servicenow

Performance Analytics Advanced

participant guide

ServiceNow provides this document and the information therein “as is” and ServiceNow assumes no responsibility for any inaccuracies. ServiceNow hereby disclaims all warranties, whether written or oral, express or implied by law or otherwise, including without limitation, any warranties of merchantability, accuracy, title, non-infringement or fitness for any particular purpose.

In no event will ServiceNow be liable for lost profits (whether direct or indirect), for incidental, consequential, punitive, special or exemplary damages (including damage to business, reputation or goodwill), or indirect damages of any type however caused even if ServiceNow has been advised of such damages in advance or if such damages were foreseeable.

TRADEMARKS

ServiceNow and the ServiceNow logo are registered trademarks of ServiceNow, Inc. in the United States and certain other jurisdictions. ServiceNow also uses numerous other trademarks to identify its goods and services worldwide. All other marks used herein are the trademarks of their respective owners and no ownership in such marks is claimed by ServiceNow.

ServiceNow Performance Analytics Advanced

Table of Contents

Module 1: Adoption Journey 5

Lab 1.1: Analytics for Custom Applications 17

Module 2: Bucket Groups and Scripts 33

Lab 2.1: Bucket Groups 44
Lab 2.2: Scripted Breakdown 51
Lab 2.3: Breakdown Relations...63

Module 3: Breakdown Rollup and Security 69

Lab 3.1: Breakdown Rollup 78
Lab 3.2: Breakdown Security...92

Module 4: Report Building Techniques 99

Lab 4.1: Database View...107
Lab 4.2: Multiple Data Sets and Interactivity 118
Lab 4.3: Data Sources...133
Lab 4.4: Report Ranges 135

Module 5: Responsive Dashboards...143

Lab 5.1: Responsive Dashboards..163
Lab 5.2: Interactive Analysis...171

Module 6: Spotlight..177

Lab 6.1: Configure Spotlight for Incident Management..............195

Module 7: Analytics on Platform Metrics.................................205

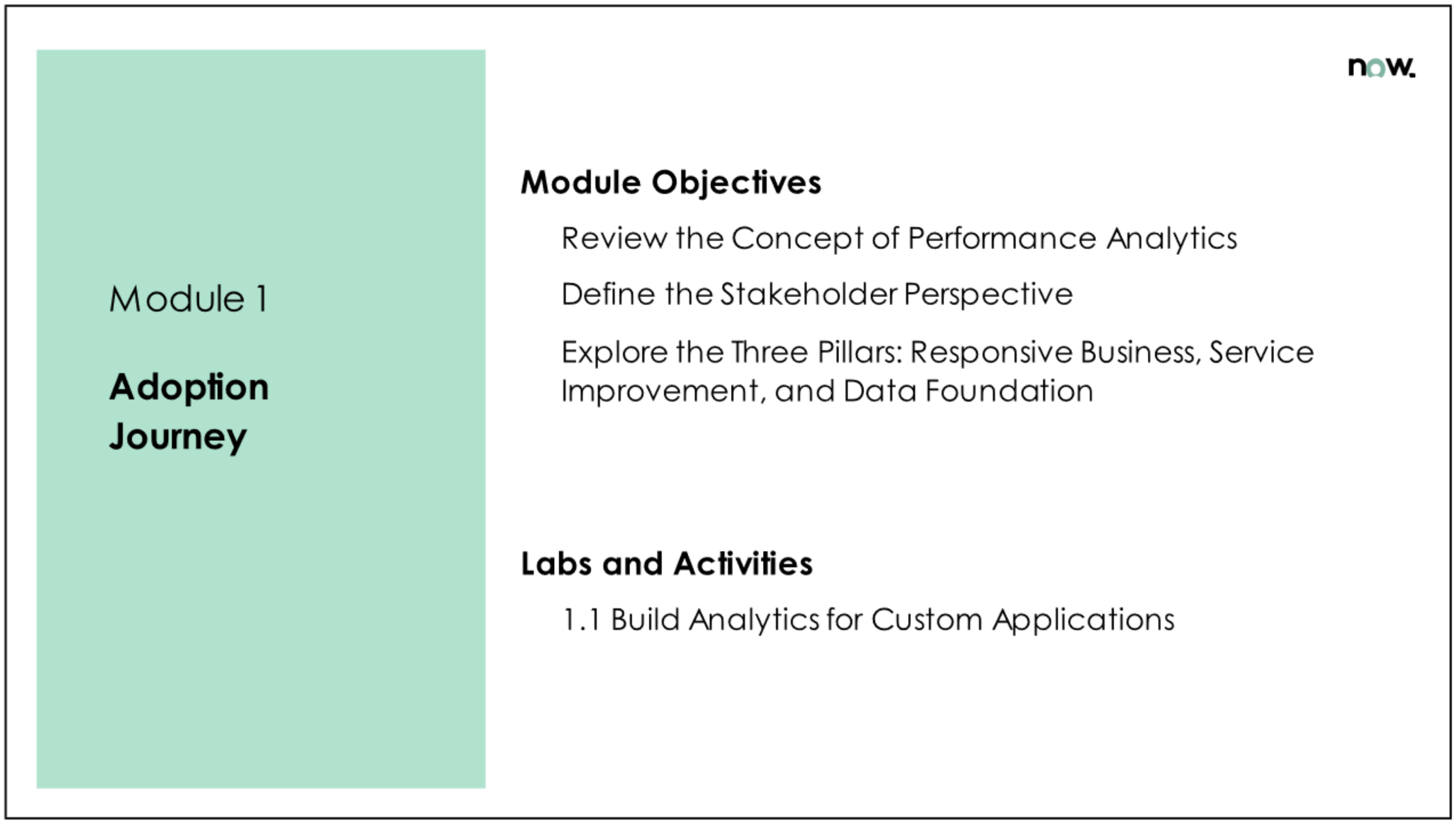
Lab 7.1: Metric Reporting and Trending 224

Module 8: Advanced Formula Indicators 231

Lab 8.1: Building Index Indicators...248

Module 9: Text Analytics 255

Lab 9.1: Configuring Text Analytics..268



Module 1

Adoption Journey

Module Objectives

Review the Concept of Performance Analytics

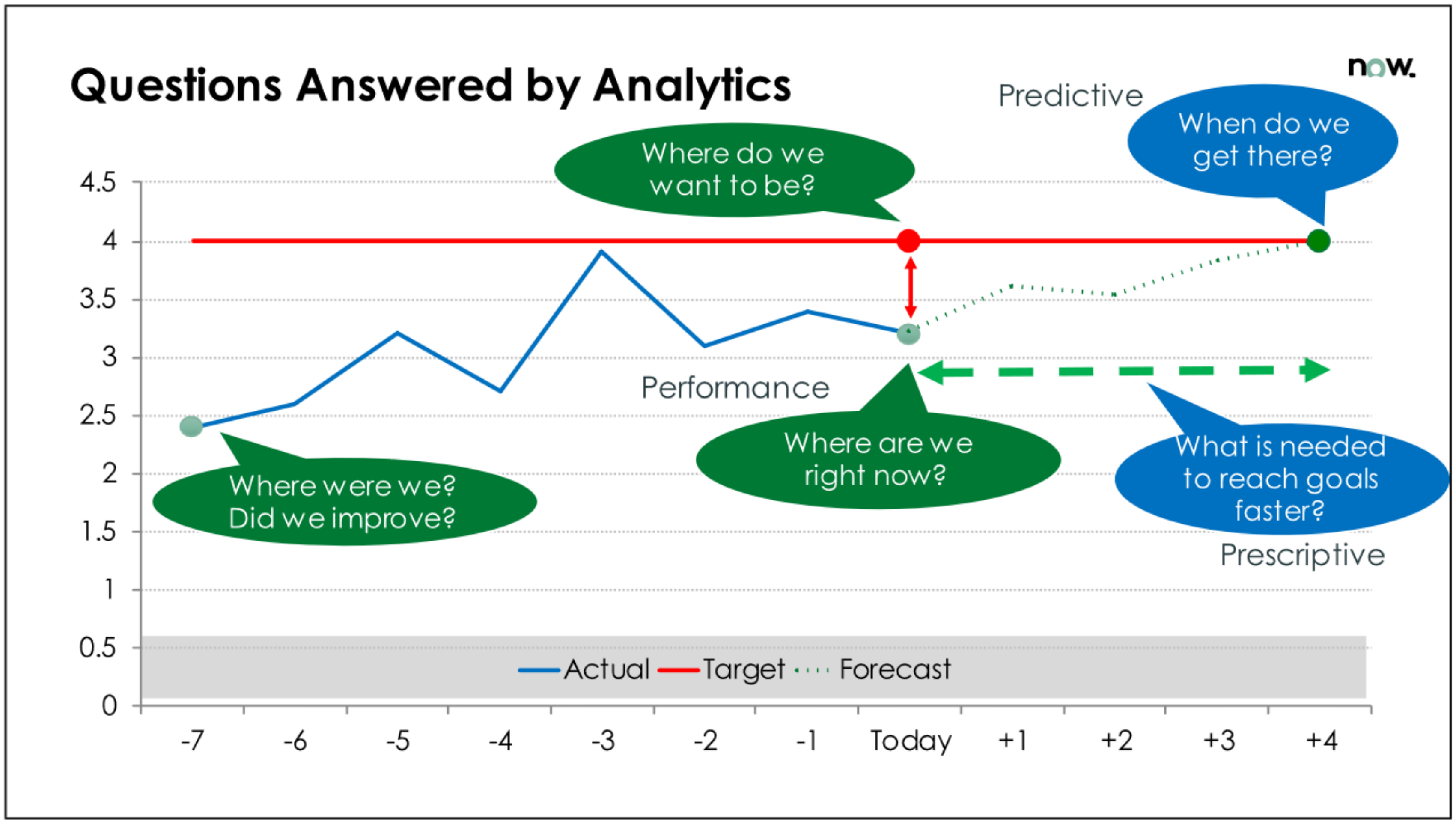
Define the Stakeholder Perspective

Explore the Three Pillars: Responsive Business, Service Improvement, and Data Foundation

Labs and Activities

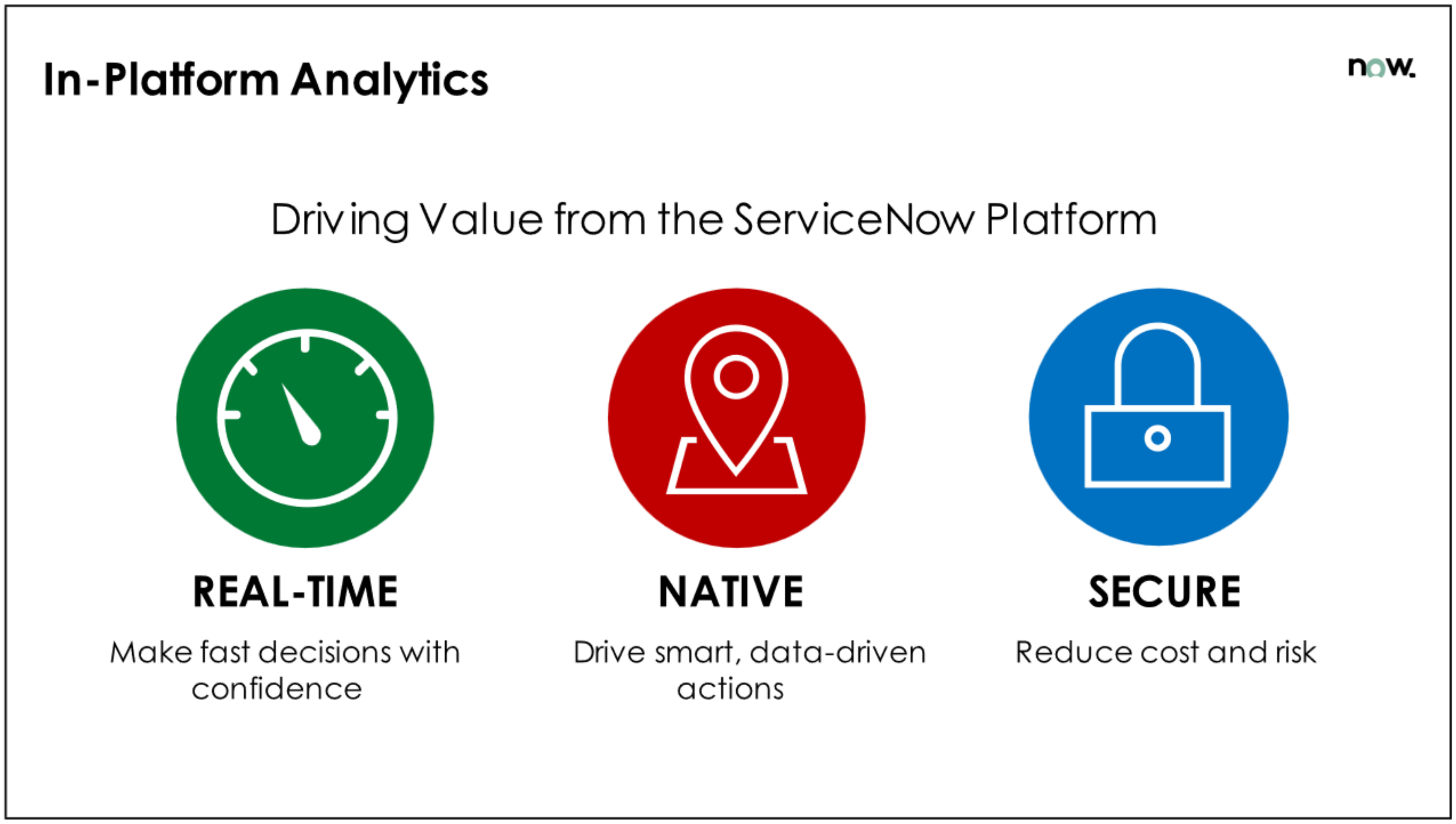
1.1 Build Analytics for Custom Applications

In this introductory module, the discussion focuses on illustrating how Performance Analytics drives value by presenting information and data that impact your business in a transformative way. The Adoption Journey or Customer Adoption Journey is about understanding how to use business intelligence proactively and in a way that drives the business goals and delivers value.



