

アプリケーションと インフラ性能観測の基本

NRU 301

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

Safe Harbor

This presentation and the information herein (including any information that may be incorporated by reference) is provided for informational purposes only and should not be construed as an offer, commitment, promise or obligation on behalf of New Relic, Inc. ("New Relic") to sell securities or deliver any product, material, code, functionality, or other feature. Any information provided hereby is proprietary to New Relic and may not be replicated or disclosed without New Relic's express written permission.

Such information may contain forward-looking statements within the meaning of federal securities laws. Any statement that is not a historical fact or refers to expectations, projections, future plans, objectives, estimates, goals, or other characterizations of future events is a forward-looking statement. These forward-looking statements can often be identified as such because the context of the statement will include words such as "believes," "anticipates," expects" or words of similar import.

Actual results may differ materially from those expressed in these forward-looking statements, which speak only as of the date hereof, and are subject to change at any time without notice. Existing and prospective investors, customers and other third parties transacting business with New Relic are cautioned not to place undue reliance on this forward-looking information. The achievement or success of the matters covered by such forward-looking statements are based on New Relic's current assumptions, expectations, and beliefs and are subject to substantial risks, uncertainties, assumptions, and changes in circumstances that may cause the actual results, performance, or achievements to differ materially from those expressed or implied in any forward-looking statement. Further information on factors that could affect such forward-looking statements is included in the filings New Relic makes with the SEC from time to time. Copies of these documents may be obtained by visiting New Relic's Investor Relations website at ir.newrelic.com or the SEC's website at www.sec.gov.

New Relic assumes no obligation and does not intend to update these forward-looking statements, except as required by law. New Relic makes no warranties, expressed or implied, in this presentation or otherwise, with respect to the information provided.

自己紹介

- •New Relic 株式会社 シニアテクニカルサポートエンジニア •伊藤 覚宏 (いとう あきひろ)
- •OSS監視ソリューションのテクニカルサポート、クラウド環境の構築・設計、 運用設計の経験を活かしお客様をサポートいたします。

-専門:監視、クラウドアーキテクト、AWS、VMware、Zabbix

- •インフラエンジニア
- •クラウドエンジニア
- •サポートエンジニア

https://8card.net/virtual_cards/XEO7QJF4GpyOYaDU-HMxEg_38702273134





本日のゴール

• 近年のアプリケーションに必要と言われている可観測性についての理解を深め

- New Relicを使って簡単に可観測性を実現する方法を知る
- アプリケーションのエラーやパフォーマンス問題に対し、迅速に対処する方法を

知る

る

- サーバーからアプリケーションまでの統合的なパフォーマンス解析を理解する。
- 狼少年とならない適切なアラート設定を検討する。

アジェンダ

- 時間(目安) 内容
- 15:00 15:10 New Relic APMの紹介とハンズオン環境のご案内
- 15:10 15:35 ハンズオン(1): アプリケーションパフォーマンスの測定
- 15:35 15:40 New Relic Infrastructure の紹介とハンズオン環境のご案内
- 15:40 16:05 ハンズオン(2):アプリケーションパフォーマンスとホストパフォーマンス
- 16:05 16:10 ダッシュボード・アラートのご紹介とハンズオン環境のご案内
- 16:10 16:50 ハンズオン(3):有意義なアラート設定
- 16:50 17:00 まとめ、アンケートご記入

New Relic APM

- アプリケーションの可視化 -

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

New Relic APM





New Relicの導入ステージ



トラブルシューティングの迅速化



エラーの増加を発見

エラーの原因をNew Relic上で追求

/oops.jsp

2020年8月5日 10:54

Error details Filed tickets JSON NR

Stack trace [Show framework code]

org.apache.http.client.HttpResponseException: Internal Server Error

ame.storffont.serviceprov.ServiceProv.getHeppone cmm.storffont.serviceprov.CompositeviceProv.stAVIL storeficet.serviceprov.CompositeviceProv.stAVIL ()odd.madvoc.ArticaReguest.invoke ()odd.madvoc.ArticaReguest.invoke me.storffont.lasidstaTerroptor.invokeProve)odd.madvoc.ArticaReguest.invoke ()odd.madvoc.ArticaReguest.invoke ()odd.madvoc.ArticaReguest.invoke ()odd.madvoc.ArticaReguest.invoke ()odd.madvoc.interroptor.serviceprov.invokeProve)odd.madvoc.interroptor.serviceprov.invokeProve ()odd.madvoc.interroptor.serviceprov.invokeProve)odd.madvoc.interroptor.serviceprov.invokeProve ()odd.madvoc.interroptor.serviceprov.invokeProve ()odd.madvoc.interroptor.serviceprov.invokeProve ()odd.madvoc.interroptor.serviceprov.invokeProve ()odd.madvoc.interroptor.serviceprov.invokeProve ()odd.madvoc.ArticaReguest.invoke ())odd.madvoc.ArticaReguest.invoke ())odd.madvoc.ArticaReguest.invoke ())odd.madvoc.ArticaReguest.invoke ()))odd.madvoc.ArticaReguest.invoke ()))	<pre>(ServiceFromy, jews:50) (ValidsetComponent, jews:50) (ValidsetComponent, jews:50) (ActoroBerguer, jews:160) (ActoroBerguer, jews:141) (ActoroBerguer, jews:142) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (BaseLotonscoptor, jews:143) (BaseLotonscoptor, jews:143) (ServiceSoft(gisterosptor, jews:13) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143) (ActoroBerguer, jews:143)</pre>
jodd.madvoc.component.MadvocController.invoke jodd.madvoc.BadvocServletFilter.doFilter	(MadvocController.java:164) (MadvocServletFilter.java:111)

Host	lp-172-31-19-138
instance	ip-172-31-19-138:8080
Referring app	WebPortal
Timestamp	1596592453000 (2020年8月5日 10:54)
URL:	/coupons/01defacc-2a9a-41bf-8e07-9b1e 100cd360/isvalid
Error attri	ibutes
actionRequi	est jodd.madvoc.ActionRequest@33dc3a 54
httpRespon Code	^{ve} 500
jvm.thread_	http.nio.8080.exec.1984

パフォーマンスの最適化



パフォーマンスの劣化を発見

ボトルネックとなっている場所をコー ドレベルで解析

Mc	Refresh the cach	wethous Market	XML N4	sourcen) JSON NRI Delete this trace File a ticket		
To Exe	Apr 25, '16	9:39	am 3	000 mi 140 mislano si thei chu suan		
To	Summary Tra	ace deta	in Detai	ase queries		
Ca						
Zo	Expand perform	ance pro	blems C	olapse all		
To	Duration (ms) Du	aration (0	Segment	Drilldown	
Us	3.000		100.00%	TopicsController#show		
Ro	1,230		41.10%	16 cells to Login Service production - Api=V1=TokensController#refresh (https://login.newnelic.com/api/v1/tokens/refresh)		
	19.0		0.63%	11 fast method calls		
La	91.5		17.13%	8 calls to Login Service production - Api:/V1::TokensController#refresh (https://login.newrelic.com/api/v1/tokens/refresh)		
LN.	21.0		0.70%	10 fast method cajis		
	16.0		0.53%	5 calls to SQL - SELECT		
09	25.0		0.83%	8 fast method cells		
LH	1,160		38.73%	layouts/application Template		
Se	63.0		2.10%	Login Service production - Api:/V1::TokensController#refresh (https://login.newrelic.com/epi/v1/tokens/refresh)	-8	
	64.0		2.14%	Login Service production - Api:V1::TokensController#refresh (https://login.newrelic.com/api/v1/tokens/refresh)	-8	
Po	0		0.00%	layouts/_head.html.erb Partial		
Dri	0		0.00%	common/_special_font_face.html.erb Partial		
6.	126		4.21%	common/_discourse_stylesheet.html.erb Partial		
26	253	1	8.38%	4 calls to Login Service production - Api:V1=TokensController#refresh (https://login.newrelic.com/api/v1/tokens/refresh)		
To	131		4.37%	common/_google_universal_analytics.html.arb Partial		
Sin	128		4.27%	application/_header.html.erb Partial		
	65.0		2.17%	Login Service production - Api::V1::TokensController/Prefresh (https://login.newretic.com/api/v1/tokens/refresh)	-8	
On	64.0		2,14%	Login Service production - Api:/VI::TokensController#refresh (https://login.newrelic.com/api/vI/tokens/refresh)	-8	
Usi	130		4.34%	common/_discourse_javascript.html.erb Partial		
Sho	131		4.37%	.common/_google_analytics.html.erb Partial		
ŵ						

