ForeSee Trigger Code Master Implementation and Specification Guide

Desktop, Mobile Web and SessionReplay

Table of Contents

Shrinkwrap License Agreement

STOP! PLEASE READ BEFORE INSTALLING ANY SOFTWARE.

By installing the ForeSee Software (as defined below), you, the purchaser of the ForeSee Software and Services (as defined below), on behalf of yourself and your affiliates, agents, employees and representatives (collectively, the "Subscriber") hereby agree as follows:

1. ForeSee Results, Inc., a Delaware corporation ("ForeSee") shall collect, process, and/or compile data from Subscriber's websites and other sources, using its proprietary software (such software, together with all later releases, updates and upgrades thereto, collectively, the "ForeSee Software"), and use such information to provide the Subscriber with access to data and reports, measuring customer satisfaction and future behaviors of users of the Subscriber's web site(s) (a "Measurement"). In addition, ForeSee shall provide any additional measurements ("Additional Measurements") requested by the Subscriber for a price agreed between the parties (the Measurement and any Additional Measurements collectively, the "Services").

2. Subscriber Licenses. To enable Subscriber to receive the Services, ForeSee grants Subscriber, and Subscriber accepts, a non-exclusive, non-transferable, license to install, store, operate and/or use the version of the current release of the client portion of ForeSee's proprietary software and Services, including: (a) any program routines or programming code that may be distributed as part of the Services, including but not limited to survey trigger code, mobile software development kit and SessionReplay code (the "Subscriber Software"); (b) any URL link that may be distributed as part of the Services (the "Subscriber Survey URL"); (c) ForeSee's data access portal, including the analytics tools and automated reporting described in an applicable Statement of Work (the "Online Portal"); and (d) ForeSee's aggregated customer satisfaction data (the "Aggregated Indices," described in Section 4.1). Subscriber agrees to use the Subscriber Software, Subscriber Survey URL, the Online Portal, and the Aggregated Indices for no other purpose than the provision of the Services to the Subscriber. Subscriber Software may be installed, stored, and operated only on websites identified in a Statement of Work or otherwise approved in writing by ForeSee.

3. Subscriber shall own all data, Measurements and Additional Measurements that result from the Services hereunder, but SUBSCRIBER HEREBY GRANTS TO FORESEE A PERPETUAL, EXCLUSIVE, ROYALTY FREE, FULLY PAID-UP, WORLDWIDE LICENSE, WITH THE RIGHT TO SUBLICENSE, TO USE SUCH DATA, MEASUREMENTS AND ADDITIONAL MEASUREMENTS IN THE PERFORMANCE OF THE SERVICES AND IN THE CREATION OF INDICES WHICH ARE COMPILED FROM AGGREGATED DATA AND MEASUREMENTS (THE "AGGREGATED INDICES"). FORESEE SHALL BE THE SOLE OWNER OF THE AGGREGATED INDICES. FORESEE HAS THE RIGHT TO USE SUBSCRIBER'S NAME IN DESCRIBING THE PARTICIPANTS OF THE AGGREGATED INDICES. ForeSee is acting only as the "data processor," and Subscriber remains exclusively the "data controller" and "data owner." As such, it is Subscriber's obligation to comply with any/all applicable privacy laws and regulations. ForeSee shall at all times comply with its Privacy Policy stated at http://www.foresee.com/privacy-policy.shtml. Subscriber shall ensure that no individualized data is passed by Subscriber to ForeSee in violation of Subscriber's published privacy policy, and Subscriber shall be solely responsible for violations of the same.

4. The Subscriber shall be responsible for the installation and maintenance of any Subscriber Software pursuant to the instructions provided to the Subscriber by ForeSee, and for Subscriber's equipment necessary to access ForeSee's Online Portal, including all computer equipment, software, telecommunications, including high-speed connections to the Internet, to the extent they are needed to operate and/or access the Services from Subscriber's site of business. For all Web Measures, Subscriber is responsible for the operation and maintenance of Subscriber's web environment. For all In-Location Measures, Subscriber is responsible for incorporating the Subscriber Survey URL into Subscriber's receipt, or otherwise facilitating delivery of the Subscriber Survey URL. For all Call Center Measures, Subscriber is responsible for incorporating the survey invitation into Subscriber's Call Center workflow. For email delivered surveys, the Subscriber Survey URL may be delivered by Subscriber or ForeSee, as described in the applicable Statement of Work. Subscriber is responsible for ensuring that any email addresses provided by Subscriber may be used without violation of law or regulation. If Subscriber purchases ForeSee's SessionReplay product, Subscriber shall be responsible for identifying web pages that collect and/or display potentially secure information, such as names and account information ("Personal Data").

5. The price and payments terms that Subscriber shall pay to ForeSee are contained in a separate written document which is herein incorporated in its entirety by reference.

6. Subscriber agrees that FORESEE SHALL NOT BE LIABLE ON ACCOUNT OF ANY ERRORS, OMISSIONS, DELAYS, OR LOSSES UNLESS CAUSED BY FORESEE'S GROSS NEGLIGENCE OR WILLFUL MISCONDUCT. IN NO EVENT SHALL FORESEE BE LIABLE FOR INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES. IN NO EVENT SHALL THE TOTAL AGGREGATE LIABILITY OF FORESEE FOR ANY CLAIMS, LOSSES, OR DAMAGES ARISING UNDER THIS AGREEMENT EXCEED THE TOTAL CHARGES PAID TO FORESEE DURING THE TERM, EVEN IF FORESEE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH POTENTIAL CLAIM, LOSS, OR DAMAGE. THE FOREGOING LIMITATION OF LIABILITY AND EXCLUSION OF CERTAIN DAMAGES SHALL APPLY REGARDLESS OF THE SUCCESS OR EFFECTIVENESS OF OTHER REMEDIES.

7. Notwithstanding any statement to the contrary in any written document incorporated herein by reference or the pattern and practice of ForeSee and Subscriber, the term of this Agreement shall continue only for such periods for which ForeSee receives the agreed payment from Subscriber (the "Term").

8. This Agreement may be terminated: (1) by either party upon the material breach by the other party of any of such other party's obligations hereunder, which breach has not been cured within 15 days after the breaching party has received notice thereof, or (2) by ForeSee upon 10 days' notice in the event of any delinquency of Subscriber in payment of amounts due hereunder. In the event of any such termination, the license(s) granted to the Subscriber pursuant to Section 2 shall terminate immediately.

9. Subscriber may not sell, transfer, assign, or subcontract this License Agreement to another party without the prior written consent of ForeSee. Any act in violation of the foregoing shall be null and void.

10. Subscriber shall be responsible for and shall promptly reimburse ForeSee for the payment of all sales, use, excise, or value-added tax or other assessments (including interest and penalties, but with respect to penalties, only if such penalties are incurred as a result of actions in which Subscriber has concurred or participated) and duties imposed by all governmental entities based on the Services rendered by ForeSee to Subscriber, provided that ForeSee shall be liable for any income taxes on or measured by net income or gross receipts of ForeSee.

11. Each party shall, at its own expense, comply with any governmental law, statute, ordinance, administrative order, rule, or regulation relating to its duties, obligations, and performance under this Agreement and shall procure all governmental licenses and pay all fees and other charges required thereby.

12. This contract is made under the laws of the State of Michigan and shall be interpreted, construed and enforced in accordance with the law of Michigan, without giving effect to its choice of law principles. The parties hereby agree to the exclusive jurisdiction of the state courts of Michigan and any federal court sitting in Michigan.

13. If any provision of this Agreement is held by a court of competent jurisdiction to be illegal, invalid, unenforceable, or otherwise contrary to law, the remaining provisions of this Agreement shall remain in full force and effect.

14. Survival. Sections 3, 5 and 6 shall survive the termination of this Agreement.

Implementation Summary

You have been provided a JavaScript library which will be hosted on your web or content server. This library contains JavaScript, HTML, CSS, images and one .swf file. The purpose of this JavaScript library is to present a survey invitation to your site visitors based on the configured requirements. Once the site visitor has become eligible, based on those requirements, the code will then present the survey invitation and subsequent survey questionnaire (hosted by ForeSee) to the site visitor.

In most cases, the ForeSee Implementation team will have worked with your business and possibly a technical contact to pre-configure the JavaScript library so that there are minimal changes required to implement this code.

Some of the pre-configured requirements are:

- When and where to present an invitation
- Where to prevent an invitation from appearing
- How frequently to present the invitation
- The look and feel of the invitation

This library may also contain logic to collect additional client specific data points that are available on the site to be passed along with the survey results through the use of customer passed parameters (CPPs).

The basic implementation is simple:

1. Load ForeSee files delivered in .zip to web or content server.

2. Modify files path in the JavaScript files to point to the proper location of the hosted files. **NOTE:** You will want to change the path in the following JavaScript files: foresee-alive.js, foresee-test.js, foresee-tracker.js, foresee-trigger.js, foresee-qualifier.js (if applicable).

3. Add a single line of JavaScript to your pages (with the path pointing to the proper file location):

   ```
   <script type="text/javascript" src="/foresee/foresee-trigger.js"></script>.
   ```

4. Test the code to ensure the survey functions as expected.

Things that may impact implementation and require additional effort:

- Separating Trigger Code by type
- Separating Trigger Code files by Environment
- Hosting the code from a third party domain
- Using a tag manager
- Changing sampling parameters
- Adding to the exclude list
- Adding new customer passed parameters

The most common implementation issues:

- Invite not displaying? Check the code, Check cookies, Check files paths, Check pooling.
- Tracker frozen, survey is not presenting? Check files paths.
- Text boxes are missing on test page (fsrtest.html)? Check files paths.
- Code is working but survey looks like it's a placeholder? Survey is not completed or mapped yet.

When in doubt please feel free to contact our implementation team.

Introduction

You have been provided a JavaScript library (trigger code) that can be pre-configured to accommodate the presentation of surveys for Desktop and Mobile (Tablet and Phone) measures. This same code can also be configured to contain the SessionReplay product. *This document does not contain instructions for mobile app implementations.*

This document is a guide to implementing the trigger code to meet your business objectives. However, each engagement has its own unique characteristics so if you need some personal guidance please do not hesitate to contact our implementation team.

Whenever possible, the ForeSee implementation team tests the trigger code via proxy technique on your **production** website as long as it is publicly available. It is tested in IE, Firefox, Chrome and Safari to ensure the survey invitation process and survey display meets the business objectives communicated by our clients.

Because every clients' testing environment, deployment process and testing process is unique it is highly recommended that additional testing be done PRIOR to deploying this code to production.

