



Full Stack Observability with New Relic

Liam Hurrell: Technical Trainer - New Relic University
<http://nru.to/liam-linkedin>

December 2020

Login

NRU TRAINING ACCOUNT

one.newrelic.com

User: learn@newrelicuniversity.com

Password: pointless-meeting

NB. You may need to log out of your own New Relic account first
or use an incognito window

Slides are at: [**http://nru.to/g**](http://nru.to/g)

Housekeeping



Record Session

Today's training session may be being recorded.

If so a link to the recording will be emailed to all registrants shortly after.



Questions/Comments

Please be sure to ask questions via the 'questions panel'



Survey

There will be a brief survey after the training course for any additional feedback

We'd love to hear what you think?

Please take one minute to share your feedback with us

Survey Day 1

<https://www.surveymonkey.com/r/nrufsoworkshopday1>

Survey Day 2

<https://www.surveymonkey.com/r/nrufsoworkshopday2>

Agenda

Day 1 AM	Dashboarding and NRQL Alerting	2 hours
	Break	
Day 1 PM	Full Stack Observability Part 1 (Backend & Logs)	2 hours
Day 2 AM	Full Stack Observability Part 2 (Infra & Alerts)	2 hours
	Break	
Day 2 PM	Full Stack Observability Part 3 (Browser and Synthetics)	2 hours

Agenda: Dashboarding and NRQL Alerting

- 1 Introduction to NRQL and Dashboards
- 2 Data Explorer
- 3 Query Builder : Starting Queries
- 4 More Advanced Queries
- 5 Chart Challenge
- 6 Using NRQL for Alerting

Agenda

10:00 AM Introduction to NRQL and Dashboards

10:10 AM Data Explorer Lab

10:20 AM Chart Builder : Starting Queries Lab

10:40 AM More Advanced Queries Lab

11:00 AM Break

11:15 AM Chart Challenge Lab

11:40 AM Using NRQL for Alerting Lab

Event data in NRDB

- Event data is collected (in NRDB) from across your stack
- Can include custom events
- Events are discrete (individual) and not regular

Dimensional data: can query after the event

- Each event collects metadata (attributes) about the event that occurred

Example

Event type: Transaction

Metadata: Transaction Type - Response Code - Request URI - Duration - App ID

Main Event Types

Browser

APM

Synthetics

Mobile

Infrastructure

PageView

Transaction

SyntheticCheck

MobileRequest

SystemSample

NRQL Syntax

Build your query around your chosen Event Type

SELECT function(attribute) FROM Event_Type WHERE condition FACET group by

Browser

PageView

APM

Transaction

Synthetics

SyntheticCheck

Mobile

MobileRequest

Infrastructure

SystemSample

Lab: Data Explorer

Experiment with Data Explorer

Choose the 'Demotronv2' account option at the top as this has the best data available

- 1** Build 4 x queries by selecting the following options
 - Event type: **PageView**, Plot **Duration (Median)**, Dimension **PageUrl**
 - Event type: **Transaction**, Plot **Count (*)** Dimension **Name**
 - Event type: **SystemSample**, Plot **CPUpercent Average**, Dimension **hostName**
 - Event type: **JavascriptError** Plot: **Count *** Dimension: **errorMessage**
- 2** Try the 'RAW Data' option/button and explore the attributes for these Event Types
 - **Transaction, PageView, SystemSample**
- 3** Mouse over the NRQL query you have built, and click to '**Edit in Query Builder**'
 - Try changing the timeperiod eg. SINCE 1 day ago
 - Try changing the FACET

Lab: NRQL Query Builder - Your first query

- 1** Create a Dashboard in the NRU Training Account
 - Name: '*your initials*-dashboard'

- 2** Add a Chart and type these queries in the Query Builder and view the results:
 - SELECT * FROM Transaction
 - PageView
 - SystemSample

Aggregate your data

Average() Max() Min() Count()

```
SELECT count(*) FROM Transaction
```

Throughput for my application

```
SELECT max(duration) FROM Transaction
```

Slowest transactions for my application

Group your data

FACET

```
SELECT count(*) FROM Transaction FACET appName
```

Transactions grouped by different applications

```
SELECT count(*) FROM Transaction FACET httpResponseCode
```

Transactions grouped by response codes

Query part of your data

WHERE

```
SELECT average(duration) FROM Transaction WHERE transactionType='Web'  
FACET name
```

Average duration of web transactions for my application, group by transaction name

```
SELECT count(*) FROM Transaction WHERE httpResponseCode !='200' FACET  
httpResponseCode
```

Count all transactions for my application, group by httpResponseCode exclude 200's

Lab: Starting Queries

Start with these Transaction query examples:

1

FROM _____ SELECT _____

SELECT Count all Count () FROM the Transaction events*

2

FROM _____ SELECT _____ FACET _____

Count all Transactions grouped by name

3

FROM _____ SELECT _____ WHERE _____ FACET _____

Count all Transactions from a specific (app appName = '...') grouped by name

Lab: Starting Queries

Start with these PageView query examples:

1

FROM _____ SELECT _____

Count all the PageView events on your site

2

FROM _____ SELECT _____ FACET _____

Count all Pageviews grouped by country

3

FROM _____ SELECT _____ WHERE _____ FACET _____

See all Pageviews from a specific country grouped by cities

Lab: Query Builder

Basic Mode

Choose account: Demotronv2 then use the Query Builder to answer these questions and add the charts to your team's dashboard.

We want to answer the following **PageView performance** questions:

- 1 What's the worst (max) page load duration in my app called 'WebPortal' (appname) in the past 24 hours
- 2 How does my average page load duration over past 7 days compare with the 7 days before
(The time specified by COMPARE WITH is relative to the time specified by SINCE or UNTIL)
- 3 Which cities (facet) are experiencing the worst page loads in my app called 'WebPortal' over the past 24 hours

Dashboards: 12 column layout and other features

Find the 'Buried dashboard NRU Copy' and explore it:

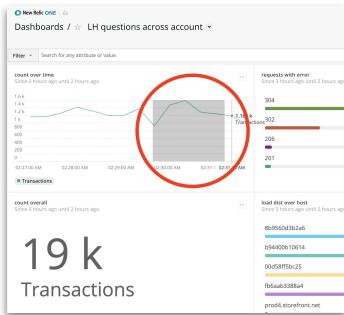
1

Is the needle correlated on the Timeseries charts? _____ Drag zoom across one of the charts

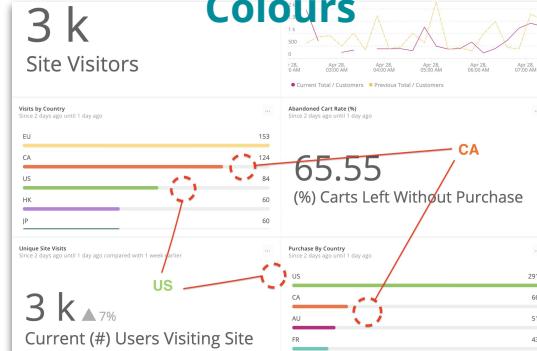
2

Reset the charts to their individual time periods using TimePicker > Default

Brush zoom



Consistent Facet Colours



Correlated Needle



What's different about these queries?

Pageview Performance

1. What are my worst page load durations broken out by?

Browser, City, & Country code

Comparing attribute values

2. What is the average Back end response time (backendDuration) and what is the average Front end load time (duration) - so we can compare them?

