

# Target Enablement Workshop

Enabling Testing and Experience Targeting



Navigating life together

# **Testing Overview, Types, and Best Practices**

# What is Experimentation

The action or process of trying out new ideas, methods, or activities

Steps:

1. Pick a topic
2. Isolate your variables
3. Make hypothesis
4. Plan your data collection
5. Conduct your experiment methodically
6. Collect data
7. Analyze your data and come to conclusion

# Experimentation with Web Design

## Steps:

1. Pick a topic – Identify Starting Point on Website
2. Isolate your variables
3. Make hypothesis
4. Plan your data collection - Be Mindful of KPI's
5. Conduct your experiment methodically – Determine Layouts and Consider Inclusion and Exclusion and Build Test in Adobe Target
6. Collect data – Ensure Adobe tracks what you need
7. Analyze your data and come to conclusion - Conduct analysis using Adobe Analytics

# Inception

## Identify a Starting Point on Website

- Roadblocks: Is something making a step or process longer/harder?
- Distractions: Is anything on the page luring visitors away from a step or process?
- Unclear Goals: Is something happening on the page that makes it hard to discern the primary action to take?
- Dead Ends: Are there pages or areas of the site that end without a CTA?
- Unclear Steps: Pages or actions that are unclear in their intent
- How many people come to the pages we are looking at?

*Ultimately, you want to make sure you do a pre-test analysis to understand your traffic before trying a test.*

\*thank you to Sultan for this high level information

# Inception

## Isolate your variables

- Are there any campaigns driving people to the experience to be tested? If so, what is the Audience? What is the Creative/Messaging?
- Are there campaigns driving traffic to a different page but with an audience that may end up in this experience? If so, what is the audience? What is the Creative/Messaging?

# Design

## Make Hypothesis

- Provide the Hypothesis that is to be proven out in this test. It is best to structure your hypothesis by following the below structure:

"If \_\_\_\_\_[I do this] \_\_\_\_\_, then \_\_\_\_\_[this]\_\_\_\_\_ will happen."

# Inception

## Plan your Data Collection

- Based on what we know about the customers coming to this page, what are the key actions the audience should be taking?
- What KPI's will prove success/failure of Hypothesis?
- Are the KPI's directly answering the hypothesis, or are we coming to any conclusions with that KPI?
- Is there enough visits to the page in order to collect sample size in an appropriate amount of time?



# Build and Test

Conduct your experiment methodically

- Based on hypothesis, what layout should I consider?
- Based on hypothesis, are there specific audiences we are isolating the test too?
- If we are not isolating the audience, are there specific segments we need to look at to help support the hypothesis?
- Is the page intended for a smaller audience? If visits to page are low, are there ways we can execute this test to ensure we can come to a conclusion?

# Build and Test

## Collect data

- Do the layouts collect the KPI's the same way as the control (Apples to apples comparison)?
- Will the data we are collecting be able to segment the data the way it needs to?
- If there are segments being collected, do we have an even distribution, if possible, across control and tests?
- What frequency will we do reporting to monitor test execution?

# Analyze

Analyze your data and come to conclusion

- Did the test reach statistical significance?
- Based on results, what is the standard deviation and sample size for the results?
- Be insights focused when reading out. Instead of what was the resulting data, look at what insights/learnings did the data provide?
- Did the hypothesis get proven?

# What not to do.

Don't:

- Have an N=1 mindset (if I would do things a certain way, then everyone will)
- Have unattainable goals for a test
- Assume you know your audience without doing an analysis
- Ignore sample size or standard deviation
- Ignore secondary metrics
- Change pages during a test
- Compare two similar, but not equal metrics for KPI

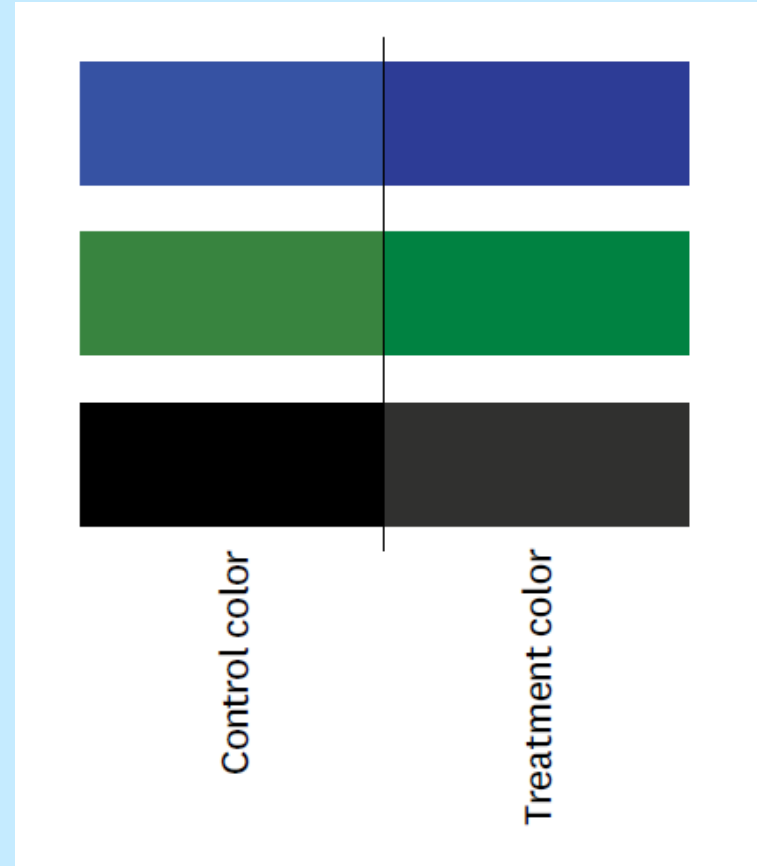
# Best Practices

Do:

- Keep testing simple
- Have small incremental changes
- Make sure page has good volume
- Ensure test plans are filled out for a test
- View results objectively
- Focus readouts on insights gained and not who won.

# Small Changes can have a huge impact:

Bing's experiments showed that slightly darker blues and greens in titles and a slightly lighter black in captions improved the users' experience. When rolled out to all users, the color changes boosted revenue by more than \$10 million annually.



# Experiments can guide investment decisions.



“Should there be 3, 10, or 50 people working on that project? Big found that every 100-millisecond difference in performance of website had a .6% impact in revenue. 100-millisecond speed up was found to be worth \$18M in annual revenue”

# Address the Definition of Success:

Every business group must define a suitable evaluation metric for experiments that aligns with its strategic goals. That might sound simple, but determining which short-term metrics are the best predictors of long-term outcomes is difficult. It requires close cooperation between senior executives who understand the strategy and data analysts who understand metrics and trade-offs.





# Beware of Low Quality Data:

It doesn't matter how good your evaluation criteria are if people don't trust experiments' results. Getting numbers is easy; getting numbers you can trust is hard!