It is not **un**common that our clients need a code update from ForeSee prior to going live. The implementation services team is here to support you during this process.

Basic Implementation Steps

Whether you are implementing survey trigger code for your website, mobile website, or survey trigger code that includes SessionReplay the basic implementation is the same:

- Host the ForeSee assets (JavaScript, HTML, CSS, images and one .swf file.) on your web or content server.
- Insert a JavaScript reference included globally across your website.

This document will walk you through the basic steps for implementing the trigger code in addition to the specific information we think is valuable for testing the implementation.

Step 1: Receive survey trigger code

These instructions are typically shipped with a link to a zip file (like the one below) containing the ForeSee survey trigger code library. The library typically includes JavaScript, HTML, CSS, images and one .swf file that make up the survey invitation experience. Unless you have requested sample code, the library sent to you is pre-configured to meet the specific business needs and should not require additional configuration unless requirements have changed after delivery. The code has been tested on the production site(s) to confirm proper survey behavior, however some changes may be required if your DEV and QA environments do not mirror the behavior of production.

https://demo.foreseeresults.com/_code/sample/foresee_[clientname]_[date].zip

Step 2: Save files to the Web or Content Server

This is "first-party" code meaning it's hosted by you, the client. In the majority of implementations it is a requirement that the files be hosted on the domain being measured. There are some exceptions to this rules and if hosting on a separate domain is a requirement please read the following section (instructions here).

The files in the zip are typically packaged in a folder called "/foresee/" (all lower case). Extract the /foresee/ folder from the zip file and save to the web server so that the files are available at: www.[yoursite.com]/foresee/. If you place the files in a different location you will need to update the files paths within the JavaScript code (instructions here).

Step 3: Insert JavaScript code reference(s) into your site

Remember: Unless previously discussed, this reference should be on every page of the site. Global JavaScript includes work great!

There are two options for implementing. Either one will work equally well. The first option is our standard and uses just one line of JavaScript so it requires a little less implementation. The second option may be useful if you have areas of your site that tend to load slowly.

Option 1:

This is the standard implementation option using just one file reference. In most cases this method will work fine.

Paste the following line of code as high up on the page as allowed. We recommend placing it between the <head> and </head> tags for optimum performance, however, this code will typically run fine from anywhere on the page.

```
<HEAD>
<script type="text/javascript" src="/foresee/foresee-trigger.js"></script>
</HEAD>
```

Note: Be sure to change the files path to point to the appropriate location of the files.

Option 2:

This option should be used if the site uses a large amount of scripts and has areas that tend to load slowly. The OnExit survey relies on timers to know when the user has left the site before presenting the survey. Pages that load slowly may cause a timer to falsely identify an “exit” condition. This second option allows for the larger foresee-trigger.js file to be placed at the bottom of the site after other slower loading scripts, while the smaller foresee-alive.js is placed near the top above slower loading scripts. The foresee-alive.js is a very small file used only to alert the code that the session is still alive and to check for the foresee-trigger.js file to load before considering an exit condition.

Paste the reference to the foresee-alive.js file as high up on the page as allowed. We recommend placing it between the <head> and </head> tags for optimum performance.

The second reference to the foresee-trigger.js file can be placed anywhere lower on the page.

```
<HTML>
<HEAD>
<script type="text/javascript" src="/foresee/foresee-alive.js"></script>
</HEAD>
<BODY>

</BODY>
<script type="text/javascript" src="/foresee/foresee-trigger.js"></script>
</HTML>
```

Note: Be sure to change the files path to point to the appropriate location of the files.

Step 4: Testing Basic Survey Behavior and Presentation

Recommended Testing:

- Ensure you can receive an invitation and get a survey.
- Ensure that the survey invitation does not present in areas of the site that have been deemed excluded areas.
- Ensure no errors or conflicts are occurring on the site.
- Ensure that the survey invitation, tracker (used for desktop onExit) and any other assets specific to the implementation look and load correctly (CSS loaded).
- Ensure you are able to get redirected to a ForeSee hosted survey questionnaire.

The basic steps to test:

1. **Clear cookies and cache** (In some browses, such as IE, clearing cookies can be difficult. Be sure "preserve favorites" is not checked when clearing cookies, the site you are testing is likely a "favorite").

2. Within your browser, navigate to the hosted fsrtest.html test page. Take note of the Loyalty Factor *(required page views)* and sampling percentage. See the next section for more details on the fsrtest.html page.

3. Load a page that contains the ForeSee code. When you navigate to a page equal to the number specified as your Loyalty Factor (assuming 100% sampling) you should see an invitation (as long as the page is not on the exclude list). If not using 100% sampling you have a chance to receive the invite on any page once the loyalty factor requirement has been met.

4. Depending on the survey requirements once accepting the survey invitation the questionnaire may either present immediately (InSession) OR when the user leaves the site (onExit). When configured for onExit of site, once invited, the survey questionnaire can be triggered by (1) closing the parent browser, (2) navigating to a different domain in the parent browser or (3) navigating to a different subdomain or protocol that does not contain the foresee code.

Some websites may have custom scenarios or logic set up beyond these steps. In this case, it may be useful to discuss the business requirements ahead of time to have a better idea of how to test.

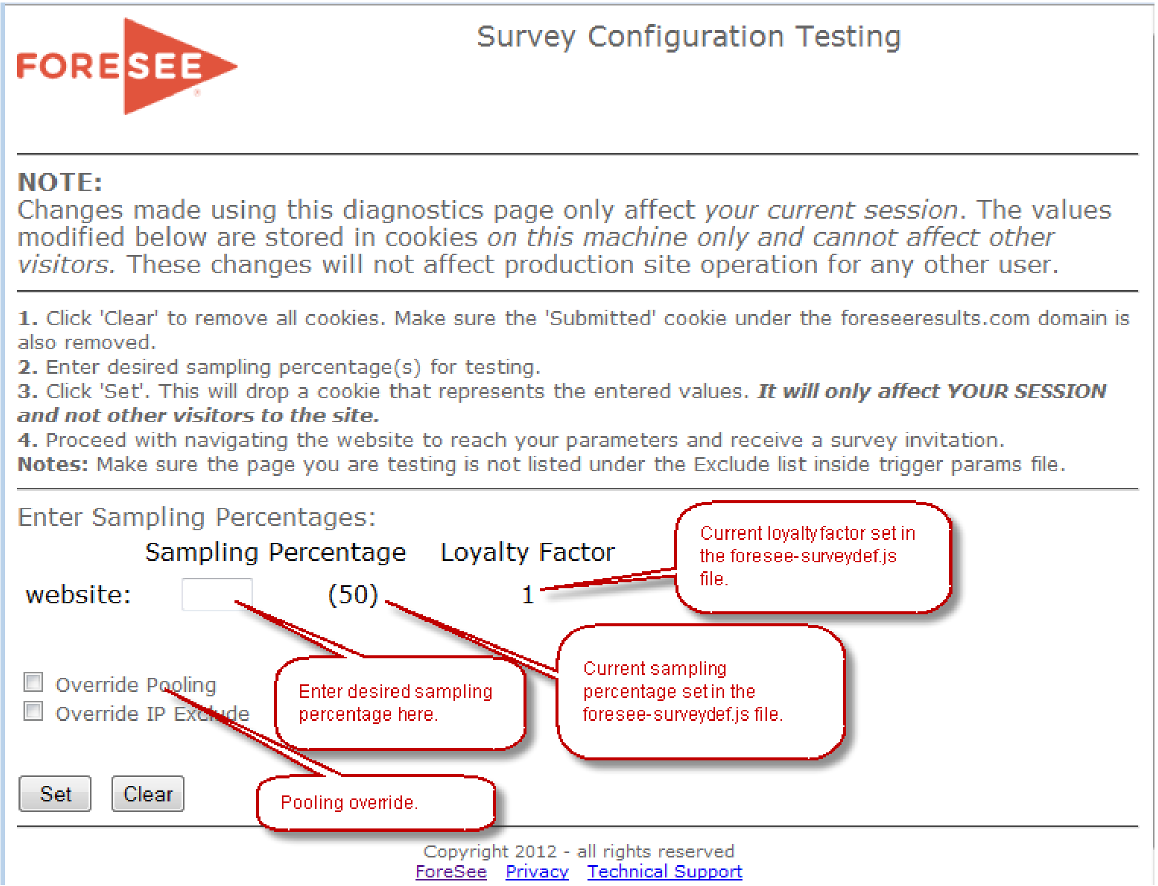
More detailed testing and troubleshooting steps can be found below.

Detailed Implementation Testing Steps and Troubleshooting

The survey and SessionReplay code is shipped pre-configured for the business rules related to your **production** environment. This means that in order to receive a survey invitation it is possible that a very low sampling percentage or pooling number will decrease the chance of your tester receiving an invitation. This may make testing difficult. Therefore, in order to receive the invitation/survey while testing an administration page (*Figure 3*) is included with this code package which will allow the invitation to be displayed at a frequency of your choice and **for your session only.** There is an option to override the Pooling percentage (if less than 100%) which is necessary for the session to be recorded and the survey invitation to appear. See more about Pooling below.

Important: Using this test page does not change the sampling percentage in the JavaScript code and therefore doesn't affect your site visitors. Please keep in mind that when you clear cookies and cache between each test you may need to navigate back to this test page to reset the increased sampling parameters.

If the code has been implemented using the default directory structure, this page would be located at http://www.[yoursite].com/foresee/fsrtest.html. If the files have been moved to a different location adjust the path to the test page accordingly. Instructions for use are provided directly on the page. This page must be loaded to the same location as the other ForeSee files to function properly. It will not work if it cannot pull in the required JavaScript files.



Survey Configuration Testing

NOTE:
Changes made using this diagnostics page only affect *your current session*. The values modified below are stored in cookies *on this machine only and cannot affect other visitors.* These changes will not affect production site operation for any other user.

1. Click 'Clear' to remove all cookies. Make sure the 'Submitted' cookie under the foreseeresults.com domain is also removed.
2. Enter desired sampling percentage(s) for testing.
3. Click 'Set'. This will drop a cookie that represents the entered values. ***It will only affect YOUR SESSION and not other visitors to the site.***
4. Proceed with navigating the website to reach your parameters and receive a survey invitation.
Notes: Make sure the page you are testing is not listed under the Exclude list inside trigger params file.

Enter Sampling Percentages:

	Sampling Percentage		Loyalty Factor
website:		(50)	1

□ Override Pooling
□ Override IP Exclude

Current loyalty factor set in the foresee-surveydef.js file.

Enter desired sampling percentage here.

Current sampling percentage set in the foresee-surveydef.js file.