NRQL: Multi Count & Multi FACET

Is Back end or Front end Browser Response time taking more time to complete?

```
SELECT average(backendDuration), average(duration)
FROM PageView
WHERE appName = 'WebPortal'
TIMESERIES 1 minute
```

Query Builder: NRQL Mode

Multi-Count and Multi-Facet

```
SELECT function(attribute), function(attribute) FROM eventType FACET PageUrl, UserAgentName
```

Labelling

```
SELECT function(attribute) AS 'my label' FROM ...
```

Since, Until, Compare With, TimeSeries

```
SELECT average(duration), max(duration) FROM PageView SINCE 1 week AGO TIMESERIES AUTO
```

Grouped by time range: FACET by time span

```
FACET weekDayof(timestamp)
```

Aggregators: Percentiles

```
SELECT percentile(duration, 5, 50, 95) FROM PageView TIMESERIES AUTO
```

FACET CASES

```
SELECT ...FACET CASES (WHERE attribute operator value, WHERE attribute operator value, ... )
```

More examples in our docs site

<https://docs.newrelic.com/docs/query-data/nrql-new-relic-query-language/getting-started/nrql-syntax-components-functions#functions>

Lab: More advanced NRQL Queries

Try some of these queries - choose DemotronV2 account

Use these event types:

1 PageView

- What is the worst (**max**) page load **duration** grouped (**FACET**) by PageUrl, countryCode, userAgentName

2 Transaction

- **Count httpResponseCodes** WHERE the httpResponseCode = '500' and group the results by **appName** and **errorMessage**

3 Transaction

- Create a **Table** of average duration of response time labelled AS 'response time', and average databaseDuration AS 'DB time' and FACET by name and request.method

4 PageView

- Using a (count*) create a **Pie chart** showing the split of different browsers (userAgents) used or deviceType

5 SystemSample

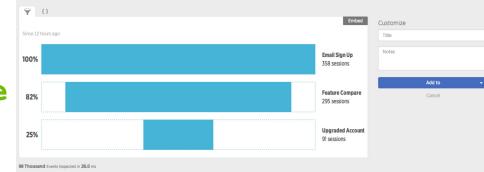
- Show me average cpuPercent usage across my hosts

Funnel

```
SELECT funnel(attribute, Step A, Step B)
```

Example:

```
SELECT funnel(session, WHERE pageUrl LIKE '%/' AS 'Homepage'
```



```
WHERE pageUrl LIKE '%/browse/plans' AS 'Click Product',
```

Step A

Label

```
WHERE pageUrl LIKE '%login.jsp' AS 'Login',
```

Step B

Label

```
WHERE pageUrl LIKE '%/checkout' AS 'Confirm')
```

FACET CASES

Break out your data by more complex conditions

`SELECT ...`

`FACET CASES (WHERE attribute operator value, WHERE attribute operator value, ...)`

Eg.

Group your PageView data into categories like < 1 second, from 1 to 10 secs, and > 10 seconds

```
SELECT count(*) FROM PageView  
FACET CASES (WHERE duration < 1, WHERE duration > 1 and duration < 10, WHERE duration > 10)
```

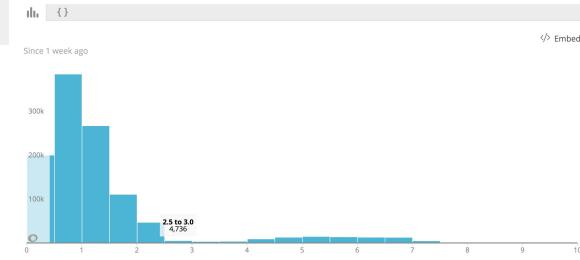
Page Load Time by Category query

```
SELECT average(duration) FROM PageView FACET CASES( WHERE pageUrl like '%/index.html' AS  
'Home Page', WHERE pageUrl like '%/browse/%' AS 'Product Page', WHERE pageUrl like  
'%/shoppingcart' AS 'Cart Page', WHERE pageUrl LIKE '%/checkout' AS 'Purchased')  
TIMESERIES since 1 day ago
```

Histogram

```
SELECT histogram(attribute, ceiling [, number of buckets])
```

```
FROM Event_Type SINCE 1 hour ago
```



Example:

"Show me a Histogram of **response times** ranging up to **10 seconds** over **20 buckets**"

```
SELECT histogram(duration, 10, 20) FROM PageView SINCE 1 week ago
```

Lab: Histogram

- 1** Fill in the blanks _____ and write the following query in a New Relic account

"Show me a Histogram of back-end response times ranging up to 2 seconds with 10 buckets":

```
SELECT histogram(_____, ___, ___, _____) FROM _____ SINCE 1 week ago
```

- ## 2 Refine the settings of your histogram:

- Adjust the ceiling to a smaller value and increase the number of buckets to get smaller steps
 - When you're happy with the distribution save your new Histogram chart

NRQL Alerts

Query results

- Queries must return a number.
- The alert condition works by evaluating that returned number against the thresholds you set.

Threshold Types

Static

- simplest type
- Create a condition based on the numeric value returned
- **FACET** optional

Baseline

- self-adjusting condition based on the past behavior of the monitored values
- Cannot use **FACET**

Outlier

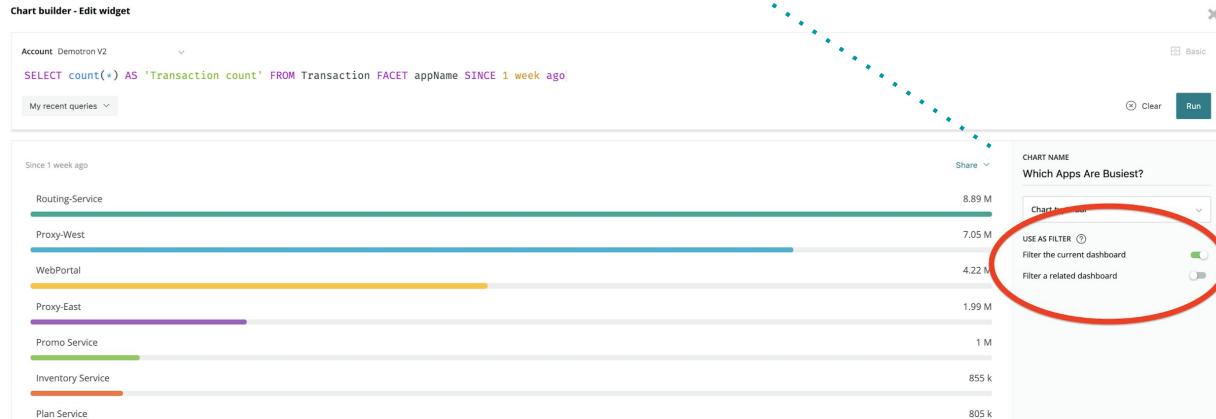
- Looks for group behavior and values that are outliers from those groups
- Uses **FACET ***

Interactive Dashboards & Collaboration

Linking FACETS to filter charts across a Dashboard

Link One ('general') Dashboard to another ('specific') Dashboard

Ways to share charts with teams or others



Lab: Chart Challenge!

Add as many of these charts! use DemotronV2 account and WebPortal application:

- 1 How many live visitors are on my site right now?

```
uniqueCount(session) AS 'Site Visitors'
```

- 2 Throughput grouped by Transaction

```
FACET name
```

- 3 What response codes are we seeing?

```
FACET HttpStatusCode
```

- 4 Average response time grouped by Transaction

```
FACET name
```

- 5 Show some percentile breakdowns of Transaction response times

```
percentile(duration, 5, 50, 95)
```

- 6 Show a distribution (histogram) of Transaction response times

```
histogram(duration, 0.5, 30)
```

- 7 Use the multi-count function to show how the front end and back end performance compare right now (past 10 mins) ?

```
average (duration),  
average (backendDuration)
```

- 8 You have added a custom attribute to your Transaction event type in the WebPortal app to store shopping cart total value:

PurchasedCartGrandTotal

- Show total shopping cart value in the past hour
 $\text{sum}(\text{PurchasedCartGrandTotal}) / 100$

Lab: Advanced NRQL ideas

Try some of these queries - choose DemotronV2 account

Histogram + FACET = **Heatmap**

- 1 Try a Heatmap of your pageloads or response times using this function and adding a FACET eg. pageUrl or name
 - `histogram(duration, width: 10, buckets: 20)`

- 2 Go further with FACET using **FACET CASES**

```
SELECT count(*) FROM PageView FACET CASES (WHERE duration < 1, WHERE duration > 1 and duration < 10, WHERE duration > 10)
```

Try an example like this for Pageload times by category:

```
SELECT average(duration) FROM PageView FACET CASES( WHERE pageUrl like '%/index.html', WHERE pageUrl like '.....', TIMESERIES
```

- 3 For data sets with a wide range of values try **log scaling** them by wrapping the attribute with `log10`:

```
SELECT percentile(log10(duration), 5, 95) FROM Transaction TIMESERIES
```

Lab: Build My Dashboard!

Build a custom dashboard in your own New Relic account using the Chart Builder. Choose a focus like Front End, Back End, Hosts, E-commerce, or a combination.

