Article
Tani Frankel · Jun 23, 2016 7m read

Web Service Business Operation Client – Response Timeout Behavior

When calling out to web services there are several settings of the Business Operation that play together in controlling what will happen when a response is not returned in the desired time.(Note this is relevant also for example for a non-SOAP simple HTTP call)

The 3 main settings involved are:

- *Response Timeout*

Specifies the timeout for getting a response from the remote web server.

- *Retry Interval*

Number of seconds to wait between attempts to connect with a destination outside Ensemble.

- *Failure Timeout*

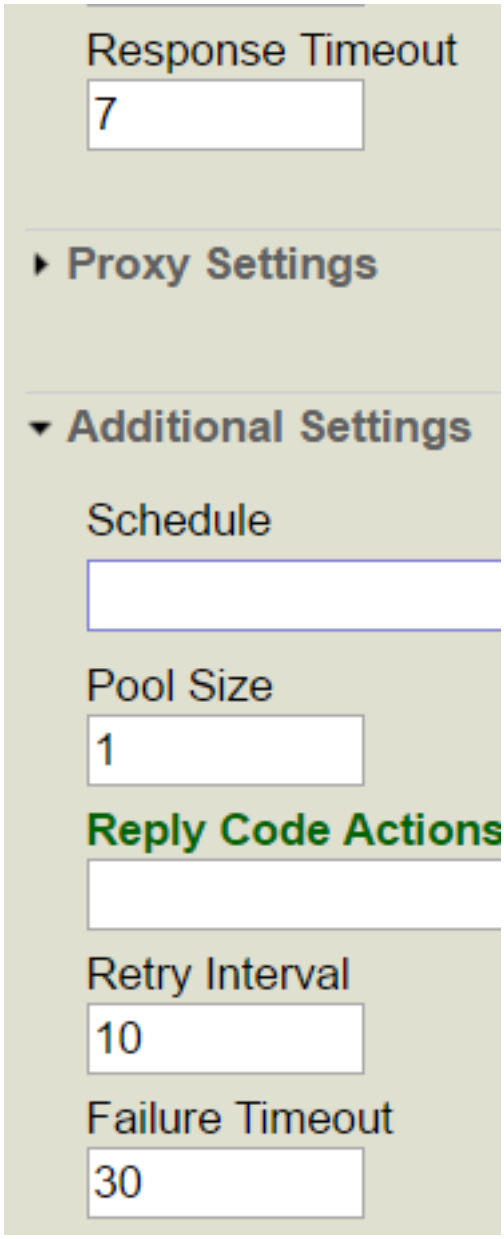
Total number of seconds to keep trying to connect with a destination outside Ensemble. After this number of seconds has elapsed, the business operation discards the message data and returns an error code.

Put together in words these play together as follows –

We will wait for a response from the Web Server for 'Response Timeout' seconds. If no response has been received by that time, we will call the Web Server again after 'Retry Interval' seconds have elapsed. We will continue attempting this 'Failure Timeout' seconds have gone by since the first attempt started.

To illustrate let's take the following example –

Assume the following settings:



In words –

Response Timeout - Waiting for 7 seconds for response

Retry Interval - Retrying every 10 sec.

Failure Timeout – "Give up" retrying after 30 seconds

So assuming the Response comes back after exactly 8 sec. then the following scenario will take place –

1. At 00:00 we will make the first call
2. At 00:07 since no response is returned we will acknowledge internally that a "Response Timeout" error has occurred [and log an "Error event" in the Event Log], and will attempt again according to the retry policy and settings – the "Failure Timeout" has not arrived yet, so a "Need to Retry flag" is raised.
3. [At 00:08 the Web Server will return a response but this response will not be received by us since we've already errored with a timeout]

4. At 00:10 the Retry Interval has arrived and since the "Need to retry flag" is raised we will call the Web Server again.
5. At 00:17 we will again have arrived at our "Response Timeout" with no response (note as mentioned before the response sent back at 00:08/step #3 is "ignored/discarded"), so again we will internally denote this as an error (though this time will not add another Event log Error entry, only the first attempt will create this, not every retry failure), and since we have not reached the "Failure Timeout" yet, have "Need to Retry flag" up yet again.
6. [At 00:18 the Web Server will return a response which again will not be received by us]
7. At 00:20 another retry interval elapses and we will call again for our 3rd attempt.
8. At 00:27, no response, again "Response Timeout" error and need to Retry (not reached "Failure Timeout" yet).
9. [At 00:28 a non-received response is sent by server]
10. At 00:30 another Retry Interval comes along and we make our 4th (and last) attempt.
11. At 00:37 a "Response Timeout" occurs again – this time the "Failure Timeout" has passed so we do not raise the "Need to Retry flag" but rather give up – we log an Error event in the Event Log noting the Failure timeout has elapsed, and also return an error from the Business Operation to the calling item.

The following is some "evidence" from a sample call per the scenario above.