Pooling override.

Copyright 2012 - all rights reserved
ForeSee Privacy Technical Support

Figure 1 - Survey Test Page

Verify Survey is working (Desktop and Mobile Websites)

Note: Tools used in this testing are Firefox with add-ons: Firebug, Firecookie and JSView. Internet Explorer 8 +, Chrome and Safari have similar tools.

1. **Clear cookies and cache** (In some browses, such as IE, clearing cookies can be difficult. **Tip:** Be sure "preserve favorites" is not checked when clearing cookies, the site you are testing is likely a "favorite").

2. **Get a survey invitation to appear.** Using the fsrtest.html test page, enter 100 in the sampling percentage box to set a cookie which will overwrite the sampling for your test. Check the override pooling option if you have SessionReplay. Take note of the Loyalty Factor *(required page views),* then either visit a few pages or reload the page to meet the page view requirement. The page view requirement will be listed on the test page under Loyalty Factor (see Figure 1 – Survey Test Page).

   Load a page that contains the ForeSee code. When you navigate to a page equal to the number specified as your Loyalty factor (assuming 100% sampling) you should see an invitation. If not using 100% sampling you have a chance to receive the invite on any page once the loyalty factor requirement has been met.

   Tip: Do not reload pages too quickly. The code loads after the DOM ready event so there may be a slight delay from the time you open a page to the invitation displaying based on page load time.

   If you do not see an invitation with 100% sampling when meeting the Loyalty Factor requirement:

   a. View source and verify the JavaScript references are on the page (see Step 3 above) and are being loaded correctly.

      i. Search for '**trigger'** and you should see a reference to foresee-trigger.js.

      ii. Confirm that the foresee-trigger.js file and foresee-surveydef.js file have been loaded * (The foresee-trigger.js file dynamically loads the foresee-surveydef.js **after** the DOM Ready event.)

      **We recommend reviewing the content of each file to ensure that the JavaScript has been called correctly and is not displaying the html for a 404 file not found page. A 404 may occur if the assets are not in the location the file path references in each of the JavaScript files.*

   b. Ensure the following cookie is loaded:

      fsr.s – this is our session level cookie. The contents should look similar to:

```
fsr.s {"v2":-2,"v1":1,"rid":"d036702-53702257-2645-8fd6-
921ea","to":3,"c":"http://www.yoursite.com/","pv":1,"lc":{"d0":{"v":1
,"s":true}},"cd":0,"sd":0,"l":"en","i":0}
```

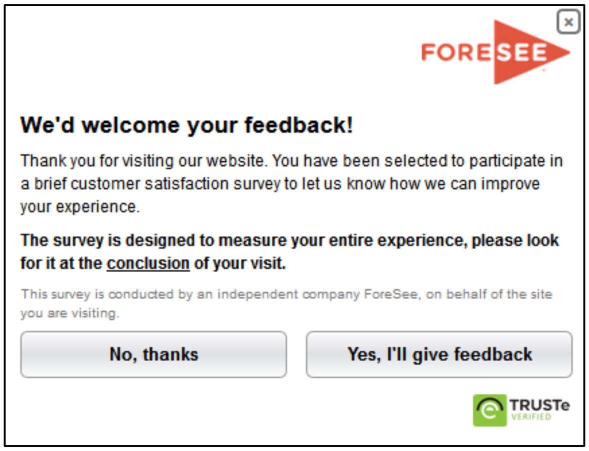
   c. Verify that you have cleared the cookies and cache.

      i. There should not be a persistent fsr.r cookie set to the domain.

      ii. The fsr.s cookie should not have an "i" value of 1 or -1. If it does, the invitation has displayed (or attempted to) previously and you will not be invited.

      iii. Clear cookies and try again if either of these situations are true.

If you see the invitation, here are examples of what you would see:



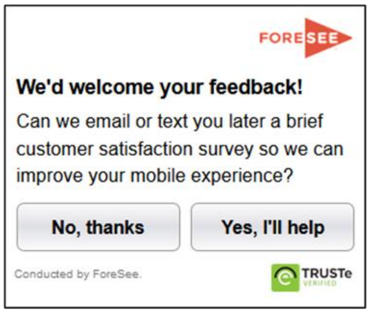
We'd welcome your feedback!

Thank you for visiting our website. You have been selected to participate in a brief customer satisfaction survey to let us know how we can improve your experience.

The survey is designed to measure your entire experience, please look for it at the conclusion of your visit.

This survey is conducted by an independent company ForeSee, on behalf of the site you are visiting.

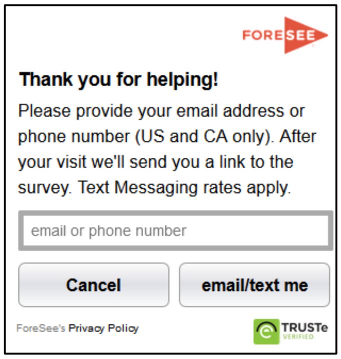
Figure 2a –Desktop invite



We'd welcome your feedback!

Can we email or text you later a brief customer satisfaction survey so we can improve your mobile experience?

Figure 2b – Mobile invite #1



Thank you for helping!

Please provide your email address or phone number (US and CA only). After your visit we'll send you a link to the survey. Text Messaging rates apply.

Figure 2c - Mobile invite #2

When the invitation is displayed, the fsr.r (repeat days) cookie is dropped. This cookie prevents a user from being invited again for X days *(default 90)*. **Tip:** If the repeat days was set to have a different number of days for a user declining, the value will change when they decline. The value of the fsr.r cookie will look similar to the following:

```
fsr.r {"d":90,"i":"1292956831475_954940","e":1293563385893,"s":1}
```

This fsr.r cookie will need to be cleared after each test or you will be unable to receive another invite.

3. **Accept the invitation** (**Tip:** Once you accept the invitation, there will be a value in the fsr.s cookie "i":1. If you decline the invitation the value will be:"i":-1. This setting indicates that there has already been an invitation displayed, if the “i” value is set to 1 or -1 the fsr.s cookie will need to be cleared to receive another invitation.)

   - **In Session Survey (Desktop and Mobile)**
     If you are using an “in session” method to present the survey, you will see the questionnaire immediately after accepting the invitation. ***At this point In Session Testing is complete.*** The survey questionnaires are hosted by ForeSee and tested prior to going live so there is no need to submit a survey to complete the testing of the code. If a survey (or placeholder) does not open, please contact support.

   - **OnExit Survey (Mobile)**
     If you are using the mobile onExit method you should receive a second invitation (figure 2c) immediately following the first (figure 2b). (**Tip:** To test mobile you must either be testing on a mobile device OR overwriting the browsers’ user agent on the desktop to appear as a mobile device. Otherwise the mobile invitation will not display on the desktop.) The second invitation provides a place for the user to enter either an email, phone number or both (Text messaging is currently only supported for US and CA).

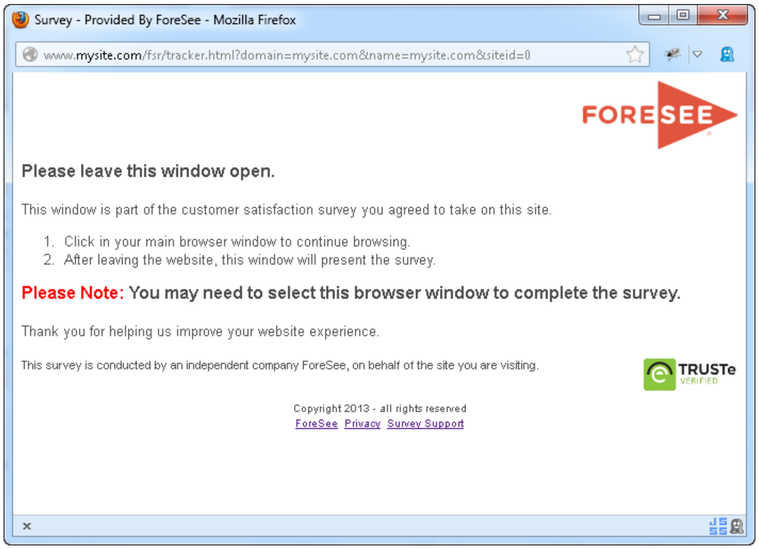
     a. Enter an email address or telephone number into the box and submit.

     b. Navigate away from the website by closing the browser or navigating to another domain.

     c. After a configurable period of time (configured by ForeSee, default 5 minutes) a survey link should be texted or emailed to the address or number entered. Click through the link to open the survey.

     At this point Mobile (Phone & Tablet) OnExit testing of the code is complete. The survey questionnaires are hosted by ForeSee and tested prior to going live so there is no need to submit a survey to complete the testing of the code. If a survey (or placeholder) does not open, please contact support.

- **OnExit Survey (Desktop)**
  If you are using the **desktop** "onExit" method to present the survey, you should see the tracker.html file launched in a new window. This window is used to monitor the parent window and display the questionnaire when an exit condition occurs.


Survey - Provided By ForeSee - Mozilla Firefox
www.mysite.com/fsr/tracker.html?domain=mysite.com&name=mysite.com&siteid=0
FORESEE
TRUSTe VERIFIED

Please leave this window open.

This window is part of the customer satisfaction survey you agreed to take on this site.

1. Click in your main browser window to continue browsing.
2. After leaving the website, this window will present the survey.

Please Note: You may need to select this browser window to complete the survey.

Thank you for helping us improve your website experience.

This survey is conducted by an independent company ForeSee, on behalf of the site you are visiting.

Copyright 2013 - all rights reserved
ForeSee Privacy Survey Support

Figure 3 – On Exit Tracker window

4. **Observe the Tracker Window.** Depending on the browser type and version the tracker may be positioned either in front of behind the parent window. The verbiage on the window should provide instructions on how to navigate the window if it presents above the parent (**Note:** *In IE you may see a hide this window button on the tracker, this will likely not be visible on other browsers unless future browser updates allows for the functionality)*.

   When the tracker is opened the fsr.a cookie should have loaded and its value should continue to be updated. The fsr.a cookie is the trackers' timer. Its purpose is to actively monitor the parent window to determine when the user leaves the site. Once this cookie loses its connection to the parent the survey will load within the tracker window.

5. **Invoke an Exit Condition.** Exit conditions happen when the following events take place: (1) closing the parent browser, (2) navigating to a different domain in the parent browser, (3) navigating to a different subdomain or protocol that does not contain the foresee code.

   The following files should have been loaded in the tracker window: foresee-tracker.js, foresee-surveydef.js and foresee-tracker.css. Note: foresee-trigger.js file will **NOT** be loaded in this window. (**Tip:** If these files have not been loaded, the files path in the foresee-tracker.js may be configured incorrectly causing the tracker to freeze and unable to load the survey.)

