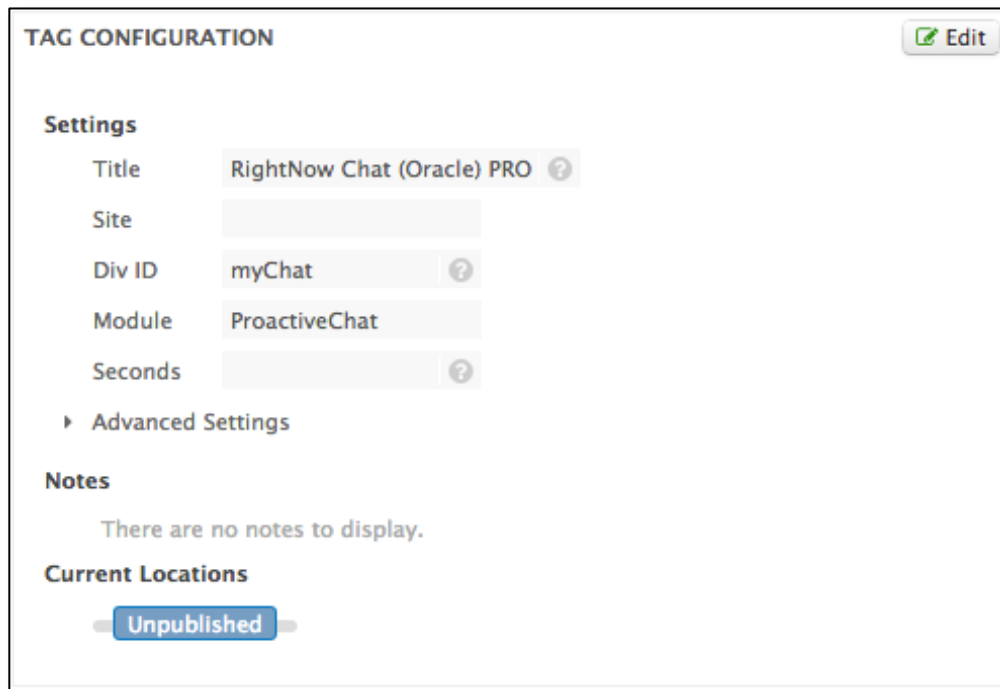


## Overview

This document covers an implementation for a client of the Oracle RightNow Chat solution across multiple pages in their web site. The client and Tealium wanted to use only one tag instance. There were many customizations that had to be applied both to the configuration of the chat itself and the logic applied to tracking chat actions.

## Tag Configuration

- Site: Value of `https://SITE.widget.custhelp.com/...`
  - Can be empty and then use a mapping set by an extension e.g. JavaScript extension "Set Chat Environment" discussed below
- Div ID: states the HTML element ID in the page where the proactive chat should appear
- Module: The only Module currently supported is "ProactiveChat"
- Seconds: Number of seconds before proactive chat will appear
  - Can be empty and then use a mapping set by an extension e.g. Lookup Table extension "determine chat trigger time"



The screenshot shows a "TAG CONFIGURATION" window with an "Edit" button in the top right corner. The "Settings" section includes the following fields:

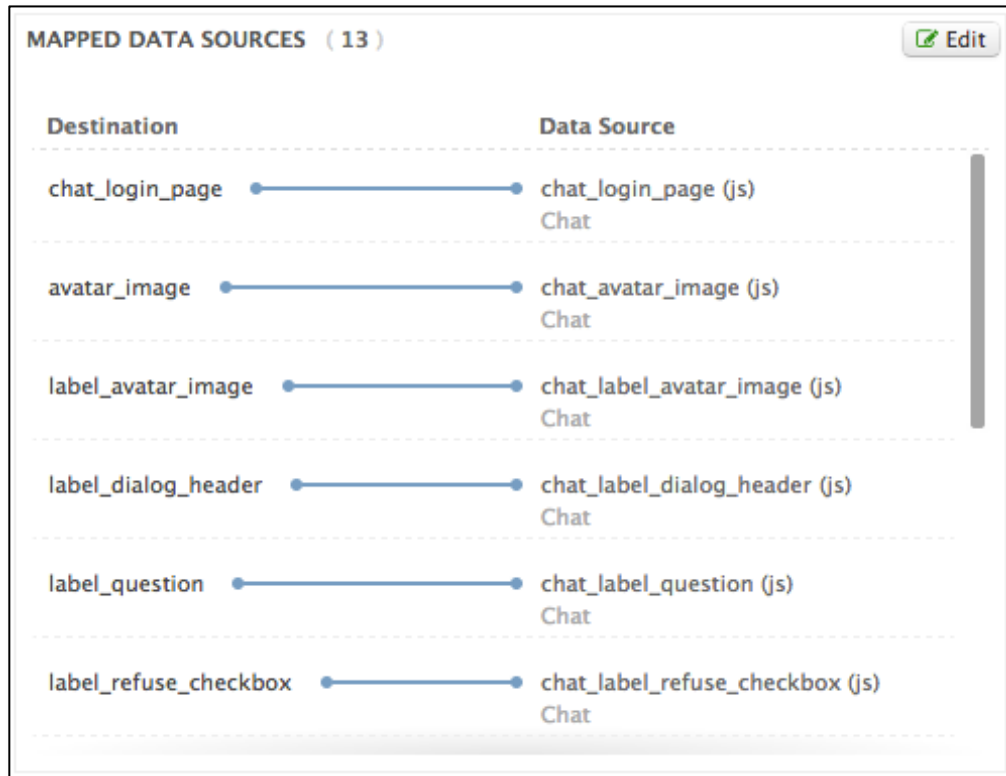
- Title: RightNow Chat (Oracle) PRO
- Site: (empty)
- Div ID: myChat
- Module: ProactiveChat
- Seconds: (empty)

Below the settings is a section for "Advanced Settings" which is currently collapsed. Underneath is a "Notes" section with the text "There are no notes to display." At the bottom, the "Current Locations" section shows a toggle switch set to "Unpublished".

## Mapped Data Sources

- chat\_login\_page: Statically set by Set Data Values extension "Chat Specs" to `"/app/chat/chat_launch/chat_source_page/"+b.key_page_name`. The `b.key_page_name` will be discussed later in detail. This value is part of a URL call to the chat solution that will tell the agent from which page the chat launched.
- chat\_avatar\_image: Statically set by Set Data Values extension "Chat Specs" to an empty string ""
- chat\_label\_avatar\_image: Statically set by Set Data Values extension "Chat Specs" to an empty string ""
- chat\_label\_dialog\_header: Statically set by Set Data Values extension "Chat Specs" to an empty string ""
- chat\_label\_question: Statically set by Set Data Values extension "Chat Specs" for the opening question dialog e.g. "Is there anything we can help you with today?"

- chat\_label\_refuse\_checkbox: Statically set by Set Data Values extension "Chat Specs" for the text that appears on the "Do not ask me again" modal when the customer attempts to close the chat modal
- chat\_label\_title: Statically set by Set Data Values extension "Chat Specs" to the label of the chat modal e.g. "LIVE CHAT SUPPORT"
- chat\_logo\_image: Statically set by Set Data Values extension "Chat Specs" to an empty string ""
- chat\_modal: Statically set by Set Data Values extension "Chat Specs" to true (boolean)
- chat\_login\_page\_height: Statically set by Set Data Values extension "Chat Specs" to set the height of the chat window e.g. "520"
- seconds: Statically set by lookup table extension called "determine chat trigger time"



MAPPED DATA SOURCES ( 13 ) Edit

Destination	Data Source
chat_login_page	chat_login_page (js) Chat
avatar_image	chat_avatar_image (js) Chat
label_avatar_image	chat_label_avatar_image (js) Chat
label_dialog_header	chat_label_dialog_header (js) Chat
label_question	chat_label_question (js) Chat
label_refuse_checkbox	chat_label_refuse_checkbox (js) Chat

## Extensions

- Key Page Look Up: the current page URL is compared and a key page name (b.key\_page\_name) value is output. It is very important to have matches on all URLs so that the key page name is output. It is required as part of the input data into the chat system so that it can also perform a look up and tell the agent from which page the chat was initiated. The importance of this cannot be expressed enough. This output is used for the "chat\_login\_page" mapping.

**Title:** Key Page Look up

**Lookup Value In:** url

**Destination:** key\_page\_name (js) Default

**Variable Type:** String

**Match Type:** Contains

The **Note** is an optional field for additional information. The content of the **Note** is not passed to the destination.