Use the following Dashboards for ideas:

- Questions across account
- Browser Overview
- Real time Dashboard
- Overall Application Health
- Tell me about my Application Servers
- Diagnostic Overview

Use the following Charts/ Questions for ideas:

- What cities are our users in? - PageView
- What countries are our users in? - PageView
- Which apps are being used most? - Transaction
- Average app server response time of all apps timeseries? - Transaction
- Average duration page load - PageView

NRQL Alerts Use Cases Outlier using FACET

A host experiences an increase in CPU % above others

- `SELECT average(CPU percent)
FROM systemSample FACET
hostname`

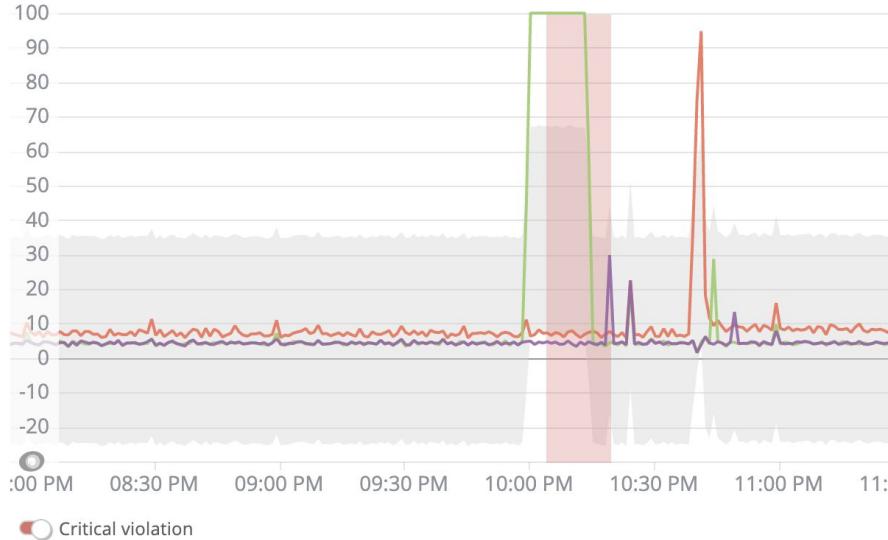
A load-balanced application has misbehaving instances:

- `SELECT average(duration)
FROM Transaction WHERE
appName = 'MY-APP-NAME'
FACET host`

NRQL> `SELECT average(cpuPercent) FROM SystemSample FACET hostname`

> What's possible with NRQL alerting?

PREVIEW CHART



THRESHOLDS

Threshold Type: Static Baseline Outlier [?](#)

Number of expected groups: 1 Change

Trigger violation when groups o

When any facet(s) deviate from the center of the group



for at least

5

minutes by

30

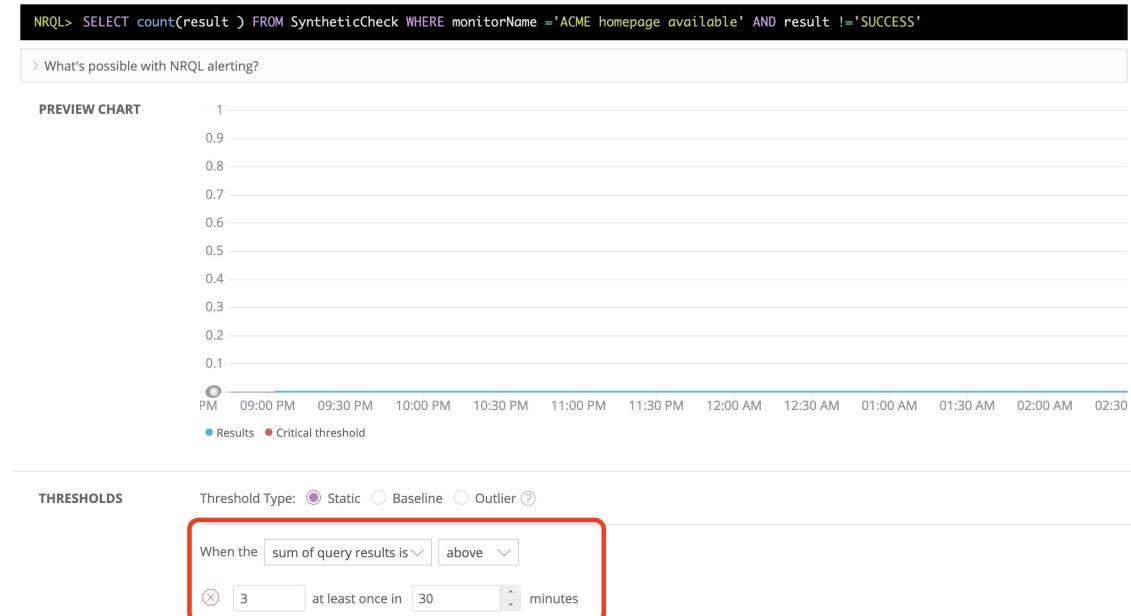
or more

NRQL Alerts Use Cases

Static Value - Sum of Query Results > 'some threshold value'

Avoid receiving multiple alerts
when intermittent/ network
issues cause repeated failures of
a Synthetics check (flapping)

- ```
SELECT count(result) FROM SyntheticCheck WHERE monitorName = 'ACME homepage available' AND result != 'SUCCESS'
```



# Lab: NRQL Alerts

**Create an NRQL Alert in the NRU Training account, then discuss with a partner:**

1. Create a NRQL Alert condition of your choice

*eg. Alert me when duration of my Pageview events take longer than 7 seconds*

---

2. Check our docs site for more NRQL use case examples:

<https://docs.newrelic.com/docs/alerts/new-relic-alerts/defining-conditions/create-alert-conditions-nrql-queries#examples>

3. Have you used NRQL Alerts in your environments already, if so how?

4. What kind of NRQL Alerts might be useful for you or your teams? (make notes)
- 
-

# Full Stack Observability

## Part 1

Troubleshooting Backend Performance  
APM, Distributed Tracing & Logs



# Full Stack Observability Part 1 Troubleshooting Backend Performance APM, Distributed Tracing & Logs

Liam Hurrell: Technical Trainer - New Relic University

Date

# Agenda: Troubleshooting Backend Performance

- 1 How does APM work? What's an agent?
- 2 Charts, Transactions, Errors, Databases, External services Lab
- 3 Using Distributed Tracing Lab
- 4 Troubleshooting Challenge Lab
- 5 Using Logs
- 6 Apdex and Key Transactions Lab
- 7 Alerting on backend services Lab

# New Relic One

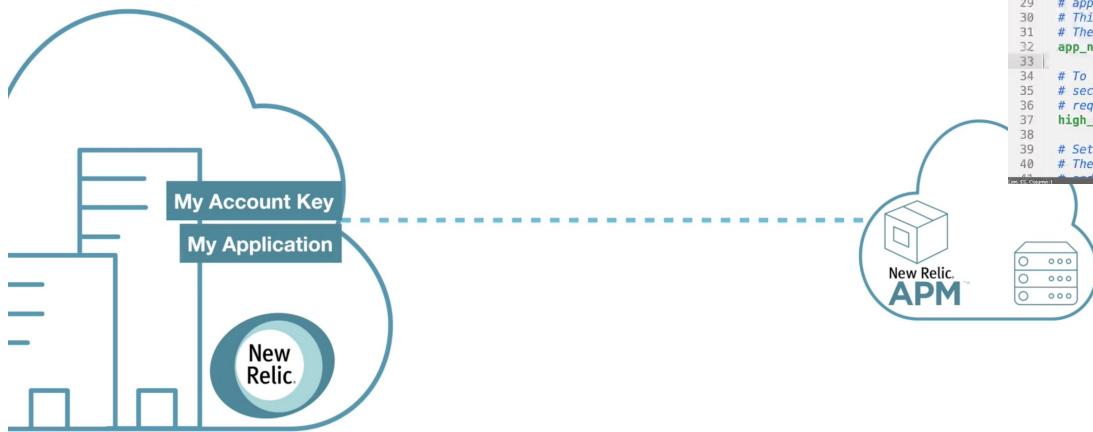
## How does New Relic work?