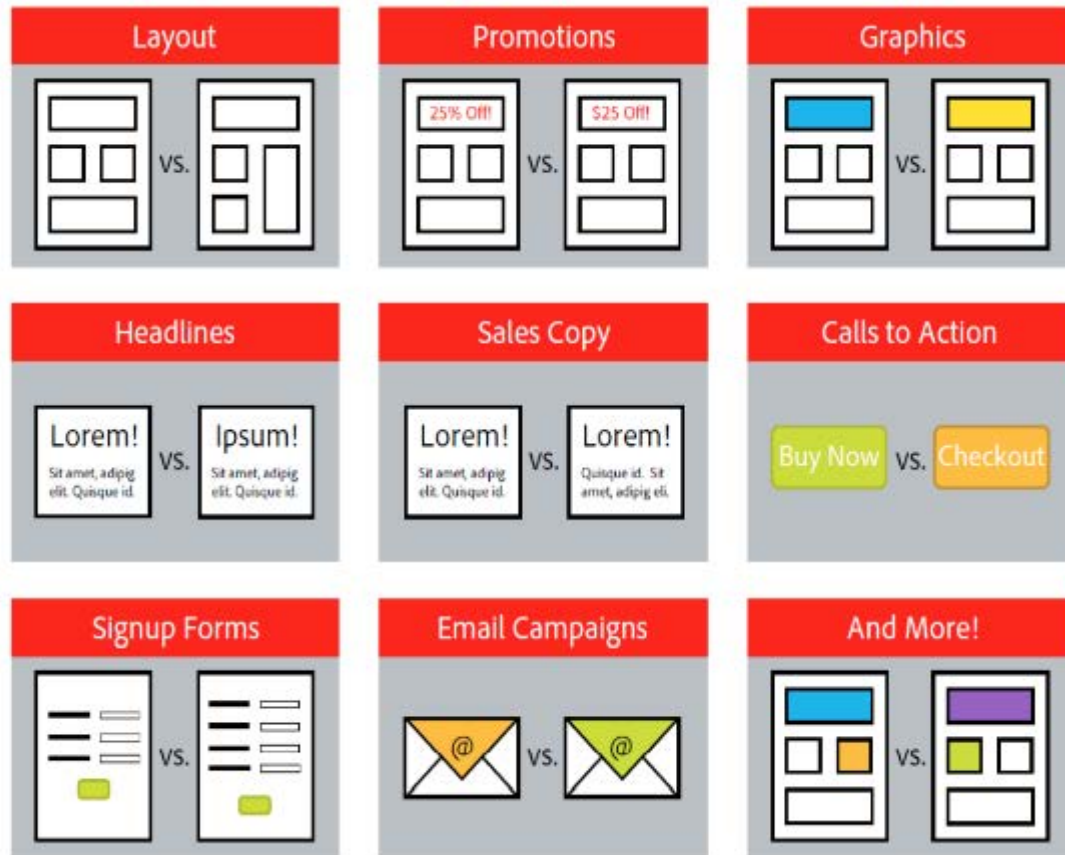
# Adobe Target Test Types

TYPE	DESCRIPTION
<b>A/B</b>	Testing to compare two or more versions of a page to see which performs better.
<b>Multivariate (MVT)</b>	Testing to compare multiple variations of different elements within a single web page. Provides context as to which elements are most influential at driving conversion.
<b>Experience Targeting</b>	Also called rules based targeting. Creating unique user experiences targeted to specific audiences in order to promoted engagement and a lift in conversion. Typically ‘Experience Targeting’ is an outcome of A/B testing.
<b>Auto Allocation</b>	An A/B test in which traffic is automatically directed to the best performing experience over time. This is ideal for tests that are considered “high risk” or for tests that are short term, such as promotions.

# What and Where to Test

The greater the change, typically the greater the impact (positive or negative). In order of priority, Adobe recommends testing:

- Inclusion/exclusion of elements
- Layout
- Functionality
- Navigation
- Calls to Action
- Headlines
- Copy
- Images
- Entire page

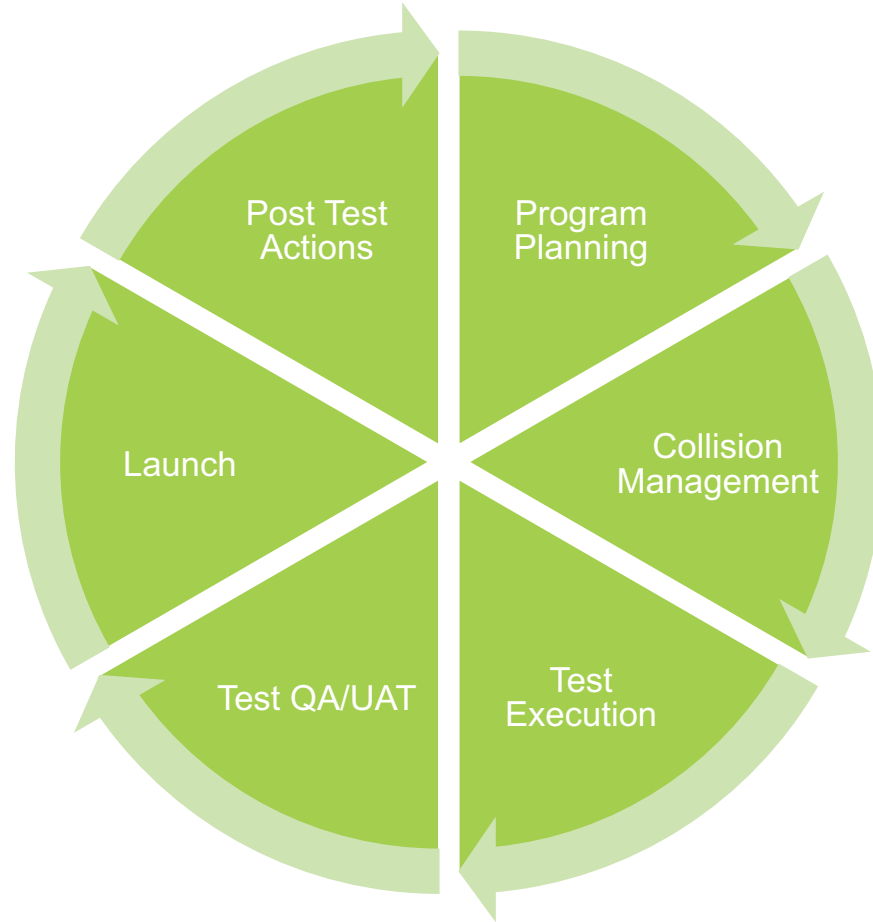




Navigating life together

# Testing Process

# Testing Execution Process





Navigating life together

# **Test KPI's, Duration, MDL, and Confidence**

# Test KPIs

## Events to consider

### Conversion Variables

Form Start %

Form Complete %

Form Start to Complete %

Video Start to Form Start %



# Test KPIs

Events to consider

Content Metrics:

- Video Start %
- Video Complete %
- CTA Clicks
- Time on Site





# Test KPIs

## Events to consider

### Other Considerations:

Bounce Rate

Page Load Time

404 pages



**Confidence Level:** also known as statistical significance, indicates how likely it is that an experience's success was not due to chance. “I am 90% confident that this lift is a trend” . The higher the confidence requirement, the larger the sample size needs to be

# What seems like a better trend?

75%

Conversion

50%

Conversion

# What seems like a better trend?

75%

Conversion

---

3/4 people

Completed

a Form

50%

Conversion

---

2000/4000

people

Completed

a Form

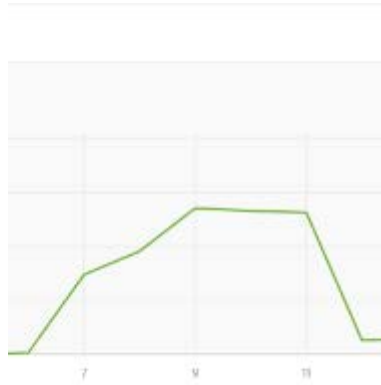
**A Statistical Trend:** Mathematically proven trend with a small or medium range in variance. A valid trend.

**A Directional Trend:** a trend that is not yet proven mathematically, so typically has a large variance in possible outcomes. It is not yet valid.

**Statistical Power:** the probability of detecting a real difference in conversion rate of a certain magnitude. “There is an 80% chance that we will reach the appropriate sample size to confirm the test results in 6 weeks”. The higher the statistical power, the more time required in a test

*80% statistical power is a typical baseline to use*

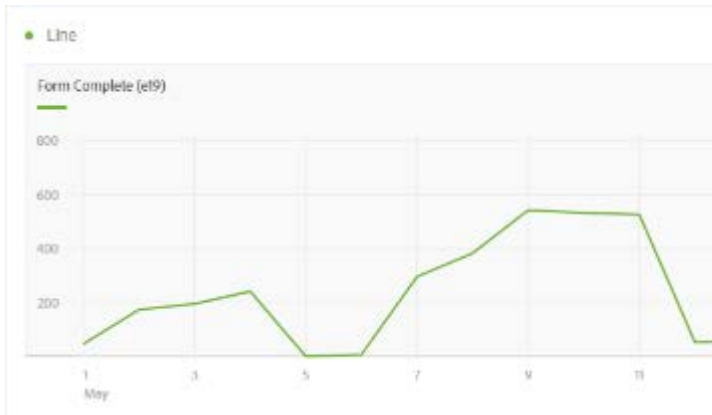
# How Confident are you that we will hit appropriate traffic?



2334 for this week averaging 333 per day

Will we average 300 form completes per day to achieve 9300 for the month?

# How Confident are you that we will hit appropriate traffic?

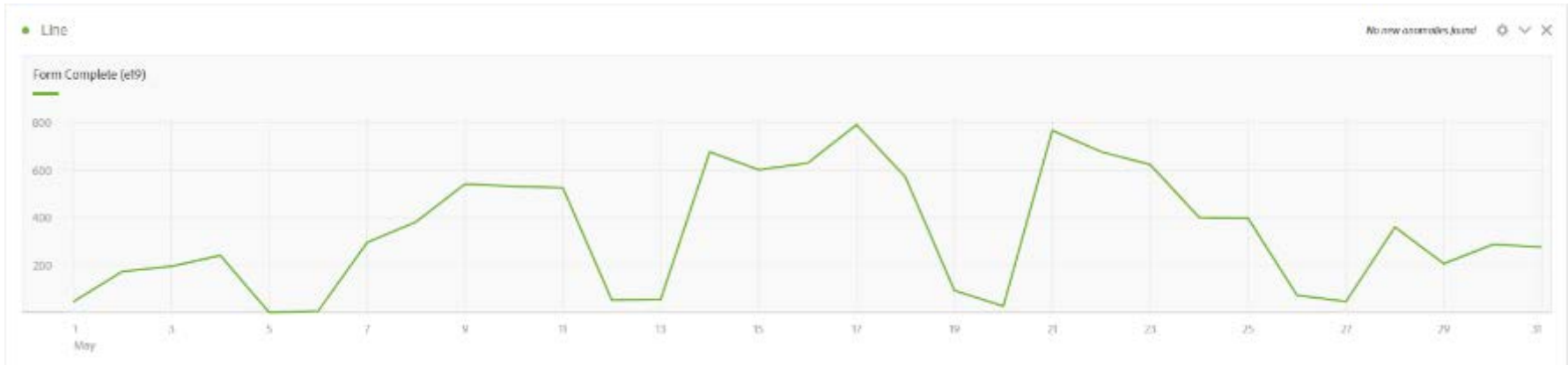


2988 for the first 2 weeks averaging 213 per day

Will we average 300 form completes per day to achieve 9300 for the month?