<b>Lookup Match:</b> page_index.jsp	<b>Output:</b> Start
<b>Lookup Match:</b> page_product.jsp	<b>Output:</b> Product
<b>Lookup Match:</b> page_verification.jsp	<b>Output:</b> Verification

- Build Default Page Name from URL: creates a base url of just the domain and pathname to be used if no key page name is output

**Title:** Build Default Page Name from URL

**Destination:** page\_base\_url (js)

**Delimiter:**  **Leading Delimiter:**  Yes  No **Default Value:**

**Join:** domain

**and:** pathname

**Sample:** dom.domaindom.pathname

**Condition:**

- Set Default Page Name: if no lookup match occurred in the "Key Page Look Up" lookup table then the b.key\_page\_name is set to variable "page\_base\_url". This "page\_base\_url" variable is created in the Join Data Values extension "Build Default Page Name from URL" above. This may be used for the "chat\_login\_page" mapping.

**Title:** Set Default Page Name

**Set:** key\_page\_name (js) **To:** Variable

**Condition:** key\_page\_name (js) is not populated

- determine chat trigger time: the current page URL is compared and the output is the number of seconds that chat should wait before it proactively appears. This value is mapped to "seconds". This allows for different pages to have a different proactive pop time.

determine chat trigger time

Lookup Value In:

Destination:  + Default

Variable Type:

Match Type:

The Note is an optional field for additional information. The content of the Note is not passed to the destination.

Lookup Match: <input type="text" value="/process"/>	Output: <input type="text" value="90"/>
Lookup Match: <input type="text" value="/summary"/>	Output: <input type="text" value="30"/>
Lookup Match: <input type="text" value="/registration"/>	Output: <input type="text" value="120"/>

- Chat Specs: Sets many of the specifications of the chat that appears to the customer. All of these values are used in the Mapped Data Sources section. The chat\_login\_page is set to: "/app/chat/chat\_launch/chat\_source\_page/"+b.key\_page\_name

Chat Specs

Set: <input type="text" value="chat_avatar_image (js)"/>	To: <input type="text" value="JS Code"/>	<input type="text" value=""/>
Set: <input type="text" value="chat_label_avatar_image (js)"/>	To: <input type="text" value="JS Code"/>	<input type="text" value=""/>
Set: <input type="text" value="chat_label_dialog_header (js)"/>	To: <input type="text" value="JS Code"/>	<input type="text" value=""/>
Set: <input type="text" value="chat_label_question (js)"/>	To: <input type="text" value="Text"/>	<input <="" td="" type="text" value="Is there anything we can help you with today?"/>
Set: <input type="text" value="chat_label_refuse_checkbox (js)"/>	To: <input type="text" value="Text"/>	<input type="text" value="Do not ask me again"/>
Set: <input type="text" value="chat_label_title (js)"/>	To: <input type="text" value="Text"/>	<input type="text" value="LIVE CHAT SUPPORT"/>
Set: <input type="text" value="chat_logo_image (js)"/>	To: <input type="text" value="JS Code"/>	<input type="text" value=""/>
Set: <input type="text" value="chat_modal (js)"/>	To: <input type="text" value="JS Code"/>	<input type="text" value="true"/>
Set: <input type="text" value="chat_login_page_height (js)"/>	To: <input type="text" value="JS Code"/>	<input type="text" value="520"/>
Set: <input type="text" value="chat_login_page (js)"/>	To: <input type="text" value="JS Code"/>	<input +b.<="" td="" type="text" value="/app/chat/chat_launch/chat_source_page/"/>

- Set Chat Environment: This dynamically sets the chat site based on the domain. If the domain contains "qa" or "staging" then the site is "SITE--tst", otherwise it is just "SITE". u.site is a reference to code within the Tealium template. This is used as the "Site" Tag Configuration value so be sure not to change the variable name.