First the server side [From the SOAP log] – you can see it received 4 calls/requests, 10 seconds apart, each time returning a response after 8 seconds from the request –

```
05/31/2016 14:18:45 **********
Input to Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV='http://schemas.xmlsoap.org/soap/envelope/'
...
</SOAP-ENV:Envelope>



05/31/2016 14:18:53 **********
Output from Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>



05/31/2016 14:18:55 **********
Input to Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>



05/31/2016 14:19:03 **********
Output from Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>



05/31/2016 14:19:05 **********
Input to Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
```

```
...
</SOAP-ENV:Envelope>


05/31/2016 14:19:13 **********
Output from Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>


05/31/2016 14:19:15 **********
Input to Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>


05/31/2016 14:19:23 **********
Output from Web service with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>
```

And now from the Ensemble BO/client side, you can see 4 attempts, 10 seconds apart, each time logging a response timeout error 7 seconds after.

Client side

```
05/31/2016 14:18:45 **********
Output from Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>

05/31/2016 14:18:52 **********
Input to Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
ERROR #5922: Timed out waiting for response
string**** SOAP client return error. method=GetResponse, action=http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
     ERROR #5922: Timed out waiting for response

05/31/2016 14:18:55 **********
Output from Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>

05/31/2016 14:19:02 **********
Input to Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
ERROR #5922: Timed out waiting for response
string**** SOAP client return error. method=GetResponse, action=http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
     ERROR #5922: Timed out waiting for response

05/31/2016 14:19:05 **********
```

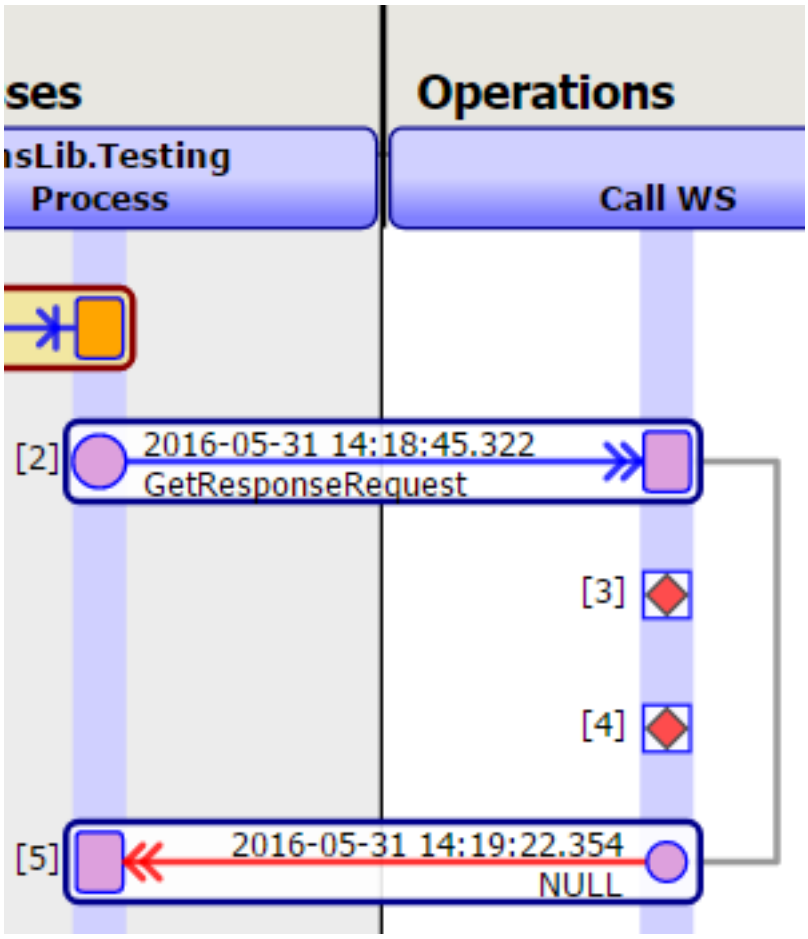
```
Output from Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>

05/31/2016 14:19:12 **********
Input to Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
ERROR #5922: Timed out waiting for response
string**** SOAP client return error. method=GetResponse, action=http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
     ERROR #5922: Timed out waiting for response

05/31/2016 14:19:15 **********
Output from Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
<?xml version="1.0" encoding="UTF-8" ?>
...
</SOAP-ENV:Envelope>

05/31/2016 14:19:22 **********
Input to Web client with SOAP action = http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
ERROR #5922: Timed out waiting for response
string**** SOAP client return error. method=GetResponse, action=http://tempuri.org/Test.WSTimeouts.WebService.GetResponse
     ERROR #5922: Timed out waiting for response
```

Here is the Ensemble Visual Trace:



And here the Event log entries –

Error	594	2016-05-31 14:19:22.354	131	8892	Call WS	ERROR <Ens>ErrFailureTimeout: FailureTimeout of 30 seconds exceeded in Test.WSTimeouts.BO.MyServiceSoap;
Error	593	2016-05-31 14:18:52.343	131	8892	Call WS	ERROR #5922: Timed out waiting for response

Sample Event Log with Trace Events on [You might need to zoom-in to read the text in the image better] –

