительность просмотра страницы	Средний % скроллинга
Новости	300	54%	3:18	69%
Статьи	423	34,12%	5:32	71,12%
Магазин	512	68,27%	0:35	41,78%
Категория	Просмотры страниц	Показатель отказов, %	Средняя длительность просмотра страницы	Средний % скроллинга
Кристина	800	63%	1:42	31%
Алеша	241	55,67%	3:39	72,48%
Миша	753	38,30%	17:36	88,01%

Рис. 512. Пример того, как это должно выглядеть на выходе

Используя дополнительный параметр, можно посмотреть статистику, например, по статьям конкретного автора и посадочной странице. Также помимо основной категории, на сайте присутствовал второй уровень вложенности, а именно категории делились еще на подкатегории. Например, «Новости» на «Медиа», «Маркетинг», «Поисковые системы» и «Сервисы».

Рис. 513. Категории и Подкатегории на сайте

Приняли решение реализовать данную задачу через Google Tag Manager, поскольку это позволяло избежать обращений к разработчикам и необходимость вносить дополнительный код на сайт.

Что мы делаем?

- создаем 3 специальных параметра в Google Analytics – «**author**», «**category**» и «**subcategory name**» с областью действия «**Hit**»;

Запоминаем индексы параметров – 1,2 и 3. Они нам пригодятся при создании тега в GTM.

+ СПЕЦИАЛЬНЫЙ ПАРАМЕТР		Поиск			
Название специального параметра	Индекс ↓	Область действия	Последнее изменение	Состояние	
author	1	Hit	5 апр. 2018 г.	Активная	
category name	2	Hit	5 апр. 2018 г.	Активная	
subcategory name	3	Hit	5 апр. 2018 г.	Активная	

Рис. 514. Специальные параметры: author, category name, subcategory name

● создаем 2 пользовательских показателя в Google Analytics – «**scrolling**» и «**timing**» с областью действия «**Hit**» для отслеживания количества скроллов и времени пребывания;

+ СПЕЦИАЛЬНЫЙ ПОКАЗАТЕЛЬ		Поиск			
Название специального показателя	Индекс ↓	Область действия	Тип форматирования	Последнее изменение	Состояние
timing	1	Hit	Время	5 апр. 2018 г.	Актив...
scrolling	2	Hit	Целое число	5 апр. 2018 г.	Актив...

Рис. 515. Специальные показатели в Google Analytics: timing, scrolling

Запоминаем индексы показателей – 1 и 2. Они нам пригодятся при создании тега в GTM.

● создаем 3 пользовательских переменных в Google Tag Manager типа «Элемент DOM»

**Метод выбора:** CSS селектор

**Селектор элементов:** брались путем копирования значения через консоль разработчика (клавиша F12 в Google Chrome)

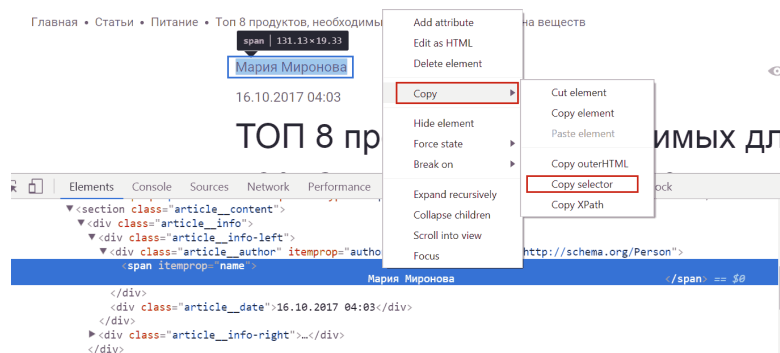


Рис. 516. Копирование CSS селектора элемента Author

Таким образом, три вышеописанные переменные имели одинаковые настройки, но разные селекторы элементов. Пример с переменной «author»:





Рис. 517. Переменная author в Google Tag Manager

Однако не все было так просто с «**category**» и «**sub category**». Копировав CSS-selector, мы убедились, что он одинаковый как для категории внутри статьи, так и для самой категории.

```
body > nav.page__breadcrumbs.breadcrumbs > span
```

Очень редко, но иногда приходится делать свои селекторы. В данном случае нам помог псевдокласс **nth-child**, который позволяет отследить определенный элемент списка из хлебных крошек. Например, **:nth-child(3)** позволяет отследить третий элемент.

Таким образом, мы для двух категорий задаем общий CSS-селектор, с единственным отличием nth-child.

- для «**category**» `.breadcrumbs .breadcrumbs__link:nth-child(2)`,

- для «**sub category**» `.breadcrumbs .breadcrumbs__link:nth-child(3)`

И получаем на выходе три переменные типа «**Элемент**

## DOM».

author	Элемент DOM
category	Элемент DOM
sub category	Элемент DOM

Рис. 518. Три переменные в GTM типа «Элемент DOM»

● создаем 4 переменных уровня данных для передачи событий, категорий, отслеживания времени пребывания и скроллинга

eventCategory	Переменная уровня данных
eventAction	Переменная уровня данных
timing	Переменная уровня данных
scrolling	Переменная уровня данных

Рис. 519. Четыре переменных в GTM типа «Переменная уровня данных»

**1. eventCategory:** Имя переменной уровня данных – eventCategory (Версия 2)

**2. eventAction:** Имя переменной уровня данных – eventAction (Версия 2)

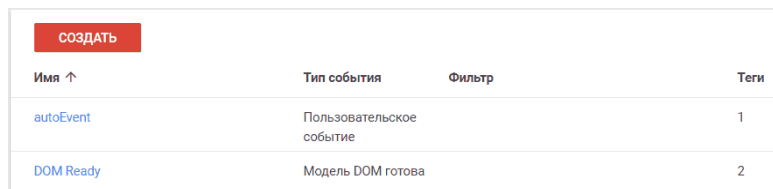
3. **timing:** Имя переменной уровня данных – time\_on\_page (Версия 2)

4. **scrolling:** Имя переменной уровня данных – percent\_of\_scrolling (Версия 2)

Далее создаются 2 триггера:

● 1 (autoEvent) с типом «**Пользовательское событие**» и условием активации «**Все специальные события**»;

● 2 (DOM Ready) с типом «**Модель DOM готова**» и условием активации «**Все события**».