# What's an agent?

## Configure agents with x 2 values

1. License key
2. Application name



2 Download the Java agent

```
agent_enabled: true
22
23 # Set the name of your application as you'd like it show up in New Relic
24 # If enable_auto_app_naming is false, the agent reports all data to the
25 # Otherwise, the agent reports only background tasks (transactions for
26 # to this application. To report data to more than one application
27 # (useful for rollup reporting), separate the application names with *
28 # For example, to report data to "My Application" and "My Application
29 # app_name: My Application;My Application 2
30 # This setting is required. Up to 3 different application names can be
31 # The first application name must be unique.
32 app_name: Application1; ApplicationRollup
33
34 # To enable high security, set this property to true. When in high
35 # security mode, the agent will use SSL and obfuscated SQL. Additional
36 # request parameters and message parameters will not be sent to New Relic
37 high_security: false
38
39 # Set to true to enable support for auto app naming.
40 # The name of each web app is detected automatically
41 # and the application data correctly assigned.
```

# Services Summary

## Explore key metrics linked to Time-picker and related links



# Lab: Explore the Services Summary

The Services Summary page in APM is a set of curated charts to help you diagnose issues and follow links to delve deeper into related metrics, and troubleshoot the problem. Explore this page for a service and answer the questions below:

1. Which charts can you zoom in on?
2. How can view individual or different Instances/JVMs?
3. How can view the slowest 5 percentiles of transactions?
4. How do know what other services or databases it is connected to?
5. Explore some slow transactions - what is a Transaction Trace?

# Lab: Explore Distributed Tracing

You have experienced some slow transactions in the WebPortal service, but aren't sure if the issue is an API call to another service.

1. Choose Distributed Tracing from the Apps menu page.
2. Filter to the service.name = webportal since 24 hours ago
3. How long is the longest Trace ('Duration') ? \_\_\_\_\_
4. Explore a Trace for the oops.jsp transaction. How many Entities/Services does it go through? \_\_\_\_\_
5. What is an Anomalous Span? Explore one \_\_\_\_\_
6. Explore an error in a Span. What are the error details? \_\_\_\_\_
7. Using links from this page, how can you get more information about this span or error?  
\_\_\_\_\_ / \_\_\_\_\_
8. Try these settings to filter Traces:
  - Longer than 2 seconds - Traces with >1 error - Since 3 days ago

# Lab: Troubleshooting a performance problem

Issues of slow response times have been reported for the 'WebPortal' service in the past 24 hours. This is a service you are not familiar with.

Use the Entity Explorer to find the Webportal service in the Demotron account.

In the Summary page set the time picker to 24 hours

1. Explore the context (Service Map, Dependencies) then dive deeper to explore any issues.
  2. What were the issues? \_\_\_\_\_ What might have been the causes?
  3. Did it impact customer experience? \_\_\_\_\_
  4. Was anyone alerted about it? \_\_\_\_\_
  5. What can we learn from it, or what can we do to improve our DevOps procedures? \_\_\_\_\_
  6. Discuss your findings \_\_\_\_\_



# Lab: Explore Logs

You want to find out what is happening with customer coupons in your WebPortal service. Choose Logs from the New Relic One homepage and set the Account to Demotronv2

1. How can you Filter the Logs to see only those from the Service Name: "WebPortal"

---

2. Filter the Logs from this Service to where 'method.name' contains "validateCoupon"

3. Can you add a column to your search results for the attribute: method.name ?

4. Without leaving this Logs page, how can you see this set of Logs from 6 hours ago?

---

5. Add another filter to this set of Logs where 'message' contains "error"

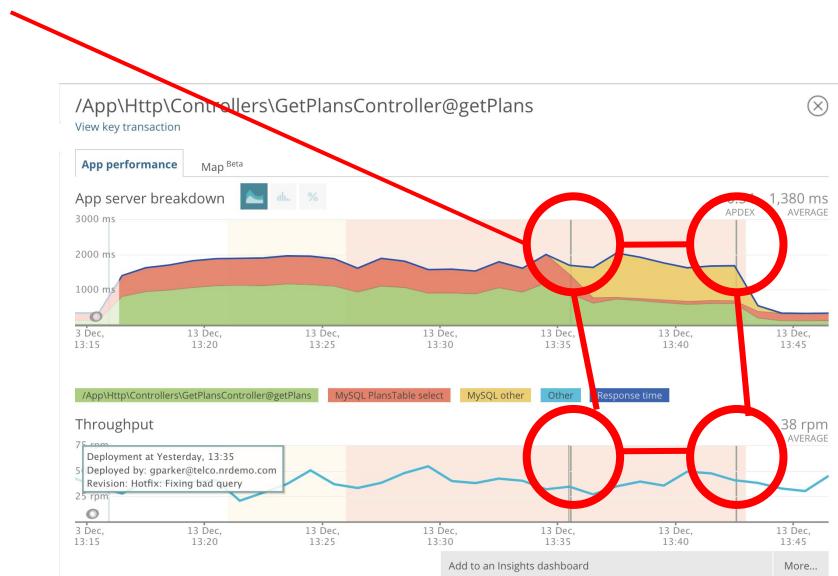
6. How can you create a 'Saved Query' for this set of Logs? \_\_\_\_\_

7. Click on a Log entry and see its attributes. How can you easily share the details of a Log entry with your team? \_\_\_\_\_

# Deployment Markers

Grey vertical lines in your metric charts

- Mark out deployments with versions, descriptors, changes
- Make Performance monitoring and troubleshooting easier
- Create using the REST API / or automate



# Lab: Explore Error Analytics

Check out the New Relic Pet Clinic Errors (NRU Training Account)  
Time period: past 24 hours/3 days

Explore the Error Trace details of 'org.apache.jasper.JasperException'  
and find answers to these questions:

- 1 How many of this transaction/class occurred within the selected time period?
- 2 What is the most recent error message?
- 3 When did it first and last occur? (is it still happening now? Is it happening regularly)
- 4 What are the error details? Response code? Would additional attributes here be useful?

# Troubleshooting back-end problems

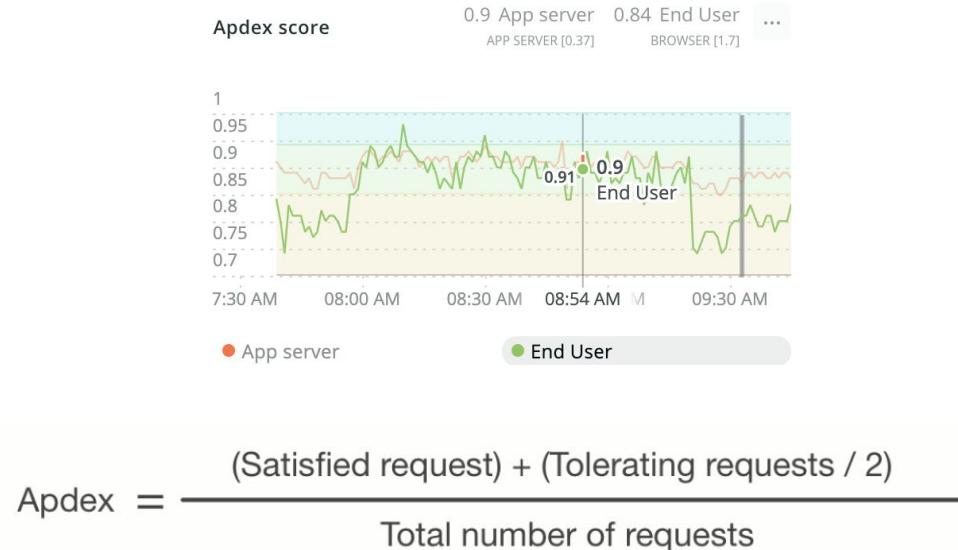
## Techniques to monitor and debug backend components

- Check your microservice **topology**  
(Service Maps, Dependencies)
- **Triage** backend issues  
(APM Service Summary)
- Review slow transactions,  
**'Transaction Traces'**,  
Database calls, Errors
- Configure your backend **Apdex Threshold**
- Identify '**Key Transactions**' in your services
- Set up backend application **Alerts**
- Configure **Deployment Markers**
- Check performance of **Third-party services**  
(Infrastructure Integrations,  
Synthetics Monitors)