イノベーションの促進

🕐 New Relic ONE** 🔡 Apps Deshboards Entry explorer APM Browser Synthesics Mobile Infrastructure Logs Alerts & Al More + - - Q. Search 💾 Query your data 😨 Feedback 🕥 🔷 +

nins Austeries inners O7/11, 7:36 jenkins 24 DD/DMMAKE SDMMAR Apdex 0.97 _{0.5} Errors 0.0218% Resp. time 299 ms CPU MA. DB NA. Memory 105.58 MB. Throughput 541 rp peppomana readpanders - All Routing-Service deployments - All Routing-Service deployments C Prev Deploy using master shz: F4228/71ec5a7a7fd7336a34bdb14c9501e2861a DDRIS Overview Change report	pm
arr 07/11,7:36 jenkins 24 DEX.OMEM.T AT DEPLOYER SEVERON Apdex 0.97 ₀₅ Errors 0.0218% Resp. time 299 ms CPU NA. DB NA. Memory 103.95 MB. Throughput 941 m Plaso MaxABY - All Routing-Service deployments - All Routing-Service deployments Coverview Change report Overview Change report	pm
asses DepUINER AI DepUINER AI Apdex 0.97 _{0.5} Errors 0.0218% Resp. time 299 ms CPU NA DB NA Memory 100.58 M8 Throughput 941 mp asses - All Bouting-Service deployments	pm
Akyments ← All Routing-Service deployments ← All Routing-Service deployments ← Prev Deploy using master sha: F422bf71ec6a7a76f7336a34b4b14c9501e2861a ← Prev x873 Overview Change report	
adg profiler ← All Routing Service deployments Deploy using master sha: f422bf71ec6a7a7fd7336a34b4b14c1501e2861a erts Overview Change report	
Deploy using master sha: f422bf71ec6a7a7fd7336a34b4b14c9501e2861a BTS Overview Change report	vious deployment
815 Overview Change report	
Overview Classifier Edvox	
Response time and throughput CPU utilization	
800 ms 4000 rpm 4 %	
00y 600 ms 1000 mm 1 X	-
transactions 400 mg 2000 rgm 2 %	
200 ms	

リリース前後のパフォーマンスを比較

マイクロサービス化されていても、依存関係を把 握しながら分析が可能





ハンズオン(1)

アプリケーションパフォーマンスの測 定

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

ハンズオン(1)アプリケーションパフォーマンスの測定

[準備]

New Relicにログインしてください。

https://login.newrelic.com/login

ユーザー: japan-handson+2021@newrelic.com

パスワード: oSz6nrupas

(オー、エス、ゼット、ロク、エヌ、アール、ユー、ピー、エー、エス)

※普段NewRelicをお使いの方はセッションが残っている場合がありますのでプライベートブラウジングをお使いください。

Chrome:シークレットウィンドウ

Firefox: プライベートウィンドウ

Edge: InPrivate ウィンドウ

IE: New Relicの一部機能はIEをサポートしていません。上記のいずれかのブラウザをご利用ください。

ハンズオン(1)アプリケーションパフォーマンスの測定

[準備] ログイン後、自動的にAPMのメニューが表示されます。 以下のアプリケーションが見えればログイン成功です。

Filter application	S		Show	labels			P	Application activity
Name	0	End user 0	Page views ©	App server 0	Throughput 0	Error % C		In progress Event log Filter
EC-site		1	1	106 ms	18.3 rpm	0%	0	
PHP Applicat	N	ame	oubleshoot				٢	ee
		ame				1 - 2 of 2		There are no violations in
	1							progress right now.
	E	C-site						4

ハンズオン(1)アプリケーションパフォーマンスの測定

[アプリケーション概要]

このハンズオンでは、PHPおよびMySQLにより構築されたジェラート屋さんの ECサイトを モニタリング対象にしています。

http://ec2-3-113-215-132.ap-northeast-1.compute.amazonaws.com/ec-cube/index.php/

NRU

新入荷 ジェラート アイスサンド





ハンズオン(1)アプリケーションパフォーマンスの測定

1. 時間の設定

データの表示期間を、過去24時間に設定してください。

2. パフォーマンスの確認

トランザクション内で最も時間が費やされているのはPHP, MySQL, 外部APIのうちどの処理レイヤーでしょうか。

PHP・MySQL・外部API

応答時間をヒストグラムで表示してください。最も分布として多い応答時間はどのレンジでしょうか。 最頻応答時間

ハンズオン(1)アプリケーションパフォーマンスの測定

"Transaction"メニューから、トランザクション毎の分析メニューを選択してください。

3. もっとも遅いTransactionは何ですか?

Transactionが発行しているSQLを確認してください。

4. リリース前後のパフォーマンスを比較(オプション)

• "Deployment"メニューより、最新のリリースがいつ実施されたかを確認してください。 直近リリース日時

最新のリリース前後のパフォーマンス比較をしてください。このリリースが原因と考えられるトラブルは発生しているでしょうか。



手順·解説

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

ハンズオン(1)アプリケーションパフォーマンスの測定

1. 時間の設定

データの表示期間を、過去24時間に設定してください。



ハンズオン(1)アプリケーションパフォーマンスの測定

- 2. パフォーマンスの確認
 - トランザクション内で最も時間が費やされているのは PHP, DB, 外部APIのうちどの処理レイヤーでしょうか。

Web tran	nsactions tir	me ()			%	118.83 i App ser	ms 0.48 s e ver browser pa	End user AGE LOAD ····	Summary>Web transactions itme では処理レイ ヤー毎に積層グラフとしてレスポンスタイムが表示 されます。
1000 ms 800 ms									
600 ms 400 ms									
200 ms	Nov 02	Nov 02	Nov 02	Nov 02	Nov 04	Nov 04	Nov 04	Nov.04	
D AM	12:00 PM	03:00 PM	06:00 PM	09:00 PM	12:00 AM	03:00 AM	06:00 AM	09:00 AM	©2008–20 New Relic, Inc. All rights reserved ON New Relic

ハンズオン(1)アプリケーションパフォーマンスの測定

- 2. パフォーマンスの確認
 - 応答時間をヒストグラムで表示してください。最も分布として多い応答時間はどのレンジでしょうか。

Web transactions histogram			
6			Summary>Web transactions itmeで 2番目のアイコンを選択するとヒストグラム表示にな ります。
3			
1 0 0 ms 50 ms 100 ms	150 ms 200 ms 250 ms 300 ms 350 m	ns 400 ms 450 ms	
• Web requests			

ハンズオン(1)アプリケーションパフォーマンスの測定

- 2. パフォーマンスの確認
 - 応答時間をヒストグラムで表示してください。最も分布として多い応答時間はどのレンジでしょうか。



ヒストグラム表示では、応答時間の頻度毎に棒グラ フが表示されます。

ハンズオン(1)アプリケーションパフォーマンスの測定

3. もっとも遅いTransaction Transactionsを開きます。

iummary	Transaction type ⑦ Web ~ S	ort by Most time consur	ning ~
	Top 20 transactions	-	Top web tra
onitor	by most time consuming		by percent of v
stributed tracing	block_search_product	97.26 %	1.6 k%
rvice map			1.4 k%
	admin_homepage_sale	0.75 %	1.2 k%
ependencies	E.		1 k96
ansactions	shopping_checkout	0.47 %	800% 600%
			400%
abases	product_add_cart	0.34 %	200%
ternal cervices	1		0%
ernal services	admin_logout	0.31 %	v 03, No 10 AM 12:1
	E.		block search
ge	admin_homepage	0.28 %	admin_logo
a labar	L.		
ITS INDOX			Thereit

ハンズオン(1)アプリケーションパフォーマンスの測定

3. もっとも遅いTransaction

Slowest average response time を選択します。

Summary	Transaction type ⑦ Web ~ S	ort by Most time consu	ming ~			
	Top 20 transactions	Most time cons	uming			
Monitor	by most time consuming	Slowest average	Slowest average response time Throughput (calls per minute)			
		Throughput (ca				
Distributed tracing	block_search_product	91.20 %	1.6 k%			
Sanica man			1.4 k%			
Service map	admin_homepage_sale	0.75 %	1.2 k%			
Depend <mark>e</mark> ncies	1		1 k96			
	shopping checkout	0.47%	800%			
Transactions	shopping_encerout	0.47 70	600%			
			400%			
Databases	product_add_cart	0.34 %	200%			
-	1		0%			
External services	admin locout	0.21.04	v 03, Nov 03			