Имя ↑	Тип события	Фильтр	Теги
autoEvent	Пользовательское событие		1
DOM Ready	Модель DOM готова		2

Рис. 520. Два триггера в GTM

Затем переходим к тегам и окончательно связываем наши триггеры и переменные.


Создаем новый тег **autoEvent UA** для передачи данных в GA с типом конфигурации Universal Analytics и добавляем в него созданные ранее переменные. Индексы специальных параметров и показателей берем из Google Analytics (см. выше).

Триггер активации – **autoEvent**. Получаем такую конфи-

гугл-аналитику:

Конфигурация тега

Тип тега

 Universal Analytics  
Google Analytics

Тип отслеживания


Событие


Категория


{{eventCategory}}

Действие

{{eventAction}}

Настройки Google Analytics 

Включить переопределение настроек в этом теге 

Идентификатор отслеживания 

UA-101539239-12

---

Специальные параметры

Индекс	Значение параметра
1	{{Author}}
2	{{category}}
3	{{sub category}}

---

Специальные показатели

Индекс	Значение показателя
1	{{timing}}
2	{{scrolling}}

---

Триггеры

Триггеры активации


 autoEvent  
Пользовательское событие

Рис. 521. Конфигурация тега autoEvent UA

Теперь осталось добавить тег **scroll tag** на определение глубины скроллинга. Для этого я воспользовался скриптом

от Анастасии Тимошенко из агентства интернет-маркетинга OdesSeo. Подробнее о том, что это за скрипт и как он работает, читайте здесь.

Тип тега – «**Пользовательский HTML**». Триггер активации – **DOM Ready**.

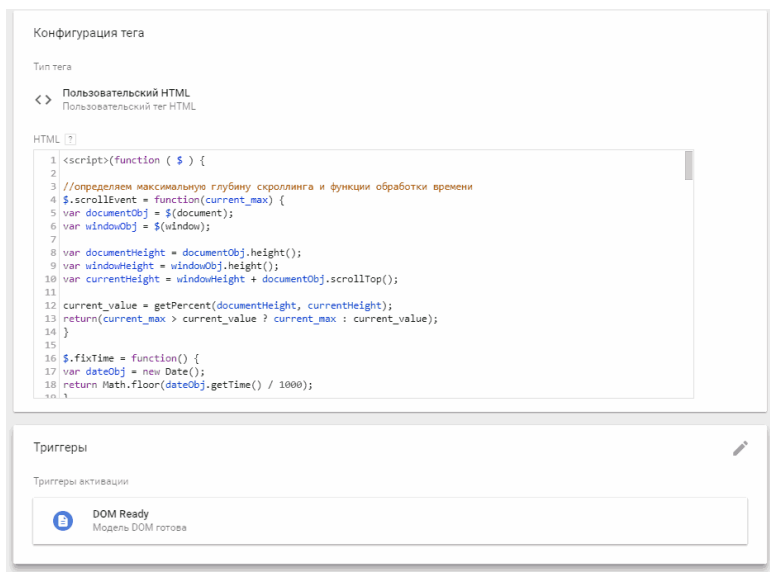


Рис. 522. Тег scroll tag, скрипт Анастасии Тимошенко (OdesSeo)

В общей сложности, в контейнере GTM получилось 4 тега (2 для решения задачи, 2 – коды счетчиков Google Analytics и Яндекс.Метрика)

СОЗДАТЬ		
Имя ↑	Тип	Триггеры активации
autoEvent UA	Universal Analytics	<> autoEvent
scroll tag	Пользовательский HTML	📄 DOM Ready
Код Universal Analytics	Universal Analytics	📄 DOM Ready
Код Яндекс.Метрика	Пользовательский HTML	👁 All Pages

Рис. 523. Теги в Google Tag Manager

Также нужно переопределить триггер для «Код счетчика Universal Analytics». Раньше там был триггер активации – All Pages, а теперь DOM Ready.

Схематично работу по созданию тегов, триггеров и переменных в Google Tag Manager можно представить так:

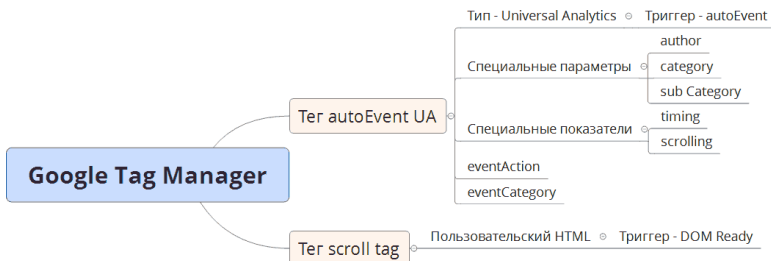


Рис. 524. Схема реализации

В Google Analytics осталось создать вычисляемый показатель, который позволит отслеживать среднюю глубину скроллинга по каждому материалу в %. Формула его такая:

$$\text{Avg. Scrolling Per.} = ( \{ \{ \text{scrolling} \} \} / 100 ) / \{ \{ \text{Просмотры страниц} \} \}$$

, где *scrolling* – количество прокруток на странице, а просмотры страниц – общее количество страниц, просмотренных посетителями. Учитываются повторные просмотры одной страницы. Размерность – проценты.