# Apdex

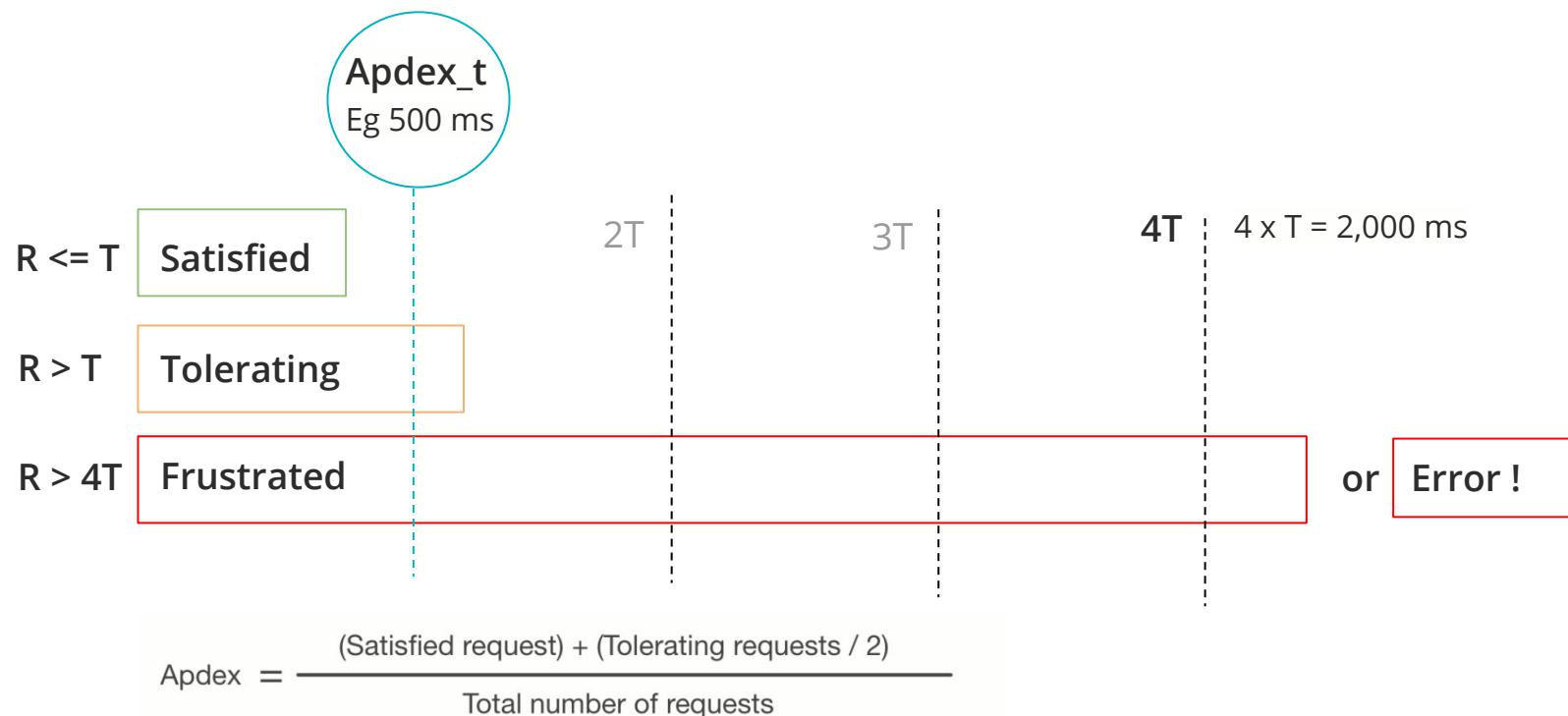
Your Apdex score measures users' satisfaction with the response time of your web applications and services

- How close are the responses staying compared to your target response time
- Score between 0 and 1
- (0.85-0.95 ideal range)



# Apdex Threshold : Apdex\_t

Set this for each of your apps



# Apdex: Setting your Apdex Threshold

Set this for each of your apps

Once you have steady data through your applications, use a query like the following to define your Apdex threshold:

- select percentile(duration, 70) from Transaction where appName=NNNNN since 24 hours ago

| Target Apdex Score | Percentile to Use |
|--------------------|-------------------|
| 0.85               | 70%               |
| 0.88               | 76%               |
| 0.9                | 80%               |
| 0.92               | 84%               |
| 0.94               | 88%               |
| 0.95               | 90%               |
| 0.96               | 92%               |
| 0.98               | 96%               |

# Apdex: Benefits

1. Combines ERRORS + RESPONSE TIME in 1 x single score
2. Takes into account USER EXPERIENCE (not arbitrary / fixed values)
3. Allows you to measure ALL APPLICATIONS / SERVICES on 1 x single scale
4. Simplifies ALERTING
5. Assess performance with little knowledge of the service / application

# Knowledge Check: Apdex



1. What does this value represent in the Apdex chart ?
2. "Apdex is a measurement index designed just for the New Relic Platform"

TRUE / FALSE ?

3. Where you can change or set your application Apdex threshold?
- 

4. If your application has a constant score of 1.0 does that mean it's performing well?

YES / NO - Explain your answer

# Key Transactions

## Are used for:

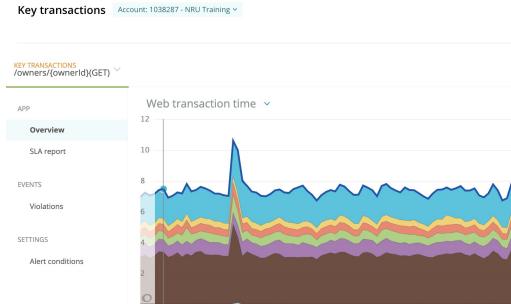
- Important Transactions
- Slow Transactions
- Transactions that behave differently to the average and skew the score

Key transactions Account: 1038287 - NRU Training ▾

| Name                                                               |
|--------------------------------------------------------------------|
| /owners/{ownerId}(GET) <small>New Relic Pet Clinic</small>         |
| get/api/restaurant <small>FooMe</small>                            |
| /owners/find (GET) <small>New Relic Pet Clinic</small>             |
| /avets (GET) testing <small>New Relic Pet Clinic</small>           |
| owners test transaction <small>New Relic Pet Clinic</small>        |
| test transaction <small>New Relic Pet Clinic</small>               |
| /owners/{ownerId}/editN20(GET) <small>New Relic Pet Clinic</small> |
| /dandelionServlet/demo <small>New Relic Pet Clinic</small>         |
| /avets (GET) <small>New Relic Pet Clinic</small>                   |

## Benefits:

- Monitor them easily and over time (with the SLA Report)
- Set a custom Apdex\_t for them
- Use to get **Alerts** on



# Lab: Key Transactions

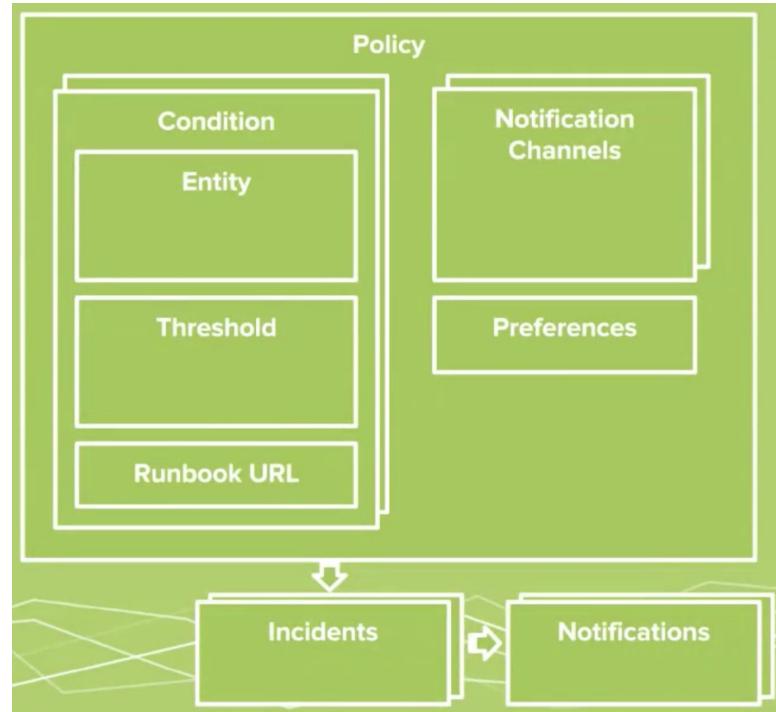
You want to monitor some important transactions in the Pet Clinic Application