# How Confident are you that we will hit appropriate traffic?



Will we average 300 form completes per day to achieve 9300 for the month? **10,545 for the month = 340 per day**

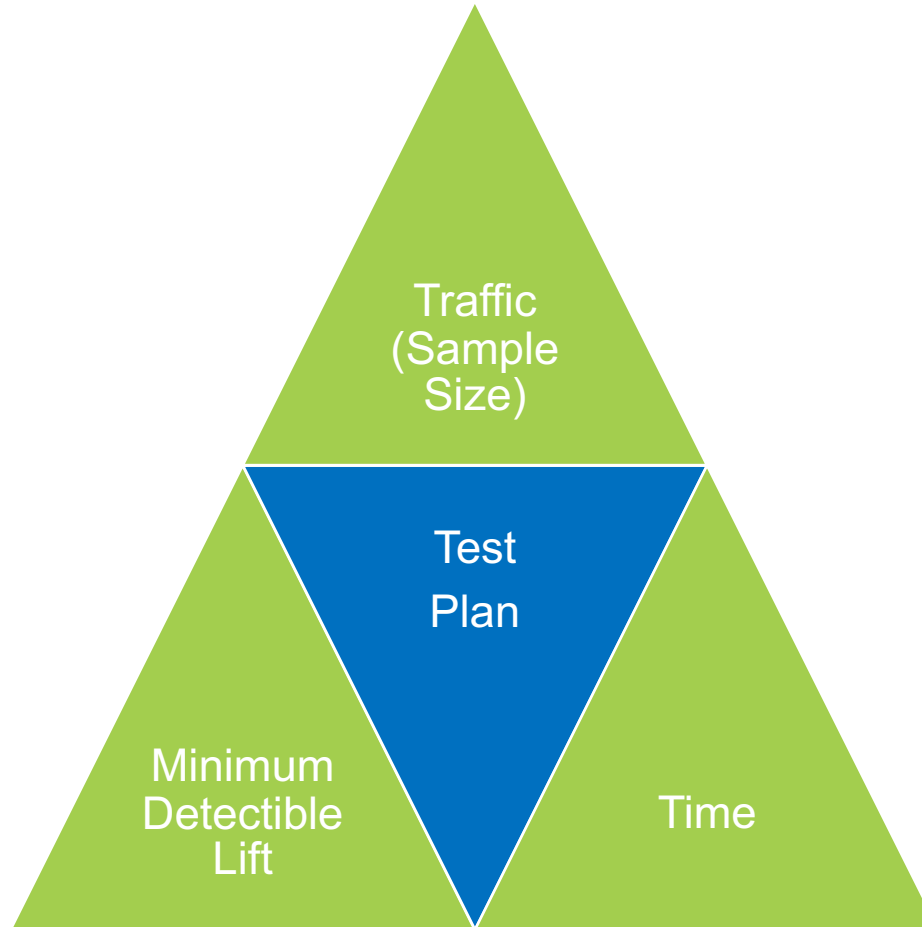
**Lift (Minimum Detectable Lift):** The amount increase needed in the test in order to ensure Confidence level will be achieved. “This test must increase conversions 5% in order to make sure we are confident the results of the test are confirmed given the amount of visitors we expect. The lower the amount of lift in detection needed, the larger the sample size needed, meaning the longer the test must run.

# What's Easier to Achieve?

Making a  
page go  
from 2% to  
2.1%  
Conversion

Making a  
page go  
from 2% to  
5%  
Conversion

# The 3 Forces in Test Planning



# Sample Size Calculator – 3 Forces at work

**Confidence Level:** also known as statistical significance, indicates how likely it is that an experience's success was not due to chance. "I am 90% confident that this lift is a trend". The higher the confidence requirement, the larger the sample size needs to be

**Statistical Power:** the probability of detecting a real difference in conversion rate of a certain magnitude. "There is an 80% chance that we will reach the appropriate sample size to confirm the test results in 6 weeks". The higher the statistical power, the more time required in a test

**Lift (Minimum Detectable Lift):** The amount increase needed in the test in order to ensure Confidence level will be achieved. "This test must increase conversions 5% in order to make sure we are confident the results of the test are confirmed. The lower the amount of lift in detection needed, the larger the sample size needed, meaning the longer the test must run.

Adobe Target

## Sample Size Calculator

Conversion Rate Metric    RPV Metric

Confidence Level: 90 %

Statistical Power: 80 %

Baseline Conversion Rate (Control Offer): 2.11 %

Total Number of Daily Visitors: 2212

Number of Offers Including Control: 2

Daily Number of Visitors per Offer: 1621

Lift (that can be detected with power (80%) probability)	5%	10%	7 %
Absolute Difference in Conversion Rate (that can be detected with power (80%) probability)	0.11%	0.21%	0.15%
Conversion Rate of Alternative (that can be detected with power (80%) probability)	2.22%	2.30%	2.26%
Sample Size per Offer (# of visitors)	225,070	60,166	121,076
Sample Size per Offer (# of conversions)	4,900	1,270	2,555
Days to Complete Test	146	38	75
Weeks to Complete Test	21	6	11

Correct for Multiple Offers (Bonferroni Correction)

# Sample Size Calculator - URL

Adobe Sample Size Calculator is a free online tool to use to calculate duration of a test. Check it out at:

<https://docs.adobe.com/content/target-microsite/testcalculator.html>

# Test Planning Exercise

US wants to do a home page A/B test (2 variations total) to boost find an x activity. How many days will it have to run if you are looking for:

- 95% confidence

- 12,534 daily visitors to the home page

- 4.6% conversion currently

- want to achieve a 5% increase in conversion (4.83% goal conversion rate)

# Test Planning Exercise

US wants to boost  
find an x e  
looking f  
-95% c  
-12,534  
-4.6% c  
-want to a goal  
conversion rate)

How can I reduce the time for the test if I do not want to change the goal conversion rate?



# Test Planning Exercise

US wants to boost  
find an x e  
looking f  
-95% c  
-12,534  
-4.6% c  
-want to a goal  
conversion rate)

How can I reduce the time for the test if I do not want to change the goal conversion rate?

Reduce Confidence

# Key Point to Consider

When planning a test duration, pick KPI/segment that has the worst sample size that needs to be statistically significant.

Example, if I want to confirm test overall, but also make sure it is confirmed on mobile, total daily visitors should be calculated using the total mobile visitor number. Similarly, if I have two metrics to look at, calculate conversion for both and use the lower performing one.



Navigating life together

# Test Plan Template

# Test Overview: Test Summary

## Objective

Explain the objective of running the test

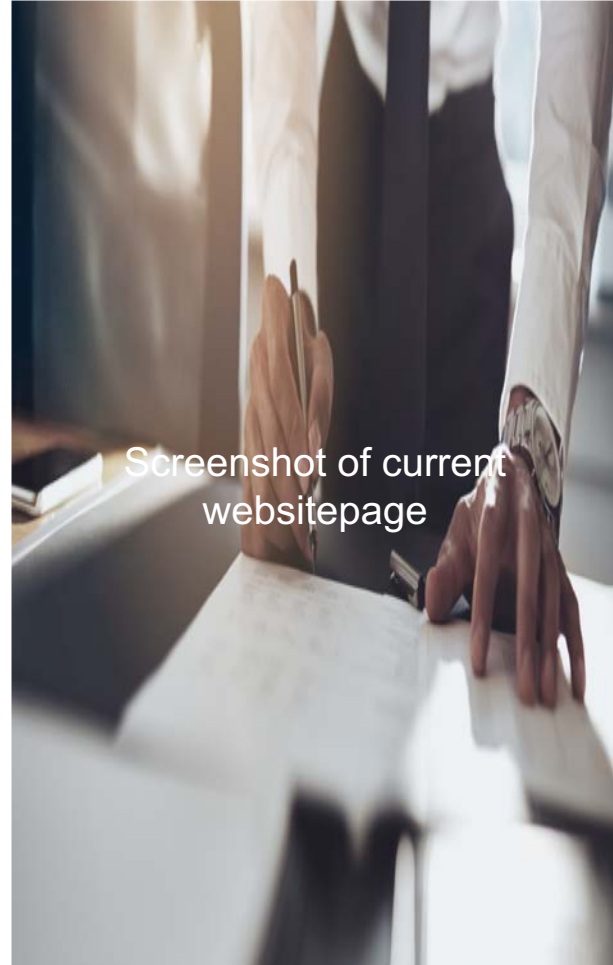
## Hypothesis

Provide the Hypothesis that is to be proven out in this test. It is best to structure your hypothesis by following the below structure:

"If \_\_\_\_ [I do this] \_\_\_\_, then \_\_\_\_ [this] \_\_\_\_ will happen."