Все, что теперь осталось, это создать кастомный отчет в Google Analytics с нужными параметрами и показателями.

author	Страница входа	Просмотры страниц	Показатель отказов	Средняя длительность просмотра страницы	Avg. Scrolling Per.
		1 418 % от общего количества: 20,93 % (6 774)	47,99 % Средний показатель для представлений: 68,50 % (29,94 %)	00:06:06 Средний показатель для представлений: 00:02:22 (157,30 %)	74,26 % % от общего количества: 284,84 % (28,07 %)
1. Зияла	/news/na-kanale-pyatnitsa-budut-vybirat-4-uchalnego-vizachista-strany/	115 (8,11 %)	65,00 %	00:07:09	29,24 % (39,38 %)
2. Кристина	/articles/rossijskaya-kosmetsevtika-obz-or-lechebnoy-kosmetiki-ot-brenda-librider-m/	86 (6,06 %)	51,90 %	00:13:13	70,12 % (94,43 %)
3. Гала	/articles/5-idealnykh-prichesok-dlya-tonkih-kh-i-redkikh-vołos/	75 (5,29 %)	34,33 %	00:03:58	89,08 % (119,96 %)
4. Наталья	/articles/pyat-byudzhetnykh-gialuronovykh-syvorotok-dlya-litsa/	58 (4,09 %)	43,40 %	00:05:52	86,72 % (116,79 %)
5. Наталья	/articles/ezhnednaya-zashchita-5-soint-sezashchitnykh-kremov-dlya-goroda-s-sp-f-50/	31 (2,19 %)	40,74 %	00:08:04	63,81 % (85,93 %)
6. Кристина	/articles/tayny-sibiri-obzor-brenda-organi-checkoy-kosmetiki-natura-siberica/	29 (2,05 %)	53,85 %	00:10:25	60,66 % (81,66 %)

Рис. 525. Специальный отчет в Google Analytics

# Отслеживание YouTube видео в Google Analytics с помощью GTM

Одним из возможных вариантов увеличения конверсии сайта является видео. Но прежде, чем заказать эффектное видео, с разными спецэффектами, неплохо было бы знать сколько пользователей его вообще просмотрели.

Не секрет, что в разных нишах конверсия будет разной. Как вариант, можно загрузить пробное видео на сайт и настроить отслеживание YouTube видео в Google Analytics через Google Tag Manager. Настроить отслеживания довольно просто, так как в GTM есть для этого специальный триггер.

Активируем встроенные переменные для отслеживания триггеров.

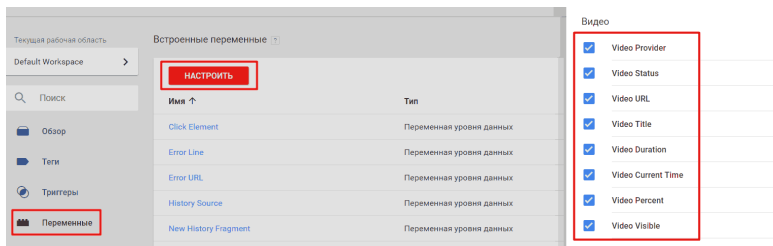


Рис. 526. Переменные типа «Видео»

● **Video Provider** – название площадки (YouTube);



- **Video Status** – статус видео. Может быть: Start, Complete, Pause, Seek (промотка), Buffering, Progress;
- **Video URL** – url-видео на площадке, пример *https://www.youtube.com/watch?v=...*
- **Video Title** – название видео;
- **Video Duration** – длина видео в секундах;
- **Video Current Time** – время в секундах, когда сработало событие;
- **Video Percent** – процент, когда происходит событие;
- **Video Visible** – может быть *true* или *false*, в зависимости от того видит ли пользователь видео, когда срабатывает событие или нет.

Создаем триггер. В GTM для отслеживания видео с YouTube есть специально встроенный триггер.

- Название – Youtube video event (название может быть любым);
- Тип триггера – Видео YouTube;
- Выбираем, что будет регистрироваться – Начало, Завершение, Приостановка, Перемотка, Буферизация;
- Ход просмотра – можно выбрать в процентах (прописываем через запятую) или по времени (в секундах);

Дополнительно:

- Включить поддержку JavaScript API для всех видео Youtube
- Условие активации триггера – В примере данный триггер будет срабатывать на всех видео.

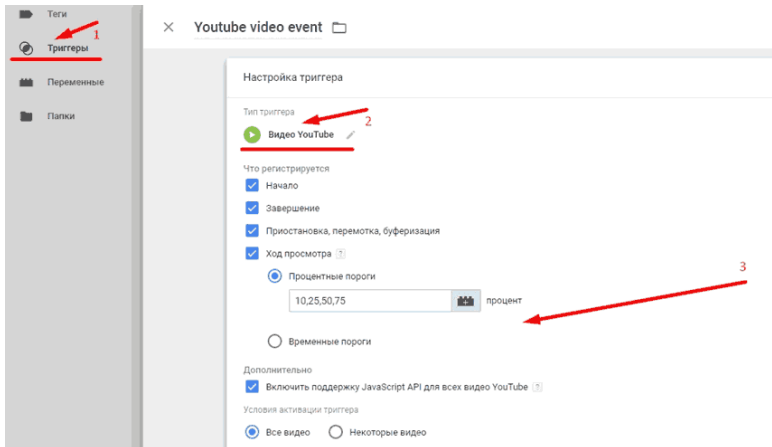
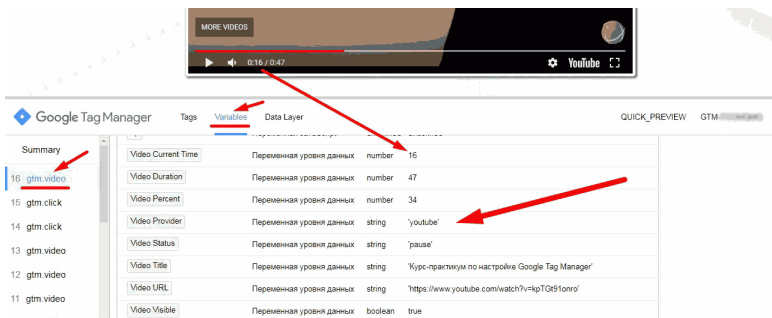


Рис. 527. Настройки триггера

Посмотреть работу триггера и значение переменных можно через режим предварительного просмотра GTM.