1. Add a Key Transaction
  2. Give it a 'custom' name
  3. Increase the Apdex threshold for this transaction
  4. Can you think of example transactions which you would add for your services?
-

# Alerts

## Terminology

- The **Alert Policy** is the container for **Conditions** and **Channels**
- Threshold violations in your conditions are called **Incidents**
- which create **Issues** which you get **Notifications** about



# Alert Policies

A Policy is a collection of

- **Conditions**

Something you want to be notified about

➤ eg Apdex is below 0.7



- **Channels**

How you want to be notified

- eg Slack, Webhooks, Mobile App

# Dynamic Baseline Alerting

Chooses the best algorithm automatically (using an ensemble system)

- Dynamic Baseline Alerts can be created on metrics data and on NRQL queries
- Automatically finds abnormalities in your environment
- Ideal for new applications where you haven't set performance levels yet

Team name - Service - Issue or Metric

New condition



# Lab: Alerts

Set up some Alerts on the things that are important to you (use the Pet Clinic App)

- 1 Set up an Alert to monitor a specific JVM
  - 2 Set up an Alert to notify you if the Apdex score on the Pet Clinic app drops. Add a warning.
- You have only recently started monitoring this application and haven't set SLA thresholds for its response times or throughput yet.
- 3 Set up an Alert condition to notify you when the application deviates from its baseline performance

We'd love to hear what you think?

Please take one minute to share your feedback with us

## Survey Day 1

<https://www.surveymonkey.com/r/nrufsoworkshopday1>

# Full Stack Observability

## Part 2

Troubleshooting Infrastructure and  
Infrastructure Alerts



# Full Stack Observability Part 2 Troubleshooting Infrastructure and Infrastructure Alerts

Liam Hurrell: Technical Trainer - New Relic University

Date

We'd love to hear what you think?

Please take one minute to share your feedback with us

## Survey Day 2

<https://www.surveymonkey.com/r/nrufsoworkshopday2>

# Lab: Explore Infrastructure

Use the NRU Training account

**1** Create a 'FoodMe-yourName' Filter Set which filters to the FoodMe application

**2** Have you used Filter Sets in your account? How do you/or might you use them? What would you filter to? (Avail zone? App name? Environment? [discuss](#))

---

**3** SYSTEM TAB: You want to see if you have the right instance types and sizes for your throughput. How can you change the 'Group By' options to show Instance sizes? Have you seen the Cloud Optimise App?

---

**4** You want to see if spikes in system or network metrics are affecting application performance. Change the default charts to view a host Metric alongside 'Application Response Time'

**5** How can you view the Query for the chart

---

---

# Lab: Explore Infrastructure

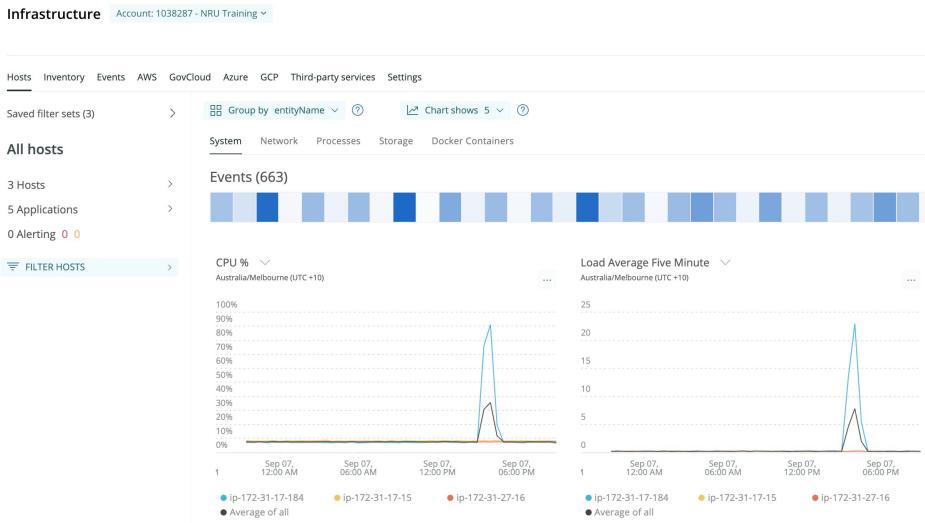
NRU Training Account: Pet Clinic App: Troubleshoot a performance problem

Visit the 'permalink' provided by the instructor and answer the following questions:

1. Is there a process on the host taking resources from the application? If so, what is the process and who ran it?

Discuss with your partner/team:

2. How did you determine which process was consuming resources?



# Smart HNR - Infra Alert Condition

## Don't trigger alerts for hosts that perform a clean shutdown

This alert condition differentiates;

- a clean shutdown from
- an unplanned outage of the server

This makes it possible to create Host-Not-Reporting on servers in an auto-scaling group and;

- Alert when a host is crashing,
- Not alert when a host is terminated by AWS auto-scaling

The screenshot shows the 'Alerts' tab in the New Relic Infrastructure Settings. It includes tabs for Host metrics, Network metrics, Process metrics, Container metrics, Process running, Storage metrics, and Host not reporting (which is highlighted). Below these are sections for Narrow down hosts, FILTER HOSTS, and Specify host. The main configuration area has a 'Define threshold' section with a radio button for 'Host not reporting for [5] minutes' and a checked checkbox for 'Don't trigger alerts for hosts that perform a clean shutdown'. A tooltip for the checkbox states: 'This can prevent false alerts when you have hosts set to shut down when not in use. Currently supported on all Windows systems and Linux systems using systemd. This requires agent version of 1.7.0 or newer. Check your version.' At the bottom, there are sections for Alert policy (Team name - Service - Issue or Metric), Condition status (Enabled), and Runbook (+ Add a runbook URL).

# Lab: Inventory, Events & Infrastructure Alerts

## Demotronv2 account

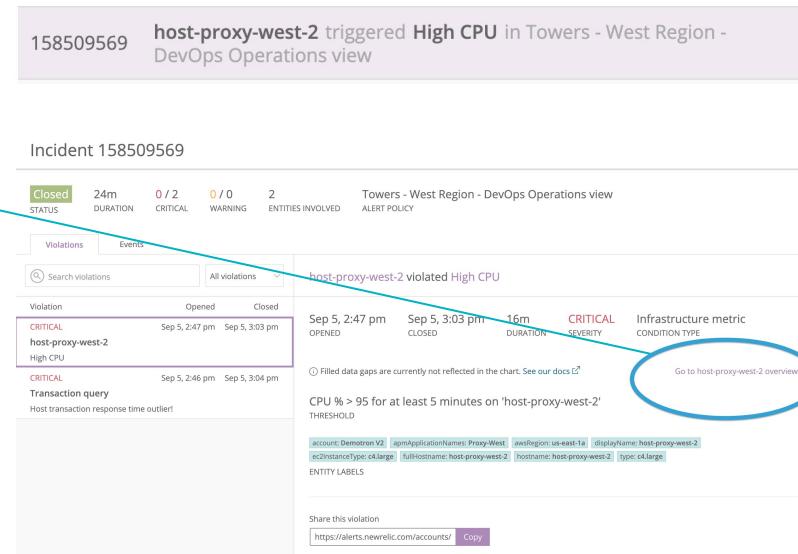
Go to ALERTS - All Incidents list - (search cpu) click on High CPU on Host Proxy West 2 - click violation link to explore events.

1. What happened? \_\_\_\_\_

2. Click on the **Inventory** Tab:  
Are any hosts vulnerable to the Heartbleed vulnerability from 2014? \_\_\_\_\_

## NRU Training account

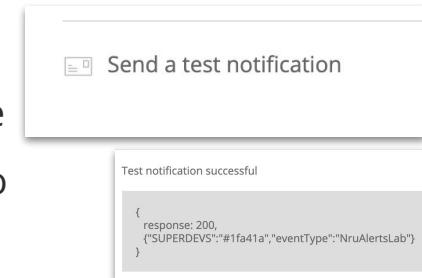
3. From Infrastructure, set up an Alert of your choice. **Discuss** with your team which one you chose and why\_\_\_\_\_



# Lab: Alerts Challenge (ensure muting rules disabled)

## Part 1 - Get your webhook Channel working

1. Work in teams if possible and choose a (family-friendly!) team name
2. Create a webhook notification channel that will send notifications to  
<https://webhook.nru.to/>
3. Customize the webhook payload so that it includes an element called 'teamName', with a value of your team's name - try a 'Notification Test'



## Part 2 - Create your Infrastructure Policy & Condition

1. Create an Infrastructure alert Condition that will trigger a violation when CPU usage on [any New Relic Pet Clinic host](#) is greater than 50% for at least 2 minutes
2. Add this Condition to an alert policy, and configure the policy to notify your webhook channel

Look at this indicator: <http://alert-indicator.nru.to>

# Full Stack Observability

## Part 3

Front End Performance and Proactive  
Monitoring with Browser and Synthetics



# Full Stack Observability Part 3 : Browser and Synthetics

Liam Hurrell: Technical Trainer - New Relic University

Date

# Front end Performance

*A framework to improve  
digital customer experience  
through the different layers  
of your stack*

## Availability

Is it up and running?