ハンズオン(1)アプリケーションパフォーマンスの測定

3. もっとも遅いTransaction

Transaction名を選択しTransaction traceを 選択します。

		esp	ponse time 🗸				Instances (?) All Instan	ces ~
			Database	MySQL atb_order select	99.93	3.00	4./9 s	*
Explorer - / Services - APM / -	🗧 EC-site 🖌 NewRelicUniversity-Japan	Show new 1	Database	MySQL dtb_authority_role select	t 0.05	2.00	2.17 ms	
Summary	Transaction type ⑦ Web ~ Sort b	y Slowest average res	Database	MySQL other	0.01	2.00	0.571 ms	
			Database	MySQL dtb_member select	0.01	1.00	0.523 ms	
Monitor	Top 20 transactions by slowest average response time		Transaction trace	25		Search		
Distributed tracing	admin homepage sale	10.1 5	sample performa	nce details		Scarch		
			Date 🗘	Transaction 🕄			Durati 🗘	
Service map	admin_logout	4.22 s	11月3日 20:19	admin_homepage_sa	ale		10,045 ms	
Dependencies			about 14 hour	's ago /ec-cube/index.php/i	nrujapanhandson/sale_chart			- 11
12	admin_homepage	3.69 s	11月3日 19:19	admin_homepage_sa	ale		10,241 ms	
Transactions			about 15 hour	rs ago /ec-cube/index.php/i	nrujapanhandson/sale_chart			
Databases	chapping chackaut	0.21 c	40%					
	snopping_cneckout	0.215	20%		©2008–20 New Relic	, Inc. All rights	reserved 🛛 🔘 New Relic.	25

ハンズオン(1)アプリケーションパフォーマンスの測定

3. もっとも遅いTransaction

時間がかかっている処理や発行された Queryを確認することができます。

		Summary Trace	Details Datab	base Queries)	
						SELECT d?_id AS id_?, d?_pre_order_id AS pre_order_id_?, d? _order_no AS order no 2, d?_message AS message?, d? _name? AS name?, 2 d?_name? AS name?, 2 d?_aAz hara? AS kana? _?, d?_kana? AS kana?, ?, d?_company_name AS company_name?, d?_postal. Code AS postal.code?, d? addd?
		Total durati 🗘	Call count 💲	Database 🗘	Database instance 💲	AS addr?_?, d?_addr? AS addr?_?, d?_birth AS birth_?, d? Quesubtotal AS subtotal ?, d?_discount AS discount_?, d?
Summary Trace Details Database Queries		4914 ms	3	MySQL	ip-172-31-26-144.ap-northeast-1.compute.internal	set charge, 2, dr.tax AS tax, 2, dr.total AS total 7, dr.payment_total AS payment_total_2, dr.payment_method AS
Time consumed by segment		6 ms	1	MySQL	ip-172-31-26-144.ap-northeast-1.compute.internal	payment_method_2, d2_note AS note 2, d2_create_date AS SET create_date_?, d2_update_date AS update_date_?, d? order_date AS order_date_?, d2_payment_date AS
	(10045)	1 ms	1	MySQL	ip-172-31-26-144.ap-northeast-1.compute.internal	SE payment_date_?, d?_currency_code AS currency_code_?, d? complete_message AS complete_message_?, d?
	MySQL dtb_order select	1 ms	2	MySQL	ip-172-31-26-144.ap-northeast-1.compute.internal	
	#Doctrine\ORM\Internal\Hydration\ObjectHydrator::hydrateAllData	0.mc	1	MISOL	in 172.21.26.144 an partheast 1 compute internal	discriminator_type AS discriminator_type_?, d?customer_id AS customer_id_?, d?country_id AS country_id_?, d?pref_id AS
	Remainder	UTIIS	1	MySQL	ip-172-51-26-144.ap-northeast-r.compute.internal	<pre>>>> pref_id_?, d?_sex_id AS sex_id_?, d?_job_id AS job_id_?, d?</pre>
	#Doctrine\ORM\UnitOfWork:createEntity					payment_id_AS_payment_id_?, d?device_type_id_AS device_type_id_?, d?order_status_id_AS_order_status_id_?, d?
	#Doctrine\ORM\Internal\Hydration\AbstractHydrator::gatherRowDati					_order_status_id AS order_status_id_?, d?_order_status_id AS
	#Doctrine\ORM\Internal\HydrationCompleteHandler::hydrationComp					order_status_id_? FROM dtb_order d?_WHERE (d?_order_date >= ? AND d?_order_date <= ? AND d?_order_status_id NOT IN (?,
	#Eccube\Controller\Admin\AdminController::convert					?, ?, ?)) AND d?_discriminator_type IN (?) ORDER BY d? _order_date ASC

ハンズオン(1)アプリケーションパフォーマンスの測定

4. リリース前後のパフォーマンスを比較 (オプション)

• "Deployment"メニューより、最新のリリースがいつ実施されたかを確認してください。

EC-site 🖌 NewRe	licUniversity-Japan 🛇 5	
Deploymen	Show instructi	ons
Our deployments t	feature reveals the impa	ct of deploy
negative impact of	TApuez, response time,	unougnpu
	< Previous	1 2
Time	Deployer	Apdex sc
Yesterday, 12:03	Systems Manager	0.9
Tuesday, 12:03	Systems Manager	0.9
Monday, 12:02	Systems Manager	0.9
Sunday, 12:02	Systems Manager	0.9
Saturday, 12:03	Systems Manager	0.9
10/29, 12:02	Systems Manager	0.9
10/28, 12:03	Systems Manager	0.9
	 EC-site NewRe Deployments impact or impact or	 EC-site NewRelicUniversity-Japan S Deployments feature reveals the impanegative impact on Apdex, response time, Previous Time Deployer Yesterday, 12:03 Systems Manager Tuesday, 12:02 Systems Manager Sunday, 12:02 Systems Manager Saturday, 12:03 Systems Manager 10/29, 12:02 Systems Manager 10/28, 12:03 Systems Manager

EVENTS>Deployments で記録されたデプロイイベ ント確認することができます。 ※デプロイの記録には RestAPIを利用し、アプリ ケーションデプロイの際に New Relicにデータを送 信する必要があります。

ハンズオン(1)アプリケーションパフォーマンスの測定

4. リリース前後のパフォーマンスを比較 (オプション)

• "Deployment"メニューより、最新のリリースがいつ実施されたかを確認してください。

Explorer / Services - APM / service map Dependencies Transactions	EC-site NewRe Deploymen Our deployments f negative impact or	licUniversity-Japan ⊘ 5 ItS Show instruct feature reveals the imposed of the second seco	tions pact of deplo <u>r</u> e, throughpu	※デプロイの記録には RestAPIを利用し、アプリ ケーションデプロイの際に New Relicにデータを送 信する必要があります。			
Databases External services		< Previou	15 1 2	curl -X POST "https://api.newrelic.com/v2/applications/\${application_id}/deployments.json" ' -H "X-Api-Key:\${api_key}" -i \ -H 'Content-Type: application/ison' \			
Triage	Time	Deployer	Apdex sc	-d \			
Errors Inbox	Yesterday, 12:03	Systems Manager	0.9	'{ "deployment": {			
	Tuesday, 12:03	Systems Manager	0.9	"revision": "\${deploy_revision}",			
Events	Monday, 12:02	Systems Manager	0.9	"changelog": "\${deploy_changelog}", "description": "\${deploy_description}"			
Errors	Sunday, 12:02	Systems Manager	0.9	"user": "\${deploy_user}"			
Violations	Saturday, 12:03	Systems Manager	0.9	}			
Deployments	10/29, 12:02	Systems Manager	0.9	<i>}</i>			
Thread profiler	10/28, 12:03	Systems Manager	0.9	©2008–20 New Relic, Inc. All rights reserved O New Rel			

ハンズオン(1)アプリケーションパフォーマンスの測定

4. リリース前後のパフォーマンスを比較 (オプション)

最新のリリース前後のパフォーマンス比較をしてください。このリリースが原因と考えられるトラブルは発生しているでしょうか。

Deployments

Show instructions

Our deployments feature reveals the impact of deployments on your application's performa negative impact on Apdex, response time, throughput, or errors.