6. **Tracker will load the survey.** ***At this point Desktop OnExit testing of the code is complete.*** The survey questionnaires are hosted by ForeSee and tested prior to going live so there is no need to submit a survey to complete the testing of the code. **Note:** Some browsers have settings which to not allow the tracker to gain focus forward when changing domains, in these cases you will need to navigate back to the survey manually. In Chrome, you may notice an alert box is used to pull the window forward. As new browsers and versions are released this behavior may vary. If a survey (or placeholder) does not open, please contact support.

Verify SessionReplay is working

To test SessionReplay follow the same steps as above, however, there are a few additional points to consider when testing:

- If your testing environment is NOT available to the public the SessionReplay code will still run. However, replay movies will not be created because the process will be unable to go back to the site and collect the assets of the page such as images and style sheets. This will cause the movie processing to be incomplete.

- **Get a survey invitation to appear with SessionReplay (Using Pooling).** When testing SessionReplay-enabled code you must navigate to the fsrtest.html test page prior to allowing the fsr.s cookie to be set by hitting a page containing code. You will need to select **Override Pooling** when entering 100 in the sampling percentage box to set a cookie which will overwrite the sampling for your test. If you do not do this it may be very difficult to get a survey.

  Pooling: Pooling is a concept introduced with SessionReplay. This concept is similar to the sampling percentage where a random number is generated to determine who should receive an invitation. However, with pooling we determine replay eligibility on the first page. If a user is not selected on the first page they will not receive an invitation. If they are selected on the first page they will then be eligible to receive the invitation if they fall within the loyalty and sampling parameters.

1. **After accepting the invitation:** Navigate around the site and collect some events by clicking on a few pages and objects.

   Tip: You can check the capturing of SessionReplay data if you open a JavaScript console and type in window.name and click enter. The data returned will start with **FSR_** followed by other data collected.

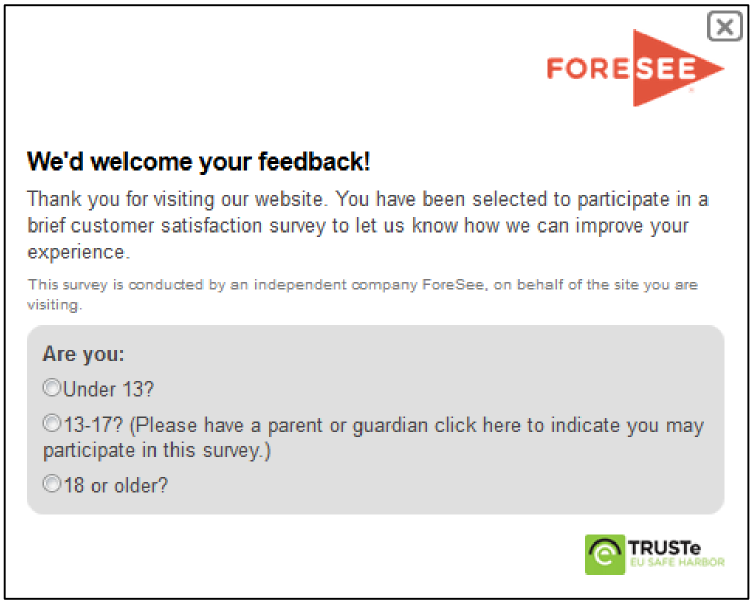
2. **Watch the replays in the ForeSee Portal the following day.** If SessionReplay seems to be recording and transmitting correctly then replays should begin showing up in the portal the following day after a survey submission. **Note**: If testing was done in an internal environment with no public access our servers will be unable to retrieve your site images, scripts and style sheets. This will result in strange broken-looking replay movie. This is to be expected in this case. Everything will look correct once deployed to public environment.

Additional Survey Options

The trigger code contains many advanced features which can be configured directly in the JavaScript. Listed below are some of the common features that may be included in the code.

Pre-invitation qualifier

The pre-qualifier page is used to determine if a user is eligible to receive a survey or not. Based on the selection the user will either be redirected to a cancel page or they will continue to the invitation. The text can be modified directly in the foresee-surveydef.js file.



We'd welcome your feedback!

Thank you for visiting our website. You have been selected to participate in a brief customer satisfaction survey to let us know how we can improve your experience.

This survey is conducted by an independent company ForeSee, on behalf of the site you are visiting.

Are you:

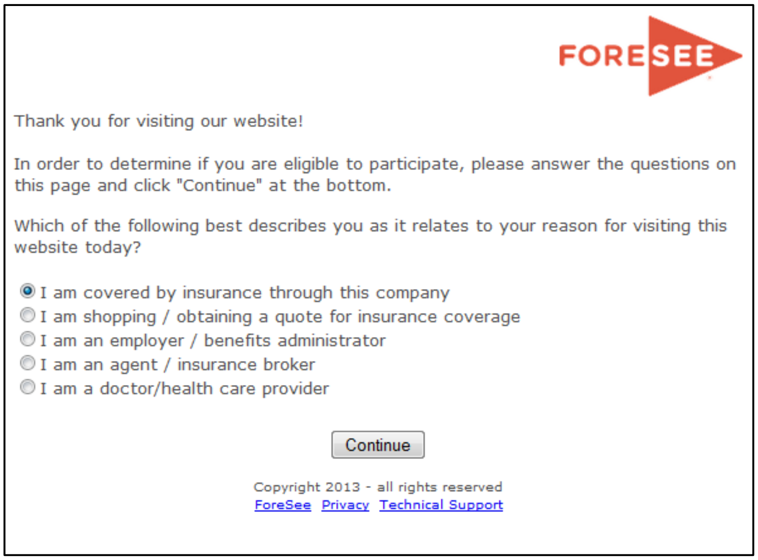
○Under 13?

○13-17? (Please have a parent or guardian click here to indicate you may participate in this survey.)

○18 or older?

Post-invitation qualifier

The post qualifier is presented after the invitation and in a separate browser window. This can be used to exclude users based on their selection or allow users to self-select options that can direct them to a specific survey. The selected answer may be passed as a CPP if the visitor submits a survey. The post invitation qualifier is typically a client hosted file (/foresee/qualifying.html) but could also be hosted by foresee. The text can typically be modified directly in the qualifying.html file.



Thank you for visiting our website!

In order to determine if you are eligible to participate, please answer the questions on this page and click "Continue" at the bottom.

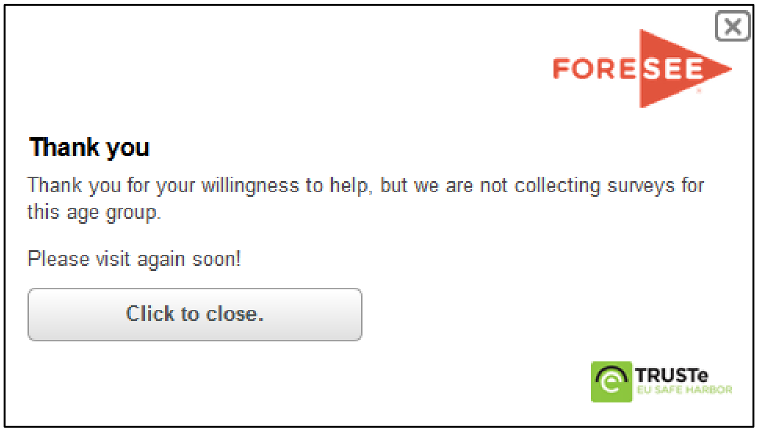
Which of the following best describes you as it relates to your reason for visiting this website today?

● I am covered by insurance through this company
○ I am shopping / obtaining a quote for insurance coverage
○ I am an employer / benefits administrator
○ I am an agent / insurance broker
○ I am a doctor/health care provider

Copyright 2013 - all rights reserved
ForeSee Privacy Technical Support

Cancel page

The cancel page (/foresee/cancel.html) is used when a user selects an answer to a qualifier question that would exclude them from being eligible for the survey. The text/html can be modified directly in the cancel.html file.



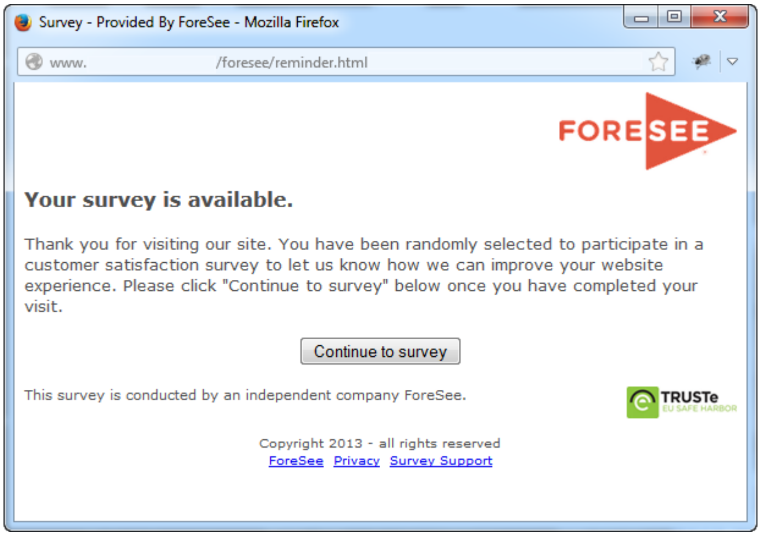
Thank you

Thank you for your willingness to help, but we are not collecting surveys for this age group.

Please visit again soon!

Reminder

The survey reminder (/foresee/reminder.html) is often used when areas of the website present an exit condition that would not typically be considered an exit. We often recommend using a reminder for sites that switch domains, sites that cannot contain code on all pages and sites that have slow page load times. The text/html can typically be modified directly in the reminder.html file.



Survey - Provided By ForeSee - Mozilla Firefox
www.   /foresee/reminder.html

Your survey is available.