## Общая переменная (таблица поиска)

Для того чтобы передавать как можно большее количество информации в Google Analytics одним тегом, нужно создать переменную таблицы поиска для переменной **Video Status**, которую мы предварительно активировали. Последовательность действий:

- Создаем пользовательскую переменную «Таблица поиска»;

- Название: **Video Status – таблица поиска** (произвольное название);

- Тип переменной – Таблица поиска;

- Входная переменная – **Video Status**;

Входные данные. Прописываем значение переменной «Video Status» (см. выше). В поле результат прописываем: как мы их хотим назвать (произвольное название)

Пример заполнения:

- start – начал просмотр;

- pause – пауза;

- buffering – буфер;

- progress – просмотр – {{ Video Percent }}%;

- seek – промотка;

- complete – просмотр до конца;

**Примечание:** при значении *progress* рекомендуем про-

писать название (пример «просмотр») – переменную процентов просмотра `{{Video Percent}}` и значок %.

По умолчанию выставляем переменную `{{Video Status}}`.

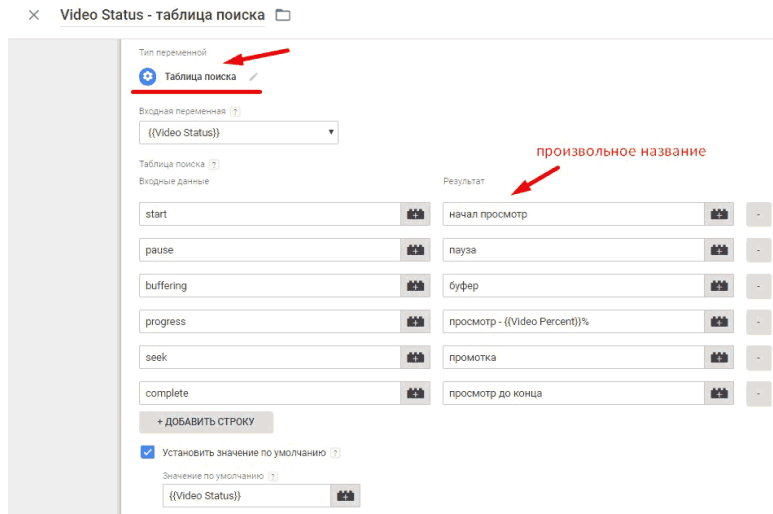


Рис. 529. Настройка переменной типа «Таблица поиска»

Проверим работу переменной через режим предварительного просмотра.

The screenshot shows the Google Tag Manager interface with the 'Variables' tab selected. A table of variables is displayed, with the following data:

Variable Name	Type	Value
Video Duration	Переменная уровня данных	number 47
Video Percent	Переменная уровня данных	number 10
Video Provider	Переменная уровня данных	string 'youtube' progress = просмотр - {{[Video Percent]}}%
Video Status	Переменная уровня данных	string 'progress'
Video Status - таблица поиска	Таблица поиска	string 'просмотр - 10%'
Video Title	Переменная уровня данных	string 'Курс-практикум по настройке Google Tag Manager'
Video URL	Переменная уровня данных	string 'https://www.youtube.com/watch?v=kpTGt91onro'

Рис. 530. Проверка переменной «Таблица поиска» в режиме отладки

На следующем шаге создаем тег.

- Название тега – **ga – Youtube video**;
- Тип тега – **Universal Analytics**;
- Категория – **Youtube video** (произвольное название);
- Действие – **Переменная Video Status – таблица поиска**, которую мы предварительно создали);
- Ярлык – Как пример, можно вставить две переменные **{{Video Title}}** и **{{Video URL}}**;
- Включаем переопределение настроек в этом теге;

Можно прописать код счетчика через переменную. Для этого ее нужно создать, в примере название **cod ua**, или вставить код напрямую.

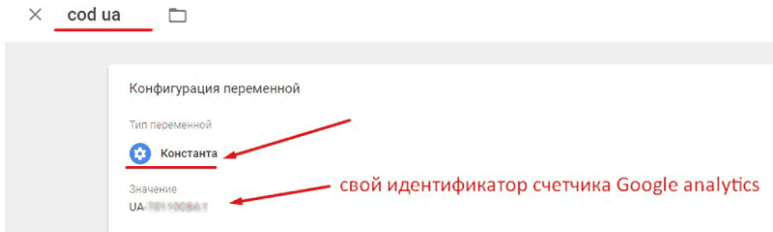


Рис. 531. Переменная типа «Константа» с идентификатором отслеживания GA

Триггером активации тега, является триггер видео, который мы предварительно создали.

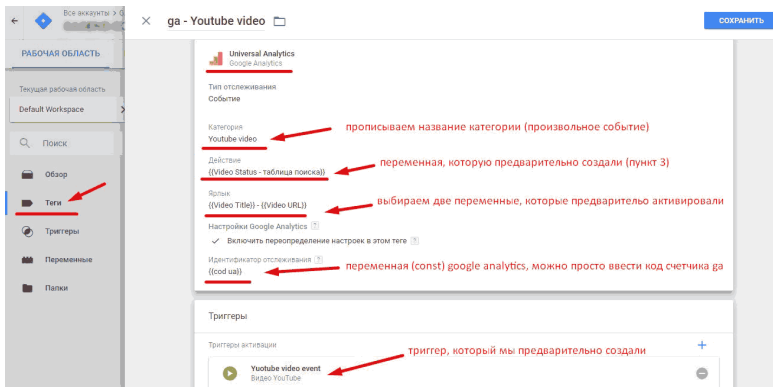


Рис. 532. Настройки тега ga – Youtube video

Перед публикацией тега проверяем корректность его срабатывания через режим предварительного просмотра.

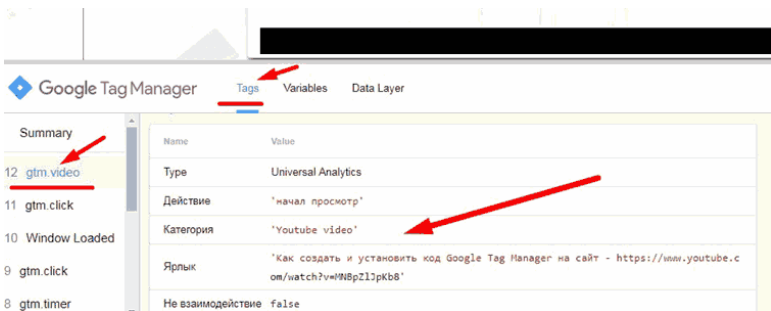


Рис. 533. Срабатывание тега

Данные о событиях с видео в Google Analytics доступны в отчете «**Лучшие события**», который находится в разделе «**Поведение**».