		End user				
Time	Deployer	Apdex score	Resp. time	Throughput		
Today, 12:02	Systems Manager	1.00 _{7.0} *	1.23 sec	1 ppm		

EVENTS>Deployments のTimeリンクをクリックするとその時刻前後でのパフォーマンスが確認できます。

ハンズオン(1)アプリケーションパフォーマンスの測定

4. リリース前後のパフォーマンスを比較 (オプション)

最新のリリース前後のパフォーマンス比較をしてください。このリリースが原因と考えられるトラブルは発生しているでしょうか。



APMエージェントは各種言語のフレームワークに対して Installを行います。 「Add more」をクリックします。



サポート言語アイコンが表示されますので、Agent導入対象の言語をクリックします。

Q Sea	arch for wh	ere you want to co	llect data from				
Conn	nect New	Relic global p	erformance data s	ets			
Free p	sublic data s	ets reporting perfo	ormance and usage stats	of commonly used technology.			
	Public	API					
		1959252					
Back	end, fro	nt-end, and m	obile applications				
Back	(-end, fro amming lan	nt-end, and m	obile applications	nce monitoring.			
Back Progra	c-end, fro emming lan	nt-end, and m suages, framework	obile applications	nce monitoring.			
Back Progra	c-end, fro amming lan	nt-end, and m guages, framework PHP	obile applications is, and customer experie definition definition	nce monitoring.	B Node JS	🙆 Ruby	ng Pyt
Back Progra	c-end, fro emming len මෙ	nt-end, and m guages, framewor PHP Golang	iobile applications (s, and customer experie () Java	nce monitoring. Microsoft.NET New Relic Browser	Node JS Node JS New Relic Synchesics	🙆 Ruby	Pyt
Back Progra	s-end, fro emming lan	nt-end, and m guages, frameword PHP Golang	iobile applications is, and customer experie description of the second second of the second of the s	nce monitoring. Microsoft.NET New Relic Browser	Node JS New Relic Synchesics	🖌 Ruby 💼 IOS	Pyt

インストール対象のアカウントを選択します。



Select the account you want to add this instrumentation to:

Select an account 🗸	
Q Şearch	
Most recent	
NewRelicUniversity-Japan	2511671
New Relic TSS	1336182

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

コピー&ペーストできるインストールコマンドが表示されますので、画面の指示に従ってインストールを行います。

NewRelicUniversity-Japan

(1) Give your application a name

You'll use this to find your data later. It's important to use a unique and meaningful name. See our docs on naming 🖒

Download your custom configuration file

This download includes your custom configuration file with license key, application name, and distributed tracing setting enabled.

Download

(3) Download and install the agent

Download the agent:



Once you've downloaded the agent, follow these steps to start the Java agent installation.

 Create a directory for your New Relic files, such as /opt/newrelic. On Windows, the New Relic files must be in a subdirectory of your application server's directory, such as CMProgram Files/Apache Software Foundation/Toncat 8.0//newrelic.

2. Copy all the New Relic files from your unzipped newnel ic directory into your new directory.

3. Replace the newrel ic.ywl file with the custom config file you downloaded.

(4) Get specific instructions for your Java set up

To use the Java agent with your JVM, you'll need to pass the - Javasent argument. Find what you're using in this list to get specific instructions. In the commands below, replace FULL_PATH_TO with the path of the New Relic directory you created, eg. /oet/newrelic. Introduction to distributed tracing E^r Open instrumentation E^r

<u>New Relic ファーストステップガイド</u> <u>New Relic APM インストール手順</u>も参考にしてください。



New Relic Infrastructure

-アプリケーションパフォーマンスと ホストパフォーマンス-

©2008-20 New Relic, Inc. All rights reserved
New Relic Infrastructure





2つの導入方法:エージェントとクラウド連携

[エージェント] 対象: OS、Kubernetes

[クラウド連携] 対象: クラウドのマネージドサービス

OSまたはKubernetesクラスタにエージェントを導入し、各種メトリックを取得





ミドルウェアとの連携: On Host Integration

MySQL	Couchbase
Cassandra	PostgreSQL
NGINX NGINX	F 5
💜 Redis	Memcached
Apache	Varnish
RabbitMQ	Consul
& Kafka	HAProxy
ORACLE OracleDB	Nagios
<u>,</u> умх	9 NFS
Elasticsearch	statsD
MSSQL	SNMP SNMP
MongoDB	

Infrastructureエージェントの追加モジュールとして 提供されており、ミドルウェアのメトリックを取得可 能

クラウドネイティブ環境に最適化されたUI



Kubernetes クラスタの可視化

AWSマネージドサービスのパフォーマンス (EBSの例)

Amazon EBS - Telco NR Demo



アプリケーションとインフラの相関関係も簡単に把握

		at an
Order-Status 2 open violations, 0 of 6 hosts alerting	Order-Composer 2 open violations, 0 of 4 hosts alerting	Order-Intake
Apdex 0.83 Throughput 804.01 rpm Error rate 0.00% Web resp. time 0.04 s	Apdex 0.50 Throughput 147.07 rpm Error rate 0.00% Web resp. time 1.05 s	Apdex 0.11 Throughput 139.46 rpm Error rate 0.00% Web resp. time 2.23 s
Related hosts (6) show all	Related hosts (4) show all	Related hosts (4) show all
Order-Packaging	Inventory 0 open violations, 0 of 4 hosts alerting	Order-Assembly 0 open violations, 0 of 4 hosts alerting
Apdex 0.95 Throughput 135.46 rpm Error rate 0.00% Web resp. time 1.07 s	Apdex 1.00 Throughput 142.82 rpm Error rate 0.00% Web resp. time 0.22 s	Apdex 1.00 Throughput 146.04 rpm Error rate 0.00% Web resp. time 0.09 s
Related hosts (2) show all	Related hosts (4) show all	Related hosts (4) show all
Bubble-Wrap 0 open violations, 0 of 2 hosts alerting	Email-Notification	Packing-Room 0 open violations, 0 of 2 hosts alerting
Apdex 1.00 Throughput 138.48 rpm Error rate 0.00% Web resp. time 0.19 s	Apdex 1.00 Throughput 131.12 rpm Error rate 0.00% Web resp. time 0.04 s	Apdex 1.00 Throughput 141.69 rpm Error rate 0.00% Web resp. time 0.15 s
Related hosts (2) show all	Related hosts (2) show all	Related hosts (2) show all

アプリケーションとそれを稼働させているインフラの状態 を一覧表示(サービスマップ)

K8sのPodを選択するとそのPod内で動いているア プリケーションの状態を表示





ハンズオン(2)

アプリケーションパフォーマンスとホス トパフォーマンス

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

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

[事前準備] ハンズオン(1)と同じユーザーでNew Relicにログインしてください。 [Infrastructure]をクリックします。

O New Relic ONE [™] Account: 25116	571 - Nev	vRelicUniversity-Japan ~		D Query your data	🖳 Instant Observability	🗄 Apps 🥥 Get started	Q 🕜 💀 🕫 💶 🤇
Explorer Browse data Dashboard	ls Aler	ts & Al Errors Inbox APM Browser	Infrastructure	Mobile Synthetics More -	t	🗗 Copy permalink 🗸 🧹 🕻	Since 60 minutes ago ~
Hosts Inventory Events AWS	GovCl	oud Azure GCP Third-party services	ettings Kubernetes	Navigator view Lookout view			
Saved filter sets (2)	>	🔡 Group by entityName 🗸 🕐	Chart shows 5	~ ⑦			
All hosts		System Network Processes S	torage Docker Conta	ainers			🔂 Navigator view
1 Host	>	Events (97)				_	0 24
1 Application	>						
0 Alerting 0 0							
₩ FILTER HOSTS	>	CPU % \checkmark Japan (UTC +9)		Load Average Five Minute \lor Japan (UTC +9)		Memory Free % \checkmark Japan (UTC +9)	
		100%		3		50%	
		90%	~~~~	2.5		45%	
		70%			~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	40% 35%	
		60%		2		30%	
		50%		1.5	1	25%	
		40%		1	/	20%	
		30%				15%	
		10%		0.5		596	
		0%		0		0%	
		a adam and a ada			laa aalaa		and an and an a

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

[事前準備]

- ハンズオン(1)と同じユーザーでNew Relicにログインしてください。 左上のアカウント名が「New Relic TSS」と表示されている場合は [NewRelicUniversity-Japan 2511671]を 選択してください。

O New Relic ONE TH	Account: 2511671 - Ne	wRelicUniversity-Japan ~				
Explorer Browse da	Q, Search		APM Browser	Infrastru	icture Logs M	
	Most recent				10.000	
Hosts Inventory	NewRelicUniversity-Ja	apan 2511671	d-party services	Settings	Kubernetes N	
Saved filter sets (2	NewRelicUniversity-Ja New Relic TSS	apan-Dora 3139014 1336182	ame 🗸 🕜) می	Chart shows 5 ∨	
All hosts			Processes	Storage	Docker Contai	
1 Host	>	Events (97)				
1 Application	>					
0 Alerting 0 0						
₹ FILTER HOSTS	>	CPU % \checkmark Japan (UTC +9)				
		100% 90% 80%		г ~	~~~	