## Functionality

Is it working right?

## Performance

Is it working fast enough?



# Availability

## Is it up and running?

1. Look for Throughput drops
  - a. [Browser Summary charts / Dashboards / Browser Alert \(Throughput drops\)](#)
2. Set up 'availability alerts' using Synthetics
  - a. [URL not reachable / API reports incorrect payload](#)
3. Track uptime against service level agreements (SLAs)
  - a. [Using Synthetics SLA reporting](#)

# Functionality

Make sure it's working for your customers

1. Triage the JavaScript errors
  - a. Prioritise - view by name, frequency, browser, custom attributes
2. Set up 'scripted browser' monitors in Synthetics
  - a. Are the critical user paths through your site working (login, checkout, search, etc.)
3. Set up Alerts on your Synthetics monitors
  - a. Get notified if something affects user experience (even if back-end or infrastructure)

# Speed

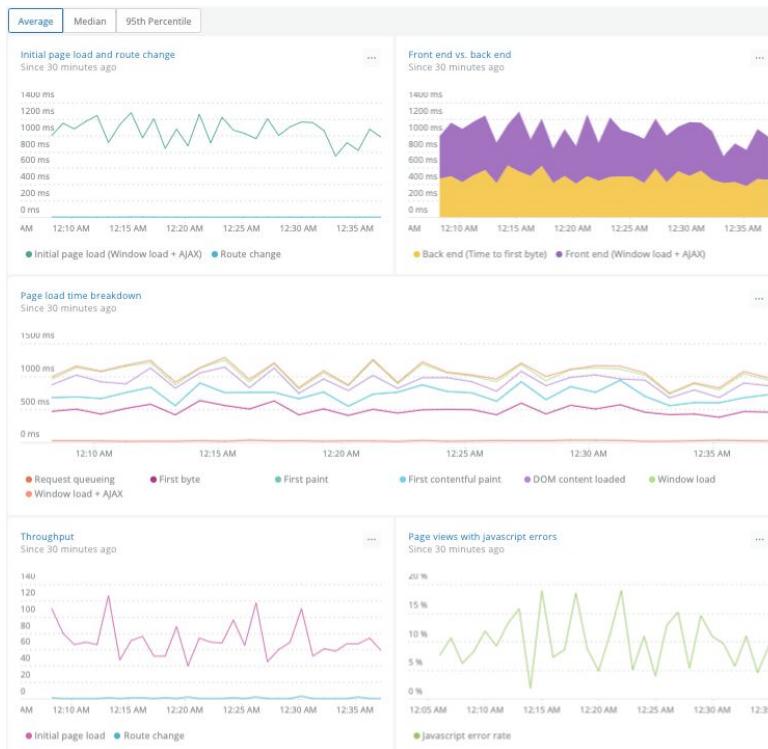
Optimise experience and minimise bounce rates

1. Page Load times
  - a. Use Percentile & Histograms to avoid skewed averages
  - b. Look at First Paint/First Contentful Paint
2. Configure front end Apdex Threshold
  - a. Are the critical user paths through your site working (login, checkout, search, etc.)
3. Optimise front-end code / manage 'page bloat' with Session Traces
  - a. Check for blocking JS, slow AJAX calls / Session Trace Waterfall
4. Set up page load speed alerts
  - a. Use Dynamic Baseline Alerting to set healthy page load thresholds

# Troubleshooting front-end problems

What correlates in Summary page?

- Drops in Throughput
- Spikes (small?) in Front End time
- Wide gaps between first byte and DOM content load/Page Load
- Drops in Apdex
- Spike in JS Errors
- Explore Median & 95th Percentiles  
(averages can be skewed by outliers)



# Focus on 'User-centric' performance metrics

## Measure when end users perceive your pages are starting to render

### First Paint:

- First pixel renders on a screen  
(background?)

### First Contentful Paint:

- Any content from the DOM  
(text, images, non-white canvas, or SVG)



# Lab: Browser Performance - Summary Page

Use Demotron v2 account - WebPortal App - past 60 mins

Front End performance issues are more subtle in charts than back end spikes. Zoom in.

1. Can you find some increases in front end time unrelated to back end time time? or wide gaps between first byte and DOM content loaded? which charts did you use? \_\_\_\_\_
2. Change the Page Load time chart to show 95th percentiles. How slow are some pages? \_\_\_\_\_
3. Do any spikes correlate to JS Errors? Which common JS Error is occurring? \_\_\_\_\_
4. Use Page views menu: What parts of the site are slow? / \_\_\_\_\_ / \_\_\_\_\_
5. Find a slow 'Session Trace' for this page
6. What was causing the slow page loads? Was it based on a particular Browser type? \_\_\_\_\_
7. What experience do you have of using New Relic Browser? (discuss) \_\_\_\_\_
8. Make some notes about Browser and how you can use it  
\_\_\_\_\_  
\_\_\_\_\_

# Lab: Custom charts & alerts Front End Performance

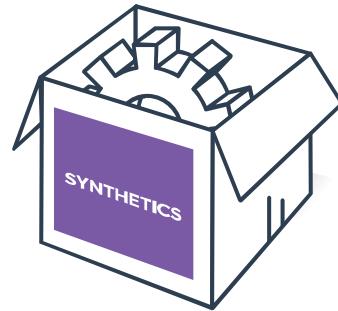
Create the following charts & alert policy in NRU Training account & Pet Clinic App

1. Monitor median baselines by comparing current versus historical performance of firstContentfulPaint
  - a. `SELECT percentile(firstContentfulPaint, 50) as 'First contentful paint' FROM Pageview WHERE appName=WebPortal SINCE 1 day ago COMPARE WITH 2 days ago TIMESERIES`
2. Find outliers using the 95th percentile
  - a. `SELECT percentile(firstPaint, 50) as 'First paint', percentile(firstPaint, 95) as 'First paint', percentile(duration, 50) AS 'Window load + AJAX' FROM BrowserInteraction WHERE appId=123456 TIMESERIES SINCE 3 days ago UNTIL now`
3. Create an NRQL alert if a user doesn't see content or pixels within 7 seconds
  - a. Try using NRQL: `SELECT percentile(firstPaint , 99) FROM BrowserInteraction`
  - b. Threshold: Static When Query Returns a value above 7 seconds

# Synthetics

## Proactive Monitoring

**Proactive**  
Find problems before  
users see them

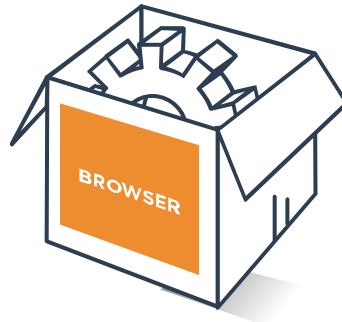


**Consistent, &  
predictable**  
Good for alerting

# RUM and Synthetics

## RUM-Syn work best together

Wide variety  
browsers devices  
locations



Valuable real-world  
performance  
debugging info

# Synthetics

## 4 x Monitor Types

- **Availability** - **Ping Monitor** - HTTP client - HEAD/GET requests to single url
- **Functionality**
  - **Scripted Browser** (Selenium Scripts) - test navigation / funct'y
  - **API Test** (HTTP object)
- **Performance** - **Simple Browser** (Chrome) - single url - waits for page load event