Категория – **Youtube video** (соответствует названию, которое мы прописали в теге)

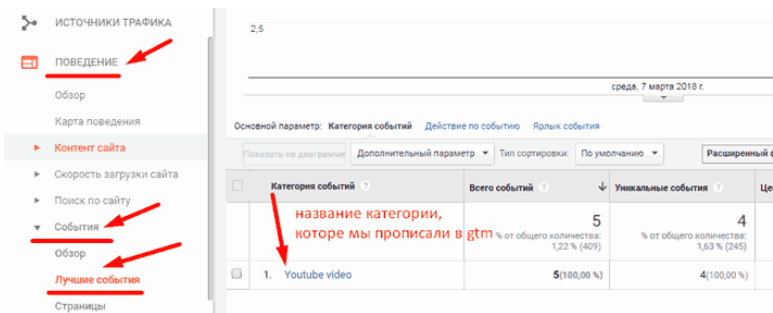


Рис. 534. Отчет «Лучшие события»

Переходим на вкладку «**Действие по событию**». В GTM мы прописывали переменную «*Video Status – таблица поиска*».

Основной параметр: Действие по событию | Ярлык события | Другое

Показать на диаграмме | Дополнительный параметр | Тип сортировки: По умолчанию

<input type="checkbox"/>	Действие по событию	Всего событий	Уникальные события
	{{Video Status - таблица поиска}}	5 % от общего количества: 1,22 % (409)	4 % от общего количества: 1,63 % (245)
<input type="checkbox"/>	1. промотка	2 (40,00 %)	1 (25,00 %)
<input type="checkbox"/>	2. начал просмотр	1 (20,00 %)	1 (25,00 %)
<input type="checkbox"/>	3. просмотр - 10%	1 (20,00 %)	1 (25,00 %)
<input type="checkbox"/>	4. просмотр до конца	1 (20,00 %)	1 (25,00 %)

Рис. 535. Действие по событию

В ярлыке событий вы увидите название видео и url-адрес. Это значения переменных, которые мы прописали в ярлыке событий тега в Google Tag Manager.

среда, 7 марта 2016 г.

Основной параметр: Ярлык события | Другое

Показать на диаграмме | Дополнительный параметр | Тип сортировки: По умолчанию

<input type="checkbox"/>	Ярлык события	Всего событий	Уникальные события
	{{Video Title}} - {{Video URL}}	1 % от общего количества: 0,24 % (411)	1 % от общего количества: 0,41 % (246)
<input type="checkbox"/>	1. Как создать и установить код Google Tag Manager на сайт - https://www.youtube.com/watch?v=MNBpZUjpkB8	1 (100,00 %)	1 (100,00 %)

Строк на ст

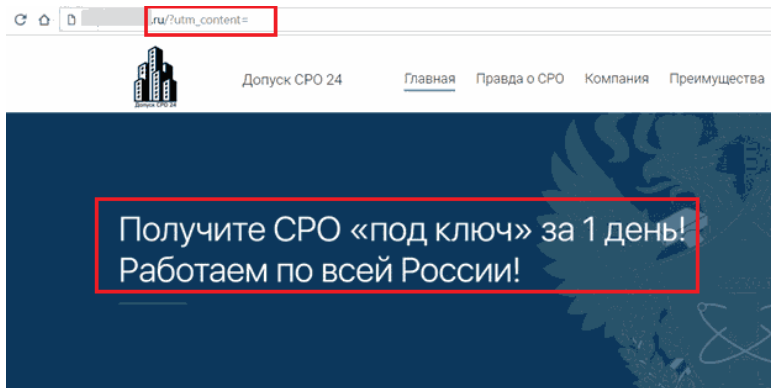


Рис. 536. Ярлык события

# Подмена контента на сайте с помощью GTM

Подмену контента на сайте (заголовок, подзаголовок, телефона, акции, метро и т.д.) в зависимости от условия в параметре URL (источника, рекламного объявления, запроса, гео и т.д.) можно осуществлять различными способами. Например, на базе *.php* (если знакомы с этим языком или есть разработчик) или с помощью сторонних сервисов, таких как **YAGLA**, **Adfor** и другие (если есть лишние \$). В этом материале мы разберем *бесплатный* способ подмены заголовка на сайте с помощью Google Tag Manager.

Пример. На сайте на первом экране есть заголовок:



The image shows a browser window with a URL bar containing a red box around the parameter `.ru/?utm_content=`. Below the browser, the website header includes a logo of a building, the text "Допуск СРО 24", and navigation links: "Главная", "Правда о СРО", "Компания", and "Преимущества". The main content area features a dark blue banner with a red box around the text: "Получите СРО «под ключ» за 1 день! Работаем по всей России!".

Рис. 537. «Получите СРО «под ключ» за 1 день! Работаем по всей России!»

Нам необходимо изменять этот заголовок в зависимости от параметра URL. В качестве параметра возьмем метку **utm\_content** (можно взять любую другую, хоть нестандартную, например, *m\_change*), в которую будем помещать значения других заголовков. Чтобы проще и быстрее реализовать задуманное, лучше всего подготовить заранее отдельную таблицу с метками и их значениями. Вот как это может выглядеть в Excel:

utm_content	Заголовок на сайте	Конечный URL
zag1	Получите СРО «под ключ» за 111 день! Работаем по всей России!	site.ru/?utm_content=zag1
zag2	Получите СРО «под ключ» за 222 дня! Работаем по всей России!	site.ru/?utm_content=zag2
zag3	Получите СРО «под ключ» за 666 дней! Работаем по всей России!	site.ru/?utm_content=zag3
zag4	Никогда не получите СРО! Мы не работаем в принципе!	site.ru/?utm_content=zag4

Рис. 538. Пример подмены заголовков в таблице Excel

Например, вы сделали пост в социальной сети или разместили рекламные кампании, разбили свой товар или услугу на категории (мебель -> один заголовок, кухни -> другой заголовок и т.д.). При переходе по одной из ссылок (столбец «*Конечный URL*») пользователь будет видеть подмененный заголовок (из столбца «*Заголовок на сайте*»).