## Methodology

Explain the number of experiences, how they will differ, and segments you will use.



# Test Overview: Test Specs

## SUCCESS METRICS

### Primary Conversion Metrics

- Ex. Form Submit Conv. %

### Secondary Conversion Metrics

- Ex. Form Complete (e19)

## SEGMENTS

### Test Segments

- Provide which segments will see the test. Put All Visitors if there will be no specific segments

### Reporting Segments

- Provide how the test analysis should be broken down in order to maximize insights from test. For example, will it be insightful to see how mobile users did in the test compared to how Desktop users did?

## GOALS

Goals for what percentage change is planned to be observed

- Ex +4% conversion rate

# Test Overview: Financial Drivers (when possible)

## CURRENT FINANCIALS

Current estimated revenue or business loss for page

- Ex. \$3M Annually in Sales form Leads
- Ex. \$2M annually in Costs for Call Center to answer login questions

Revenue/Cost Per Current Test Metric

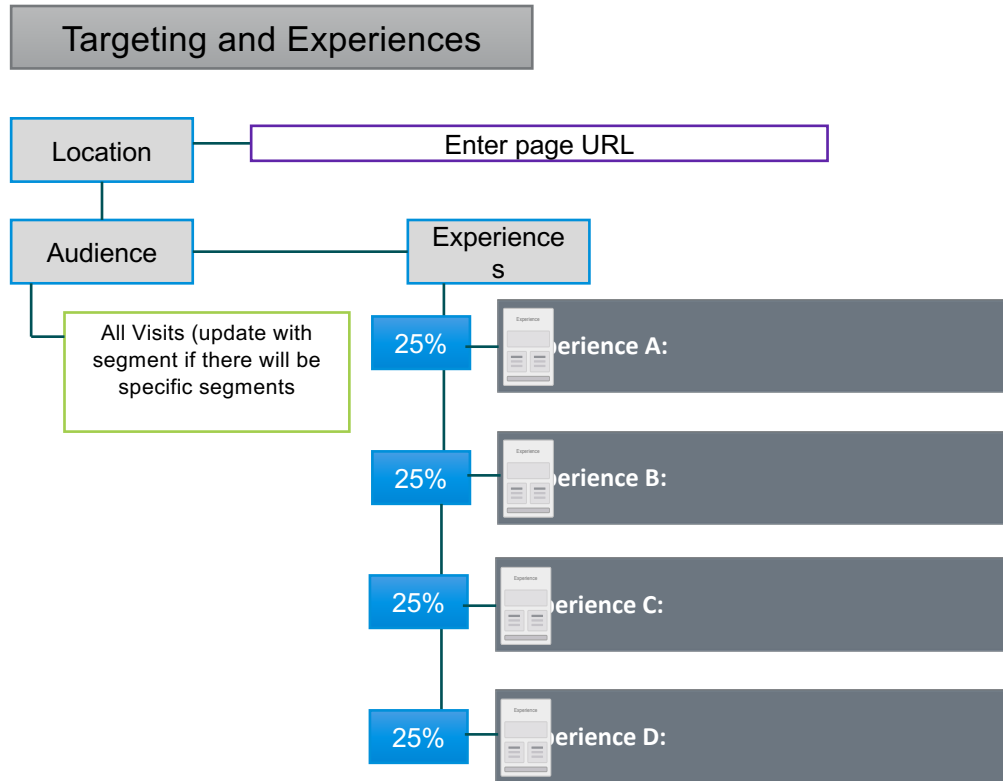
- Ex. \$12.50 in revenue per lead captured
- Ex \$4 in Call Center expensed per Login Failure

## GOAL

Based on Goal and Current financials breakdown, explain what the expected cost savings or revenue generation you plan on seeing

- Ex 4% lift in lead conversion=\$120k in new revenue annually
- Ex 6% drop in Login Failures = \$120k in cost savings by reducing login failure rates

# Test Overview: Test Specs



## Performance Measurement

### Primary Conversion Metric(s)

- Ex. Form Submit Conv. %

### Secondary Conversion Metric(s)

- Ex. Form Complete (e19)

## Statistical Recommendations

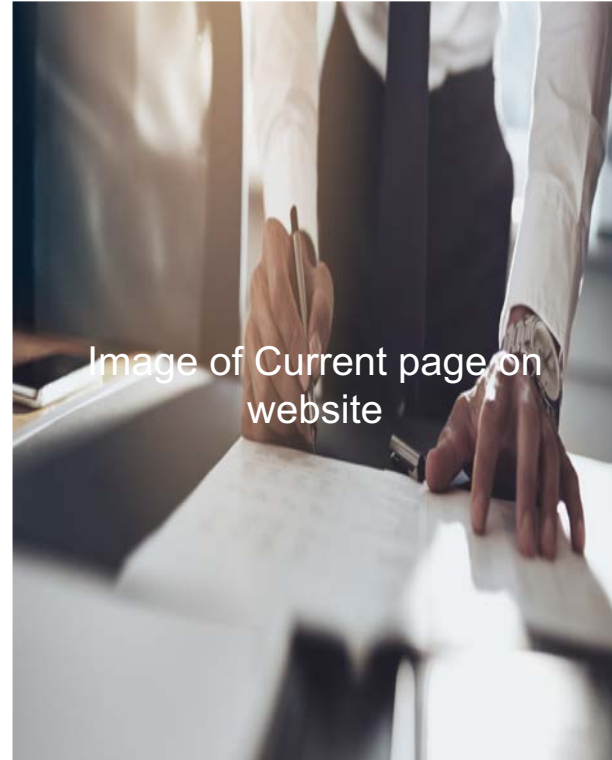
Test Duration  
**2wks**

Confidence  
**+95%**

# Test Overview: Experience Mocks

## Experience A (Control)

- Default Content

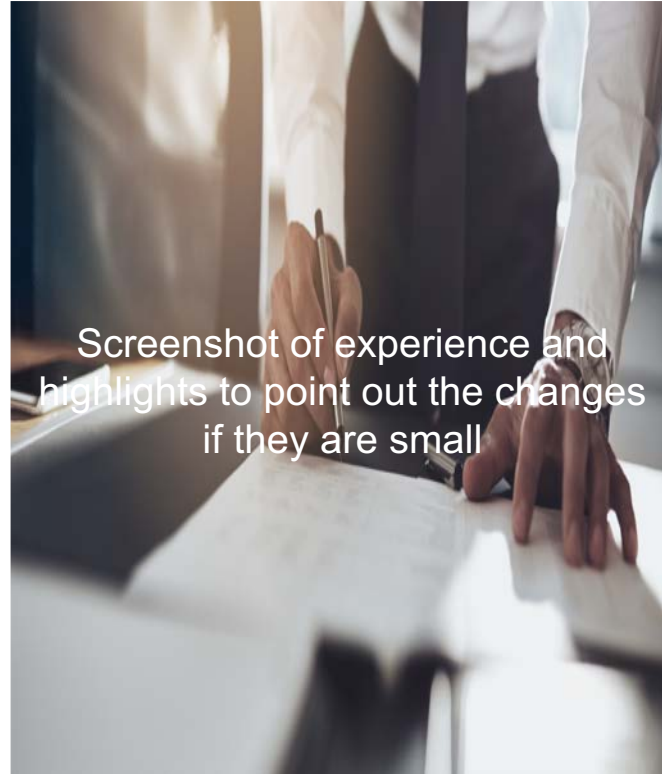




# Test Overview: Experience Mocks

Experience B: Title/Name of experience

- Explanations of changes to the page for this experience



# Test Results: Overall Performance

Primary KPIs					
KPI	Segment	N	Conversion Rate	%Lift	Confidence
KPI 1 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%
KPI 2 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%

Secondary KPIs					
KPI	Segment	N	Conversion Rate	%Lift	Confidence
KPI 1 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%
KPI 2 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%

# Test Results: Segment Performance

Primary KPIs					
KPI	Segment	N	Conversion Rate	%Lift	Confidence
KPI 1 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%
KPI 2 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%

Secondary KPIs					
KPI	Segment	N	Conversion Rate	%Lift	Confidence
KPI 1 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%
KPI 2 Name	Segment 1	Number of Conversions	%	%	%
	Segment 2	Number of Conversions	%	%	%

# Test Findings



- Summarize findings of past slides. Be objective and do not assume any theories on this page. The point here is to have a summary page for executives to be able to review and know what happened at a high level without having to review all of the KPIs

# Test Theories



- The point of this page is to work out theories based on findings. This should be filled out as a group during the final readout. The test findings will be input into these theories
- Design Theories: These should be about design elements, general usability, or speed of the immediate surroundings. Sometimes a test result is simply explained by being faster, less buggy, or prettier
- Customer Mindset Theories: This is the predicted state of mind of the customer in this particular session. What do they care about? What emotion could be affecting them and why?
- Who Theories: This is a theory on who your user might be as a person and what would this imply about their behavior while in this test? What are their likes and dislikes?