ServiceNow Performance Analytics presents actual business intelligence data, targets, and predictive statistics in order to answers these questions regarding key business processes:

- What is the past performance trend, where did we start from, and ultimately, have we improved?
- What is the performance right now? Are we where we want to be and how do we compare to the target? Have we achieved the target or is there a target Gap?
- Is there a correlation to other KPIs? What other trends and factors impact our current performance? What specific KPIs do we need to focus on in order to effect faster change?
- What is the forecasted trend? And how soon can we reach out target?



In-Platform Analytics

Driving Value from the ServiceNow Platform

REAL-TIME

Make fast decisions with confidence

NATIVE

Drive smart, data-driven actions

SECURE

Reduce cost and risk

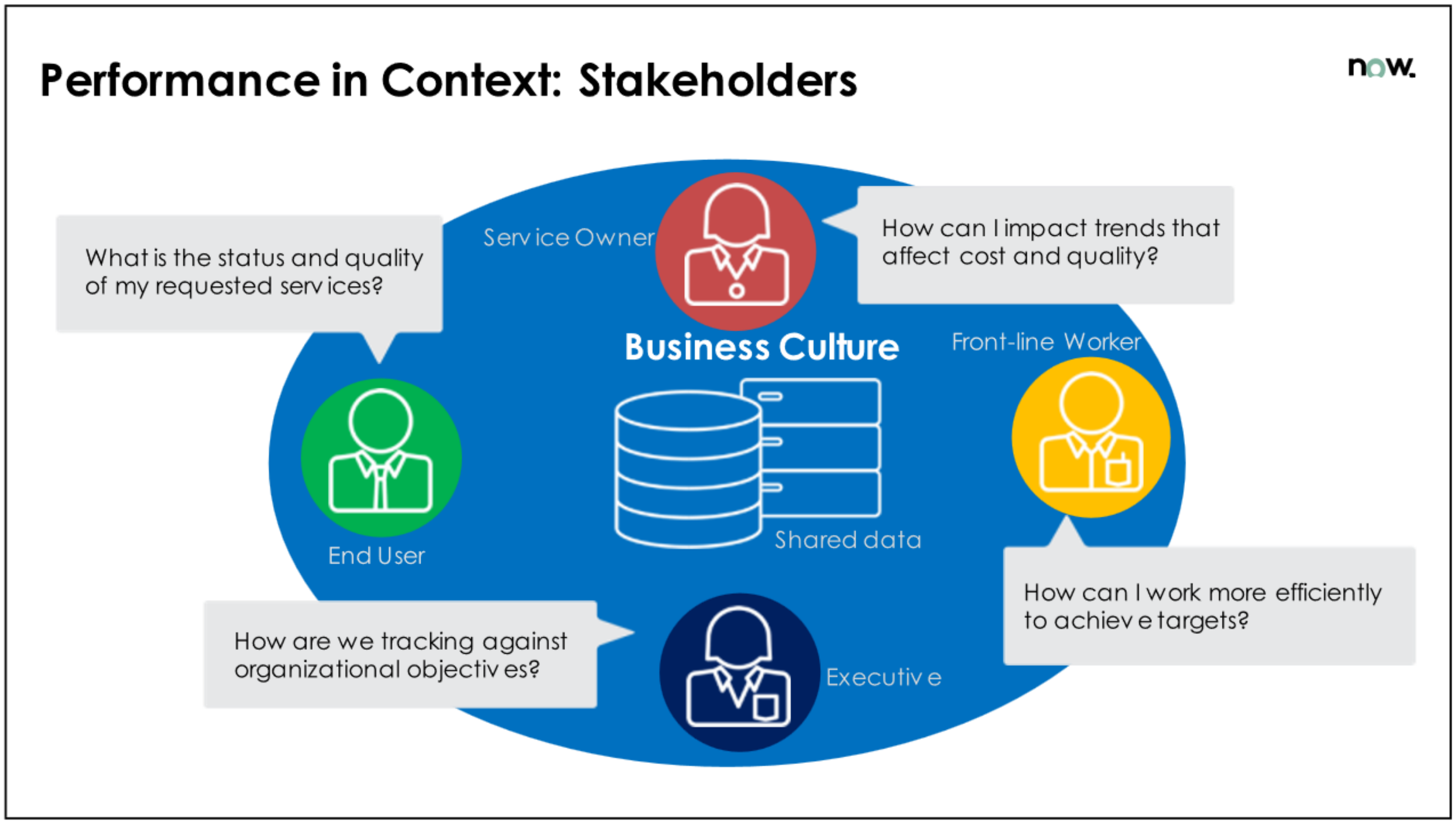
Performance Analytics focuses on making Service Management manageable by providing real-time insight into performance and helping to identify what actions are required to improve performance and quickly identify and influence trends.

REAL TIME: Because your data stays in-platform at all times, it retains its integrity. Your scorecards and dashboards show to-the-minute information, empowering you to adapt at the speed of business and make fast decisions with confidence.

NATIVE:

- Out of the box content is precisely tuned to and optimized for ServiceNow workflows
- Includes best-practice metrics developed in tandem with application teams
- Direct access to live data, both at score and at record level
- Existing ACLs and Roles apply to dashboards
- Mobile: relevant PA information automatically appears in ServiceNow's mobile app

SECURE: No data is duplicated or leaves the secure ServiceNow enterprise cloud. Data is protected at all times by the same access control lists you implement across the platform – reducing your cost and risk.



Performance in Context: Stakeholders

What is the status and quality of my requested services?

How can I impact trends that affect cost and quality?

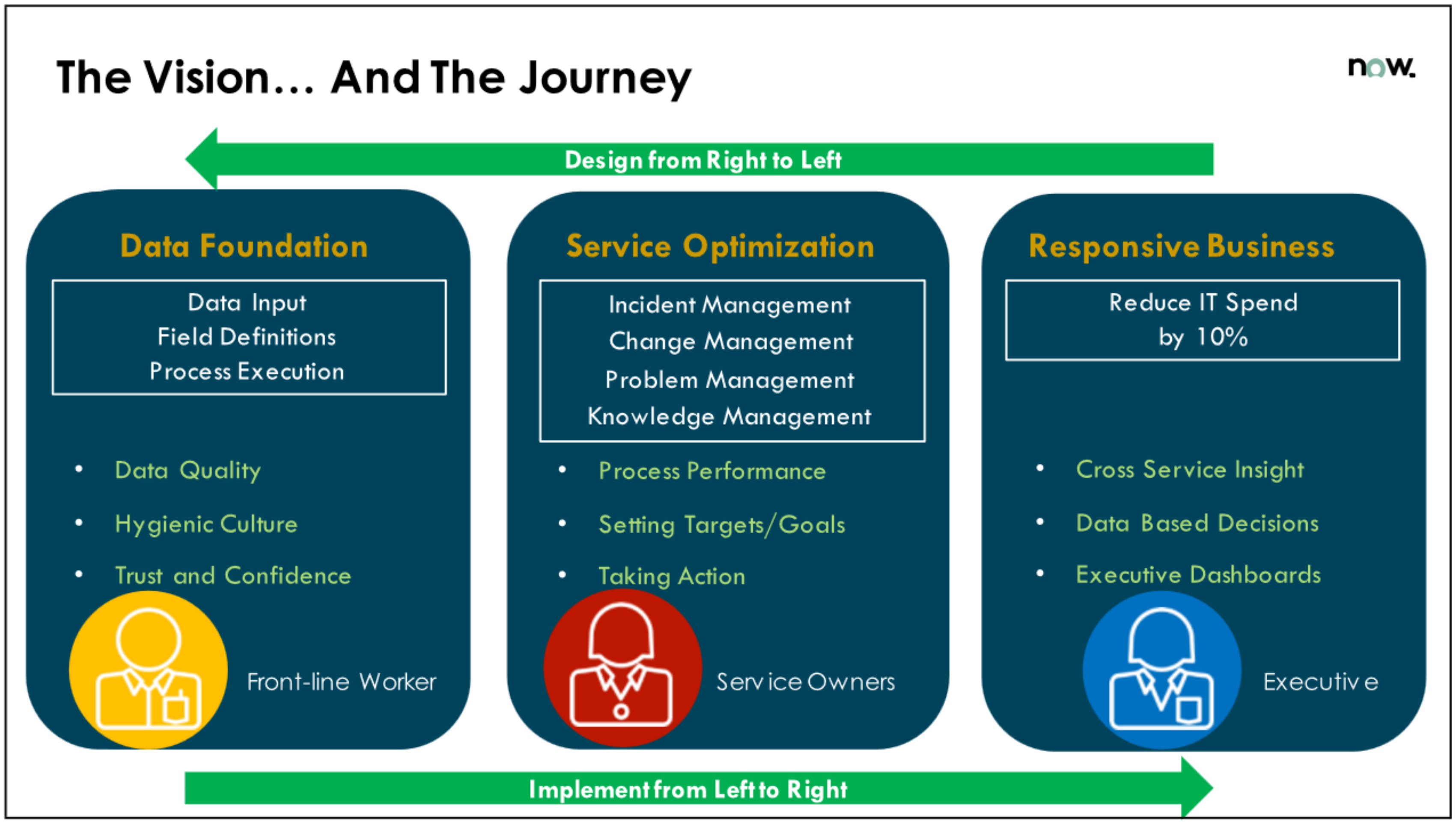
How can I work more efficiently to achieve targets?

How are we tracking against organizational objectives?

Stakeholders have different responsibilities leading to different business goals. Every role pictured above will have different questions that will help them meet their objectives:

- An Executive looks for information around corporate governance.
- The Service or Process Owners wants to see trend information related to the quality and cost of the services delivered.
- Front-line workers need relevant information that helps them make the right decisions about their current activities
- End users want insight into the activity around their requests.

All of these objectives require a slightly different representation of the data, but it’s all coming from the same central platform leading to stakeholder alignment.



