



Error Analyser

Powered by Dynatrace Real User Monitoring

What is it?

Real User Monitoring from Dynatrace captures an enormous amount of information about users that access your application – both for their behaviour and their performance. Whilst analytics platforms can point towards drop-offs in user journeys (revenue generating or otherwise), Dynatrace brings the context of **if this happened due to errors or performance**.

The Error Analyser is an application which looks to quantify:

- How many users **received an error**?
- How many users then **abandoned their journey**?
- How many of those users were “lost” and **did not return** to complete their journey?

Error analysed - Invalid Form Key. Please refresh the page.


Based on the last 7 days...

 **286** Impacted users

Total number of users that received the error.

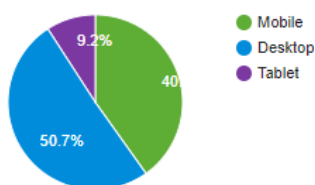
 **166** Unconverted users

Of the impacted users, the number who did not convert in that session.

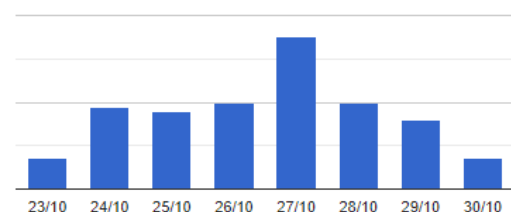
 **142** Lost users

Of the users who did not convert, how many did not return later to convert.

Lost user breakdown by channel



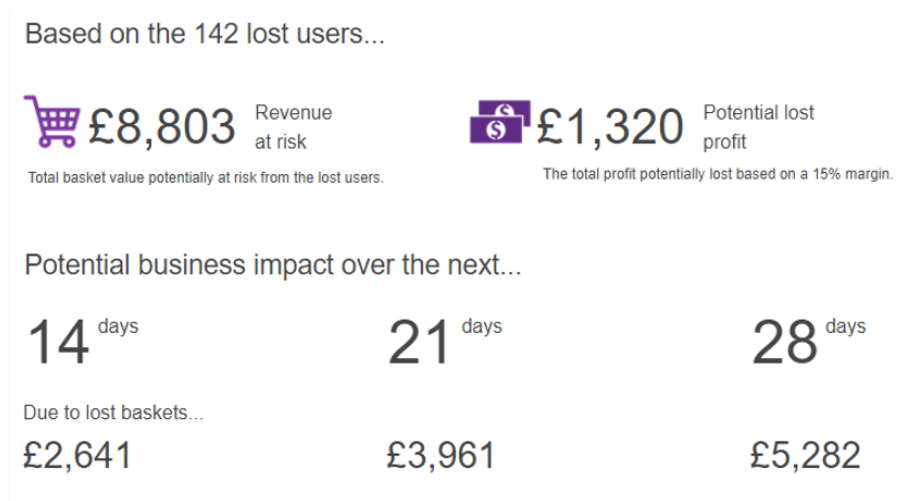
Occurrence of error over time



Based on these “lost” users, the Error Analyser can then assess the business impact in the context of 1 or more use cases:

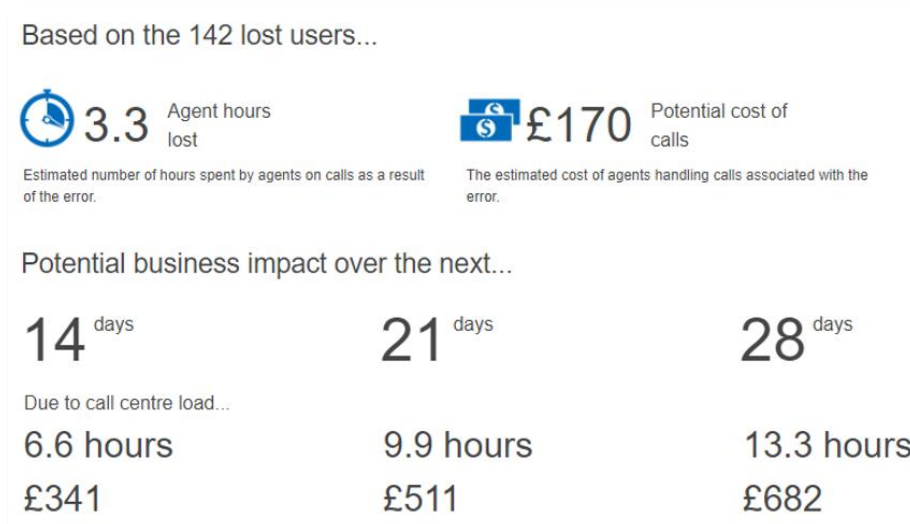
- **Lost orders:** Quantify the revenue at risk from users that have not converted (and did not return) after encountering an error.
- **Agent hours:** Estimate the hours (and cost, where relevant) to be spent by support staff, dealing with enquiries due to an error.
- **Incurred costs:** Total up the potential costs incurred from an error hitting a user, for example a penalty due to an SLA.

Lost orders



For a revenue generating application, Dynatrace can capture the basket/order value attached to a specific user. For the “lost” users who quit their purchase due to an error and then don’t return to complete it, the Error Analyser shows the total value attached to these users over the last 7 days (and if relevant, can take your net profit margin into account).