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

ホスト[ip-172-31-26-144.ap-northeast-1.compute.internal]について以下の情報を調べてください。

1. ホスト情報の確認

ホストのOS種別を確認してください



2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

ホスト[ip-172-31-26-144.ap-northeast-1.compute.internal] について以下の情報を調べてください。

- 3. ホストイベントの確認
 - 過去1週間以内にミドルウェアのアップデートが行われていますか その内容を確認してください。



手順·解説

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

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

- 1. ホスト情報の確認
 - ホストのOS種別を確認してください

Hosts Inventory Events AWS (GovCloud Azure GCP Third-party services Settings Kubernetes	Hosts> HOST > からホストを選択し Navigator view L [V]をクリックしてホスト情報を展開しま
Saved filter sets (2)	Search hosts	⊕ Add hosts \$\$\$\$
All hosts	ip-172-31-26-144.ap-northeast-1.compute.internal	
1 Host	Metadata Operating System linux Linux Distribution Amazon Linux 2	Distribution: Amazon Linux 2
0 Alerting 0 0	Agent Version 1.20.7 Kernel Version 4.14.248-189.473.amzn2.x86_64 CPU Count 2	が確認できます。
	System Memory 2,056 MB)
	Applications EC-site	

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。

O New Relic ONE** Account: 2511	671 - Nev	vRelicUniversity-Japan ~	🛛 🔡 Apps ⊘ Get started	Q (?)	₽ ₽ 10+ 0 -	
Explorer Browse data Dashboar	ds Alert	ts & Al Errors Inbox APM Browser Infrastructure	🗗 Copy permalink 🗸 📢	Since	60 minutes ago	
Hosts Inventory Events AW	5 GovCl	oud Azure GCP Third-party services Settings Kub H Group by entityName v ⑦ L Chart K	>		5 minutes 30 minutes 60 minutes	TIME PICKER から24hoursを選択して過去24時
All hosts		System Network Processes Storage Doc	>	l	3 hours 6 hours	間の情報を表示させます。
1 Host 1 Application	> >	Events (97)	>	0	12 hours	
0 Alerting 0 0	>	сри % 🗸	> Memory Free % 🗸		3 days 7 days	
		Japan (UTC +9) 100% 90% 80% 50% 50%	Japan (UTC +9) 50% 45% 40% 33% 30% 25%	~	Set custom	

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。



マウスオーバーすることによって値が表示される ので、一番高い山を探します。 CPU 使用率が等しい場合は LoadAverage を参 考に負荷が高かった時間を特定します。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。



細かな確認を行うために、対象の時間帯を マウスドラッグして選択します。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。



Processタブを表示することで 全体のリソース使用率だけではなく、 プロセス毎のCPU使用率やメモリ使用率を確認 することができます。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。



Inventory>metadata/host_aliases から ホストのinstance-id を確認しておきます。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。

Explorer Browse data Dasht	boards Alerts & Al Errors Inbox	APM Browser Infrastructure Logs Mobile Sy	nthetics More - 🖉 🔁 Copy permalink -	
Hosts Inventory Events	AWS GovCloud Azure GCP	-party services Settings Kubernetes Navigator v	iew Lookout view	
Active Integrations (2)		0	0	Third-party services > MySQL dashboard
INTEGRATION NAME	M DASHBOARDS	♀ CREATE ALERT	EXPLORE DATA	を選択し、MySQLの情報を確認します。
لا کې MySQL	MySQL dashboard	Set alert Alert API Info	MySQL data	
Apache	Apache dashboard	Set alert Alert API Info	Apache data	

Available Integrations (25)

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。

Hosts Inventory Events AWS GovCloud Azure GCP Third-party services

〒 FILTER

Q Search a key to filter by	
Attributes	3
> cluster.nodeType	1
∨ Host	1
Q Search in Host	
Include Exclude	
node:i-0e4f0324c79ece5d9:3306	
> software.version	1
Labels	2

MySQLはHOST ID ではなく、nodeとして表示されるので、先ほど Inventoryで確認した instance-idでFilterします。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン

2. パフォーマンス情報の確認

過去24時間のうちで、一番CPU負荷が高かったのはいつですか そのタイミングで一番CPU消費が高かったプロセスを確認してください。 そのタイミングのMySQLコネクション数を確認してください。

50		0.1
Nov 03, Nov 03, Nov 03, Nov 03, Nov 03, O2:30 PM 03:00 PM 03:30 PM 04:00 PM	C	Nov 03, Nov 03, Nov 03, Nov 03, 02:30 PM 03:00 PM 03:30 PM 04:00 PM
• Queries		• node:i-0e4f0324c79ece5d9:3306
Active Connections by node		Max Connections by node
Active Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9)		Max Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9)
Active Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9)		Max Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9)
Active Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9) node:I-0e4f0324c79ece5d9:3306	3	Max Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9)
Active Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9) node:I-0e4f0324c79ece5d9:3306		Max Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9) 20 15 10
Active Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm,Japan (UTC 9) node;i-0e4f0324c79ece5d9:3306		Max Connections, by node Since Nov 3, 02:30 pm until Nov 3, 04:26 pm.Japan (UTC 9) 20 15 10 5

TIME PICKERの設定は維持されています。 Max Connections, by node を確認すること で、CPU負荷が高かったタイミングでのコネ クション数を確認することができます。

ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン

3. ホストイベントの確認

 過去1週間以内にミドルウェアのアップデートが行われていますか その内容を確認してください。



ハンズオン(2)アプリケーションパフォーマンスとホストパフォーマン ス

- 3. ホストイベントの確認
 - 過去1週間以内にミドルウェアのアップデートが行われていますか
 その内容を確認してください。

C TIME PICKER 7d ✓			
Events (145)			-
Q Search events			
CATEGORY	Loading More Eve	nts	
agents (4) applications (2)	Oct 30th 9:42:04 AM	Package modified: libcrypt ip-17	~
kernel (15) metadata (3)	Oct 30th 9:42:04 AM	Package modified: elfutils-default-y	
packages (38) services (75)	Oct 30th 9:42:04 AM	Package modified: elfutils-libelf i	~
_) sessions (8)	Oct 30th 9:42:04 AM	Package modified: glibc-all-langpac	~
	Oct 30th	Package installed kernel 2 in 17	

CATEGORY>packages を選択して絞り込む事で、パッケージに関するイベントのみを表示できます。 [V]で詳細を確認することで、変更前、変更後のパッケージバージョンを確認することができます。



ダッシュボード・アラート

- 収集したデータの活用-

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





ダッシュボードを活用する場面

• 複数アプリケーションやアプリケーションとインフラのメトリックの相関関係など、様々なソースからの

データを一つの画面で把握したい場合 (データの選択と集約)

- チームで定めた KPIに対する実測値を把握したい場合 (データの加工)
- ・ 集めたデータを目で見てわかりやすい形式で表示したい場合 (データのビジュアライズ)
- New Relicアカウントを横断的に表示する場合

①データの選択と集約

デフォルトの画面で表示されているグラフをダッシュボードに追加することで、見たい情報のみを簡単に集約することが可能



"Add to a dashboard"を押すだけ!

GC Execution I Java Response time Web external

②データの加工

基本編: チャートビルダーを使い、自分で見たいデータを選択する

view a chart with	Enter an event type, attribute, keyword, or select a shortcut	(元)	Narrow results to finder in a limit of which a
next hu Colori	SHORTCUTS	Respo	onse time
over by Selectly	APM Transaction time breakdown	> The res	sponse time of web transactions or non-web
	APM Response time	> transac	ctions in your app. Segments in the chart vary ding on which agent you are using. Some
	APM Error rate	charts time th	may have an independent line for response nat represents the relationship between
	APM Throughput	> respon	ise time and total time.
	Browser SPA load times percentile		
	Browser SPA load times histogram	>	
	Browser SPA throughput	>	
	Browser Browser vs. Synthetics performance		
	Host CPU usage	5	

見たいデータをドロップダウンで選 択

グラフ例

der 💿



チャートビルダーの使い方 チャートに表示したいデータを選択 データの絞り込み(オプション) この例では、TransactionというEventから 例.特定のアプリのデータのみを表示する durationという属性を選択し、平均値を出す Account Demotron V2 Data Type: Events Metrics = Narrow results to appName = 'Plan ...rvice' × Ent View a chart with Transaction : duration : average X 1 Last 30 mins × Clear Facet by name × Limit 10 Compare no comparison 1 比較のための過去の データのグループ分け(オプション) 表示させるデータ期間 データを重ねて表示 例トランザクション名ごとにデータを分けて表示 (オプション)

②データの加工(続き)

応用編: 自分でクエリを書いて、見たい情報をチャートに表現

- NRQLという独自の言語を使用
- チャートビルダーよりも柔軟なデータ加工が可能

Account Demotron V2 V		Switching to basic mode will erase any	changes you make here.	🗄 Basic
SELECT average(duration) FROM Transaction	WHERE appName = 'Plan Service' TIMESERIES FACET `name` LIMIT 10 SINCE	1800 seconds ago EXTRAPO	OLATE	
My recent queries $~\sim~$			🛞 Clear	Run
S	ELECT <u>function(attribute)</u> [AS 'label'][,]			

 NRQL構文
 FROM event

 [WHERE attribute [comparison] [AND|OR ...]][AS 'label'][, ...]