The Vision… And The Journey

Design from Right to Left

Data Foundation

Data Input
Field Definitions
Process Execution

- Data Quality
- Hygienic Culture
- Trust and Confidence

Front-line Worker

Service Optimization

Incident Management
Change Management
Problem Management
Knowledge Management

- Process Performance
- Setting Targets/Goals
- Taking Action

Service Owners

Responsive Business

Reduce IT Spend
by 10%

- Cross Service Insight
- Data Based Decisions
- Executive Dashboards

Executive

Implement from Left to Right

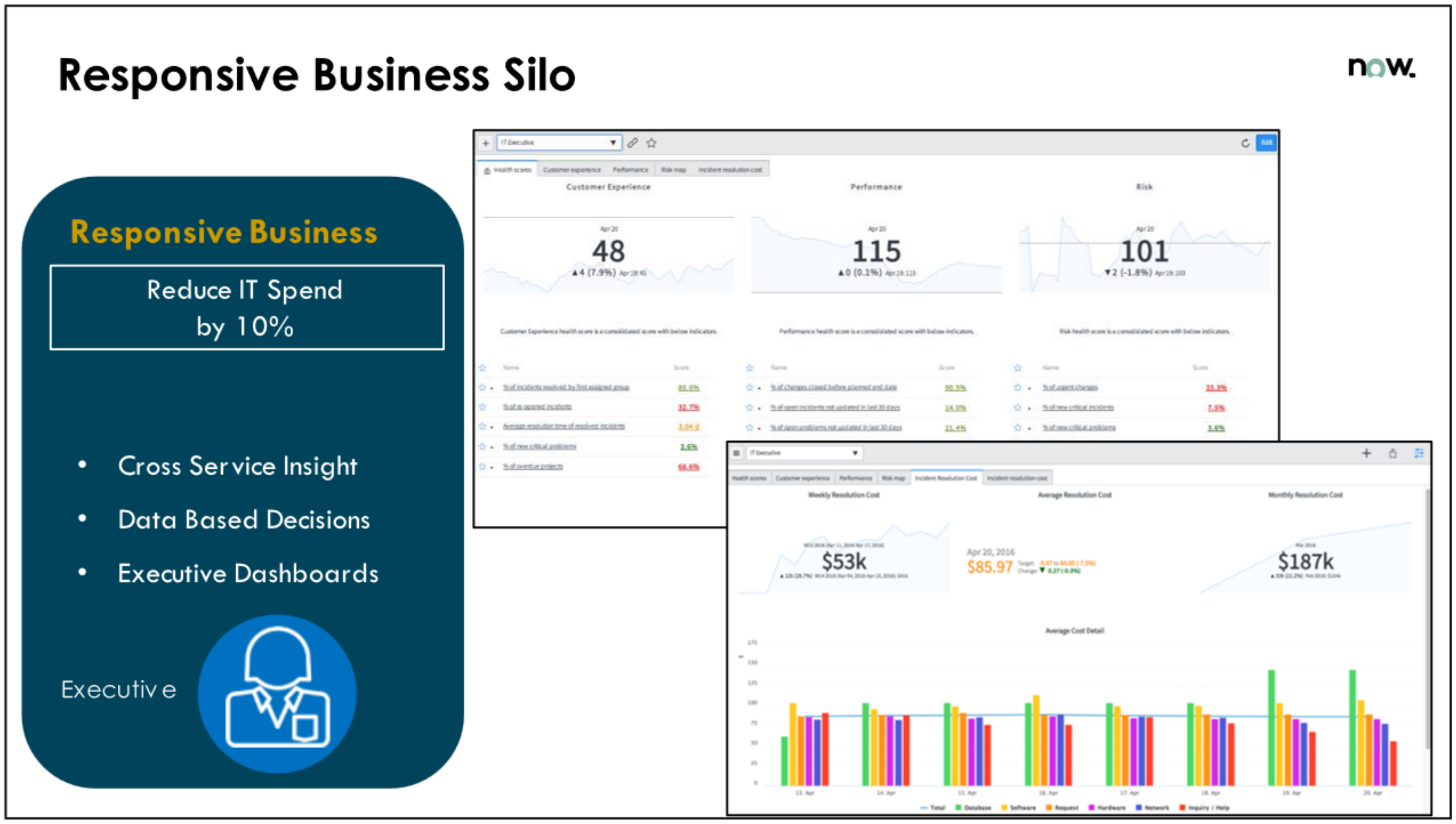
Performance Analytics provides a new view of Analytics and Business Intelligence built upon:

- Responsive Business: This is the end goal of any Performance Analytics implementation. It consists of Executive level business health dashboards tailored to business goals that enable cross service insight and empower smart business decisions based on real time data.

- Service Optimization: Service and Process owners focus on the health and trend of individual processes. They need trend information and targets to measure and drive towards process optimization and stakeholder alignment. Process trend information presented here bubbles up into executive style metrics on the Executive dashboards.

- Data Foundation: To empower Service Owners, technicians and front line workers focus on ticket and record processing. It is of key importance that field definitions and data input are standardized so that processes are executed in a consistent manner thus guaranteeing high data quality and providing a trusted data foundation.

The above represent the three stages of the Adoption Journey. The Journey needs to be designed with the end-goal in mind from right to left, but implemented from left to right – starting with a solid data foundation and bubbling up the underlying data to the Service Owner and eventually to the Executive dashboards.



Responsive Business Silo

Responsive Business

Reduce IT Spend by 10%

- Cross Service Insight
- Data Based Decisions
- Executive Dashboards

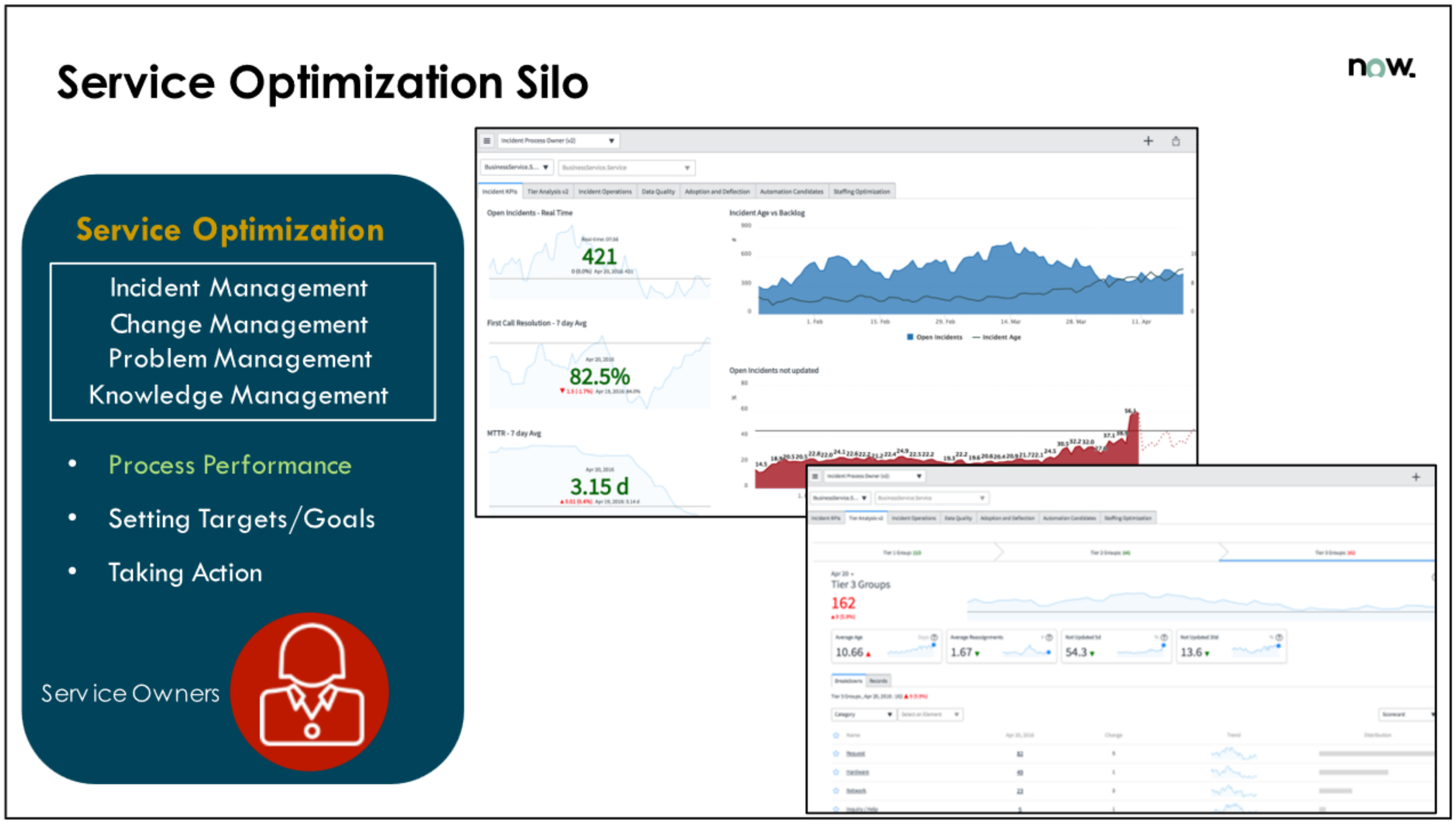
Executive

Because of the shared platform and the single data source, you can create tailored views to specific stakeholders. The Responsive Business outcome, or silo, of the Adoption Journey is geared toward the business executives.

At this level, the spotlight is on executive metrics or health scores – the end state of the Customer Journey. Pictured here, you can see a few examples that represent the pillars of service: Customer Experience, Performance, and Risk. Each of these high-level metrics is backed up by Indicators from multiple processes.

In our example, one of the executive goals is to Reduce IT spend by 10%. The corresponding targeted metrics are within the Incident Resolution Cost tab of the dashboard. You can see here how much we are spending on Ticket Resolution daily, weekly, and on a monthly basis.

By driving this benchmark, you work towards reduction in IT spend.



Service Optimization Silo

Service Optimization

Incident Management
Change Management
Problem Management
Knowledge Management

- Process Performance
- Setting Targets/Goals
- Taking Action

Service Owners

The Service Improvement is one step below the Responsive Business in terms or the data hierarchy.

In the context of our example, if the goal is to reduce spend, one way to achieve it is by optimizing the performance of relevant IT processes.

At this level you are looking at very high level view of key leading and lagging indicators related to the Incident Management process and visualized on Incident Process Owner dashboard:

- First Call Resolution is a leading indicator
- MTTR (Mean Time to resolution) is a lagging indicator
- The Open Incidents widgets presents the incident backlog related to Incident Management

Leveraged here is the ability to see real time scores, correlated multiple processes in the same graph, set targets, and even view trend predictions.

Looking into the Tier Analysis tab, you can see different visualizations that segment data in a a meaningful way and ultimately allow the Service Owner to take concrete action by revealing the backlog analyzed by tier and incidents segmented by assignment group and category.

Data Foundation Silo

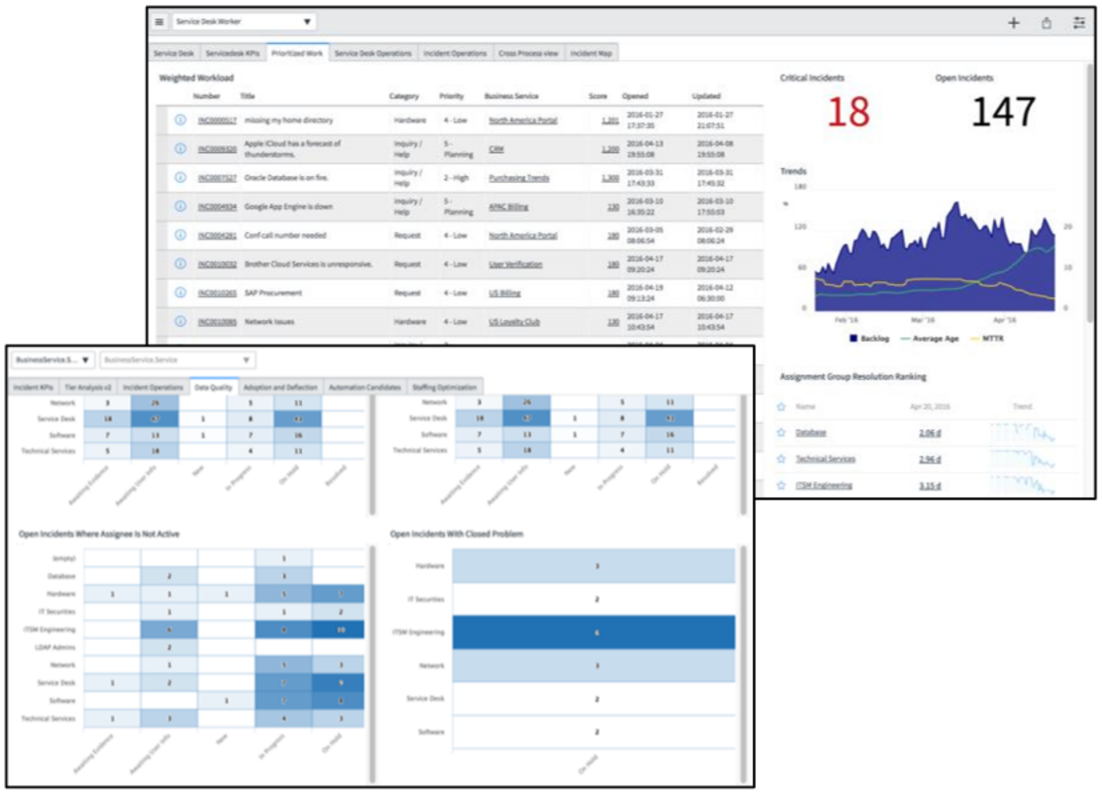
now.