# Follow up tests



- Based on findings, are there any follow up tests that should be done to confirm any of the Theories?

# Collision Management

## Why should we consider the impact of test overlap?

Any time individual visitors are in multiple simultaneous tests, there is possibility for the test experiences to impact one another.

### Risks to Consider:

- Two simultaneous 2-variation AB tests (A,B and A,B) make 4 user experiences (AA, AB, BA, BB).
  - *Since the test sample size is only enough to determine significance for a two-way split, the impact of the second test cannot be reliably measured in this scenario.*
- If the combined experiences have not been QA'd together, there is a possibility of the different variation code conflicting and causing broken user experiences.
- Multiple experiences that have not been QA'd together might also result in mixed content messages that could result in a poor user experience.

# Collision Management

## 2 Types of Collisions

### Development Collision:

- Utilizing a front end tool with simultaneous code running on a common page for a single user could have unknown interaction with content or functionality of the site

### Experience Collision:

- Having multiple teams using the same visitor base - there will likely be times where a test is either on the same page for a user or in the site flow for a user. This causes usability interaction with the tests where the user experiences a different experience than the hypothesis accounted for.



# Option 1: All Traffic, All Tests

Do not split traffic and allow multiple campaigns to overlap user groups and pages

**With this option you are making one of the following assumptions:**

1. Running multiple campaigns simultaneously does not change how a user interacts with the site when considering both primary and secondary metrics
2. The changes within each test variation will be equally felt across a randomly distributed population and therefore have a similar impact across all variations of other active tests
3. You are able to statistically account for both primary and secondary interactions in the final test read

# Option 1: All Traffic, All Tests

Do not split traffic and allow multiple campaigns to overlap user groups and pages

## Pros

- Can be a good option for sites running few simultaneous tests that rarely overlap
- This is the most efficient approach
  - it minimizes the traffic needed and test duration time

## Cons

- Simultaneous tests can impact one another and create noise that is difficult to analyze
- It's more difficult to calculate test run times when multiple tests overlap
- This method could cause extra analysis time if tests are too close in the funnel because tests are more likely to impact one another and require cross-test analysis

# Mitigate Risk with This Approach

## Options & considerations for the All Traffic, All Tests approach

- When there is little to no overlap of the test population, running simultaneous tests should not cause an issue.
- When the tests are closer together in the user experience, or impact the same user decision, there is more chance of interference.
- While it isn't a perfectly clean approach, one of the great things about testing is the randomization of experience qualification. The influence is mitigated by the fact that we expect equal exposure to all other experiences.

# All Traffic, All Tests

For high traffic sites, this approach is not recommended

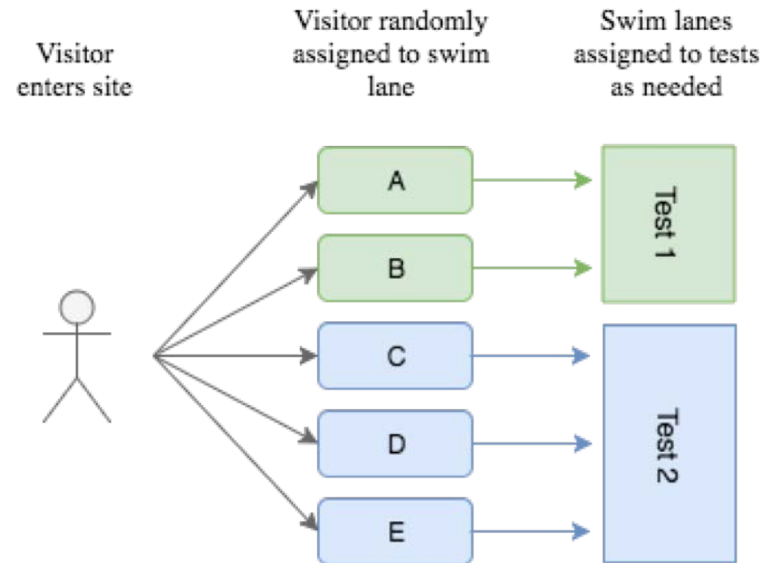
However, there are a few exceptions:

- The test experiences are far away in the funnel
- You're testing concepts that are generally agreed not to interact
- You have the data and expertise to calculate impact to primary and secondary metrics

# Option 2: Utilize Swim Lanes

Store an identifier for each user that can later be targeted as needed for future tests.

- All visitors are assigned to a group on site entry (called 'swim lanes').
- Tests then target one or more 'swim lane' depending on individual test needs.



★ NOTE: The visitor remains in the assigned swim lane throughout his or her entire visitor duration

# Option 2: Utilize Swim Lanes

## Approaches to swim laning

**Lanes:** Utilize a set number of ‘lanes’ from which tests can run in and potentially to assign to specific testing groups. There can be any number of lanes, but it is best to keep these to manageable quantities.

This also provides the ability to assign specific names. For example, if one team will always be using a specific lane, you can assign an identifiable name.

**Random Number:** Utilize a random and static number assigned to each user. This number is a decimal number and one that can be targeted within an audience.

The benefit of this approach is that you can have an almost infinite number of swimlanes, and each test can be targeted to precisely the amount of traffic required. In general, and in this case, the random number can be 0-99, and targeting would utilize a combination of greater than and/or less than qualifiers to create specific groups for each test.

# Option 2: Utilize Swim Lanes

## Test Duration & Scheduling - Swim Lanes vs. No Swim Lanes

*Example: You have two tests that need to run on the homepage during the month of September.*

You know, after calculating test run time, that both tests will need to run for 10 days each. However, best practice is to run all tests for at least 14 days to account for seasonality. So, by taking a swim lane approach, you will be able to run two tests in 20 days instead of 2 tests in 28 days

### Swim lane approach:

#### **Test 1 - September 1-20**

Targets users with a visitor ID ending in 0-4

- A: Control
- B: Test

#### **Test 2 - September 1-20**

Targets users with a visitor ID ending in 5-9

- A: Control
- B: Test

*The swim lane approach allows you to save 10 days worth of traffic.*

### Non-swim lane approach:

#### **Test 1- September 1-15**

Targets all visitors

- A: Control
- B: Test

#### **Test 2- September 16-30**

Targets all visitors

- A: Control
- B: Test

# Option 2: Utilize Swim Lanes

## Swim Lane Execution

In Target, you would create a profile script that would assign a random number (1-100) to each visitor and save it in their profile. Then when the test is set up you can choose which of those "lanes" to include in the targeting.

Example script:

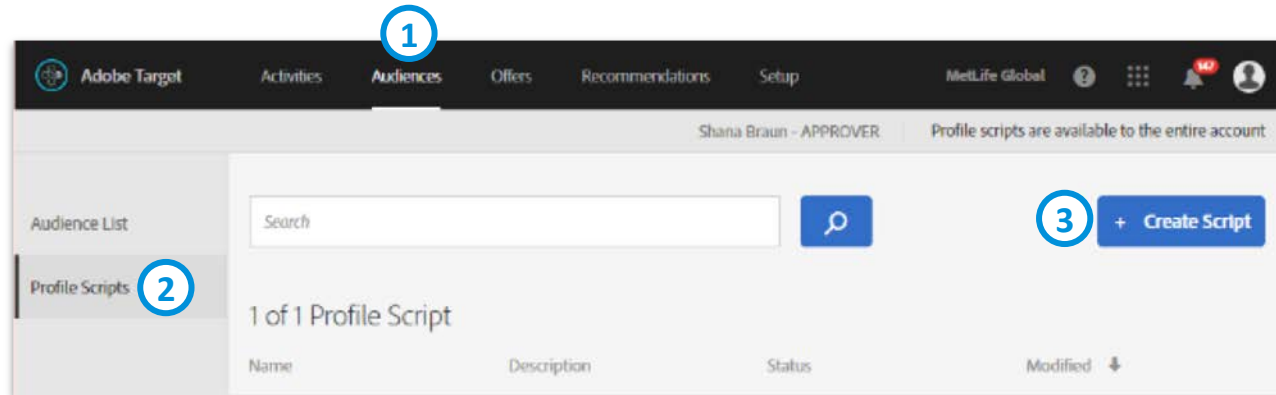
```
if (!user.get('swimlane')) {  
  var ran_number = Math.random() * 99 + 1;  
  return ran_number;  
}
```