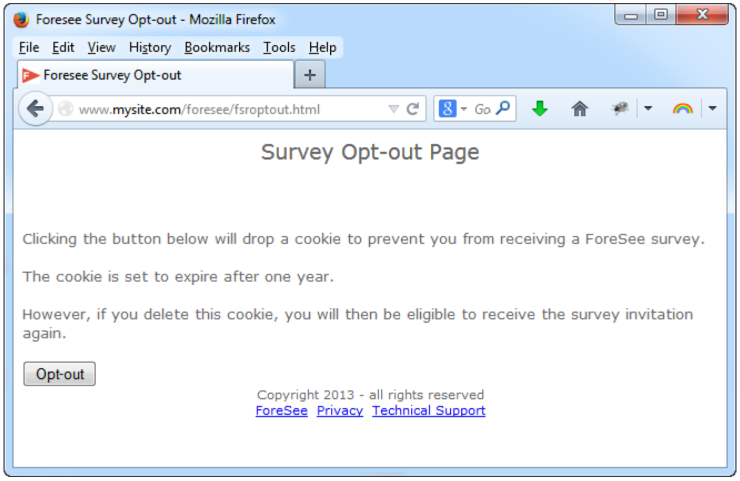
Thank you for visiting our site. You have been randomly selected to participate in a customer satisfaction survey to let us know how we can improve your website experience. Please click "Continue to survey" below once you have completed your visit.

This survey is conducted by an independent company ForeSee.

Copyright 2013 - all rights reserved
ForeSee Privacy Survey Support

Opt-out

The opt-out page (/foresee/fsroptout.html) is sometimes used internally to an organization if they would like to allow employees to opt-out of the survey. The link to the opt-out page would need to be supplied to the employees. The page itself will drop a cookie that expires after one year which will prevent the invitation from coming up on the specified site. If the user clears cookies, they will need to go back to this page and opt out again.



Foresee Survey Opt-out - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Foresee Survey Opt-out
www.mysite.com/foresee/fsroptout.html
Go

Survey Opt-out Page

Clicking the button below will drop a cookie to prevent you from receiving a ForeSee survey.

The cookie is set to expire after one year.

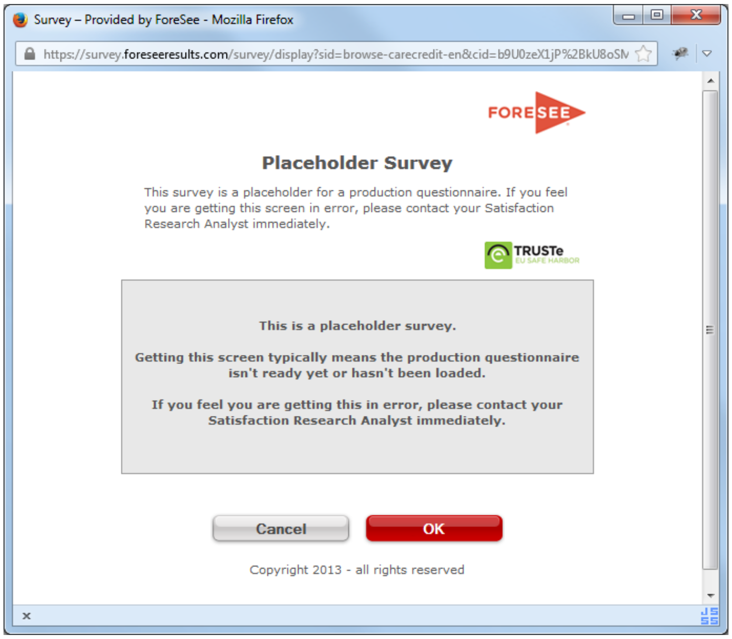
However, if you delete this cookie, you will then be eligible to receive the survey invitation again.

Opt-out

Copyright 2013 - all rights reserved
ForeSee Privacy Technical Support

Survey Placeholder

On occasion the survey code is delivered prior to the survey questionnaire being approved. In this case you may see a survey placeholder instead of the client approved survey. Once the survey questionnaire has been approved by the client a database mapping exercise at ForeSee will allow the production survey to appear. **Note:** If you believe this placeholder has come up in error or this is showing up after you are in production please contact your analyst to get the survey mapped. This database mapping does not require a code change in the survey trigger code.



Survey – Provided by ForeSee - Mozilla Firefox
https://survey.foreseeresults.com/survey/display?sid=browse-carecredit-en&cid=b9U0zeX1jP%2BkU8oSM
FORESEE
TRUSTe
EU SAFE HARBOR

Placeholder Survey

This survey is a placeholder for a production questionnaire. If you feel you are getting this screen in error, please contact your Satisfaction Research Analyst immediately.

This is a placeholder survey.

Getting this screen typically means the production questionnaire isn't ready yet or hasn't been loaded.

If you feel you are getting this in error, please contact your Satisfaction Research Analyst immediately.

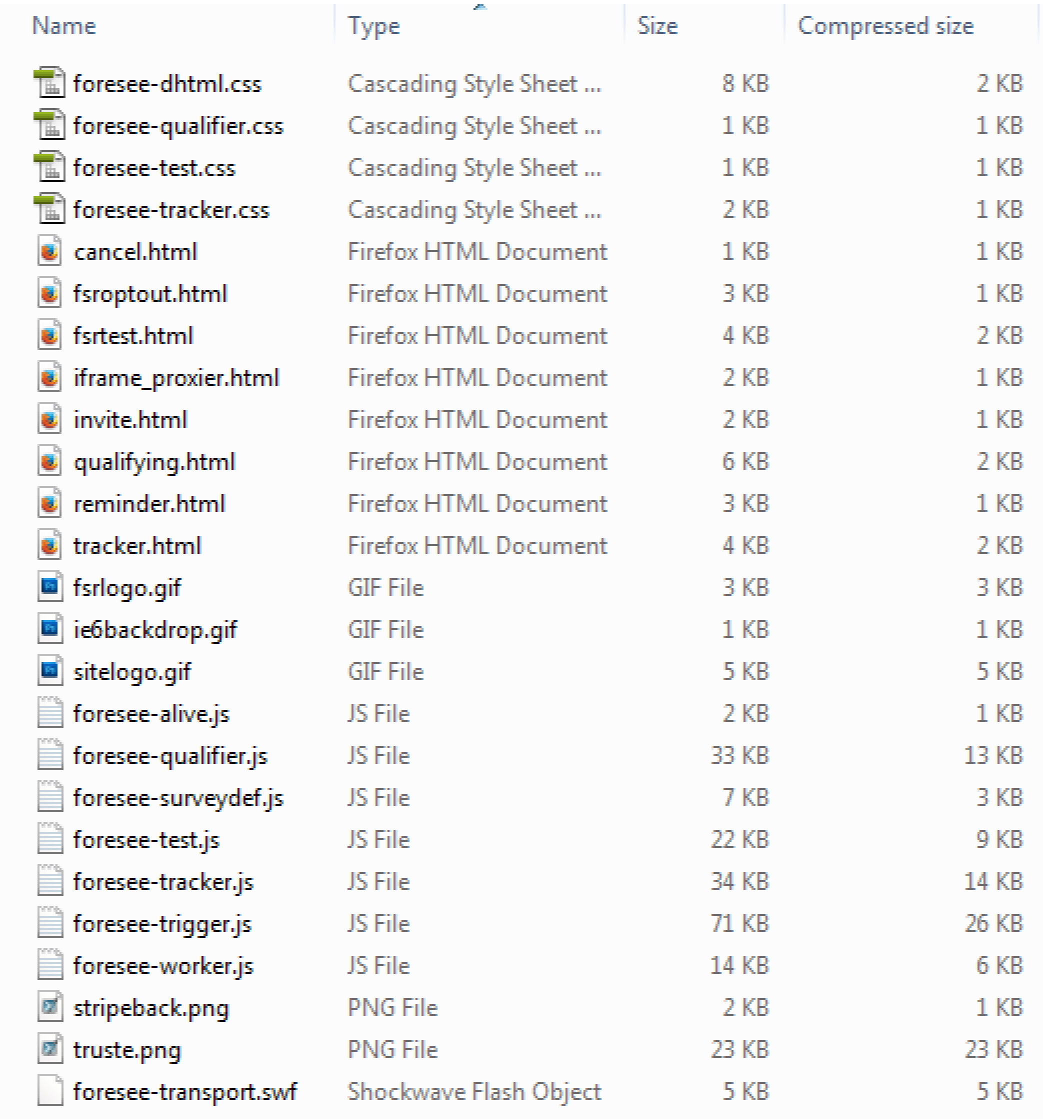
Cancel OK

Copyright 2013 - all rights reserved

The ForeSee Trigger Code File Library

The ForeSee JavaScript library will be hosted on your web servers. This library contains JavaScript, HTML, CSS and Image files as well as a SWF file for SessionReplay. The purpose of this JavaScript library is to present a survey invitation to users based on the configured site requirements. Once the site visitor has become eligible, based on those requirements, the code will then present the survey invitation and subsequent survey questionnaire (hosted by ForeSee) to the site visitor.

In most cases, the ForeSee Implementation team will have worked with your business and possibly a technical contact to pre-configure the JavaScript library so that there are minimal changes required to implement this code.



Name	Type	Size	Compressed size
foresee-dhtml.css	Cascading Style Sheet ...	8 KB	2 KB
foresee-qualifier.css	Cascading Style Sheet ...	1 KB	1 KB
foresee-test.css	Cascading Style Sheet ...	1 KB	1 KB
foresee-tracker.css	Cascading Style Sheet ...	2 KB	1 KB
cancel.html	Firefox HTML Document	1 KB	1 KB
fsroptout.html	Firefox HTML Document	3 KB	1 KB
fsrtest.html	Firefox HTML Document	4 KB	2 KB
iframe_proxier.html	Firefox HTML Document	2 KB	1 KB
invite.html	Firefox HTML Document	2 KB	1 KB
qualifying.html	Firefox HTML Document	6 KB	2 KB
reminder.html	Firefox HTML Document	3 KB	1 KB
tracker.html	Firefox HTML Document	4 KB	2 KB
fsrlogo.gif	GIF File	3 KB	3 KB
ie6backdrop.gif	GIF File	1 KB	1 KB
sitelogo.gif	GIF File	5 KB	5 KB
foresee-alive.js	JS File	2 KB	1 KB
foresee-qualifier.js	JS File	33 KB	13 KB
foresee-surveydef.js	JS File	7 KB	3 KB
foresee-test.js	JS File	22 KB	9 KB
foresee-tracker.js	JS File	34 KB	14 KB
foresee-trigger.js	JS File	71 KB	26 KB
foresee-worker.js	JS File	14 KB	6 KB
stripeback.png	PNG File	2 KB	1 KB
truste.png	PNG File	23 KB	23 KB
foresee-transport.swf	Shockwave Flash Object	5 KB	5 KB

Figure 4 – Library of Files (files may differ per implementation requirements)

Explanation of files

Our survey code package includes many files to support many features. Some files included may not be used in your specific implementation, however, we have compartmentalized the code to minimize loading of code on your website.