Data Foundation

Data Input
Field Definitions
Process Execution

- Data Quality
- Hygienic Culture
- Trust and Confidence

Front-line Worker

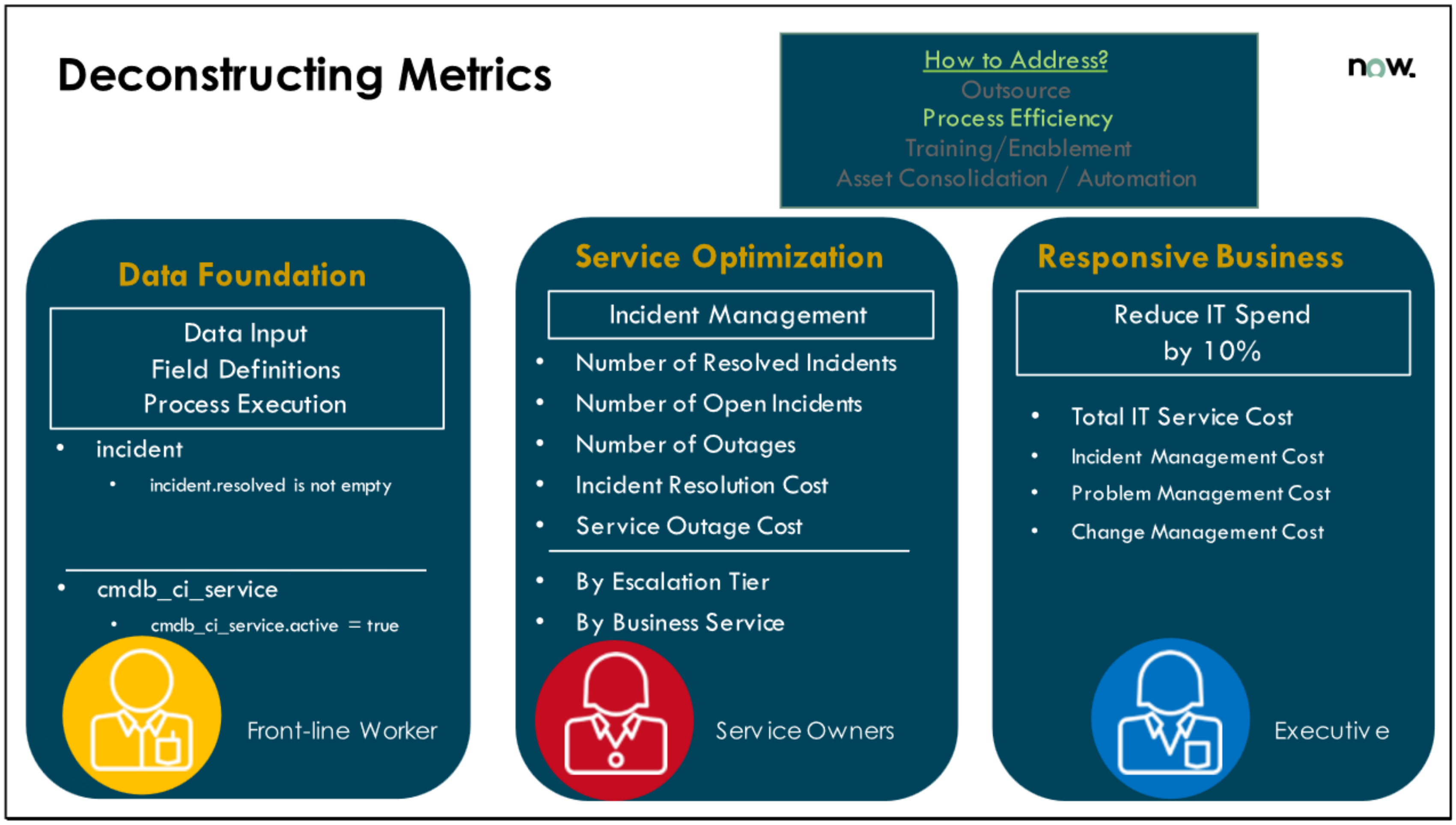


The Data Foundation silo is about how the individual front-line worker or technician contributes to data quality. It is a critical piece of Performance Analytics as it is the corner stone of good, trusted data that you can confidently bubble up to Service Owner and Executive Dashboards.

To promote good data, you need structured processes and workflows, enforcement of mandatory fields, well defined dictionary of fields, and shared standards.

Shown here is a Worker dashboard that allows them to work on the right task at the right time. This capability uses Spotlight intelligence to identify high priority work. As a result, front-line workers are not just ticket processors but know what they should work on and why.

Additional tabs with relevant data drill-downs allow to deep dive into the data and gain additional process insights. The Data Quality tab shown here employs advanced reports for comprehensive analysis of the current incident backlog.



Deconstructing Metrics

How to Address?
Outsource
Process Efficiency
Training/Enablement
Asset Consolidation / Automation

now.

Data Foundation

Data Input
Field Definitions
Process Execution

- incident
  - incident.resolved is not empty
- cmdb_ci_service
  - cmdb_ci_service.active = true

Front-line Worker

Service Optimization

Incident Management

- Number of Resolved Incidents
- Number of Open Incidents
- Number of Outages
- Incident Resolution Cost
- Service Outage Cost
- By Escalation Tier
- By Business Service

Service Owners

Responsive Business

Reduce IT Spend
by 10%

- Total IT Service Cost
- Incident Management Cost
- Problem Management Cost
- Change Management Cost

Executive

To reduce IT Spend by 10%, you have chosen the Process Efficiency route as your strategy. Next, you need to identify items and processes that help quantify items that contribute to cost and can be influenced in order to reduce cost. Here is a sample implementation plan:

Executive Metrics

- Executives need a Service Cost higher level metric. This metric can be further quantified into Incident Management Cost, Problem Management Cost, Change Management Cost, etc.

For each identified process area, you need to go a level deeper to the Service Owner:

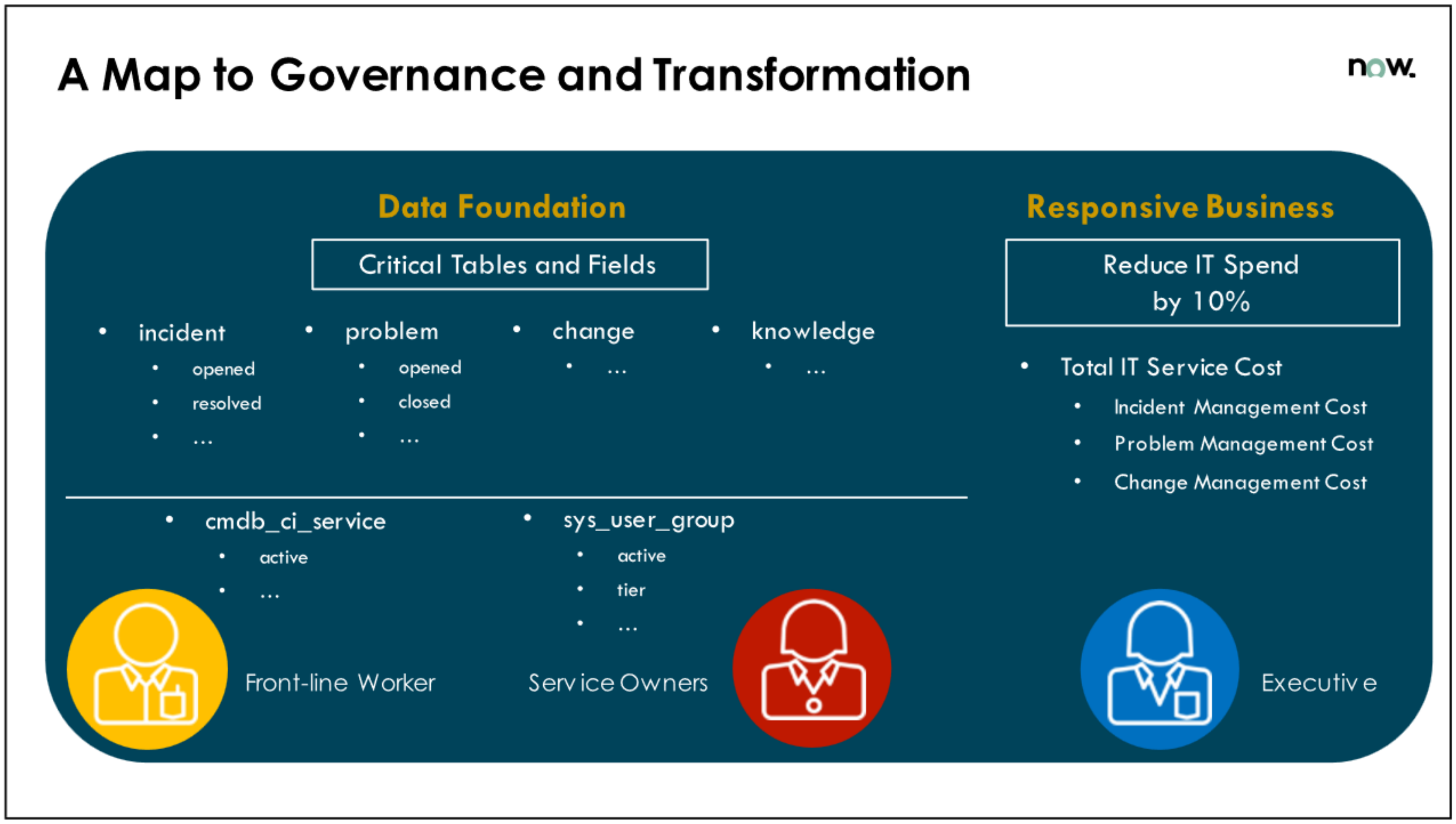
Service Owner Metrics:

- Service Owners need to see trends that quantify and directly or indirectly impact cost. For the Incident Management Process, this involves building a few key process indicators such as:
  - Number of Resolved incidents – there is a cost attached to each incident
  - Number of Open incidents – it lets you predict Service cost
  - Number of outages - Linked to incidents

The above can be segment by Tier or Business Service to enable Service Owners to take action.

Data Foundation Validation

To implement Executive and Process metrics, you need a consistent measure of what things like Resolved Incidents, Open Incidents, etc. are. For instance, a Resolved Incident may be defined as an incident for which the resolved timestamp is not empty. And if you are segmenting this data by Business Service, then you need to have a shared definition of where Shared services are stored.



A Map to Governance and Transformation

Data Foundation

Critical Tables and Fields

- incident
  - opened
  - resolved
  - …
- problem
  - opened
  - closed
  - …
- change
  - …
- knowledge
  - …

- cmdb_ci_service
  - active
  - …
- sys_user_group
  - active
  - tier
  - …

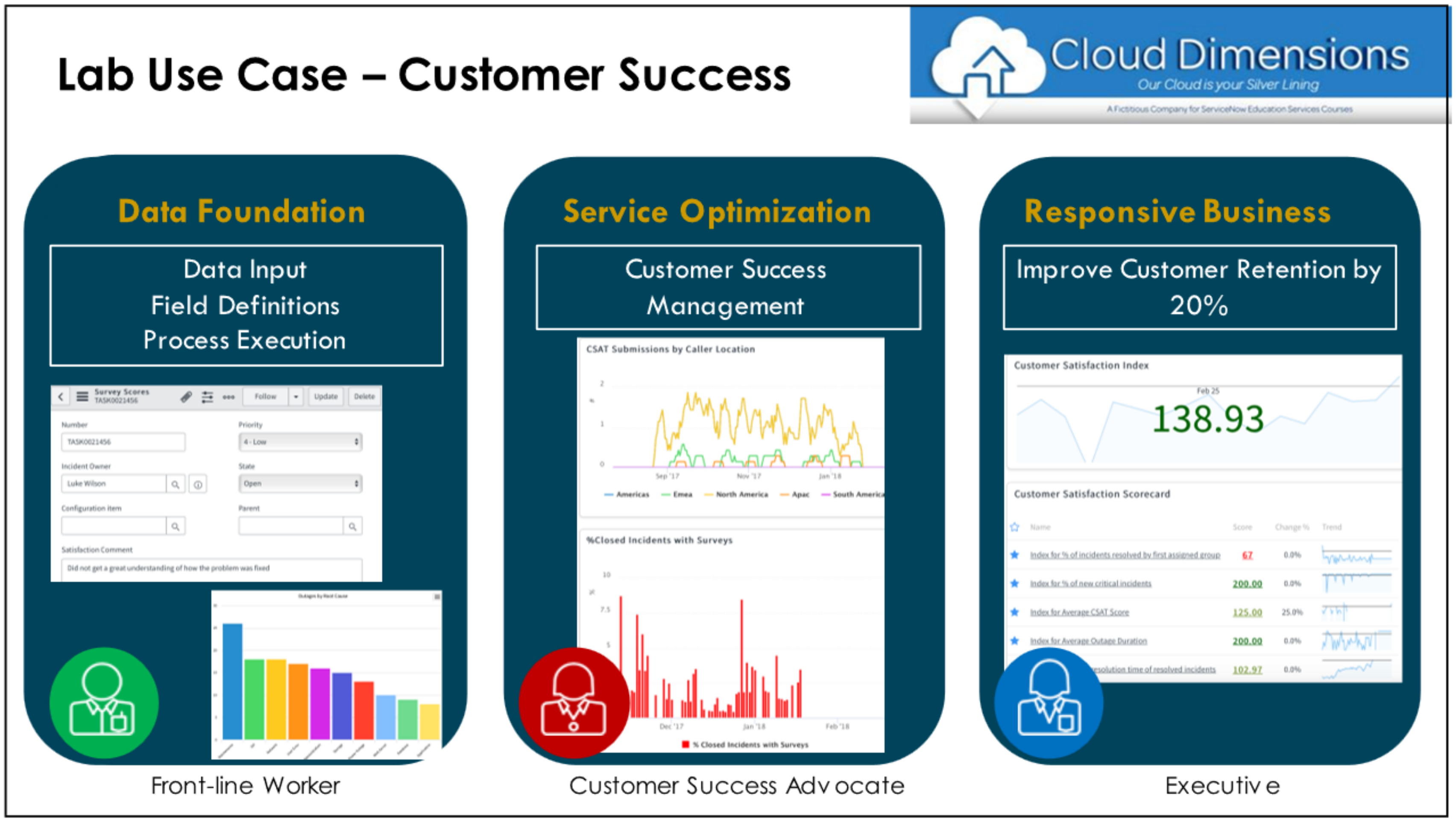
Responsive Business

Reduce IT Spend
by 10%

- Total IT Service Cost
  - Incident Management Cost
  - Problem Management Cost
  - Change Management Cost

To complete the exercise of quantifying Service Cost, metrics deconstructing should be done for every service that contributes to Service Cost. Problem and Change Management Costs should be broken down to the corresponding process indicators. In addition, data governance and standardization should be validated to make sure that all process workers are using the right data definitions and ensuring high data quality.