- The script name should match the word in quotes in line 1. So this script would be named "swimlane"

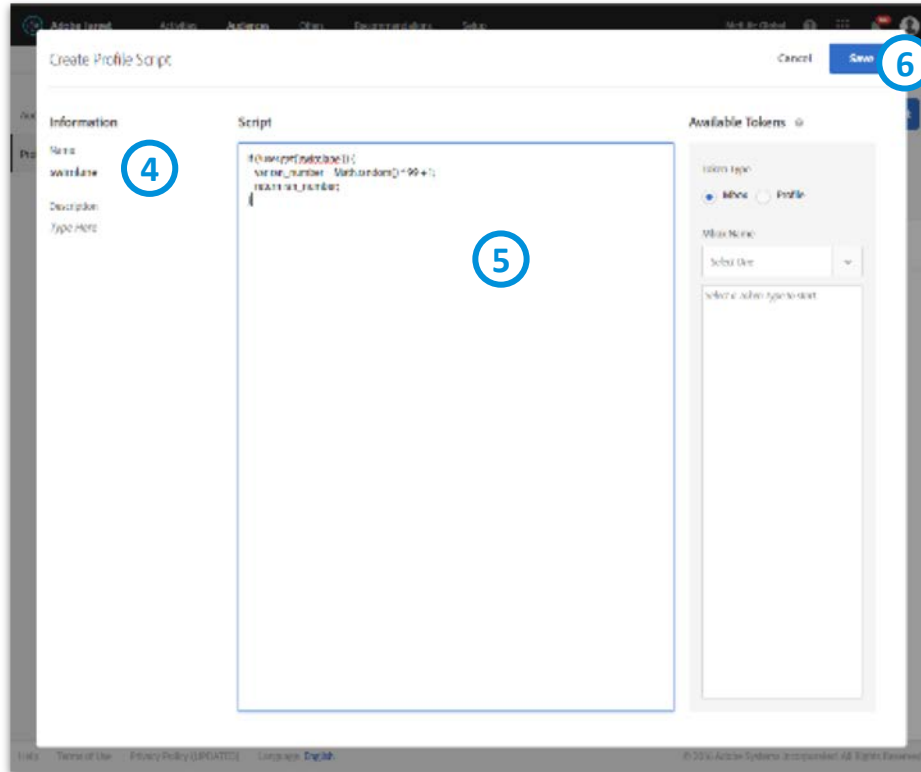


# Creating a Swimlane Profile Script

To create a profile script in Target, first navigate to the Audiences > Profile Scripts section, then click on Create Script.



# Creating a Swimlane Profile Script



Next provide the script name, script code, and save.

Example script (lanes 1-100):

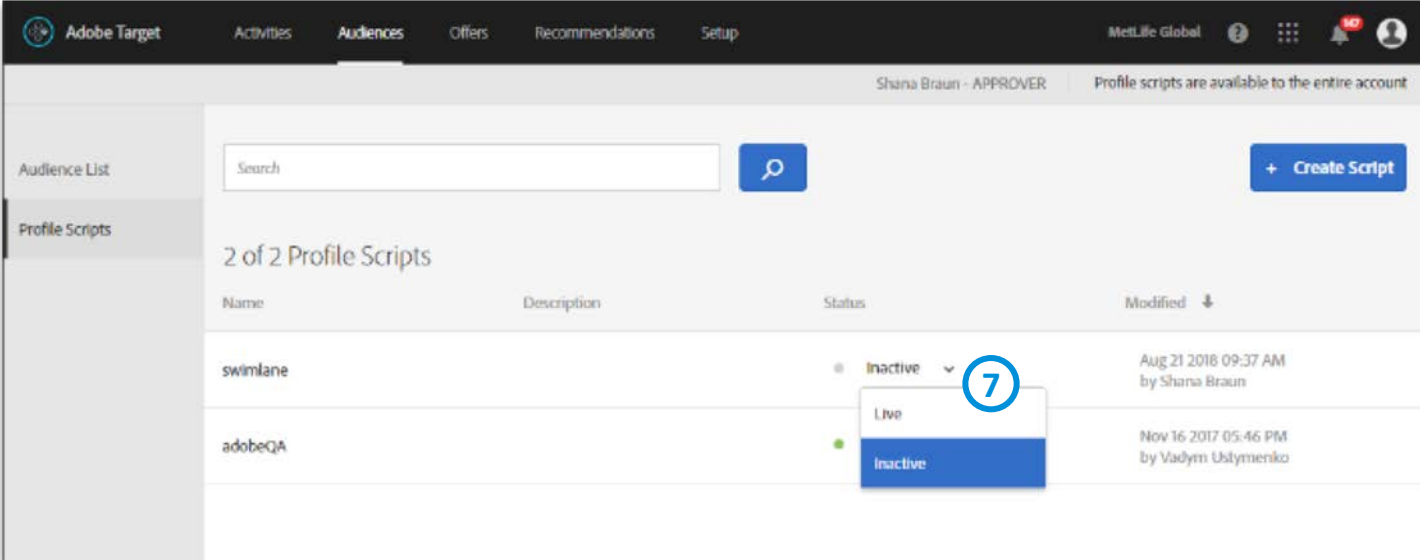
```
if (!user.get('swimlane')) {  
    var ran_number = Math.floor(Math.random() * 100) + 1;  
    return ran_number;  
}
```

\* When using this script, the profile script name must match the value in quotes on line 1 of the script (in this case, “swimlane”).

# Creating a Swimlane Profile Script

After saving, use the status column in the script list to set the script to “Live” status.

Look for the green light icon to indicate the script is working. All users will now receive a swimlane.



The screenshot displays the Adobe Target interface. The top navigation bar includes 'Adobe Target', 'Activities', 'Audiences', 'Offers', 'Recommendations', and 'Setup'. The user is logged in as 'Shana Braun - APPROVER' and has access to 'Profile scripts available to the entire account'. The left sidebar shows 'Audience List' and 'Profile Scripts'. The main content area shows a search bar and a '+ Create Script' button. Below, there is a table of profile scripts:

Name	Description	Status	Modified ↓
swimlane		Inactive	Aug 21 2018 09:37 AM by Shana Braun
adobeQA		Inactive	Nov 16 2017 05:46 PM by Vadym Ustymenko

A dropdown menu is open for the 'swimlane' script, showing options for 'Live' and 'Inactive'. A red circle with the number 7 highlights the 'Live' option.

# Targeting with the Profile Script

The screenshot shows the 'Create Audience' dialog box. At the top, there are fields for 'Audience Name (Required)' and 'Enter name', with 'Cancel' and 'Save' buttons on the right. Below these is a 'Rules' section. The first rule is under the 'Visitor Profile' dropdown and consists of three parts: a dropdown menu set to 'swimlane', a comparison operator dropdown set to 'is greater than or equal to', and a 'static value' dropdown set to '1'. Below this rule is an 'Add Rule' button. The second rule is also under the 'Visitor Profile' dropdown and consists of three parts: a dropdown menu set to 'swimlane', a comparison operator dropdown set to 'is less than or equal to', and a 'static value' dropdown set to '50'. Below this rule is another 'Add Rule' button. At the bottom of the rules section, there is a final 'Add Rule' button. The logic between the two rules is indicated by an 'AND' dropdown menu.

When creating an audience for targeting, the swimlane profile script can be found under “Visitor Profile” rules.

Add rules to restrict the audience to a range of swimlane values.

In this example screenshot, the audience will target lanes 1-50, or half of the total traffic

# Option 2: Utilize Swim Lanes

## Swim Lane Execution

Swim lane management would need to be done by the COE who is in charge of collision management. The COE's responsibilities are to:

- Document the tests and check for collision
- Keep track of swim lanes being utilized
- Determine number of swim lanes needed for a particular test
- Communicate and resolve issues with swim lane shortages by moving and adjusting tests run dates

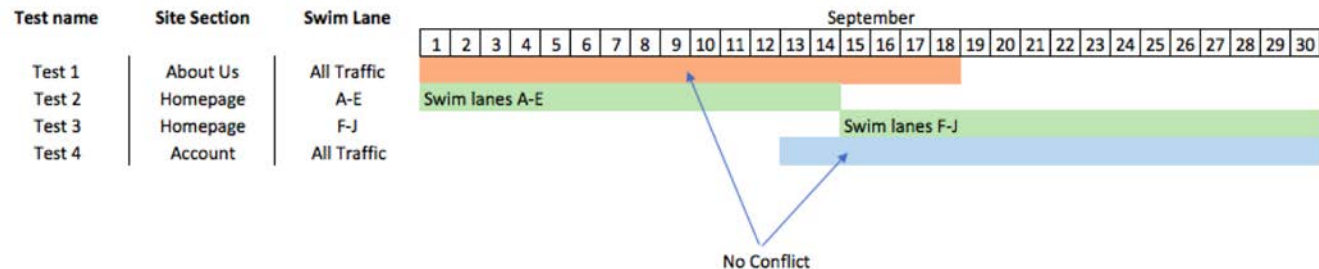
# Option 2: Utilize Swim Lanes

## Swim Lane Documentation/Test Queue

Documentation for swim lanes typically lives in the roadmap and/or collision management document. For example, if you have 10 swim lanes labeled A-J, label each test with the dedicated swim lanes within the document.