foresee-dhtml.css	Style sheet for survey invitation.
foresee-qualifier.css	Style sheet for qualifying.html and reminder.html.
foresee-test.css	Style sheet for fsrtest.html.
foresee-tracker.css	Style sheet for tracker.html.
cancel.html	Cancel page used when using qualifier page and disqualifying visitor.
fsroptout.html	Used to set "survey shown" persistent cookie that lasts for 1 year.
fsrtest.html	Used to set sampling rate for testers session only.
iframe_proxier.html	Iframe user in conjunction with the foresee-worker.js to transmit compressed CORS POST transmissions for SessionReplay.
invite.html	Used as template for designing invitation. Actual invite is in foresee-surveydef.js.
qualifying.html	Used when asking a qualifying question PRIOR to presenting survey.
reminder.html	Used often to present a survey reminder to users PRIOR to presenting survey.
tracker.html	Used when displaying survey onExit of site.
fsrlogo.gif	ForeSee Logo for invitation and tracker.html.
ie6backdrop.gif	Used for invite display in IE6.
sitelogo.gif	Client logo.
foresee-alive.js	JavaScript functionality used for onExit scenarios to prevent early presentation of survey.
foresee-test.js	JavaScript functionality for fsrtest.html test page.
foresee-qualifier.js	JavaScript functionality for qualifying.html page.
foresee-surveydef.js	JavaScript functionality for survey sampling parameters, this is the configuration file.
foresee-tracker.js	JavaScript functionality for tracker.html (used for onExit).
foresee-trigger.js	JavaScript functionality for survey presentation. Initializes survey process.
foresee-worker.js	JavaScript for a web-worker used to compress the SessionReplay event log when using CORS POST transmission in supported browsers.
stripeback.png	Used for invite display in IE6.
truste.png	Trustee logo.
foresee-transport.swf	Flash transport used to send SessionReplay information to ForeSee

Important Switches within the code

There are some important parameters within our code of which you should be aware. These parameters can occur in multiple files so if you make a change you'll want to be sure to include it in all the listed files.

Disabling the code

"Turning off" the trigger code

Change the enabled flag to false (This needs to be made to the foresee-trigger.js file only).

```
var $$FSR = {
   'timestamp': '__buildtimestamp__',
   'version': '__version__',
   'build': '__buildnumber__',
   'enabled': true,
```

"Turning off" SessionReplay

Change the sessionreplay flag to false (This needs to be made to the foresee-trigger.js file only). **Note: T**his setting is always set as enabled by default even if the SessionReplay product was not built into the code. This setting is not an indicator of whether SessionReplay has been included.

```
var $$FSR = {
   'timestamp': '__buildtimestamp__',
   'version': '__version__',
    'build': '__buildnumber__',
    'enabled': true,
    'frames': false,
    'sessionreplay': true,
```

Saving files in a different directory structure

If you do not wish to save the files using the “/foresee/” folder structure and prefer your own directory structure, be sure to save these files to a directory that is accessible to all web pages where satisfaction is to be measured.

If doing this the **‘files’** path in the code will also need to be modified in the following files:

- foresee-alive.js
- foresee-test.js
- foresee-tracker.js
- foresee-trigger.js
- foresee-qualifier.js (if applicable)

The code found in *Figure 5* is located at the top of each of the files listed above. The **‘files’** path will need to be changed to match your directory structure.

```
var $$FSR = {
   'timestamp': '__buildtimestamp__',
   'version': '__version__',
   'build': '__buildnumber__',
   'enabled': true,
   'frames' : false,
   'sessionreplay': true,
   'auto' : true,
   'encode' : true,
   'files': '/foresee/',
```

Figure 5 – Files Path

Options for the **‘files’** parameter include:

```
      'files': '/foresee/',  //relative to the domain of the site being measured
      'files': 'https://www.site.com/foresee/',  //absolute path
      'files': '//www.site.com/foresee/',  //absolute path adopting protocol
```

Separating Trigger Code files by type

If you have a need to separate the code files into different directories a **‘files’** path can be defined for the different types of files. *Figure 6* is an example of how this can be done. Two important things to note if this is being done:

1. There may be paths within the HTML files that need to be adjusted as well (i.e. tracker.html).

2. The HTML files, especially the tracker.html, need to be hosted on the domain being measured.

Reminder: The paths below would need to be modified in each of the following files:

- foresee-alive.js
- foresee-test.js
- foresee-tracker.js
- foresee-trigger.js
- foresee-qualifier.js (if applicable)

```
var $$FSR = {
   'timestamp': '__buildtimestamp__',
   'version': '__version__',
   'build': '__buildnumber__',
   'enabled': true,
   'frames' : false,
   'sessionreplay': true,
   'auto' : true,
   'encode' : true,
   //'files': '/foresee/',
   'js_files': '/foresee/js/',
   'image_files': '/foresee/images/',
   'html_files': '/foresee/html/',
   'css_files': '/foresee/css/',
```

Figure 6 – Breaking out the files

Separating Trigger Code files by Environment

If you need to have a different files path based on the environment you can do this by adding them under the 'sites' declaration. **Tip:** the code uses a regular expression string match so you do not need to call out each environment if the files paths are the same. Also, the sites declaration is evaluated from top down so be careful not to add something that will match unexpectedly first. **Example**: mysite.com (no www) would match before qa.mysite.com.

Reminder: The scripts below would need to be modified in each of the following files:

- foresee-alive.js
- foresee-test.js
- foresee-tracker.js
- foresee-trigger.js
- foresee-qualifier.js (if applicable)

```
var $$FSR = {
    {omitted code}
    'pools': [{
        path: '.',
        sp: 100 // CHANGE ONLY WHEN INCLUDING SESSION REPLAY
    }],
    'sites': [{
        path: 'www.mysite.com',
        domain: 'mysite.com',
        files: '/scripts/prod/foresee/'
    }, {
        path: 'qa.mysite.com',
        domain: 'mysite.com',
        files: '/scripts/qa/foresee/
    }],
    storageOption: 'cookie',
    nameBackup: window.name
};
```

Figure 7 – Different files path for each environment

Hosting the code from a third party domain

If you do not wish to save the files under the domain being measured there are some things to be aware of:

1. You can host all the files on a third-party domain except for the html files. HTML files must be hosted ON the domain being measured allowing them to properly access the ForeSee cookies.
2. You will need to split up the html files (see Separating Trigger Code files by type). The html path will need to be an absolute path in the 'files' path. (Change in foresee-alive.js, foresee-test.js, foresee-tracker.js, foresee-trigger.js, foresee-qualifier.js).
3. You will need to modify all paths in the tracker.html file by changing the location of the JavaScript and the image path references to be to absolute paths to the file location of server.
4. You will need to turn off our call to load the CSS (instructions below).**
5. You will need to load the content of foresee-dhtml.css within your global CSS or add a separate call to the foresee-dhtml.css on your pages.

   **To turn off our invite CSS call:

   In foresee-survyedef.js

   Find:

   ```
   CSS: 'foresee-dhtml.css',
   ```

   Replace with:

   ```
   CSS: false,
   ```

Changing the Survey Behavior

Loyalty Factor and Sampling Percentage

The behavior of the survey is controlled by the foresee-surveydef.js file located in the zip file you received. The two most common parameters that may need adjusting are the LF *(loyalty factor)* and the SP *(sampling percentage)*. The loyalty factor indicates the minimum number of pages a user must visit to receive and invitation. The sampling percentage indicates the percentage (random chance) they will be invited. If asked to change this number for a survey with multiple definitions you will want to be aware of the name of the definition and check the include URLs to ensure it aligns with the survey you are trying to change.

```
FSR.surveydefs = [{
    name: 'browse,
    invite: {
        when: 'onentry'
    },
    pop: {
        when: 'later'
    },
    criteria: {
        sp: 10, // sampling percentage
        lf: 3  // page view requirement (loyalty factor)
    },
    include: {
        urls: ['.']
    }
}];
```

Figure 8 - Sampling percentage and page view requirement

Invite Excludes

The ForeSee code should always be placed globally across all pages. However in some cases you may not wish an invitation to come up based on certain criteria. To suppress the invitation, page exclusions can be utilized. Page exclusions are formatted in a comma separated list based on specific URLs, referring domains, user agents, browsers, cookies and/or variables. Each of these methods can be set up in the foresee-surveydef.js file. **Note:** Excluded pages still count towards the Loyalty Factor requirement (minimum page views).

Note: Code should not be intentionally left off a page in an attempt to exclude it. Leaving code off a page may cause an early exit condition causing the survey questionnaire to present for users who had already accepted an invitation.

Tip: Rather than trying to exclude URLs individually, use regular expressions or use part of the URL that is common to a group of pages rather than the individual page.

```
invite: {

        exclude: {
            urls: ['/support','/checkout'],
            referrers: ['google.com'],
            userAgents: ['Firefox/22.0', 'MSIE'],
            browsers: [{"name":"IE", "version":7 }],
            cookies: [{name: '__utmc',value: '1'}],
            variables: [{name: 'testvar',value: ['1']}]
         },
```

Figure 9 - Excluding the invitation

Global Excludes/IP Excludes

Global Excludes are used in situations when the ForeSee code should not execute on the site. These are often utilized when clients are trying to exclude internal IP addresses or have a special user agent assigned to an internal bot checking the site. The Global excludes can be based on URLs, variables, cookies or user agents. It is possible to set up an exclude based on IP address, however, this requires the IP address of the user to be collected by the site and passed as JavaScript variable. The variable containing the IP can then be added in the variable exclude with a list of IP values that should be blocked. **Note:** We recommend using regular expressions for the IPs if excluding a large number of them. **Tip:** The parameter called ipexclude should be modified to the name of the variable used to exclude the IPs. This parameter is connected to the fsrtest.html page and allows your internal testers to set a cookie overwriting the exclude temporarily from the test page.

```
FSR.properties = {
   {omitted code}
    exclude: {
            urls: ['/support','/checkout'],
            referrers: ['google.com'],
            userAgents: ['Firefox/22.0', 'MSIE'],
            browsers: [{"name":"IE", "version":7 }],
            cookies: [{name: '__utmc',value: '1'}],
            variables: [{name: 'fsr$ip',
                          value: [/^141\.214\./,/^198\.(109|110|111)\./]
                        }]
},
ipexclude: 'fsr$ip',
```

Figure 10 - Excluding the invitation globally based on different exclude criteria

Using a tag manager

Implementing foresee via a tag manager can be as simple as adding the JavaScript reference to the site. We suggest utilizing the tag manager to dynamically add the following JavaScript reference `<script type="text/javascript" src="/foresee/foresee-trigger.js"></script>`. This will allow the ForeSee files to be enabled or disabled directly through the tag management system.