Ссылка обязательно должна содержать параметр запроса, иначе подмена не произойдет. Давайте перейдем к настройке в GTM. Последовательность действий:

- Создаем пользовательскую переменную типа «URL»;
  - Название – **paramURL** (любое произвольное);
  - Тип компонента – Запрос;
  - Ключ запроса – **utm\_content** (тот, который вы определили для себя, может быть другим);
- Сохраняем.

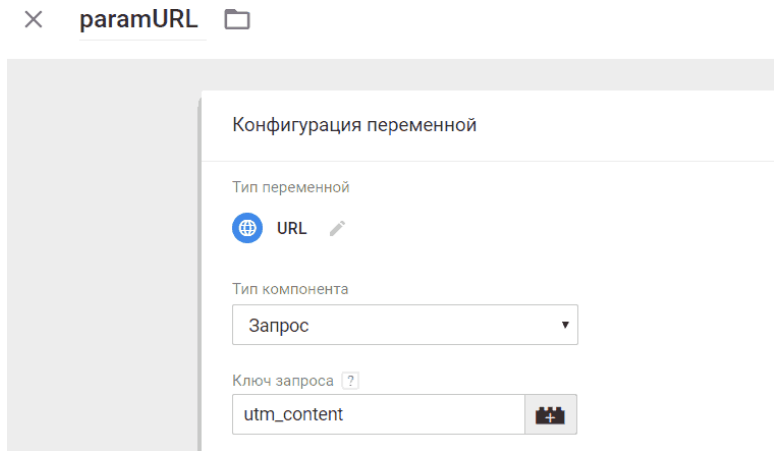


Рис. 539. Пользовательская переменная типа «URL»

Далее создаем еще одну пользовательскую переменную «**Таблица поиска**» из типа «**Утилиты**».

- Название – **podmeniZag** (может быть любым);
- Входная переменная – **{{paramURL}}** (та, которую создали на шаге выше);
- Таблица поиска – в каждую из строк в поле «**Входные**

**данные»** вводим атрибут ключа запроса `utm_content` (см. таблицу выше), а в поле **«Результат»** вводим значение параметра `utm_content`, то есть наши подмененные заголовки;

● Установить значение по умолчанию (галочка) – добавляем исходный заголовок.

Сохраняем переменную.

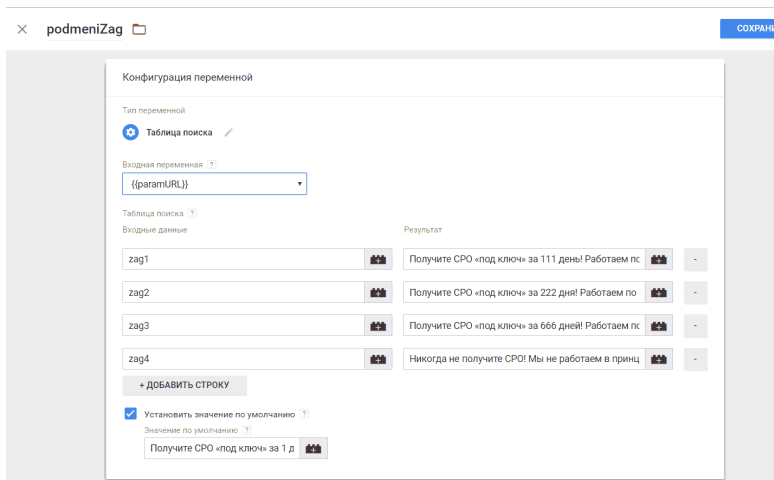


Рис. 540. Переменная «Таблица поиска» с нашими значениями

После этого создаем триггер типа **«Просмотр страницы»**, который будет активироваться только тогда, когда значение переменной `utm_content` определено. В противном случае если в ссылке нет параметра запроса, и оно принима-

ет значение «**undefined**» (**не определено**), то контент под-  
меняться не будет, и тег не срабатывает.

- Название – **undefined** (может быть любым);
- Тип триггера – Просмотр страницы;
- Условие активации триггера – Некоторые просмотры страниц;

Активация триггера при наступлении события и выполнении всех этих условий: **paramURL не равно undefined**.

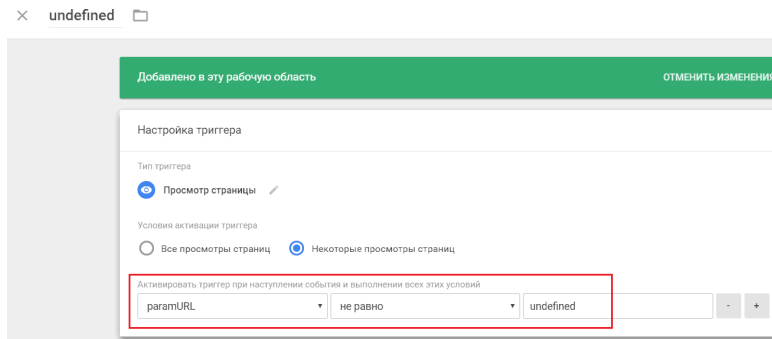


Рис. 541. paramURL не равно undefined

Сохраняем триггер.

Теперь нам осталось создать тег, который будет срабатывать при заданном условии активации триггера и подменять контент на сайте с помощью специального скрипта на значение табличной переменной.