Title: Set Chat Environment

```

1  if(b["dom.domain"].indexOf("qa")>=0 || b["dom.domain"].indexOf("staging")>=0){
2    u.site="SITE--tst";
3  }else{
4    u.site="SITE";
5  }

```

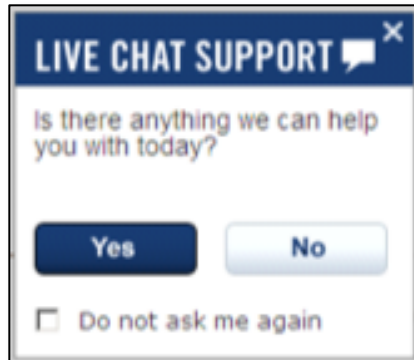
### Sample URL call to the chat solution

Here is a sample call that occurs with the above configurations

*[https://SITE.widget.custhelp.com/ci/ws/get/w/2/co/\[\[{"chat\\_login\\_page":"/app/chat/chat\\_launch/chat\\_source\\_page/Start", "chat\\_login\\_page\\_height":520, "div\\_id":"myChat", "instance\\_id":"spac\\_0", "label\\_question":"Is there anything we can help you with today?", "label\\_refuse\\_checkbox":"Do not ask me again", "label\\_title":"LIVE CHAT SUPPORT", "modal":true, "module":"ProactiveChat", "seconds":"60", "type":2}\]\]/https\\_request/1](https://SITE.widget.custhelp.com/ci/ws/get/w/2/co/[[{)*

Notice how the values set in extension "Chat Specs" are passed through in this URL. Notice the "chat\_login\_page" has a value ending in "Start" that will be output from the "Key Page Look Up" lookup table. Also notice the "seconds" has a value of "60" that was an output from the "determine chat trigger table" lookup table.

The final invite window should look something like this:



## Custom Tracking

The out of the box functionality for the chat solution did not meet the needs of the client and needed to be updated from default. Tealium worked with an Oracle consultant to update the functionality. It should be noted that what was implemented was/is not fully supported by Oracle in that any customizations should be handled by the client. Oracle did provide the foundation of this code to Tealium though and the final solution worked very well.

Changes required:

- chatOffered: default functionality is to pop chat once and never again. Client wanted this changed so that it required user action to reject/refuse before chat not be popped again. Now we just erase the chat cookie so that chat can pop again on a later page (set the "noChat" cookie with an expiration date 10 days previous).
- chatAccepted: default functionality is such that if a user moves to another page that a second chat can appear to the user. This is not the desired functionality so it was changed to set a cookie that tells the chat solution not to pop a second chat on subsequent pages while still allowing the first chat to persist across pages (set the "noChat" cookie to "RNTLIVE" which signifies chat in session).
- chatRejected (when user clicks the "X" icon or "No" button): default functionality is to not pop chat again. Client wanted this changed, as their belief was that if a user clicked "No" that the business still wanted the opportunity to pop again because the customer wasn't actually refusing the chat. Now we just erase the chat cookie so that chat can pop again on a later page (set the "noChat" cookie with an expiration date 10 days previous).
- chatRefused (when user checks the "Do not ask me again" checkbox): default functionality is to not pop a chat again for 30 days and this was retained. The belief is that if a customer refuses to take a chat then they should not be bothered again. This is only documented so the difference between Rejected and Refused is understood. If this needs to be changed the code below should be clear enough to be updated.

The reasoning for using the `setInterval` isn't remembered. This may need to be changed to a `setTimeout` so that the large chat library has time to load before this code is run. This should be the last chat extension to run.

Note: when copied into your code editor, be sure that the single and double quotes are not converted to Microsoft quotes which are handled differently than true quotes.

```
if(typeof _tealium_chat=="undefined"){
  _tealium_chat=function(){
    if(typeof spac_0!="undefined"){
      if(typeof(spac_0.prototype)==="undefined"){
        spac_0.prototype={
          chatOffered: spac_0.chatOffered,
          chatAccepted: spac_0.chatAccepted,
          chatRejected: spac_0.chatRejected
        }
        spac_0.chatOffered=function(){
          spac_0.prototype.chatOffered.call(this);
          RightNow.Client.Util.setCookie("noChat","", -10, '/', "", "");
        }
        spac_0.chatAccepted=function(){
          RightNow.Client.Util.setCookie("noChat", 'RNTLIVE', "", '/', "", "");
          spac_0.prototype.chatAccepted.call(this);
        }
        spac_0.chatRejected=function(){
          RightNow.Client.Util.setCookie("noChat","", -10, '/', "", "");
          spac_0.prototype.chatRejected.call(this);
        }
      }
    }
  }
  setInterval("_tealium_chat()",1000);
}
```