Type	ID	Time Logged	Session	Job	Source	Text
Trace (queue)	712	2016-06-05 17:09:53.224		15200	Call WS	begin wait: -1
Trace (queue)	711	2016-06-05 17:09:53.224		15200	Call WS	dequeueing from Call WS with timeout=-1
Trace	710	2016-06-05 17:09:53.208		15200	Call WS	event '^Ens.Event("EnsLib.Testing.Process")' already exists
Trace	709	2016-06-05 17:09:53.208		15200	Call WS	enqueueing #205 with body Class '' and body Id '' to EnsLib.Testing.Process at priority 6
Error	708	2016-06-05 17:09:53.208	203	15200	Call WS	ERROR <Ens>ErrFailureTimeout: FailureTimeout of 30 seconds exceeded in Test.WSTimeouts.BO.MyServiceSoap; status from last attempt was ERROR #5922: Timed out waiting for response
Trace	707	2016-06-05 17:09:53.208	203	15200	Call WS	calling OnError E
Trace	706	2016-06-05 17:09:53.208	203	15200	Call WS	calling OnError R
Trace	705	2016-06-05 17:09:53.208	203	15200	Call WS	GetResponse() returned with status ERROR #5922: Timed out waiting for response, Retry=1, SuspendMessage=0, DeferResponse=0, Response=
Trace	704	2016-06-05 17:09:46.208	203	15200	Call WS	Retrying Request Header Id 204 - try #4
Trace (queue)	703	2016-06-05 17:09:46.208	203	15200	Call WS	dequeuing timed out
Trace (queue)	702	2016-06-05 17:09:43.208	203	15200	Call WS	begin wait: 3
Trace (queue)	701	2016-06-05 17:09:43.208	203	15200	Call WS	dequeueing from SystemSignal:15200 with timeout=2.971029
Trace	700	2016-06-05 17:09:43.208	203	15200	Call WS	calling OnError R
Trace	699	2016-06-05 17:09:43.208	203	15200	Call WS	GetResponse() returned with status ERROR #5922: Timed out waiting for response, Retry=1, SuspendMessage=0, DeferResponse=0, Response=
Trace	698	2016-06-05 17:09:36.182	203	15200	Call WS	Retrying Request Header Id 204 - try #3
Trace (queue)	697	2016-06-05 17:09:36.182	203	15200	Call WS	dequeuing timed out
Trace (queue)	696	2016-06-05 17:09:33.172	203	15200	Call WS	begin wait: 3
Trace (queue)	695	2016-06-05 17:09:33.172	203	15200	Call WS	dequeueing from SystemSignal:15200 with timeout=2.989901
Trace	694	2016-06-05 17:09:33.172	203	15200	Call WS	calling OnError R
Trace	693	2016-06-05 17:09:33.172	203	15200	Call WS	GetResponse() returned with status ERROR #5922: Timed out waiting for response, Retry=1, SuspendMessage=0, DeferResponse=0, Response=
Trace	692	2016-06-05 17:09:26.168	203	15200	Call WS	Retrying Request Header Id 204 - try #2
Trace (queue)	691	2016-06-05 17:09:26.168	203	15200	Call WS	dequeuing timed out
Trace (queue)	690	2016-06-05 17:09:23.152	203	15200	Call WS	begin wait: 3
Trace	689	2016-06-05 17:09:23.152	203	15200	Call WS	event '^Ens.Event("SystemSignal:15200")' created
Trace (queue)	688	2016-06-05 17:09:23.152	203	15200	Call WS	dequeueing from SystemSignal:15200 with timeout=2.971514
Error	687	2016-06-05 17:09:23.152	203	15200	Call WS	ERROR #5922: Timed out waiting for response
Trace	686	2016-06-05 17:09:23.152	203	15200	Call WS	calling OnError R
Trace	685	2016-06-05 17:09:23.152	203	15200	Call WS	GetResponse() returned with status ERROR #5922: Timed out waiting for response, Retry=1, SuspendMessage=0, DeferResponse=0, Response=
Trace	684	2016-06-05 17:09:16.136		15200	Call WS	dequeuing from Call WS found message #204 with body Class 'Test.WSTimeouts.Msg.GetResponseRequest' and body Id '94' and SessionId '203'
Trace (queue)	683	2016-06-05 17:09:16.136		15200	Call WS	dequeuing received an event
Trace (queue)	671	2016-06-05 17:09:08.197		15200	Call WS	begin wait: -1

Here you can observe some of the "inner workings" of the above described scenario –

At Log ID #684 the initial call is made – at 17:09:16.

Then 7 seconds later (09:23) we get the response timeout error (#685). The operation then logs the error (#687) and decides to wait for another 3 seconds until the retry interval; 10sec retry interval minus 7sec response timeout (#688-#690).

After the 3 second wait elapses (at 09:26; #691) the 2nd attempt is made (#692), with the same result and the same subsequent behavior, until the 4th try (#704). After the 4th try fails (09:53; #705) another attempt is not made since the failure timeout (30sec) has exceeded.

#Best Practices #Business Operation #SOAP #Ensemble

Source URL:https://community.intersystems.com/post/web-service-business-operation-client-%E2%80%93-response-timeout-behavior