Note: In some occasions the tag manager may be hosting the ForeSee files off domain. In this case the html files must remain on the domain being measured. See Hosting the code from a third party domain.

We do have some additional options available when using tag managers that can be discussed with the implementation team.

SessionReplay Sampling – Pooling

With the introduction of SessionReplay we introduced the concept of **pooling** to limit the number of visitors we would be recording at any one time. It works by selecting people into (or out of) of the pool on the very first page. As a result we need to adjust the sampling percentage up for the survey to account for the fact that fewer people will now be eligible to take it. We start by adjusting it in unison with the pooling number so if the pooling sp is 20% (1/5), we multiply the current sampling rate by 5. Then we need to adjust it up for the average number of pages viewed being larger than 1 to account for the increased probability users who are now not in the pool had of getting invited previously."

Adjusting the pool (when testing only):

In the foresee-trigger.js file you will find:

```
'pools': [{
      path: '.',
      sp: 100
}],
```

You will want to adjust the pooling `'sp'` to 100 (when testing only) in order to receive the invitation as you normally would.

At this point you can use the fsradmin.html file that comes with the code (See Step 5) to set the sampling percentage for your specific survey in order to see the invitation.

Testing outside of production (using other domains, IP addresses and localhost)

If you are testing in a different environment or using a different host than production then you may need to modify the 'sites' parameter in the following files:

- foresee-alive.js
- foresee-test.js
- foresee-tracker.js
- foresee-trigger.js
- foresee-qualifier.js (if applicable)

The ‘sites’ definition in *Figure 11* is for a production environment and is meant to work in any qualified domain. The regular expression in the `'path'` area determines the *root* domain in which the ForeSee cookies are dropped. For example if your site is ‘www.xyz.com’ or ‘products.xyz.com’ then the cookies will be dropped in ‘xyz.com’ domain.

This is the recommended method for production web sites. It is important in most cases that the cookies be dropped to the *root* domain, especially if the site has sub-domains and is using the onExit method for presenting a survey.

```
var $$FSR = {
    {omitted code}
    'pools': [{
        path: '.',
        sp: 100 // CHANGE ONLY WHEN INCLUDING SESSION REPLAY
    }],
    'sites': [{
        path: /\w+-?\w+\.(com|org|edu|gov|net|co\.uk)/
    }, {
        path: '.',
        domain: 'default'
    }],
    storageOption: 'cookie',
    nameBackup: window.name
};
```

Figure 11 – Production Environment

The 'sites' definition in *Figure 12* is for a test environment and is meant to work in an environment WITHOUT a qualified domain (localhost or IP address). The dot (.) in the 'path' area is meant to be a wild card and drops the cookies into whatever domain it thinks it is. For example, if you're testing in your LOCALHOST environment then the cookies will be dropped in the domain 'LOCALHOST'.

```
var $$FSR = {
    {omitted code}
    'pools': [{
        path: '.',
        sp: 100 // CHANGE ONLY WHEN INCLUDING SESSION REPLAY
    }],
    'sites': [{
        path: /\w+-?\w+\.(com|org|edu|gov|net|co\.uk)/
    }, {
        path: '.',
        domain: 'default'
    }],
    storageOption: 'cookie',
    nameBackup: window.name
};
```

Figure 12 – Test Environment

Flash and DHTML

If your site is using Flash, it may be necessary to add *wmode="opaque" or "transparent"* in the **object** tag (for IE) and the **embed** tag (for Firefox) in order to prevent the DHTML invitation from displaying UNDERNEATH the Flash movie.

Customer Passed Parameters (CPPs)

Customer-passed parameters (CPPs) are used to send data points available on your website along with the survey response. The value of these parameters can be used for segmenting your respondent data. You may send as many parameters as you like using the methods below. We suggest you limit them to meaningful data that will be used for specific reporting.

Note: IE limits the number of characters in a URL string to 2084 including the URL to our survey. If additional CPP's cause the URL to exceed this limit they may not be passed with the survey and will be unavailable in the portal.

Method 1: The CPP API

The API in *Figure 13* can be used to manually set a CPP on your site. This line can be duplicated as many times as you want, but be sure it's placed below the call to foresee-trigger.js on the page. The foresee-trigger.js must be loaded first or it will fail due to the FSR object not being defined.

In the below example, the **name** should be replaced by the name of the CPP to be passed and viable in the portal. The **value** can be either a static hardcoded value, or a dynamic variable which can be customized to pass whichever data is preferred.

```
<script type="text/javascript">FSR.CPPS.set('name','value')</script>
```

Figure 13 – Set CPP

Method 2: Adding CPPs to the code

This method of adding CPPs keeps all the CPP logic directly in the codes' configuration file (foresee-surveydef.js file). Using this method is helpful because it keeps all the logic in one place and helps ensure the CPP logic isn't accidently removed during a site change.

The following snippets of code can be added to the FSR.properties section of the foresee-surveydef.js file.

Tip: It's a good idea to always set up an "init" value as "N", "None", "Null", "False", etc. so that the CPP is not left blank in the portal if a value isn't present. It is not possible to segment on blank values.

*<u>Passing the value of a **variable**</u>*

The code in *Figure 14* would set a CPP called **'category'** passing the value of a variable on the site called 'cat'.

```
FSR.properties = {
      cpps: {
        category: {
            source: 'variable',
            name: 'cat',
            init: 'none' //initial value of CPP if no variable exists
        }
    }
```

Figure 14 – Pass a Variable

*Passing the value of a **cookie***

The code in *Figure 15* would set a CPP called **'authenticated'** using a cookie with the name 'auth'.

```
FSR.properties = {
      cpps: {
        authenticated: {
            source: 'cookie',
            name: 'auth'
            init: 'false' //initial value of CPP if no cookie exists
        }
    }
```

Figure 15 – Pass a Cookie

*Creating a CPP using a **URL***

The code in *Figure 16* would set a CPP called **'support'** with a value of 'true' when the URL contained a string like '/support'**.**

```
FSR.properties = {
      cpps: {
        support: {
            source: 'url',
            init: 'none', //initial value of CPP if pattern isn't present
            patterns: [{
                regex: '/support',
                value: 'true'
            }]
        }
    }
```

Figure 16 – URL

*Creating a CPP using a **Meta Tag***

The code in *Figure 17* would set a CPP called **'CountryCode'** with the value of the Meta tag named **DCSext.CountryCode**.

```
FSR.properties = {
      cpps: {
        CountryCode: {
            source: 'meta',
            init: 'none', //initial value of CPP if no meta tag exists
            name: 'DCSext.CountryCode'
        }
    }
```

Figure 17 – Meta Tag

Creating a CPP using a URL query string parameter

The code in *Figure 18* would set a CPP called **'CountryCode'** with the value of the Meta tag named **DCSext.CountryCode**.

```
FSR.properties = {
      cpps: {
        CountryCode: {
            source: 'meta',
            init: 'none', //initial value of CPP if no meta tag exists
            name: 'DCSext.CountryCode'
        }
    }
```

Figure 18 – Meta Tag

How do I know cpps are being passed?

Method 1: fsr.s cookie

Open the fsr.s cookie and you will see your parameters inside the “cp” attribute. CPPs will be listed as name then current value to be passed.

```
"cp":{"ads":"none","icmpid":"none","cmpid":"none","icid":"none","cid":"none"},
```

Method 2: Survey URL

When you get a survey questionnaire to appear, you will find the CPP parameters in the URL string:

```
http://survey.foreseeresults.com/survey/display?sid=browse-yoursite.com-
en&cid=gcdwU50NZxxRxwJI1IEw%3D%3D&pattern=http%3A%2F%2Fwww.yoursite.com%2F&a=12930
32648063_751683&b=38790979455466765205&c=86400000&version=5.0.0&cpp[orderDate]=12%
2F22%2F2010&cpp[TLSessionID]=4EEBFE540DE2100D7A51B7D51C444613&cpp[browser]=Firefox
%203.6&cpp[os]=Windows&cpp[pv]=2&cpp[url]=http%3A%2F%2Fwww.yoursite.com%2F&cpp[ref
_url]=&cpp[locale]=en&cpp[site]=yoursite.com&cpp[section]=&cpp[referrer]=&cpp[term
s]=&cpp[sessionid]=1293032648063_751683&cpp[replay_id]=&cpp[flash]=10
```

Decreasing the cookie size

If you are particularly concerned about cookie size there are a few options for decreasing the size of the cookies.

Un-encoding of the fsr cookies

Change the encode flag to *false*.

The ForeSee trigger cookies as a default are encoded. If you would like to un-encode them to reduce the size of our cookie you may change the parameter below in the following 5 files:

- foresee-alive.js
- foresee-test.js
- foresee-trigger.js
- foresee-tracker.js
- foresee-qualifier.js (if applicable)

```
var $$FSR = {
   'timestamp': '__buildtimestamp__',
   'version': '__version__',
   'build': '__buildnumber__',
   'enabled': true,
   'frames' : false,
   'sessionreplay': true,
   'auto' : true,
   'encode' : true,
```

There are some items of interest that the ForeSee trigger code captures that may be interesting to our clients. The referring URL and search terms, the URL where the survey was presented and the visitor entry page are examples. Those items are, at some point during the visit, stored within our cookie and can increase the size significantly depending on the site. If you do not care to collect one or many of these parameters, you can disabled the collection of them in the foresee-surveydef.js file. Find the following snippet of code with the default values and change them for the items you do or do not wish to capture.