- Создаем пользовательский HTML-тег;

- Название – **Tag – podmenaZag** (может быть любым);
- HTML

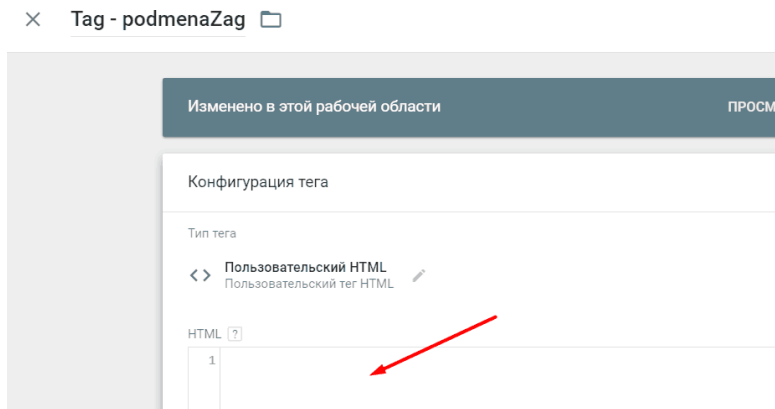


Рис. 542. Пользовательский HTML-тег

Что же добавлять в текстовое поле? Какой фрагмент кода? Перед реализацией этого шага необходимо определить атрибуты заголовка, на основании которого мы можем производить изменения. Именно от этого будет зависеть реализация пользовательского HTML в Google Tag Manager.

Поясним на конкретном примере. Переходим на наш сайт и открываем консоль разработчика, чтобы посмотреть фрагмент кода нашего заголовка.

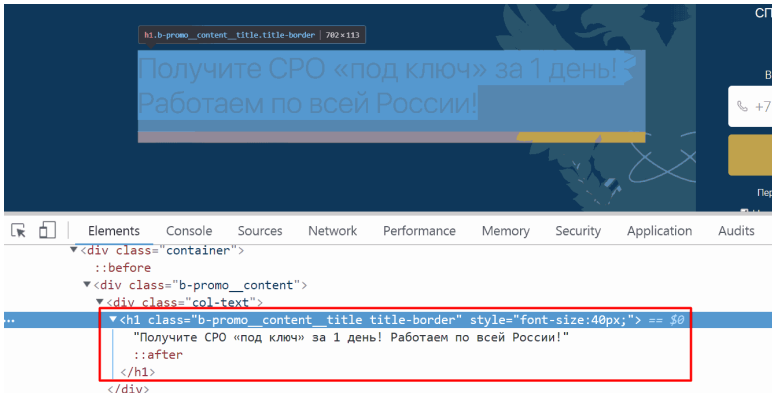



Рис. 543. Фрагмент кода заголовка в консоли разработчика

Если бы у данного заголовка `h1` был атрибут `id` со значением, например, **zagolovok** (может быть другим), то фрагмент кода, который нужно вставить тег, выглядел так:

Тип тега

<> Пользовательский HTML  
Пользовательский тег HTML 

HTML 


```
1 <script type="text/javascript">
2 document.getElementById("zagolovok").innerHTML="{{podmeniZag}}";
3 </script>
```



Рис. 544. Фрагмент JS-кода в случае, если бы у нас был атрибут `id`

В данном случае конструкция `document.getElementById("zagolovok").innerHTML = {{podmeni}}` изменяет содержимое HTML-элемента `h1` с `id = zagolovok` на новое из переменной **podmeniZag** (нашей таблицы поиска).


Но у нас нет атрибута `id`, а есть **class**. Тогда необходимо использовать другую конструкцию для подмены. Вот пример реализации:

Tag - podmenaZag 

Конфигурация тега

Тип тега

<> Пользовательский HTML  
Пользовательский тег HTML

HTML 

```
1 <script>
2 x = document.getElementsByClassName("b-promo__content__title title-border");
3 x[0].innerHTML = "{{podmeniZag}}";
4 </script>
```

Рис. 545. Фрагмент кода реализации через `class`

Сначала мы объявляем переменную `x`, которой присваиваем значение всех элементов, которые имеют заданные име-

на классов. Сам метод `getElementsByClassName()` возвращает коллекцию дочерних элементов элемента с указанным именем класса.

В данном случае класс `b-promo__content__title.title-border` – это атрибут заголовка `h1` в нашем примере (см. выше на скриншот). К узлам можно обращаться по номерам индексов. Индекс начинается с 0, поэтому в нашем примере `x[0]`, то есть у 1 дочернего элемента изменяем HTML, подставляя пользовательскую переменную `podmeniZag` таблицы поиска.

Есть еще проще вариант. Использовать библиотеку `jQuery`. Тогда конструкция будет иметь вид:

Тип тега

<> Пользовательский HTML  
Пользовательский тег HTML

HTML ?

```
1 <script>
2 jQuery("h1.b-promo__content__title.title-border").html("{{podmeniZag}}");
3 </script>
```

Рис. 546. Используем `jQuery` и функцию `.html()`

Функция `.html(newHTML)` в `jQuery` заменяет содержимое всех выбранных элементов на `newHTML`. В нашем примере она заменяет содержимое заголовка на переменную `podmeniZag`, которая содержит значения в таблице

поиска.

Есть и такой вариант. Использовать метод `querySelector()`. Он возвращает первый элемент, который соответствует указанному CSS-селектору в документе.

Тип тега

<> Пользовательский HTML  
Пользовательский тег HTML

HTML [?](#)

```
1 <script>
2 document.querySelector("h1.b-promo__content__title.title-border").innerText="{podmeniZag}";
3 </script>
```

Рис. 547. Метод `querySelector()`

В теге мы выбираем условие активации триггера `undefined` и сохраняем тег.

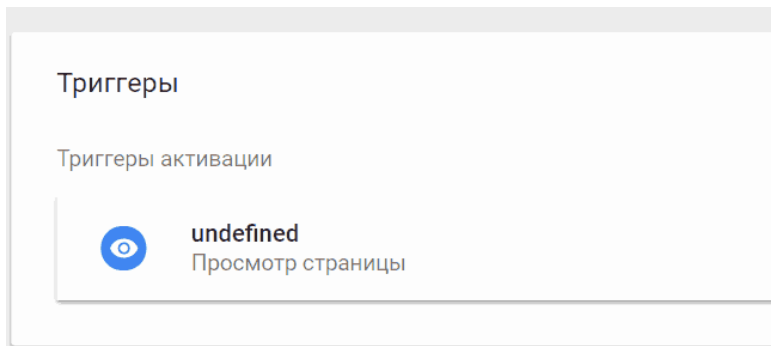


Рис. 548. Условие активации тега – `undefined`

Проверяем корректность работы в режиме предварительного просмотра.