 [FACET attribute | function(attribute)] [LIMIT number]

 [SINCE time] [UNTIL time]

 [WITH TIMEZONE timezone]

 [COMPARE WITH time]

 [TIMESERIES time]

参考: https://docs.newrelic.com/docs/query-data/nrgl-new-relic-query-language/getting-started/introduction-nrgl

Event名について

参考:

データ(Event)は種類に応じたEvent名が割り振られています 一例(他にも多数あります):

データソース	Event名	データの種類
АРМ	Transaction	トランザクションの所要時間を記録
	TransactionError	アプリで発生したエラーを記録
Infrastructure(エ ージェント)	SystemSample	OS全体のメトリックを記録
	StorageSample	ファイルシステムごとのメトリックを記録
Infrastructure(ク ラウド連携)	ComputeSample	計算リソースを提供するサービスのメトリックを 記録(AWS EC2等)
	DatastoreSample	ストレージキャパシティを提供するサービスのメ トリックを記録(AWS S3等)

https://docs.newrelic.co.jp/docs/insights/insights-data-sources/default-data/insights-default-data-other-new-relic-products

③データのビジュアライズ

加工したデータを様々なチャートタイプで表示

Since 30 minutes ago	Share 🗡	CHART NAME Enter a chart name
WebTransaction/Action/App\Http\Controllers\GetPlansController@getPlans	0.334	Chart type Bar
WebTransaction/Action/App\Http\Controllers\GetPlansController@getPlan	0.134	JSON Line
WebTransaction/Action/App\Http\Controllers\TronController@statusMessage	0.0639	Area Bar
WebTransaction/Action/App\Exceptions\Handler@render	0.0586	Bullet Billboard
		Table Pie

さらに高度なビジュアライズも

Reactを使ってリッチなダッシュボードを自分でカスタマイズすることもできま す

一部のダッシュボードはオープンソースとして公開しており、自由に利用できます

https://developer.newrelic.com/open-source/nerdpacks



アラート機能

Eventデータを使ってしきい値(動的/静的)を設定し、アラートを発報することが可能

2. Select entities	1 er
3. Define thresholds	TomcatApp \vee
When target application	
Response time (web) \lor has an average above \lor	2.5 ms
	2 ms
○ 0.001 sec at least once in ∨ 5 minutes	15 ms
Add a warping throshold	Transfer and the second s
Condition name	0.5 ms
Response time (web) (High)	0
	Oct 03, Oct 03
Add runbook URL	Critical threshold Critical violation

アラートの通知と確認

メールやSlack, モバイルアプリ等で アラートを受信 (下はSlackの例)

Incident #83551393 opened

Target

SystemSample query

Condition

NRQLテスト

Policy Tomtest

Threshold

SystemSample query result is > 3

(8 kB) -



アラートの詳細をNew Relic上で確認

Plan Service violated Check response time for web portal



Web response time > 600 milliseconds for at least 10 minutes THRESHOLD



ハンズオン(3)

アラート設定とアラートの確認

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

ハンズオン(3)ダッシュボード

[事前準備]

ハンズオン(1)と同じユーザーでNew Relicにログインします。 画面左上の[Dashboards] をクリックします


ハンズオン(3)ダッシュボード

Dashboardsメニューを選択し、右上の "Create a dashboard"を押下して新しいダッシュボードを 作成してください。

ダッシュボード名にはご自身のお名前を入れてください (日本語可)。

アカウントの選択メニューでは "NewRelicUniversity-Japan"を選択します。

2. 既存チャートの追加

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。

ハンズオン(3)ダッシュボード

- 3. 新規チャートの追加
 - 自分のダッシュボードを表示し、Create ChartからChartを作成して追加します。

4. アラートの確認

• Alertsを開き過去のインシデントを確認します。

ハンズオン(3)アラート

5. 通知設定の作成

[Alert policys]を開き[By policy]の設定で新しいAlertPolicyを作成します。

Policy名はダッシュボードと同じように自分の物とわかる名前を付けてください(日本語可)

新たに作成したAlertPolicyにNotification channelを作成します。

メールもしくはSlackを利用し自分用の通知を作成してください。

6. ベースラインアラート

 [Alert Policys] で自分で作成した Policyを選択しを選択し、[Add condition] からアラート条件の 設定画面を開いて、[Application metric baseline] を選択して、ベースラインアラートの感度設定 を確認しましょう。



手順·解説

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

ハンズオン(3)ダッシュボード

New Relic Oneの"Dashboards"メニューを選択し、右上の "Create a dashboard"を押下して新しい ダッシュボードを作成してください。

O New Re	Hic ONE" Account: 2511671 - NewRelicUniversity-Japan		🖸 Query your data 년날 in	stant Observability 🔠 Apps ⊘ Get starte	4 Q @ 🛱 🗗 🚺 🔍 -
Explorer	Browse data Dathboards Alerts & Al Errors Inbox	APM Browser Infrastructure Logs Mot	olle Synthetics More - 🏼 🖉	D ^A Copy permalink → [}- Import o	lashboard + Create a dashboard
[▼]	Filter by name, type, tags (e.g. entityType = Host)				
3	Name C	Account 0	Created By	Last Edited 0	Created On C
*	NRU#ホワイトホードサンプル	NewRelicUniversity-Japan	aito@newrelic.com	May 20, 2021	0ct 29, 2020
*	NRU#レポートサンプル	NewRelicUniversity-Japan	japan-handson@newrelic.com	Mar 17, 2021	0at 31, 2019
Start quer	rying your data [Ctrl + Shift + O]				0

ハンズオン(3)ダッシュボード

[Browse pre-built dashboards] ではプリセットの Dashboardを展開することができます。 今回は[Create a new dashboard] でDashboardを自作します。

Create a dashboard

LILI Browse pre-built dashboards Install a quickstart with dashboards made for what you're monitoring.

>

Create a new dashboard Build a dashboard from scratch and see what's happening with your data.

ハンズオン(3)ダッシュボード

ダッシュボード名にはご自身のお名前など、他の参加者と取り違え無い文言を入れてください (日本 語可)。

アカウントの選択メニューでは "2511671 NewRelicUniversity-Japan"を選択します。 [Create]をクリックすると空のグッシューザービジル・ボッセュナ

Create a dashboard

Dashboard name



Select your account

Account: 2511671 - NewRelicUniversity-Japan ~



ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。



APM をクリックします。

ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したMySQLの画面から それぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。



PHP • MySQL • Web external • Response time

ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。

Web transactions time	
Select an existing dashboard	
We excluded dashboards you don't	t have permission to edit.
Q NRU	
Dashboard Name	Account
	NewReli
NRU#ホワイトホートサンフル	

チャートに名前を付け、自分のダッシュ ボードを検索して[Copy to dashboard] をクリックします。

ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。

Average response time, by	tier (ms)	
Copy to existing dashboard	Copy to a new dashboard	
Q Search dashboards		
Last added to		
✓ 伊藤NRU#2		
Recently viewed		
NRU02		
All dashboards		2
伊藤NRU#2		
NRU02		

ダッシュボードを選択して [Copy]をクリッ クします。

同様に、[NewRelicUniversity-Japan]の MySQLからもChartを追加します。

ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。



InfrastructureのHostから追加する場合 は「・・・」から「Copy to query builder」を クリックします。

ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。

Data explorer Query builder		[Add to Dashboard]たクレ
Account: 2511671 - NewRelicUniversity-Japan ~	🔘 PromQL-style 🖃	
SELECT average(cpuPercent) FROM SystemSample TIMESER 100 SINCE 7 days ago	S FACET entityId WHERE (entityId in ('4707049641037416929')) LIMIT	
Add another query Your recent queries ~ Create alert	Clear	Run
since 7 days ago × Q Update query with this time	··· Chart name Enter a chart name	
	Chart type Line More visualizations in I/C	° 0 ∰
40 35 30	Other groups ①	
25 20 15	Visible 🕥	
10	y Axis	
0 Oct 28, Oct 29, Oct 30, Oct 31, Nov 09:00 AM 09:00 AM 09:00 AM 09:00 AM 09:00 AM 09:00 AM	4707049641037416929 Range Fit range to data Nov 01, Iov 02, Nov 03, I 11:30 PM, 00 AM 0	
4707049641037416929	WIT 0 Max Auto	
178 events inspected in 108 ms (0.0 MEPS)	Legend	