Ultimately, after completing the discovery for each process, a Data Map is created that links the process objective and executive metric to the respective core tables and fields. So as a result of going through the Adoption Journey, you will be able to map a high level Business Goal to the fields and tables that need to be protected.



Lab Use Case – Customer Success

Cloud Dimensions
Our Cloud is your Silver Lining
A Fictitious Company for ServiceNow Education Services Courses

Data Foundation

Data Input
Field Definitions
Process Execution

Survey Scores
TASK0021456

Number: TASK0021456
Priority: 4 - Low
Incident Owner: Luke Wilson
State: Open
Configuration item:
Parent:
Satisfaction Comment: Did not get a great understanding of how the problem was fixed

Follow
Update
Delete
Outages by Root Cause

Front-line Worker

Service Optimization

Customer Success
Management

CSAT Submissions by Caller Location
Sep '17
Nov '17
Jan '18
Americas
Emea
North America
Apac
South America
%Closed Incidents with Surveys
Dec '17
Jan '18
Feb '18
% Closed Incidents with Surveys

Customer Success Advocate

Responsive Business

Improve Customer Retention by
20%

Customer Satisfaction Index

138.93

Customer Satisfaction Scorecard

Name	Score	Change %	Trend
Index for % of incidents resolved by first assigned group	67	0.0%	
Index for % of new critical incidents	200.00	0.0%	
Index for Average CSAT Score	125.00	25.0%	
Index for Average Outage Duration	200.00	0.0%	
…esolution time of resolved incidents	102.97	0.0%	

Executive

In this class, you build towards a Use Case for the Customer Success Management process at Cloud Dimensions. Cloud Dimensions is a fictitious SaaS company which provides enterprise services to global customers.

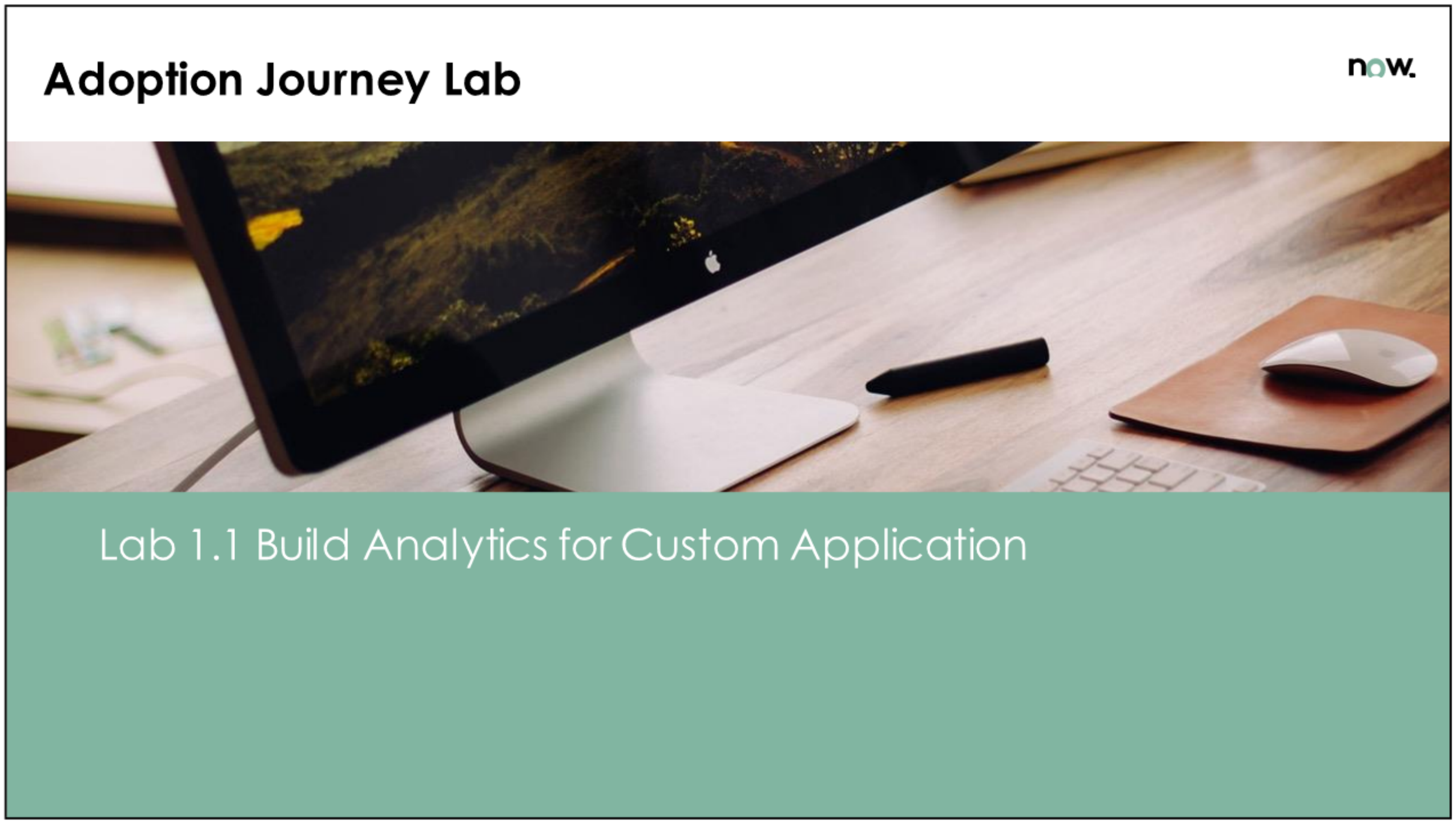
Executive Management at Cloud Dimensions is interested in improving customer retention and service satisfaction. This is a process efficiency approach which requires the implementation of process metrics for the Customer Success Management process.

The key stakeholder is the Customer Success Advocate, or Manager, who is responsible for customer relations and reporting on Incident and Change Management, Outage Management, and ultimately on Customer Satisfaction.

Throughout the lab portion of the class, you build various visualizations to present different aspects of Customer Success Management – Incident Metrics, Survey Metrics, and Change Metrics.

As part of Data Foundation, you define the core tables for Survey and Outage process measurement and implement corresponding Process Indicators.

At class completion, you are going to have a Customer Success executive dashboard featuring the Customer Success Index executive metric and multiple process metrics visualizing the many parts of Customer Success management.



Adoption Journey Lab

Lab 1.1 Build Analytics for Custom Application

Analytics for Custom Applications

Lab 1.1

⌛35m

Lab Objectives

As a Customer Success Manager, you would like to gain additional insight into customer satisfaction scores submitted through surveys. In addition, you would like to correlate Incident metrics with Survey scores. In this lab, you are going to create the following:

- Indicators to track Survey submission volume and the average Survey score
- A Formula indicator to monitor the percentage of Incidents which have a Survey
- Survey Breakdowns to help analyze the survey Scores
- Survey Collection Jobs to populate the data and perform trend analysis

Create Source and Indicators

A. Survey Table Structure

As a first step, you need to review the **Survey Scores** table structure.

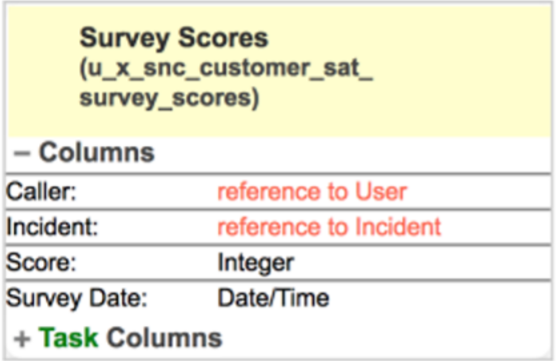
1. Log into the lab environment as a **System Administrator**.

2. Navigate to **System Definition > Tables & Columns**.

3. Select the **Survey Scores [u_x_snc_customer_sat_survey_scores]** table.

4. Click the **Schema map** button.

5. Review the **Survey Scores** list of columns.



Survey Scores
(u_x_snc_customer_sat_ survey_scores)

− **Columns**

Caller:	reference to User
Incident:	reference to Incident
Score:	Integer
Survey Date:	Date/Time

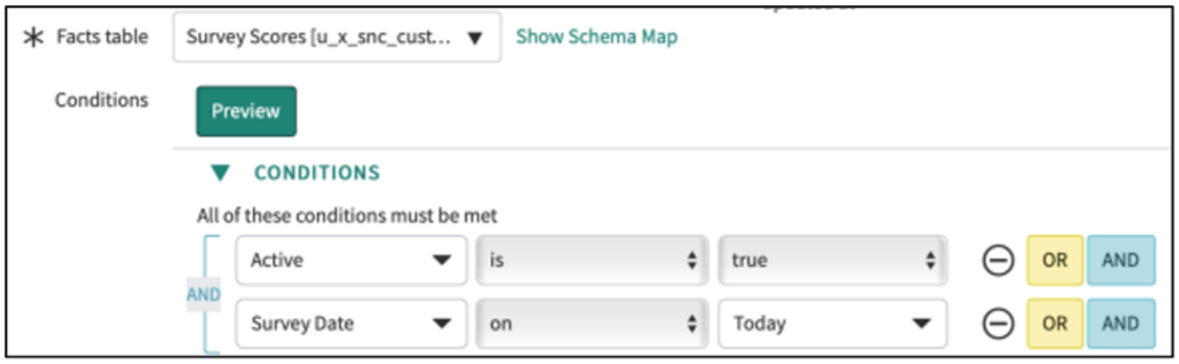
+ **Task Columns**

Question: *Which field should you use when building an Indicator to track Score magnitude?*

B. Indicator Source

In this step, you create a new Indicator source.

1. Navigate to **Performance Analytics > Sources > Indicator Sources**.
2. Click **New** and configure the source as follows:
   - Name: **survey.source**
   - Valid for frequency: **Daily**
   - Facts Table: **Survey Scores [u_x_snc_customer_sat_survey_scores]**
   - Conditions**: Active = true AND Survey Date on Today**



* Facts table: Survey Scores [u_x_snc_cust...

Conditions

CONDITIONS

All of these conditions must be met

Active is true

AND

Survey Date on Today

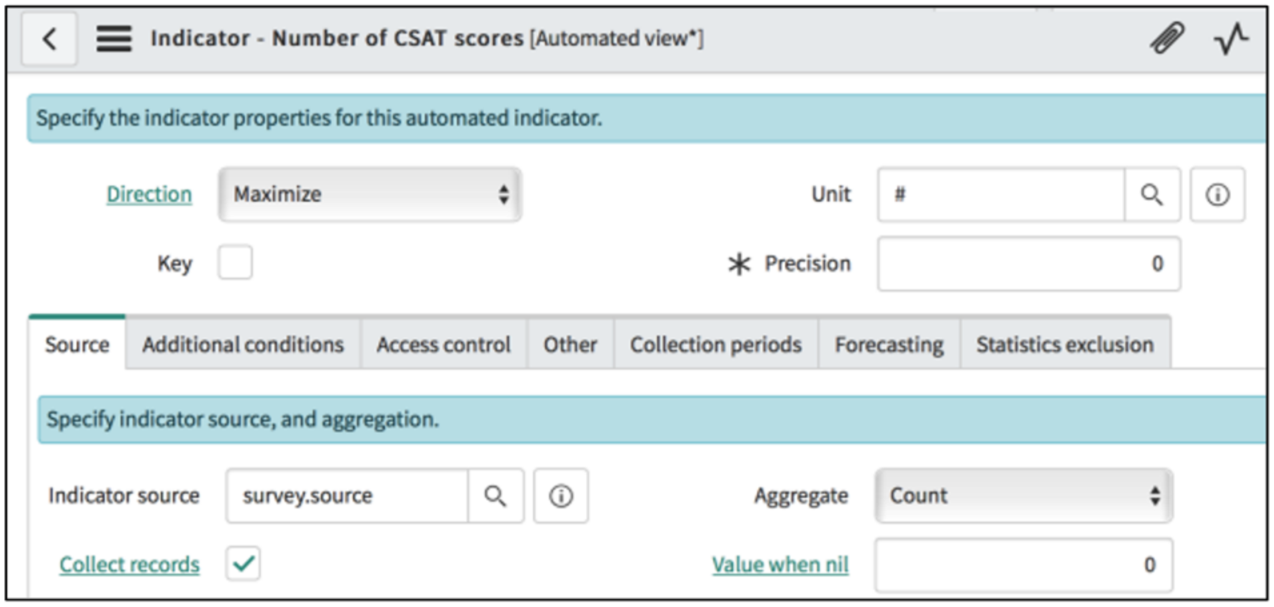
3. Click **Submit**.