← → ↻ 🏠 C [redacted].ru

Допуск СРО 24 Главная Правда о СРО Компания Преимущества Стои

Получите СРО «под ключ» за 1 день!  
Работаем по всей России!

Google Tag Manager Tags Variables Data Layer

Summary

- 3 Window Loaded
- 2 DOM Ready
- 1 Page View

Summary

Tags Fired On This Page

calltouch Пользовательский HTML - Fired 1 time(s)	GA Universal Analytics - Fired 1 time(s)	YM Пользовательский HTML - Fired 1 time(s)
--	---	---

Tags Not Fired On This Page

event form Universal Analytics	<b>Tag - podmenaZag</b> Пользовательский HTML
-----------------------------------	--

Рис. 549. Тег не сработал, переменной `utm_content` нет

Если мы переходим без метки в адресной строке, то тег не активируется и заголовок не подменяется. Все правильно. Давайте добавим к нашему url метку `utm_content=zag3`.

The image shows a browser window with a URL containing `utm_content=zag3`. The website header includes a logo and navigation links: "Допуск СРО 24", "Главная", "Правда о СРО", "Компания", "Преимущества", and "Стои". A large banner features the text "Получите СРО «под ключ» за 666 дней! Работаем по всей России!". Below the banner is the Google Tag Manager interface, displaying a "Summary" page. On the left, a list of events includes "3 Window Loaded", "2 DOM Ready", and "1 Page View". The main content area is divided into "Tags Fired On This Page" and "Tags Not Fired On This Page". Under "Tags Fired On This Page", three tags are listed: "calltouch" (User HTML - Fired 1 time(s)), "GA" (Universal Analytics - Fired 1 time(s)), and "Tag - podmenaZag" (User HTML - Fired 1 time(s)). Under "Tags Not Fired On This Page", one tag is listed: "event form" (Universal Analytics).

Рис. 550. Заголовок 3: Получите СРО «под ключ» за 666 дней! Работаем по всей России!

Тег сработал

Тег с меткой сработал, значение заголовка поменялось. Убедимся еще один раз. Теперь введем **utm\_content=zag4**.

← → ↻ 🏠 🔍 /?utm\_content=zag4

Допуск СРО 24 Главная Правда о СРО Компания Преимущества Ст...

Никогда не получите СРО! Мы не работаем в принципе!

Google Tag Manager Tags Variables Data Layer

Summary

- 3 Window Loaded
- 2 DOM Ready
- 1 Page View

Summary

Tags Fired On This Page

calltouch	GA	Tag - подменаZag
Пользовательский HTML - Fired 1 time(s)	Universal Analytics - Fired 1 time(s)	Пользовательский HTML - Fired 1 time(s)

Рис. 551. Заголовок 4: Никогда не получите СРО! Мы не работаем в принципе! Тег сработал

Все работает корректно. Публикуем контейнер GTM и радуемся подменам.

С помощью GTM вы можете настраивать правила, при котором на сайте будут подменяться не только заголовки, но и подзаголовки, номера телефонов в зависимости от источника переходов, изображения и многое другое.

Однако все же лучший способ реализации динамического контента на сайте – это серверная сторона (через тот же *.php*), а не браузерная, поскольку различные манипуляции с DOM могут привести к замедлению загрузки страницы у пользователя, да так, что он увидит эту подмену на сайте.

Допустимо использование такого приема в небольшом проекте (лендинге, несколько страничном сайте) и на малом объеме трафика (до 50–100 посетителей в день). Но если ежедневное количество пользователей гораздо больше, на сайте много заголовков и подмен, то рекомендуется реализовать все это через backend.

## В заключение

Вот и подошло к концу первое в русскоязычном сообществе электронное руководство по самому популярному диспетчеру тегов Google Tag Manager. Мы постарались добавить в него самые важные навыки и знания, которые будут необходимы вам в процессе работы.

Обе части (теоретическая и практическая) подробно изложены и разобраны с примерами и скриншотами. Все, что необходимо делать – это просто читать, делать, читать снова, переосмысливать, и снова делать. Только так можно разобраться в чем-то новом и неизведанном. Ну и должно пройти какое-то время, чтобы началась адаптация и привыкание. Продукты Google в этом плане не исключение.

К тому же, если заложить основы работы с GTM получилось даже у меня за полгода (хотя бы на уровне «новичок»), то почему не сможете вы? Я в вас верю!

Практическая часть составлена из примеров, которые встречались команде GaSend и мне во время работы с различными проектами. Разумеется, в главе 3 разбирается лишь 5% от возможностей диспетчера тегов Google. Однако не стоит забывать, что данное электронное руководство является достоянием всемирной паутины и каждого интернет-маркетолога и веб-аналитика в мире.

Вы также можете принять участие в расширении приме-



ров этой книги. Просто присылайте материалы мне на почту [ya.osipenkov@icloud.com](mailto:ya.osipenkov@icloud.com) и свое согласие на публикую в следующей версии **«Google Tag Manager для googлят: Руководство по управлению тегам»** И это будет сделано!

Увидимся в следующих изданиях и на моих вебинарах!

Понравилась книга? Подписывайтесь на мои социальные сети:

- [Vk.com – vk.com/ya.osipenkov](https://vk.com/ya.osipenkov)
- [Facebook.com – facebook.com/osipenkov.ru](https://facebook.com/osipenkov.ru)
- [Instagram.com – instagram.com/yakov.osipenkov](https://instagram.com/yakov.osipenkov)
- [Telegram: t.me/osipenkovru](https://t.me/osipenkovru)

*До встречи в январе 2019*