Agent hours



Where application issues lead to calls into the call centre, the Error Analyser is able to estimate the loss in productivity and also cost where relevant. This is achieved by providing figures such as the estimated percentage of users that call in due to errors and average call length. Error Analyser will then base this upon the real life “lost users” seen in the Real User Monitoring data.

Incurred costs

Based on the 142 lost users...



£17,750

Potential costs
incurred

The estimated revenue impact from users hitting the error in your application.

Potential business impact over the next...

14 days

21 days

28 days

Due to incurred costs...

£35,500

£53,250




£71,000

Certain errors will have a fixed cost associated with them, for example if an error indicates an SLA breach that may have a fixed financial penalty. Providing that fixed value to Error Analyser will add up the total financial impact of all errors so far in addition to the future projections by carrying on at the current rate.

Pre-requisites

For all use cases, Dynatrace requires your Dynatrace environment URL, an API token (with User Session Query Language access), the name of your session property containing the error message and the name of the action which counts as a user "converting" in your application. As an optional step (in case of many applications or duplicate properties), you can specify an application to filter the results on. Each use case then has specific information required to produce output on the business impact.

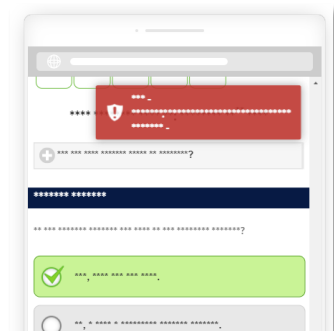
*Dynatrace environment.....?	*API token	*Error property
abc12345.live.dynatrace.cor	User session access needed	exceptionmessage
*Conversion action	Application	
loading of page /checkout/confirmation	www.angular.easytravel.com	

		
Lost orders	Agent hours	Incurred costs
Quantify the revenue at risk from users that have not converted (and did not return) after encountering an error.	Estimate the hours (and cost, where relevant) to be spent by support staff, dealing with enquiries due to an error.	Total up the potential costs incurred from an error hitting a user, for example a penalty due to an SLA.
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
*Basket property	*% of users who call in when experiencing an error	*Cost of an error
order_value	Must be an integer	Value in £
Margin (%)	*Average length of a call in minutes	
% percent	Must be an integer	
Monthly multiplication factor	Average cost of a call	
12	Value in £	



Define your errors

The error analyser centres on errors presented to users in the web UI, the more specific the better – and try to avoid low impact errors like incorrect password. Typically, these will be surfaced in a specific error "box" or "toast" such as the example on the right. These will need to be extracted as [session properties](#), CSS selectors work well for grabbing it off the page but JavaScript variables have also been useful. Please store these as a **string**.





Define your conversions

To understand if a user counts as “converted,” the error analysers asks for the name of your conversion action. This is the name of the action **as it appears in a user’s session**. [Naming rules](#) may be useful to remove variables such as order ID.

16:17:17	Loading of page /orange-trip-details.jsf	314 ms
16:17:20	Loading of page /orange-booking-review.jsf	429 ms
16:17:23	Loading of page /orange-booking-payment.jsf	369 ms
16:17:27	Loading of page /orange-booking-finish.jsf	2.34 s



Capture returning users accurately

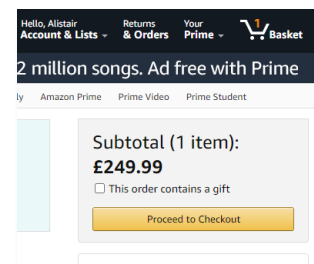
To make sure that you can tell when a user with an error returns later, please enable [user tracking](#) in the application settings.



For lost orders...

Extract your order value

To understand revenue loss, you must capture a user’s order value. This will again be captured as a [session property](#). CSS locators will work well unless the actual cart value is hidden on some pages, typically it will already be stored in a global JS variable, for example in “dataLayer.” Please store this as a **double**. Dynatrace always takes the **last** value, so it will update as they add/remove items.



Provide a profit margin (optional)

In some cases, a profit margin needs to be taken into account when analysing the business impact as this is the true cost lost to the business. In these situations, please provide an integer between 1 and 100 so that Error Analyser can tailor the output to this value.

Specify a monthly multiplication factor (optional)

Where an application allows users to purchase something with a recurring monthly cost (like a phone plan or broadband), it’s likely that the revenue property captured is that single monthly cost. In these cases, please provide a number of months to multiply that value by, for example the minimum contract term.



For agent hours...

Percentage of users that are likely to call in

Based upon existing data, what percentage of callers are likely to call in if they experience an error? This can be done by analysing a specific historical incident where the rough impact is known (in terms of number of users) and matching up in your CRM how many calls were received by users who experienced the same issue.

Average length of a call in minutes

Using your existing CRM, provide how long the average support call lasts – if possible please provide this value where it is filtered to users experiencing IT errors specifically rather than other issues related due to user knowledge.

Average cost of a call (optional)

If tracked, please provide the average cost of a call into the call centre when a user needs support. Typically, we see this in the range of between £8 and £15.



For incurred costs...

Cost of an error

This is a static value which is incurred whenever a specific error happens. For example, some legal proceedings must now be recorded and if an error happens, the proceeding must be repeated. This means every time the "recording failed to save" error occurs, legal fees are incurred to repeat it.