C. Indicator for Daily Submitted Surveys

In this step, you create a new Indicator to retrieve a daily count of submitted surveys.

1. Navigate to **Performance Analytics > Indicators > Automated Indicators**.
2. Click **New** and configure the Indicator as follows:

   Name: **Number of CSAT scores**
   Direction: **Maximize**
   Unit: **#**
   Indicators Source: **survey.source**
   Aggregate: **Count**
   Value when nil: **0**



Indicator - Number of CSAT scores [Automated view*]

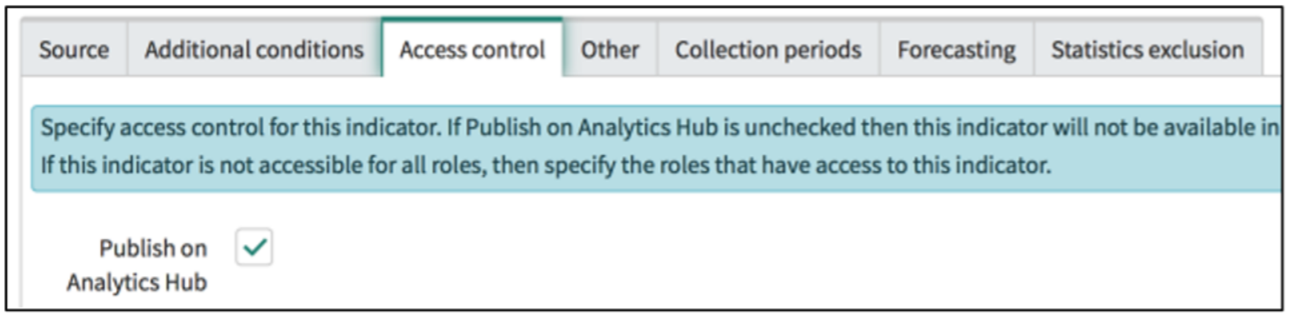
Specify the indicator properties for this automated indicator.

Direction: Maximize
Unit: #
Key: □
∗ Precision: 0

Specify indicator source, and aggregation.

Indicator source: survey.source
Aggregate: Count
Collect records: ☑
Value when nil: 0

3. Under **Access control**, check **Publish on Analytics Hub**.



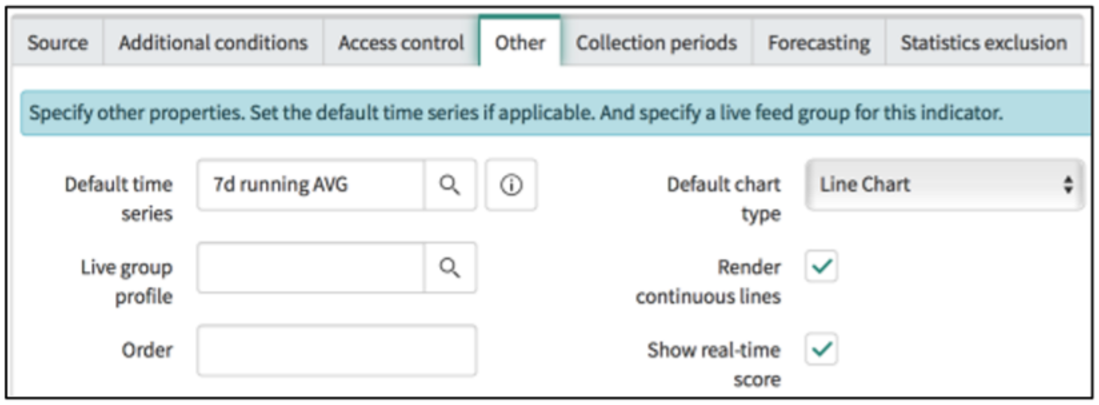
Specify access control for this indicator. If Publish on Analytics Hub is unchecked then this indicator will not be available in
If this indicator is not accessible for all roles, then specify the roles that have access to this indicator.

Publish on Analytics Hub: ☑

4. Under **Other**, check these two settings:
   - **Render Continuous lines**
   - **Show real-time score**

5. **Save** the Indicator.

6. Under **Other**, set the Default time Series to **7d running AVG**.



Specify other properties. Set the default time series if applicable. And specify a live feed group for this indicator.

Default time series: 7d running AVG
Default chart type: Line Chart
Live group profile:
Render continuous lines: ☑
Order:
Show real-time score: ☑

Note that we are using the 7d running AVG Time series to smooth out any data gaps

7. Click **Update**.

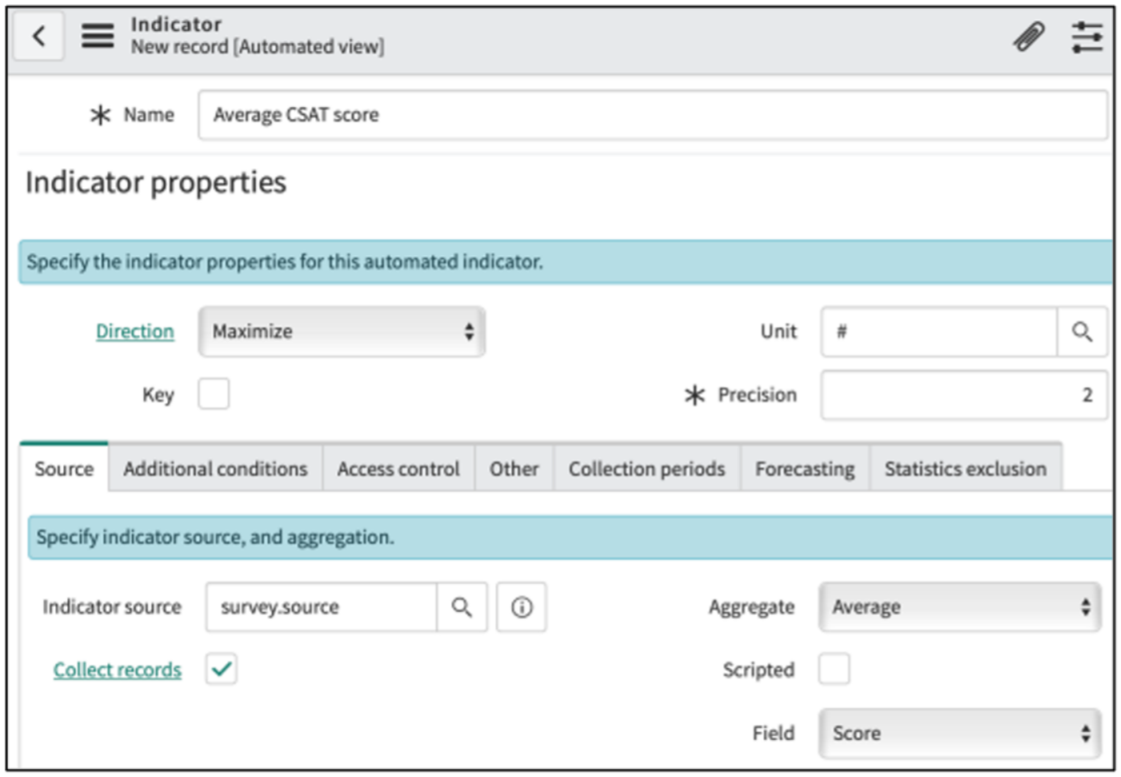
D. Indicator for Average Survey Score

In this step, you create another new Automated Indicator to trend the average Survey Score.

1. Navigate to **Performance Analytics > Indicators > Automated Indicators**.

2. Click **New** and configure the new Indicator as follows:

   Name: **Average CSAT score**
   Direction: **Maximize**
   Unit: **#**
   Precision: **2**
   Indicators Source: **survey.source**
   Aggregate: **Average**
   Field: **Score**



Indicator
New record [Automated view]

* Name	Average CSAT score		

Indicator properties

Specify the indicator properties for this automated indicator.

Direction	Maximize	Unit	#
Key	□	* Precision	2

Source | Additional conditions | Access control | Other | Collection periods | Forecasting | Statistics exclusion

Specify indicator source, and aggregation.

Indicator source	survey.source	Aggregate	Average
Collect records	☑	Scripted	□
		Field	Score

8. Under **Access control**, check **Publish on Analytics Hub**.

9. Under **Other**, check **Render Continuous lines**.

10. **Save** the Indicator.

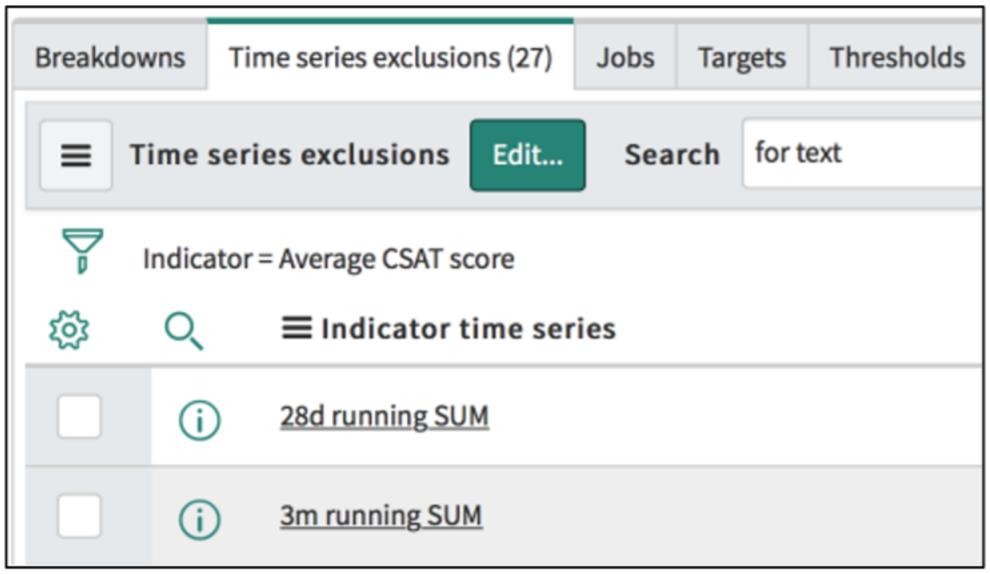
11. Under **Other**, set the Default time Series to **7d running AVG**.

12. **Save** the Indicator.

13. Navigate to the **Time series exclusions** related list.

14. Click **Edit…** and add all ***SUM** and ***SUM +** Time Series.

> *Summing the average Survey score is not meaningful in this context.*



Breakdowns | Time series exclusions (27) | Jobs | Targets | Thresholds

Time series exclusions [Edit...] Search for text

Indicator = Average CSAT score

Indicator time series

- □ 28d running SUM
- □ 3m running SUM

15. Click **Save** to return to the Indicator form.

E. Indicator for Percentage of Incidents with Surveys

In this step, you create a Formula Indicator to trend the percentage of Closed Incidents that have an associated Survey.

1. Navigate to **Performance Analytics > Indicators > Formula Indicators**.

2. Click **New** and configure the Indicator as follows:

   Name: **% Closed Incidents with Surveys**
   Direction: **Maximize**
   Unit: **%**
   Precision: **2**
   Formula:

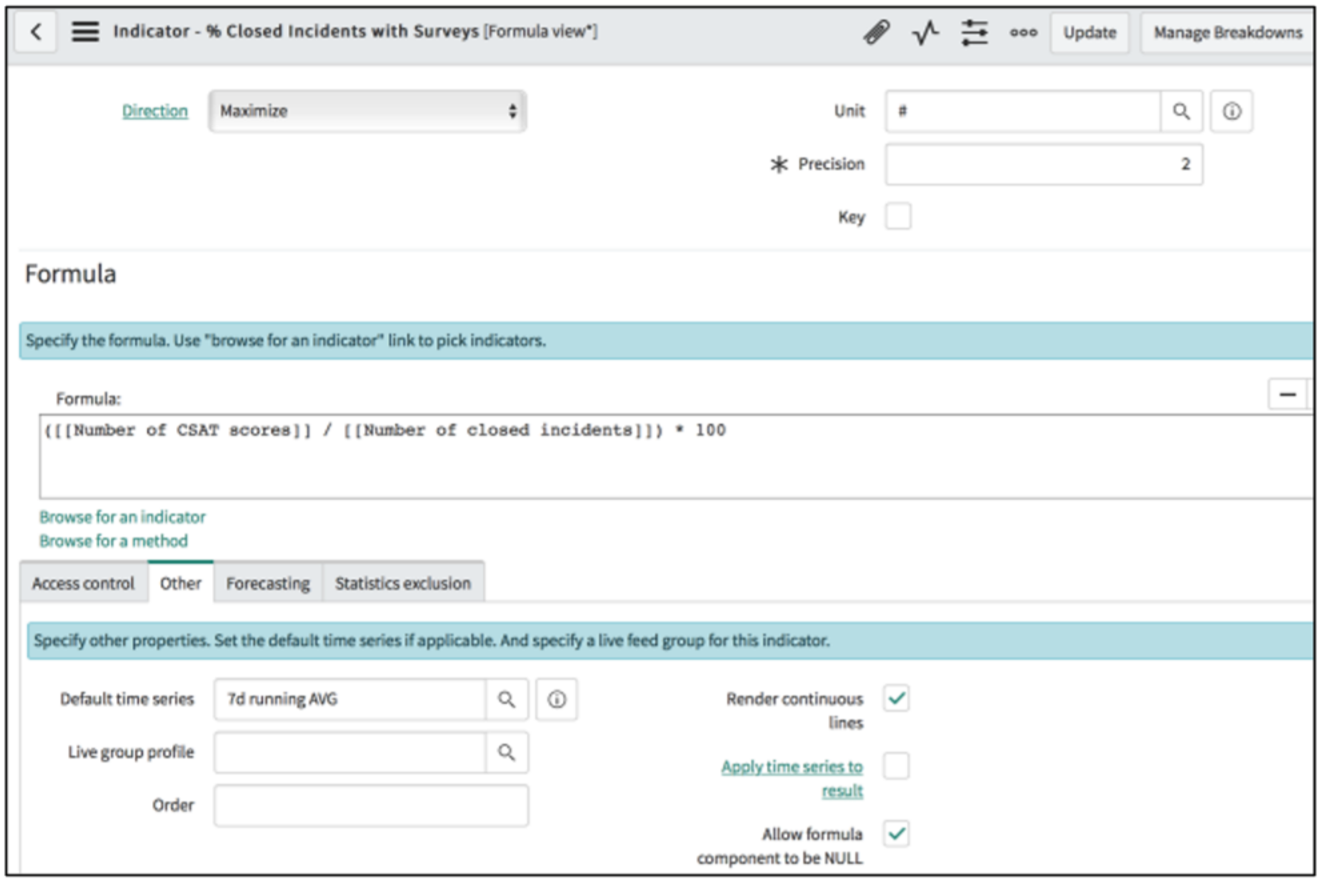
   ([[Number of CSAT scores]]
   /
   [[Number of closed incidents]])
   ***100**

3. Under **Access control**, check **Publish on Analytics Hub**.

4. Under **Other**, check:

- **Render Continuous lines**.
- **Allow formula component to be null**

5. **Save** the Indicator.



Direction: Maximize

Unit: #

✱ Precision: 2

Key: □

Formula

Specify the formula. Use "browse for an indicator" link to pick indicators.

Formula:

```
([[Number of CSAT scores]] / [[Number of closed incidents]]) * 100
```

Specify other properties. Set the default time series if applicable. And specify a live feed group for this indicator.

Default time series: 7d running AVG

Live group profile:

Order:

Render continuous lines: ☑

Apply time series to result: □

Allow formula component to be NULL: ☑

6. Under **Other**, set the Default time Series to **7d running AVG**.
7. **Save** the Indicator once again.
8. Navigate to the **Time series exclusions** related list.
9. Click **Edit…** and add all ***SUM** and ***SUM +** Time Series.
10. Click **Save** to return to the Indicator form.

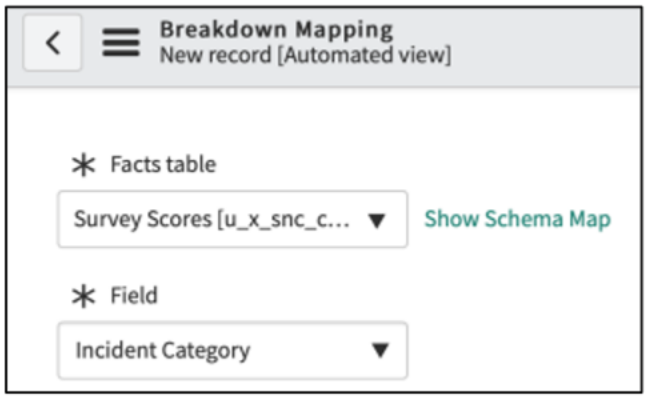
Implement Breakdowns

In this section, you need to create **mappings** for the following Breakdowns:

- Category
- Assignee
- Priority

Category Mapping

1. Navigate to **Performance Analytics > Breakdowns > Automated Breakdowns**.
2. Open the **Category** Breakdown with a source **Incident.Category**.
3. Navigate to the **Mappings** related list and click **New**.
4. Add a new mapping as follows:
   - Facts table: **Survey Scores**
   - Field:
     - Expand the **Incident** field
     - Select the **Category** field under Incident
5. Verify that your new mapping looks like this:



Breakdown Mapping
New record [Automated view]

∗ Facts table
Survey Scores [u_x_snc_c... ▼

∗ Field
Incident Category ▼

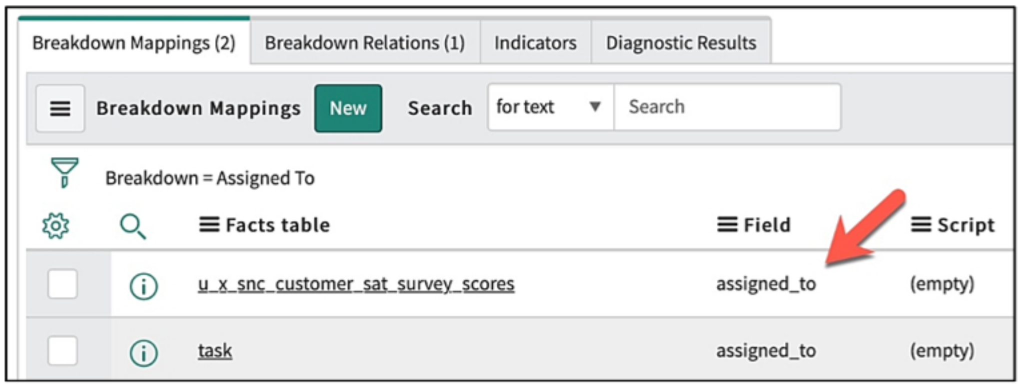
6. Click **Submit**.

Assigned To Mapping

1. Navigate to **Performance Analytics > Breakdowns > Automated Breakdowns**.
2. Open the **Assigned To** Breakdown.
3. Navigate to the **Mappings** related list and click **New**.
4. Add a new mapping as follows:
   - Facts table: **Survey Scores**
   - Field: **Incident Owner**

5. Click **Submit**.

6. Verify that your new mapping looks like this:



Breakdown Mappings (2) | Breakdown Relations (1) | Indicators | Diagnostic Results

Breakdown Mappings New Search for text Search

Breakdown = Assigned To

Facts table	Field	Script
u_x_snc_customer_sat_survey_scores	assigned_to	(empty)
task	assigned_to	(empty)

Priority Mapping

1. Navigate to **Performance Analytics > Breakdowns > Automated Breakdowns**.

2. Open the **Priority** Breakdown with a source **Incident.Priority**.

3. Navigate to the **Mappings** related list and click **New**.

4. Add a new mapping as follows:

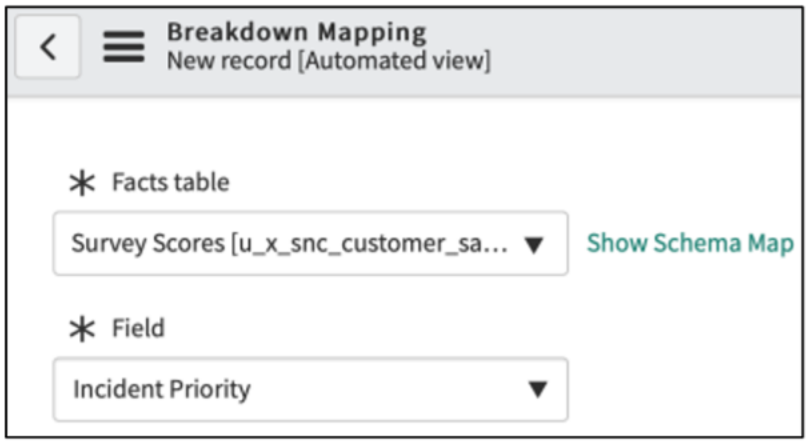
   - Facts table: **Survey Scores**

   - Field:

     - Expand the **Incident** field

     - Select the **Priority** field under Incident

5. Verify that your new mapping looks like this:



Breakdown Mapping
New record [Automated view]

Facts table: Survey Scores [u_x_snc_customer_sa…

Show Schema Map

Field: Incident Priority

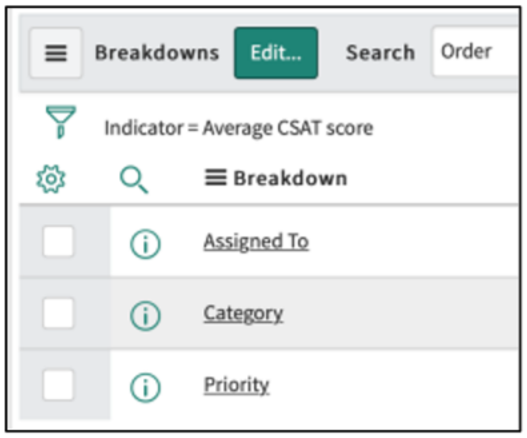
6. Click **Submit**.

Assign Breakdowns to Indicators

In this section you associate the **Priority**, **Category**, and **Assigned to** breakdowns with the three Indicators you created earlier.

Average CSAT Scores Indicator

1. Navigate to **Performance Analytics > Indictors > Automated Indicators**.
2. Open the **Average CSAT Scores Indicator**.
3. Navigate to the **Breakdowns** related list.
4. Add these breakdowns:



Indicator = Average CSAT score

Breakdown
Assigned To
Category
Priority

5. Click **Collect breakdown matrix** under the **Collect breakdown matrix** tab.
6. Click **Update**.

Number of CSAT Scores Indicator

1. Navigate to **Performance Analytics > Indictors > Automated Indicators**.
2. Open the **Number of CSAT Scores** Indicator.
3. Navigate to the **Breakdowns** related list and **Edit** to add these breakdowns:
   Assigned To
   Category
   Priority
4. Check **Collect breakdown matrix** under the **Collect breakdown matrix** tab.
5. Click **Update**.

% Closed Incidents with Surveys Indicator

1. Navigate to **Performance Analytics > Indictors > Formula Indicators**.

2. Open the **% Closed Incidents with Surveys** Indicator.

3. Navigate to the **Breakdowns** related list and **Edit** to add these breakdowns:
   Assigned To
   Category
   Priority

> ***Questions:***
>
> *Did you need to create a Breakdown Source for any of the Breakdowns and why?*
>
> *Did you need to create a Breakdown Mapping for all of the Breakdowns and why?*

Survey Data Collection

In this section, you create the following collection jobs:

- Historic job for collecting Survey Scores – going back **90 days/3 months**
- Scheduled Daily job for collecting Survey Scores

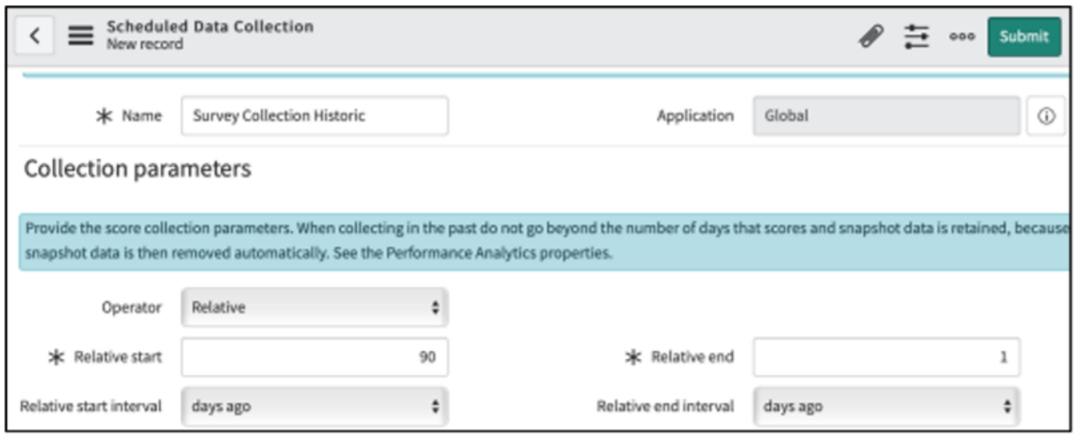
Historic Surveys Job

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Click **New** and create this job:

   - Name: **Survey Collection Historic**

   - Relative start: **90 days ago /** Relative end**: 1 day ago**



Scheduled Data Collection
New record

Name: Survey Collection Historic
Application: Global

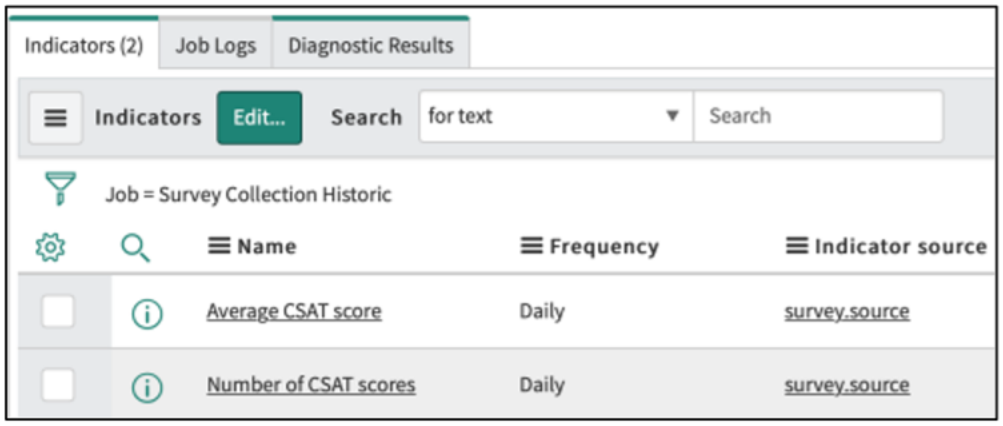
Collection parameters

Provide the score collection parameters. When collecting in the past do not go beyond the number of days that scores and snapshot data is retained, because
snapshot data is then removed automatically. See the Performance Analytics properties.