A gantt chart view is optional, but will help the document manager visualize collisions and swim lanes during the planning process.

In this example, the two homepage tests present a conflict that is solved with swim lanes:



# Option 2: Utilize Swim Lanes

Store an identifier for each user that can later be targeted as needed for future tests.

## Pros

- Tests are mutually exclusive
- Swim lanes are randomized and comprised of an accurate sample of your site as a whole
- Helps correct for 'over-testing' and allows for rest periods for swim lanes between tests

## Cons

- This method takes planning and maintenance
- Your test duration times will likely increase since swim lanes limit the traffic going to each test

# Utilizing Swim Lanes

This is the recommended approach for high traffic sites.

Utilizing swim lanes is the recommended approach as long as the following conditions can be met:

1. Swim lanes can be set up properly
2. There is a system in place for managing swim lanes
3. Traffic can support them without drastically increasing run times



# Recommendation

Move forward with implementing All Traffic, All Tests collision management strategy

Reasons we believe this strategy is best for MetLife

1. This is a good option for sites running few simultaneous tests that rarely overlap
2. This is the most efficient approach - it minimizes the traffic needed and test duration time
3. This method takes less planning and maintenance than the swim lane approach

# Recommendation

## Move forward with implementing All Traffic, All Tests collision management strategy

In general, the best guideline for mitigating collisions is to stagger tests so that multiple experiences are not running at the same time. However, some collisions are low risk.

Early and late funnel tests are typically OK to run at the same time. In addition, tests that are unlikely to impact one another can run together as well.

*Example: Drop down navigation + Account create tests*

However, keep in mind that no rule is *always* safe. So, discussions should be had for each test to mitigate collision risk.



# Important considerations

## Avoid these conflicting test scenarios

### Collision Management Considerations

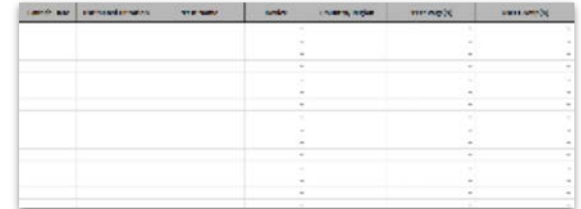
1. Multiple tests running on the same page(s)

2. Tests that could impact the amount of traffic of other tests

*a. Example: a navigation test meant to push more users to login should not be run at the same time as a login test*

3. Tests that could impact the amount of users in a defined user group should not be run at the same time as a test targeting that same user group

*a. Example: a test meant to increase logins should not be run at the same time as a test targeting logged-in users*



Test Name	Status	Start Time	End Time	Duration	Pass Rate	Fail Count
Test 1	Running	10:00	10:30	30m	95%	2
Test 2	Running	10:00	10:30	30m	90%	5
Test 3	Running	10:00	10:30	30m	98%	1
Test 4	Running	10:00	10:30	30m	92%	4
Test 5	Running	10:00	10:30	30m	96%	2
Test 6	Running	10:00	10:30	30m	94%	3
Test 7	Running	10:00	10:30	30m	97%	1
Test 8	Running	10:00	10:30	30m	93%	4
Test 9	Running	10:00	10:30	30m	91%	5
Test 10	Running	10:00	10:30	30m	99%	0

# Decision Making

## The collision decision-making cycle



Test idea is submitted either through conversation or through an intake form (WF).

The test idea is added to the collision management document

Test is considered against all tests that will overlap with it. If there are no collisions, skip to step 5. If there are collisions, move to step 4.

Refine the colliding tests until there are no more collisions. This could mean changing the scope of the test(s) or changing the run times of the test(s).

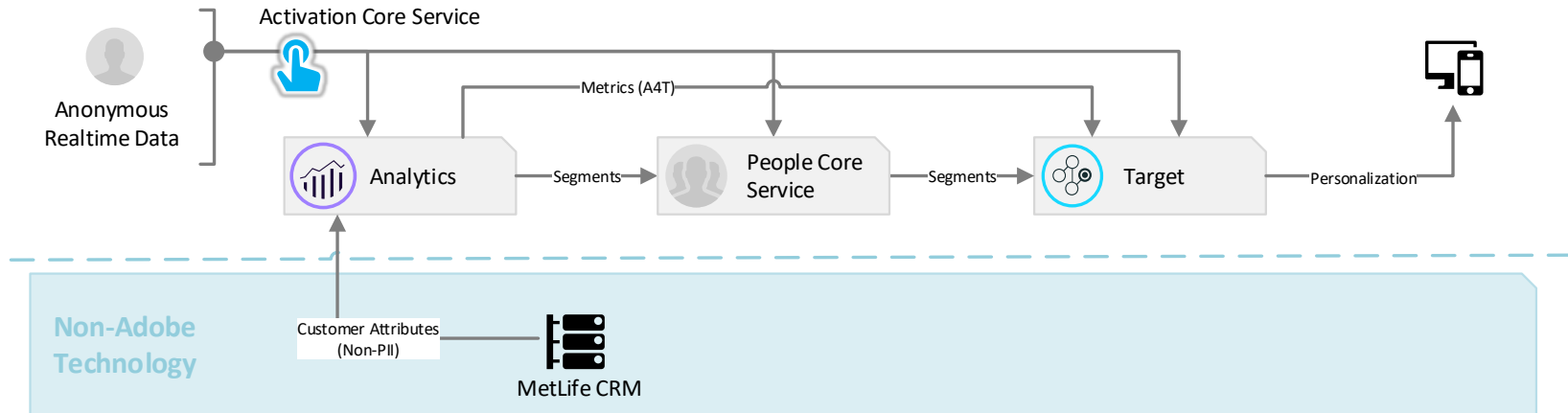
Once all collisions are managed, finalize test dates and scope of all tests in the queue.