85

ハンズオン(3)ダッシュボード

ハンズオン(1)で確認したEC-siteのAPMの画面、およびハンズオン(2)で確認したInfrastructureの画面からそれぞれ1つずつ任意のチャートを、自分のダッシュボードに追加してください。



Third-party-Services(MySQL)から追加 する場合は [・・・]から [Add to Insights dashboard] を選択します。

ハンズオン(3)ダッシュボード

- 3. 新規チャートの追加
 - 自分のダッシュボードを表示し、Create ChartからChartを作成して追加します。

Dashboards / ☆ 伊藤NRU#2 ▼ ♥ ♀ Search for any another or value.	± 1	D D Greate a c
👻 🛩 Search far any attribute or value.		Create a c
Average response time, by tier (ms) Since 20 hours ago until 19 hours ago		
20 ms		
15 ms		
10 ms		
Sms		
0 ms		
Ort \$0. Ort \$0. <t< td=""><td></td><td></td></t<>		
● Java ● Response time ● MySQL		
Max Connections, by node		
25		
20		
15		
10		
5		

ダッシュボードを開き[+]アイコンをクリッ クします。 [Add a chart]を選択してチャートを作成 します。

Add to your dashboard



デチャネテトビルダーの使い方 チャートに表示したいデータを選択 データの絞り込み(オプション) この例では、TransactionというEventから 例、特定のアプリのデータのみを表示する durationという属性を選択し、平均値を出す Account Demotron V2 Data Type: Events Metrics = Narrow results to appName = 'Plan ...rvice' × Ent View a chart with Transaction : duration : average X 1 Last 30 mins × Clear Facet by name × Limit 10 Compare no comparison 1 比較のための過去の データのグループ分け(オプション) 表示させるデータ期間 データを重ねて表示 例トランザクション名ごとにデータを分けて表示 (オプション)

ハンズオン(3)アラート

- 4. アラートの確認
 - Alertsを開き過去のインシデントを確認します。

Alert & AI を開き、Incidentsを開きます。

O New Relic ONE™	Account: 2511671 - NewR	elicUniversity-Japan 🐱	
Explorer Browse d	ata Dashboards Alerts	& AI Errors Inbox AP	PM Browser Infrastruct
Overview		Open incidents	All incidents
Issues & activity			
ALERTS			
Incidents)		
Events			
Policies			
Notification channe	els	г	here are n

89

ハンズオン(3)アラート

- 4. アラートの確認
 - Alertsを開き過去のインシデントを確認します。

O New Relic ONE ^{**}	s Dashboards	Entity explorer	APM	Browser	Synthetics	Mobile	Infrastructure	Logs	AI	[A
Applied Intelligence	Account: 251167	I - NewRelicUnive	rsity-Jap	oan 🗸						<u>ام</u>

[All Incidents] を開きインシデン トを確認します。

Overview	Open incid	ents All incidents	
ALERTS			
Incidents	(Search ir	ncidents	
Events	ID 0	Name	Open
Policies	152396483	EC-site triggered Web transaction throughput (Baseline) in アラートポリシー	
Notification channels			
Muting rules	152384781	EC-site triggered Web transaction throughput (Baseline) in アラートポリシー	

ハンズオン(3)アラート

4. アラートの確認

• Alertsを開き過去のインシデントを確認します。



New Relic Alerts ではインシデントを検 知した際に 自動的に関連がありそうな情報を表示し ます。 ※出ない場合もあります。

ハンズオン(3)アラート

5. 通知設定の作成

Muting rules

• Pollicies を開き[New alert policy]をクリックします。

● New Relic ONE [™] Account: 25	511671 - NewRelicUniversity-Japan 🐱				<u>}</u> (Query your data 🛛 📗	14 Instant Observability	🗄 Apps ⊘ Gets	tarted Q 🕜	🗟 🗗 🚺 🔘
Explorer Browse data Dashbo	oards Alerts & Al Errors Inbox A	PM Browser	Infrastructure	Logs Mobi	le Synthetics	More 🗸 🖉				Copy permalink 🔍
Overview Issues & activity	Search policies							+ New alert policy	Browse pt	e-built alerts
ALERTS	Policy				^	Conditions 🤇	Channels 🗘	Open incidents	Last \Diamond incident	
Incidents	アラートポリシー					2	2	0	Apr 20, 2:01 am	ŵ
Policies	インシデントインテリジェンス					1	0	0	Nov 3, 5:53 am	
Notification channels	ダッシュボードハンズオン用	アラートポリシー				1	0	0	2:21 am	1

ハンズオン(3)アラート

通知設定の作成 5.

Applied Intelligence Account: 2511671 - NewRelicUniversity-Japan ~

ALERT POLICY NAME にわかり易い名前を入力し、[By pollicy]を選択します。 ٠

Overview	Create alert r		他の参加者と名前がぶつからないよう
ALERTS	create alere p	oncy	に日月のホリノーとりがる日前を入れ
Incidents			てください。
Events	ALERT POLICY NAME	Give your policy a concise and descriptive name.	
Policies		通知ポリシー	
Notification channels			
Muting rules			
PROACTIVE DETECTION	INCIDENT PREFERENCE	Specify how incidents should be created when conditions in this alert policy are sent only when an incident opens. Is acknowledged, and closes.)	violated. (I
Anomaly overview			
Configuration		By policy All violations within this policy will be grouped into a single incident; only one open incident at a time for this alert policy	
		By condition All violations within a condition in this policy will be grouped into a single incident; only one incident at a time per alert condition	
		By condition and entity An incident will open every time an entity violates a condition in this policy	©2008–20 New Relic, Inc. All rights reserved

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

93

ハンズオン(3)アラート

- 5. 通知設定の作成
 - [Create alert policy]ボタンをクリックして、AlertPolicyを作成します。

NOTIFICATION CHANNELS

To get notified when incidents occur, you need to associate this policy with a notification channel. You can select a channel after you create the policy.

Cancel Create alert policy

ハンズオン(3)アラート

- 5. 通知設定の作成
 - [Notification channels] を選択し[New notification channel]をクリックします。

Applied Intelligence	ccount: 2511671 - NewR	lelicUniversity-Japan 🗸		
Overview	Search	channels		+ New notification channel
ALERTS				
Incidents	Туре	^ Channel name	Policy subscriptions	
Events	SLACK	NRU-Slack	1	面
Policies	USER	Akihiro Ito <aito@newrelic.com></aito@newrelic.com>	0	
Notification channels	USER	Akihiro Ito <qryuu.maoto@gmail.com></qryuu.maoto@gmail.com>	0	
Muting rules	USER	NRU Japan <japan-handson@newrelic.com></japan-handson@newrelic.com>	1	

ハンズオン(3)アラート

- 5. 通知設定の作成
 - [Channel Type] で [Email]を選択し自分で受信確認が出来るメールアドレスを設定してください。

lect a	charmer	type			
<u> </u>	Email		~		
nail					
apan-	handson+	-0206@ne	wrelic.cor	n	

ハンズオン(3)アラート

5. 通知設定の作成

• [policies]で先ほど自分で作成したポリシーを選択します。

O New Relic ONE™ ☐ ☐ App	s Dashboards Entity explorer	APM Browser	Synthetics M	obile Infrastructure	e Logs	Alerts & Al	More ~	Q	Search	h	Q
Applied Intelligence	Account: 2511671 - NewRelicUnive	rsity-Japan v									
Overview	Q sourch policies										
ALERTS	Search policies										
Incidents	Policy				~	Conditio	ns o	Channels		0 C	p
Events	アラートポリシー	٦					2			2	
Policies							2				
Notification channels											

Muting rules

ハンズオン(3)アラート

5. 通知設定の作成

[Notification channels]タブを選択します。
 [Add notification channels]ボタンをクリックします。

ポリシー作成 3 Incident preference: By policy Delete this policy id: 1010195 Last modified 4:37 pm by NRU Japan (i) Add a notification channel to receive alerts 0 Alert conditions 0 Notification channels

Add notification channels to this policy

Your channels tell us who to notify when incidents are opened, acknowledged, and closed.