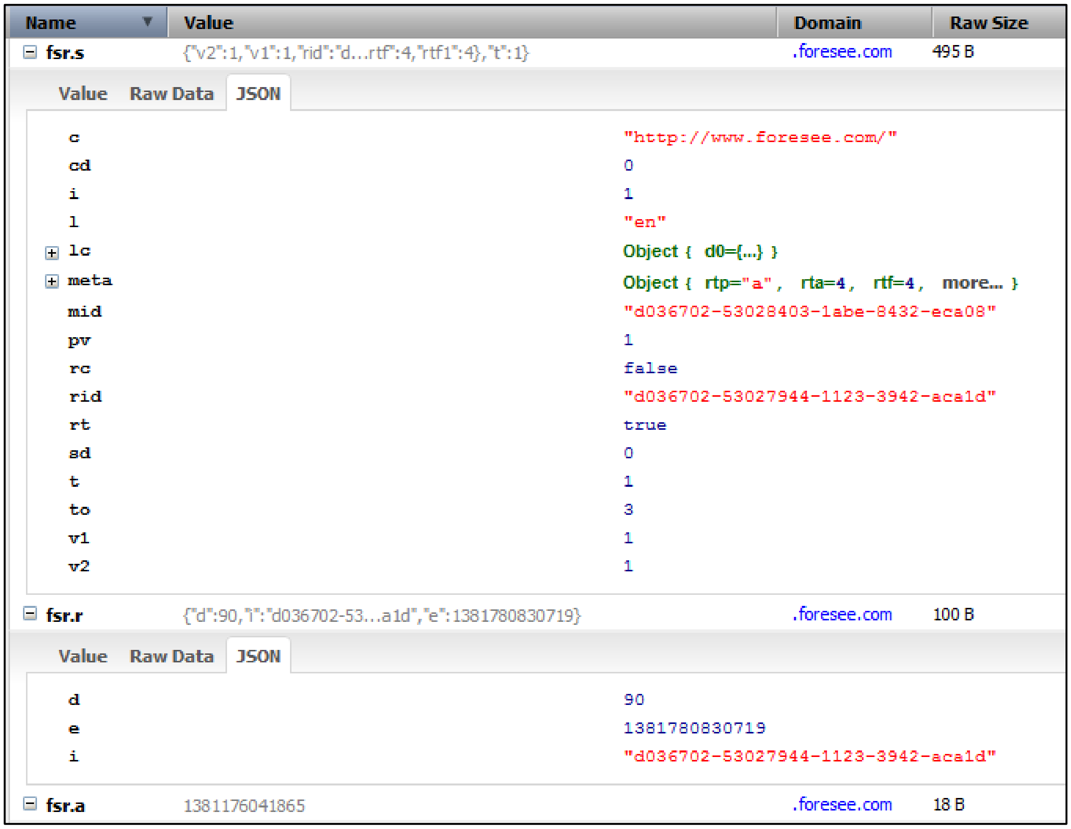
```
meta: {
       referrer: true,          //captures referring domain
       terms: true,               //captures referring domain search terms
       ref_url: true,             //captures referring url (full string)
       url: true,                 //captures URL of page where survey was presented
       url_params: false,         //captures URL params of page where survey was presented
       user_agent: false,         //captures full user agent
       entry: false,              //captures entry page of visitor
       entry_params: false        //captures entry page of visitor (full string)
    },
```

The ForeSee Cookies

ForeSee utilizes three cookies. Two session cookies (fsr.s and fsr.a) which stores information relating to the survey control and operation as well as some basic information such as the number of pages a user has viewed, and the URLs the user was on they entered and left the site.

In addition to items mentioned in the prior paragraph the ForeSee cookie can contain customer passed parameters (CPPS). This data can range from values stored within a variable, cookie, or URL parameter. To pass CPPs special code needs to be programmed into the survey configuration file in order to pass any additional data. This code is client controlled so you will know what data is being passed when the code is implemented. **Note: Foresee does not collect PII.**

ForeSee also has a persistent cookie (fsr.r) which contains information indicating when a user has received a survey invitation in order to prevent the same user from being presented with another invitation for a specified period of time.



Name	Value	Domain	Raw Size
⊟ fsr.s	{"v2":1,"v1":1,"rid":"d...rtf":4,"rtf1":4},"t":1}	.foresee.com	495 B
Value / Raw Data / JSON			
c	"http://www.foresee.com/"		
cd	0		
i	1		
l	"en"		
⊞ lc	Object { d0={...} }		
⊞ meta	Object { rtp="a", rta=4, rtf=4, more... }		
mid	"d036702-53028403-1abe-8432-eca08"		
pv	1		
rc	false		
rid	"d036702-53027944-1123-3942-aca1d"		
rt	true		
sd	0		
t	1		
to	3		
v1	1		
v2	1		
⊟ fsr.r	{"d":90,"i":"d036702-53...a1d","e":1381780830719}	.foresee.com	100 B
Value / Raw Data / JSON			
d	90		
e	1381780830719		
i	"d036702-53027944-1123-3942-aca1d"		
⊟ fsr.a	1381176041865	.foresee.com	18 B

Figure 19 - Example of the fsr.s, fsr.r and fsr.a cookies and what information they contain.

Changing your cookies to secure (https) cookies:

ForeSee is able to utilize secure cookies ONLY if the entire site being measured is secure (all https). The secure attribute for the ForeSee cookies will keep the cookie communication limited to SSL/encrypted transmission for HTTP-requests so this can be used only if the entire website being measured is https. If this setting in enabled when non-https pages are available the code will be unable to work on these pages. Please contact the implementation team for more details.

ForeSee cookies and the HTTPOnly attribure:

The purpose of the HttpOnly attribute in a cookie is to disallow client side code from reading cookie information. The ForeSee scripts are client side JavaScript code that require access to its cookie to function. Therefore ForeSee cookies do not use the HttpOnly attribute.

Performance Optimization

How the code runs

- The JavaScript files are loaded on the first page load and cached by the browser.
- Files that are not needed during the initial page load are loaded asynchronously.
- The execution of the code occurs on the DOM READY event after all the clients' content has been loaded.

Code delivery options

- Deliver code G-zip to browser.
- Set caching headers to expire after one day.
- Use alternate Implementation with the foresee-alive.js file at the top of the page and the foresee-trigger.js at the bottom.

Trigger Code Specifications (Web and Mobile)

Highlights:

- Based on latest JavaScript best practices
- Delivered fully name-spaced, minified and obfuscated
- Works across multiple tabs/browsers
- Works across protocols and sub-domains
- Code tested on client's production web site prior to delivery

Features:

- Initiate survey in browser session, on exit of site/section, or upon clickable event
- Uses three cookies: two session, one persistent (optional)
- Exclude invitation based on local URL, referring URL, userAgents, cookie, variable
- Present qualifying page prior to showing of the survey
- Fully multi-lingual enabled including invitation, tracker and survey
- Random sampling by measure and/or language
- Standard site parameters passed for all measures including, but not limited to, page views, url, browser, operating system, referring domain, search terms and device type (for mobile)

- Ability to define custom events based on URL, variable or cookie (e.g. purchase confirmation, items in cart, dollars purchased)

Browser Support:

- Internet Explorer 7+
- Firefox 1.5+
- Safari 3+
- Chrome 4+

Operating System Support:

- Win32
- Mac
- Linux
- iOS (mobile)
- Android (mobile)

Code Size (approx.):

- Uncompressed: 69 KB (with SessionReplay: 113KB)
- Compressed (G-zipped): 25 KB (with SessionReplay: 40KB)

Deployment:

- Single line of JavaScript code referenced globally
- JavaScript library hosted on clients' web servers
- Data collection hosted by ForeSee
- Code delivered as a single zip file with instructions

SessionReplay Code Specifications

Highlights:

- PII blocking

Features:

- Capture:
    - Full DOM and changes to DOM
    - Mouse Movements
    - Scrolling
    - Keyboard Strokes
    - Images

  - Most AJAX Interactions
  - Flash/Silverlight Object Initialization
- Exclude:
  - Entire Pages
  - Specific Fields (i.e. Credit Card Number, Social Security Number)
- Compression of idle time
- Off Screen Indicator
- Histogram of Activity
- Clock
- Integrated with Satisfaction Data

Limitations (limited or no recording of):

- Intranet Sites
- Flash/Silverlight Interactions
- IFrames that violate same origin policy of host page

Transmission:

Small, asynchronous posts are made back to ForeSee servers. They contain highly compressed information about the website DOM, user's mouse movements and page interactions. Once transmission begins it can happen as frequently as every second. The average packet size is only 16 bytes.

Transmission is done using CORS as the primary method (www.w3.org/TR/cors/) and AMF as a backup for browsers that don't support the CORS mechanism.

SessionReplay Privacy Protection Options

Overview

ForeSee's SessionReplay product was designed to give the client control over blocking any sensitive data, including but not limited to Personally Identifiable Information (PII). Because ForeSee is not in the business of collecting and managing PII, ForeSee prefers that this information be blocked from capture on ForeSee's servers. We assume our clients will agree that PII should be kept apart from customer satisfaction analytics. However, on a case-by-case basis at the direction of a client, ForeSee can accommodate requests for alternative management of PII data.

The following are various approaches that we provide our clients to control the way personal information is handled by SessionReplay – including the preferred option of not capturing the personal information at all. In all cases, the client is responsible for rigorously identifying any sensitive information that maybe present on their web site to assure that it's handled properly in accordance with the client's processes and procedures. We require all clients to declare which of the following options will be implemented with their SessionReplay code package via ForeSee's Secure Information Form (SIF). Clients must sign and date the SIF prior to code development.

Data Field Entry

As a default, SessionReplay will not capture any data field entry. These fields will appear as blocked with the following symbol: "*". The actual characters typed in the data field will not appear in SessionReplay.

Example: if “Joe” is typed into the “First Name” field, it will appear in SessionReplay as “***”.
ForeSee has the ability to change this default and allow certain fields deemed NOT Personally Identifiable Information (PII) to be processed. For example, some clients choose to enable the capture of input into the following data fields:

- Search Box
- City and Zip Code (but still blocking First Name, Last Name, Address and Phone)

In-line Personalization

In-line personalization is data of a personal nature that is displayed on a webpage as static text or within a drop down.

Example 1: After a visitor logs into a site, he may receive the following message: “Hello Joe, your balance is $123.45.”

Example 2: In the process of placing a purchase order (order confirmation page): “Shipping Address: Joe Smith, 123 Main Street, Ann Arbor MI 48103”.

Example 3: Visitors’ previous credit card information selectable from a drop down. Visitor may have the following in the drop down:

“Amex, ….1234, exp 10/2011”
“Mastercard, …9876, exp 11/2012”

Data Blocking

ForeSee’s SessionReplay has the ability to mask information PRIOR to it being sent to ForeSee by inserting comment that encompass the PII fields. These tags are created and inserted by the SessionReplay product and require no alterations to the page HTML.

<!--fsrHiddenBlockStart--> Name:<fname>, Zip Code:<zip code>**<!-- fsrHiddenBlockEnd -->**

Resulting in:

HTML PAGE Name : Tom Zip Code: 48105	**Data Captured on ForeSee Server** ***** *** *** ***** *****	**Data Displayed on Replay** ***** *** *** ***** *****

Note: ForeSee’s Implementation Team will review the SIF with the Subscriber prior to making any changes in the default setting, which blocks the recording of all data collection fields, and approving code to block all in-line personalization.

Contact ForeSee

If you have any questions regarding these instructions or need help troubleshooting an issue, please contact the Implementation Services Team:

Implementation Services Team

Support Email: **ImplementationServices@ForeSeeResults.com**