Navigating life together

# **Adobe Target/Analytics Setup**

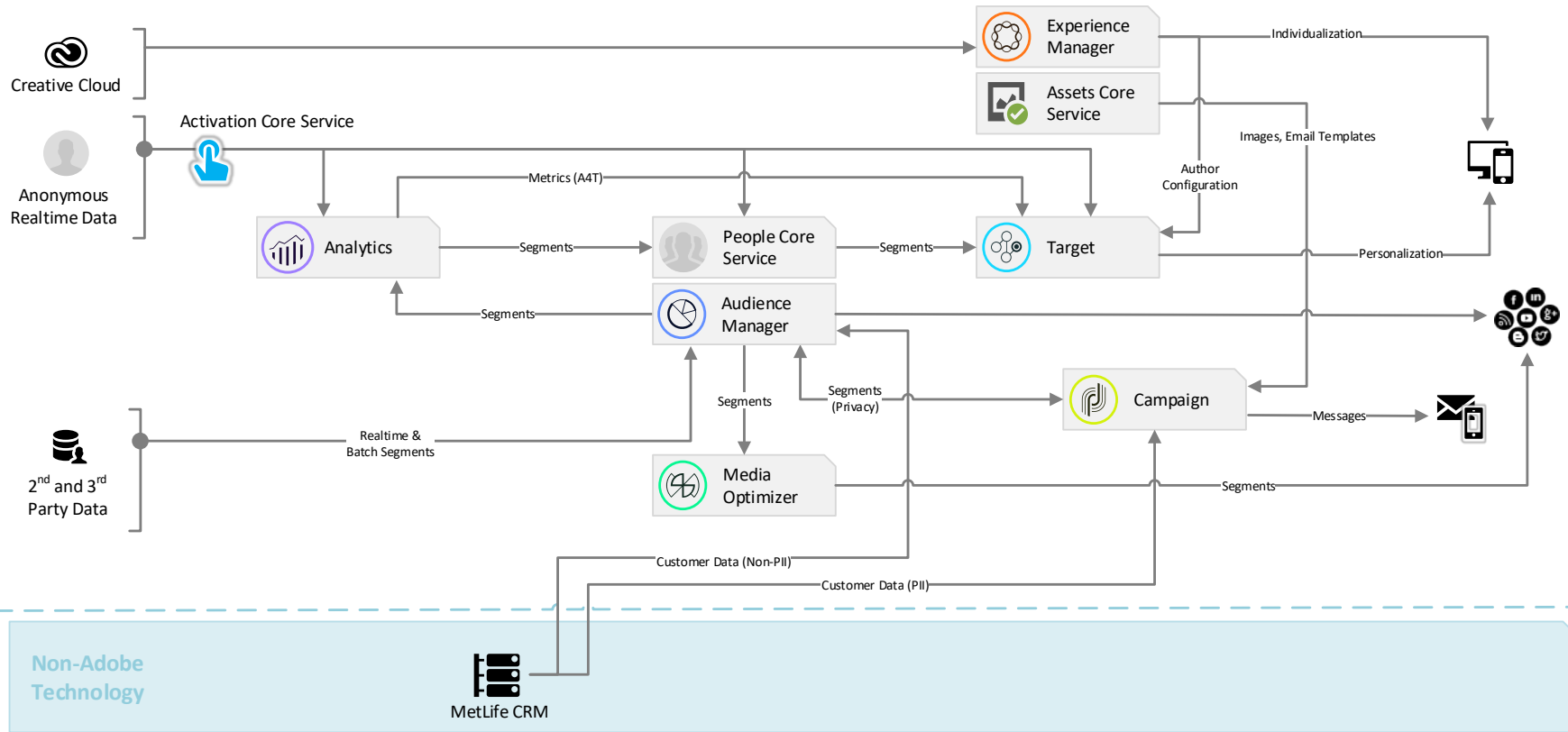
# Phase 1 Architecture – Nov Launch







# Phase 3 Architecture – Future Possibilities



# Adobe Proposed Use Cases (Long-Term)

## Phase 1

- Customize site experiences to audience segments built in Analytics
- Customize site experiences to logged-in customers based on their offline attributes

## Phase 2

- Send email / push notifications to Analytics segments
- Send immediate automated emails based on real-time Analytics triggers
- Launch A/B and Experience testing directly within AEM Author workflow
- Compose email templates in AEM, access directly in Campaign

## Phase 3

- User stitching across devices
- Build audience segments based on a combination of 1<sup>st</sup>, 2<sup>nd</sup> and 3<sup>rd</sup> Party data
- Remarket to audiences on display networks
- Leverage 3<sup>rd</sup> party data and lookalike modeling for acquisition efforts

# Topics for Setup

- Logging in
- Toggling between the two
- Verification of user access

# **Adobe Target/Analytics Overview – DTM Properties, Dependencies, Test Types**

# Topics for Setup

- DTM, How it works, and DTM Properties
- Adobe Analytics Reporting suite setup, and how it integrates with DTM
- Target, Mbox Code, Mbox Properties, and How it is integrated with DTM
- Target Activity types
- Target Audiences, and Adobe Analytics Segmentation Sharing
- Test set up best practices before launch
- Offers

# Activity Naming Conventions

Activity Naming should follow the following structure:

Environment/status-Country-LOB(if applicable)-Devteam or name-TestID (if applicable)-Name

.

For example:

QA-US-P&C-EGPlus-AB002-Landing\_Page\_Test

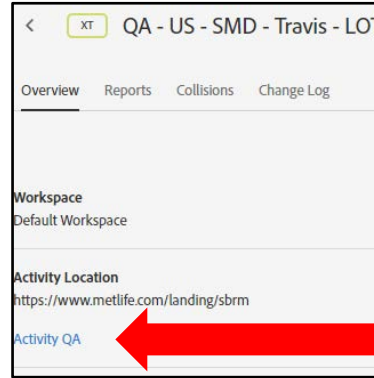
Prod-Japan-Change\_Rougo-Travis-Call\_to\_Action\_position\_test

# Testing Options

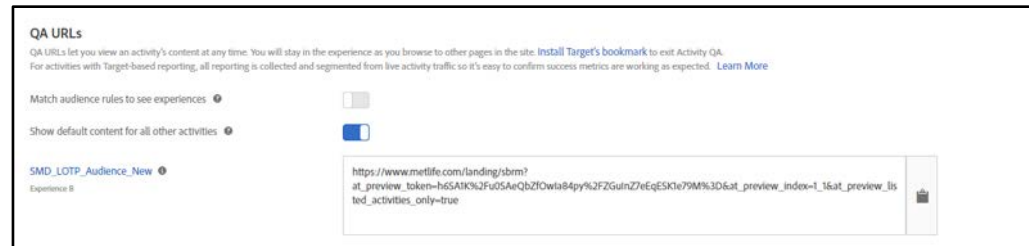
## Use the target QA link:

Click on the activity in target once it has been created. Once you click on the activity and to the activity page, there is a link for “Activity QA”. This allows you to pull up the activity view page which allows you to pull the link for the different variations of the page.

1.



2.

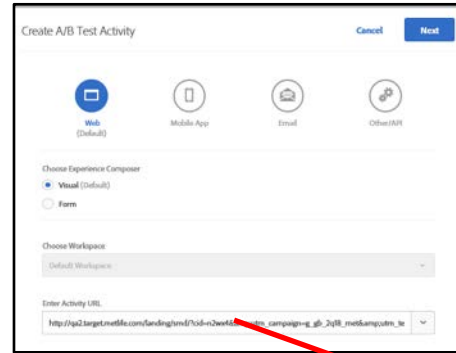


# Testing Options

## Using Lower Environment:

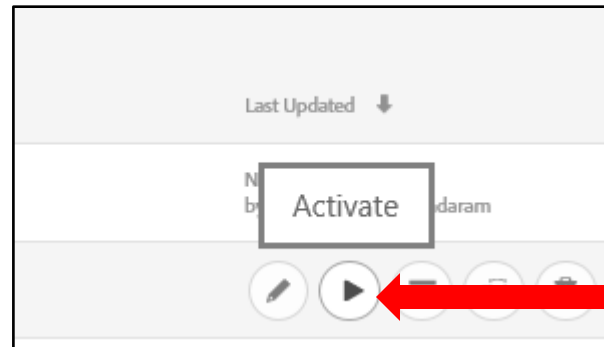
Create the test and for setting it up, use the lower environment url (for example `uat.www.metlife.com` instead of `www.metlife.com`) and publish the test to a lower environment.

1.



URL is lower environment like `qa....metlife.com` Or `uat....metlife.com`

2.





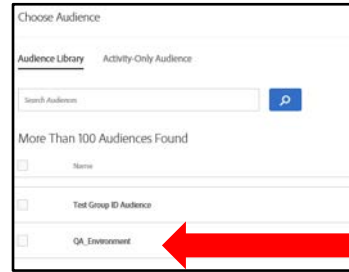
# Testing Options

## QA segment:

While creating a test, when choosing an audience to target, include the “QA environment” link. This will limit who sees the test to only people who put “env=QA” as part of the url parameters. An example url you would use to test would be structured like

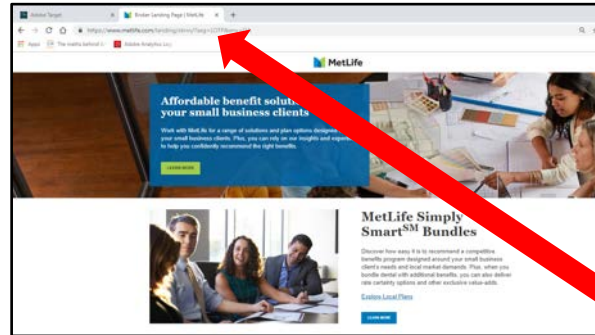
[www.metlife.com?env=QA](http://www.metlife.com?env=QA)

1.



Use “QA environment” as part of your audience when setting up test

2.



URL must contain “env=QA” as a parameter

# Deployment options for testing

## Using lower environment:

### Pro's:

- Get to see test in full working order
- Causes least amount of technical issues bringing up the test

### Con's

- Requires the page to be on lower environment
- When moved to production, must retest (sanity check)
- Could cause false positives in other project testing if not aware of this

## Using QA Segment:

### Pro's:

- Get to see test in working order in the production environment you would want it working in

### Con's:

- Most risky seeing incorrect implementation may unintentionally expose test in prod
- Sometimes false positives in tests can come up using this due to the segmentation logic of QA segment

## Using QA link:

### Pro's:

- Least risky
- Can be pulled out without activating the test
- Less issues with trying to get test to show

### Con's:

- Token technology sometimes messes with browser rendering and often needs to be flushed
- Could cause false positives

# Testing Recommendation

## Steps for testing:

1. Dev team builds test and uses QA link to view the test rendering.
2. Once test page look is ok, UAT of look should be done using the QA link.
3. Once look is approved, set up the test to run using QA segment, or a publish to lower environment (which ever is preferred for that test), and perform a secondary QA specifically around testing if other pages are affected unintentionally by the test, and to ensure that the appropriate audience is seeing the test
4. Publish to Production