ハンズオン(3)アラート

- 5. 通知設定の作成
 - [Notification channel]タブを選択します。
 - ・ 先ほど登録した自分のメールアドレスに✓を付けて[Update policy]をクリックします。

1 channel selected		
Browse all		
Email	2 channels selected	
🛅 HipChat	Browse all > Email	
PagerDuty	Search channels	
🗁 VictorOps	Select: Page None	Filter: All Selected (1)
	Japan-handson+0206@newrelic.com	
	1 - 1 of 1	
		Cancel Update policy

ハンズオン(3)アラート(オプション)

5. 通知設定の作成(オプション)

• Slackの管理権限をお持ちの場合、New RelicアプリをSlackに追加していただく事で、Slack通知が利用できます。



ハンズオン(3)アラート(オプション)

5. 通知設定の作成(オプション)

Slackの管理権限をお持ちの場合、New RelicアプリをSlackに追加していただく事で、Slack通知が利用できます。

New Relic Alerts

リアルタイム アプリケーション パフォーマンス管理。

New Relic はウェブや携帯端末のアプリケーションパフォーマンスモニターサービスで、チームはアプリの正常性および可用性を監視 することができます。

このインテグレーションは、New Relic 内でアラートがトリガーされた時に Slack に通知を投稿します。ウェブ、トランザクション、サーバー、そして携帯アラートをそれそれ違うチャンネルに投稿したい時には、別のインテグレーションの設定が必要です。

このインテグレーションを追加できるのは、チームの New Relic アカウントの管理者のみです。

まず New Relic アラートを投稿するチャンネルを選択します。	# new-relic-通知	~
	⊘ 新しいチ	ヤンネルが作成されました
	New Relic インテグレーションの追加	

ハンズオン(3)アラート(オプション)

5. 通知設定の作成(オプション)

- Slackの管理権限をお持ちの場合、New RelicアプリをSlackに追加していただく事で、Slack通知が利用できます。
- アプリで生成した[Webhook URL]をchannel 設定に入力してください。

Select a channel type	Webhook URL		
🗱 Slack	このインテグレーションを設定する場 今には、このUPLを Now Polic Alorts	https://hooks.slack.com/services/	prost in the Brancip Distant
Channel name	日には、この OKE を New Keil Alerts にペーストします。 セットアップの手順を表示		Copy URL • 再生成する
NRU-Slack			
URL			
https://hooks.slack.com	n/services/		
Team channe <mark>l</mark> (optional))		
#new-relic-通知			

Create channel

ハンズオン(3)アラート

- 6. ベースラインアラート
 - [Alert Policys] で先ほど作成したAlertPolicyを選択し[Alert conditions]タブを選択し Create a condition ボタンをクリックします。

通知設定		င့်္ပို Incident preference: By polic
0 Alert conditions	1 Notification channel	Last



This policy doesn't have any conditions

Alert conditions are the criteria for creating incidents.

Notifications are sent when incidents are created.



ハンズオン(3)アラート

- 6. ベースラインアラート
 - [Application metric baseline] を選択して、ベースラインアラートの感度設定を確認しましょう。

New condition				⊗ Cancel	
1. Categorize					
NRQL APM Brow	wser Mobile Synt	hetics Infrastructure QL conditions. Learn more			APM>Application metric baseline を 選択し Nextをクリックします。
Application metric	JVM health metri	c Key transaction metri	c Web transactions percentiles		
External service	Application metric ba	seline			
			3	Next, select entities	

ハンズオン(3)アラート

- 6. ベースラインアラート
 - [Application metric baseline] を選択して、ベースラインアラートの感度設定を確認しましょう。

2. 1 entity selected		
Search applications		
Select: All (2) None	View: All (2) Selected (1) Unselected (1)	アプリケーションを選択して Nextをクリッ
EC-site		クします。
PHP Application		
	K Back to Name and Categorize Next, define thresholds	

ハンズオン(3)アラート

- 6. ベースラインアラート
 - [Application metric baseline] を選択して、ベースラインアラートの感度設定を確認しましょう。

New condition						⊗ Cancel
1. Categorize					APM - App	lication metric baseline
2. Select entities						1 entity
3. Define thresholds Baseline Direction: New Uppe	er and lower \simeq		PHP Application ~			🖯 Lest 2 days 🗟
When any target application	Search metric names	2	0.004 0.003			
Datastore/statement/MySQ has an average value ~	Key metrics Web transaction time Web transaction throughput Transaction database time Web transaction external service time	ninutes	0.002 0.001 0 -0.001	Muhh.	stat	that
more violations	Non-web transaction time Web transaction database time Non-web transaction database time	ewer violations	Oct 30, 12:00 AM	Oct 30, 12:00 PM Average search metric	Oct 31, 12:00 AM	Oct 31, 12:00 PM
Condition Name	Custom metric Search metric names		To see values not visible	e in larger time window	is, click and drag to zoo	om the chart

評価対象とするメトリクスを選択します。

ハンズオン(3)アラート

- 6. ベースラインアラート
 - [Application metric baseline] を選択して、ベースラインアラートの感度設定を確認しましょう。

New condition				(S) Cancel
1. Categorize			APM	Application metric baseline
2. Select entities				1 entity
3. Define thresholds Baseline Direction: New Upper and lower > When any target application Web transaction throughput > Control count deviates from the baseline for at least > s minutes	PHP Application C DHP Applications 2 k 1.5 k 1 k 500 0		LI LA ARAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA	Carraya S
more violations fewer violations	-500 -1 k 1.5 k			No.
Condition Name	Oct 30, 12:00 AM	Oct 30, 12:00 PM	Oct 31, 12:00 AM	Oct 31, 12:00 PM
Web transaction throughput (Baseline) G) Add runbook URL	 Web transaction throughp To see values not visible it 	out • Average web tran in larger time window	nsection throughput is, click and drag to zoo	om the chart
			ς Back to Select entitie	es Create condition

スライダを左右に動かす事でベースラインの広さが変 化します。 実際のグラフを見ながらどの程度の外れた場合に通知 が必要なのかを検討してください。

ハンズオン(3)アラート

- 6. ベースラインアラート
 - ベースラインアラートを利用することで、あらかじめ閾値を決められない未知の障害兆候に対して、普段の挙動とは異なる変動を検知し、
 潜在的な障害の可能性に対して未然に対処することが可能となります。
 - これまでの固定された閾値による監視とベースライン監視を組み合わせて サービスが止まる前に対処が出来る運用が可能となります。
New Relic AI (Applied Intelligence)







Proactive Detection



110

Proactive Detection Dashboard





Proactive Detection



ぜひNew Relicをお試しください!

弊社サイトからサインアップ <u>https://newrelic.com/signup</u>

New Relic.

Free access to all of New Relic. Forever.

With your new account, you get:

Perpetually free access 100 GB/month of free data ingest. 1 free full access user. Unlimited free basic users.

One data platform for all metrics, logs, events, and traces Petabyte scale. Millisecond speed. Pennies per gigabyte (beyond free tier).

Easily visualize, analyze & troubleshoot your entire stack APM, Infrastructure Monitoring, Digital Experience Monitoring, Applied Intelligence, and more.

NAME

e.g. Katherine Johnson

EMAIL

name@company.com

By signing up, you're agreeing to Terms of Service and Services Privacy Notice.

Sign up

Have an account? Log in

New Relic 実践入門

希望者に無償提供中(11月15日まで)

ついに発売された New Relic の全てを理解できる 330 ページにわたる技術書籍。オブザーバビリティの基本 から New Relic One の基本機能、さらには 16のオブ ザーバビリティ実装パターンまで含めた、初心者から応 用を理解したい上級者まで対象にした New Relic の パーフェクトガイドブック。

無償提供希望はこちらのGoogle Form から

https://forms.gle/jqiYmWRYt8Hf1nHk8

アプリケーション開発者 フロントエンド開発者 モバイルアプリ開発者 インフラ管理者 プロジェクトマネージャー プロダクトマネージャー **New Relic** 実践入門 監視からオブザーバビリティへの変革 松本 大樹、佐々木 千枝、田中 孝佳、伊藤 覚宏、清水 毅、 齊藤恒太、瀬戸島 敏宏、小口 拓、東 卓弥、会澤 康二 ∞ すべてを観測し、開発の高度化と

信頼性を高める運用を実現する

 Azure
 GCP
 モノドイル
 Serverless
 Kubernetes

 ログ昭和
 高州版アラート
 AlOps
 SRE
 DevOps
 OSS 通用

 C
 Go
 Java
 .NET
 Node,js
 PHP
 Python
 Ruby

SE





Thank You

aito@newrelic.com @qryuu