# Synthetics

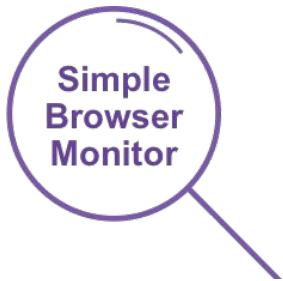
## Use cases



Track up/down availability

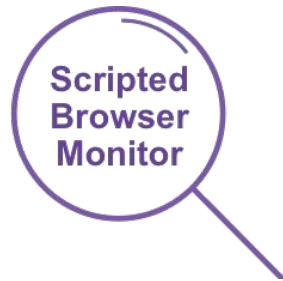
Availability

99.1% LAST 30 DAYS    99.1% LAST 7 DAYS    100% LAST 30 MINUTES



Analyze page load performance

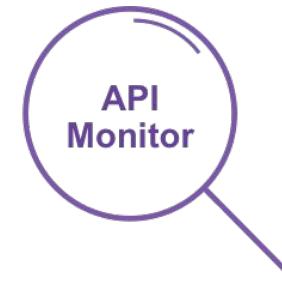
Screenshot of result



Advanced tests for complex workflows

```
function clickHome() {
 return $browser.findElement(By.linkText("Home")).click();
}

function clickBrowse() {
 return $browser.waitForAndFindElement(By.linkText("Explore New Relic"))
 .element.click().then(function() {
 $browser.waitForAndFindElement(By.xpath("//h1"));
 });
}
```



Ensure key APIs are operational

| SORT BY                               |        |
|---------------------------------------|--------|
| Average size                          | 52.0 B |
| api.newrelic.com/v2/applications.json | 52.0 B |

# Lab: Synthetics

## NRU Training Account

Use your knowledge of the different Synthetic monitor types to:

1. Set up a monitor to ensure your homepage is available
2. Set up a monitor to provide you with page performance data for your homepage
3. Create an Alert Policy to notify you if one of your monitors fail
4. Create a Scripted Browser Monitor to monitor a 3 step navigation path on your site  
(use Selenium Chrome ext. Or Katalon or copy/paste an example from  
<https://docs.newrelic.com/docs/synthetics/new-relic-synthetics/scripting-monitors/scripted-browser-examples>

# Lab: Synthetics discussion

1. Discuss with your team how have you used Synthetics before?

---

2. What ideas do you have on how you can use Synthetics in your environments?

---

---

3. Make some notes on Synthetics

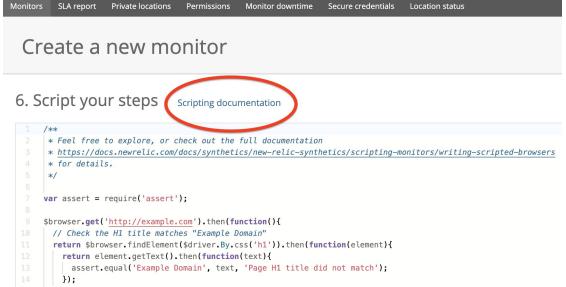
---

---

---

# Lab: Synthetics Scripted Browser (part1)

1. Create a Scripted Browser. Monitor with one Location and 5 minute schedule
2. Click “Next: Write your script” which will take you to the Browser based IDE
3. Write your script. Use an example from our docs site, or use the Selenium IDE and the New Relic plugin
4. click on your Selenium Chrome plugin icon to launch the IDE and click “Record a new test” in a ‘new project’



The screenshot shows the 'Create a new monitor' page in the New Relic interface. At the top, there's a navigation bar with links: Monitors, SLA report, Private locations, Permissions, Monitor downtime, Secure credentials, and Location status. Below the navigation, the title 'Create a new monitor' is displayed. A section titled '6. Script your steps' contains a 'Scripting documentation' link, which is highlighted with a red circle. Below this, there is a code editor window showing a JavaScript script. The script starts with a comment block and then defines a function that gets a browser instance, checks for an H1 title, and asserts its text against 'Example Domain'. The code is numbered from 1 to 14.

```
1 // Feel free to explore, or check out the full documentation
2 // https://docs.newrelic.com/docs/synthetics/new-relic-synthetics/scripting-monitors/writing-scripted-browsers
3 * for details.
4 */
5
6 var assert = require('assert');
7
8
9 browser.get('http://example.com').then(function(){
10 // Check the H1 title matches "Example Domain"
11 return browser.findElement(By.css('h1')).then(function(element){
12 return element.getText().then(function(text){
13 assert.equal('Example Domain', text, 'Page H1 title did not match');
14 });
15 });
16 })
```

# Lab: Synthetics Scripted Browser (part2)

1. Click on your Se icon to launch the IDE and click “Record a new test” in a ‘new project’
2. Give your project a name and click OK
3. Next you need to specify a valid URL and click “Start Recording”
4. Stop Recording when done and export using the New Relic Synthetics format - open this file with a text or code editor and copy into your Synthetics Monitor

The screenshot shows the Selenium IDE interface. At the top, it says "Welcome to Selenium IDE! Version 3.13.0". Below that, there's a menu with options: "Record a new test in a new project", "Open an existing project", "Create a new project", and "Close Selenium IDE". A message below the menu says "To learn more on Selenium IDE and how to use IDE project page." On the right, there's a panel titled "Set your project's base URL" with a "BASE URL" field containing "http://telco.nrdemo.com/" and a "Start Recording" button. In the center, there's a list of recorded commands for a script named "ACME-purchase-prod...". The last command in the list is a "click" action. At the bottom right, there's a "Select language" dropdown with "New Relic Synthetics" selected. The "Start Recording" button and the "Select language" dropdown are highlighted with yellow boxes.

# Lab: Synthetics Dashboard

## NRU Training Account

Build a Dashboard to show the success rates & performance of your Monitors. Use the ideas below to help you...

1. Success rate for a monitor (a percentage of the Synthetic Checks where the result is 'SUCCESS').

```
SELECT percentage(count(*), WHERE result = 'SUCCESS') AS 'Success Rate' FROM
SyntheticCheck WHERE monitorName = 'New Relic Pet Clinic' SINCE 1 week ago
```

2. What are the most common error messages for your Synthetic Monitor?

```
SELECT count(result) FROM SyntheticCheck FACET error
```

3. What is the average load time of your monitor grouped by location?

```
SELECT average(duration) FROM SyntheticCheck FACET locationLabel
```

4. What are your load times group by asset type (TIMESERIES, Bar Chart)?

```
SELECT average(duration) from SyntheticRequest facet contentCategory
```

# Lab: Synthetics and NRQL Alerts

NRU Training Account

Now that we've got our Dashboard up and running, let's build an NRQL alert condition. Take a few minutes to do this for yourself.

1. One suggestion would be "Alert me if the success rate of my monitor drops below 90% for 5 minutes"
2. HINT: `SELECT percentage(count(*), WHERE result ='SUCCESS' AS 'Success Rate")`
3. Work with a partner.... What others can you think of? List them here:  

---

---

We'd love to hear what you think?

Please take one minute to share your feedback with us

## Survey Day 1

<https://www.surveymonkey.com/r/nrufsoworkshopday1>

## Survey Day 2

<https://www.surveymonkey.com/r/nrufsoworkshopday2>

Keep learning !

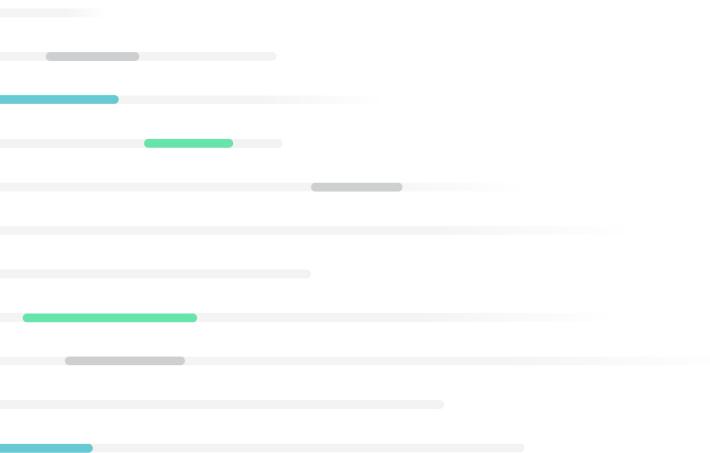
<https://learn.newrelic.com/>



# Questions

**<https://tinyurl.com/nru-signup>**

---



# Thank You

Technical Trainer - New Relic University



©2008–20 New Relic, Inc. All rights reserved