

## 3. Tagging set-up

Created by Alec Cochrane, last modified by Andrea Bull on 17/Mar/19

For each test in Target (experience or A/B or other) there should be some tagging implemented in Target to ensure that the test is monitored in Adobe Analytics as well - this way the tests can be analysed against [the full range of KPIs](#) to establish whether the changes represent a net gain or not. This should be done in a number of different ways depending on what the test is.

For the campaignName value format, please see [the guidance on Impression IDs](#).

What does this code does:

- The first part of the code defines a datalayer ('targetCampaign')
- The second part of the code adds the value of the campaign ID to the datalayer
- The third part of the code ('\_satellite.track("[DirectCallRule]")') sends a signal to Adobe through DTM that any appropriate rules can trigger a tag (in this case an Adobe Analytics tag that collects the Impression ID in custom variable prop32). This is called the 'direct call rule' (if required)
- The final part of the code is set up to update a different datalayer variable for Experience Testing to show that users are 'return' users (if required)

### 1. The test launches on load of the page and is visible to the user

If the user can see the test on load of the page (eg a billboard takeover or a completely new page) then we require the test to load an Analytics tag on load

```
var targetCampaign = targetCampaign || {};
targetCampaign = {
  page: {
    campaignName: "tt:<siteID>:<JIRATicketID>:<testType>:<placement>:<testDescription>:<variantID>"
  }
};
_satellite.track("genericTestingImpressionIDWorkaround");
```

### 2. The test becomes visible to a user on an existing overlay or part of the site using Direct Call Rules (typically TDR and B&P)

If the test becomes visible to the user upon load of an existing overlay there is already a system set up to fire a 'direct call rule' to say that the overlay has loaded. Therefore we do not need an additional direct call rule - we just need to update the datalayer:

```
var targetCampaign = targetCampaign || {};
targetCampaign = {
  page: {
    campaignName: "tt:<siteID>:<JIRATicketID>:<testType>:<placement>:<testDescription>:<variantID>"
  }
};
```

### 3. The test becomes visible to a user on a new overlay

If the test becomes visible to the user on load of a new overlay then we need to fire a different direct call rule with the datalayer trigger so that we can create an impression for the new popin in Adobe:

```
var targetCampaign = targetCampaign || {};
targetCampaign = {
  page: {
    campaignName: "tt:<siteID>:<JIRATicketID>:<testType>:<placement>:<testDescription>:<variantID>"
  }
};
_satellite.track("impression-xt-popin")
```

### 4. The test becomes visible on click

If the test becomes visible to the user on clicking a link (eg a test that directs some of users to a different url), then both variations should have the following impressions in Adobe:

```
var targetCampaign = targetCampaign || {};
targetCampaign = {
  page: {
    campaignName: "tt:<siteID>:<JIRATicketID>:<testType>:<placement>:<testDescription>:<variantID>"
  }
};
_satellite.track("genericTestingImpressionIDWorkaround")
```

## A. Experience Targeting

If the user is being presented with an experience because of some of their previous actions (Experience Targeting test) then in addition to the above we require an additional datalayer variable to be added to the page to collect this information:

```
FordPersonalisation.ptemplate.set("return");
```

## B. Click tracking

Usually we will not add additional click tracking to a test as predominantly clicking on any links created within the test will be tagged as they would have been anyway. The exception to this rule is for popins where we'd like to track the click on the popin to be able to tell that it was here the user clicked. For those clicks we'd work in a similar way to the load but with a different direct call rule:

```
var targetCampaign = targetCampaign || {};  
targetCampaign = {  
  page: {  
    campaignName: "tt:<siteID>:<JIRATicketID>:<testType>:<placement>:<testDescription>:<variantID>"  
  }  
};  
_satellite.track("cta-click-xt-popin")
```