Operator: Relative
Relative start: 90
Relative end: 1
Relative start interval: days ago
Relative end interval: days ago

- Active: **unchecked**
- Run: **On Demand**

3. **Save** the Job.
4. Navigate to the **Indicators** related list and add these indicators:
    - Number of CSAT Scores
    - Average CSAT Score



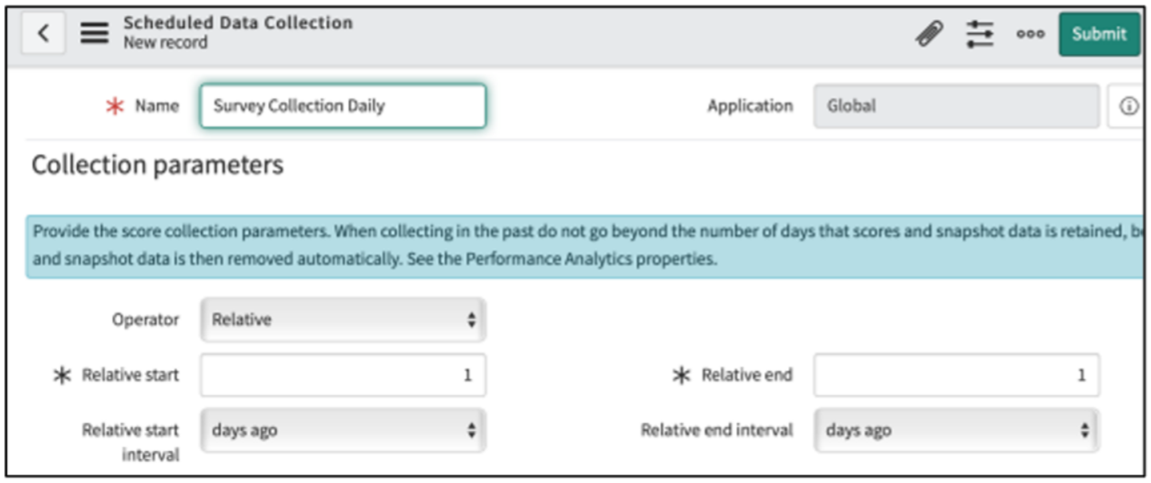
Job = Survey Collection Historic

Name	Frequency	Indicator source
Average CSAT score	Daily	survey.source
Number of CSAT scores	Daily	survey.source

5. Click **Execute Now** to execute the Survey Historic Collection job.

Daily Surveys Job

1. Navigate to **Performance Analytics > Data Collector > Jobs**.
2. Click **New** and create this job:
    - Name: **Survey Collection Daily**
    - Relative start**: 1 days ago /** Relative end**: 1 days ago**



Name	Survey Collection Daily	Application	Global

Collection parameters

Provide the score collection parameters. When collecting in the past do not go beyond the number of days that scores and snapshot data is retained, b
and snapshot data is then removed automatically. See the Performance Analytics properties.

Operator	Relative		
Relative start	1	Relative end	1
Relative start interval	days ago	Relative end interval	days ago

- Active: **checked** (The job will run daily)

3. **Save** the Job.

4. Navigate to the **Indicators** related list and add these indicators:

    - Number of CSAT Scores

    - Average CSAT Score

> ***Question:*** *Why is it redundant to execute the Daily Job right now?*

Historic Incidents Job Execution

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Open the **[PA Incident] Historic Data Collection** job.

3. Set the Relative start to **3 months ago**.

4. Click **Execute Now**.

> *Note that this job collects ~ 50K scores and is going to take 3-4 minutes to complete.*

Daily Incidents Job Activation

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Open the **[PA Incident] Daily Data Collection** job.

3. Select the **Active** checkbox under the **Job parameters** section.

4. Click **Update**.

Job Verification

1. Navigate to **Performance Analytics > Data Collector > Job Logs**.

2. Make sure that the two historic jobs you initiated have status of **Collecting** or **Collected**.

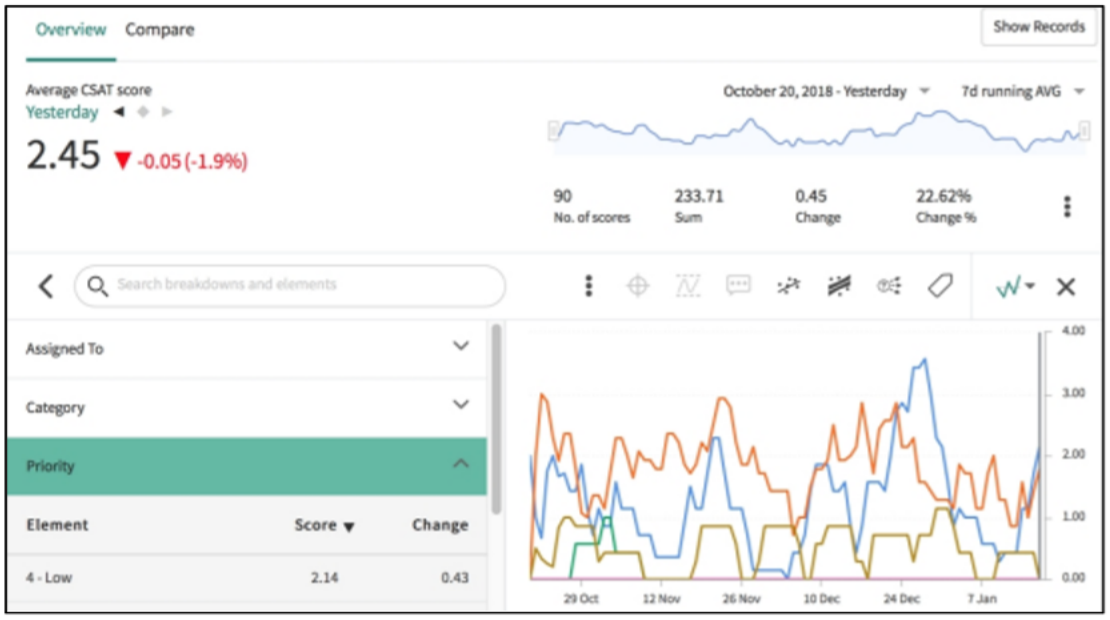
Analytics Hub Data Verification

In this section, you verify that data is available for the three indicators created earlier.

1. Navigate to **Performance Analytics > Analytics Hub**.

2. Make sure that all collection is complete and open the **Average CSAT scores / 7d running AVG** Indicator.

3. Verify that you have data for approximately the last 3 months and across all breakdowns.



Average CSAT score
Yesterday
2.45 ▼ -0.05 (-1.9%)

October 20, 2018 - Yesterday
7d running AVG

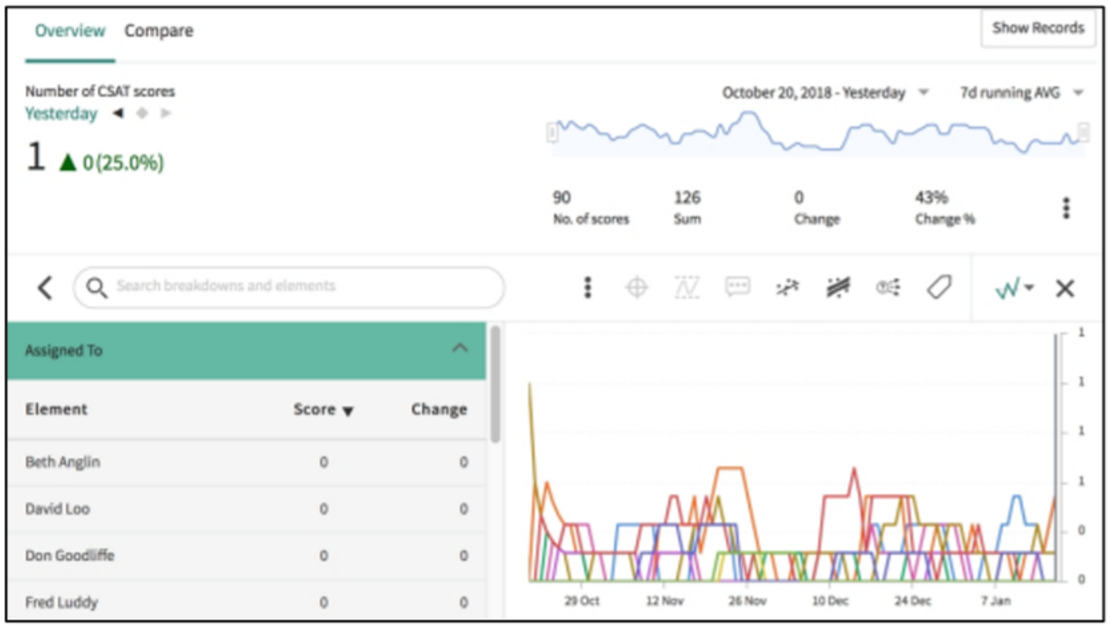
90	233.71	0.45	22.62%
No. of scores	Sum	Change	Change %

Assigned To
Category
Priority

Element	Score ▼	Change
4 - Low	2.14	0.43

4. Return to the Indicator list and open **Number of CSAT scores / 7d running AVG**.

5. Verify that you have data for approximately the last 3 months and across all breakdowns.



Number of CSAT scores
Yesterday
1 ▲ 0 (25.0%)

October 20, 2018 - Yesterday
7d running AVG

90	126	0	43%
No. of scores	Sum	Change	Change %

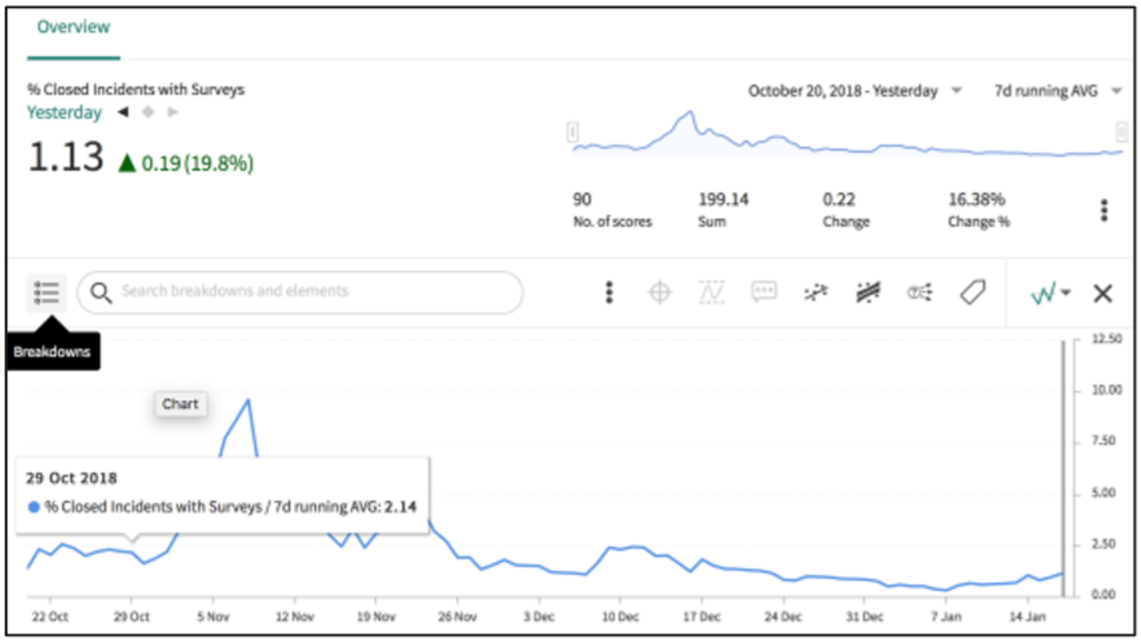
Assigned To

Element	Score ▼	Change
Beth Anglin	0	0
David Loo	0	0
Don Goodliffe	0	0
Fred Luddy	0	0

> ***Note:** Your actual scores and trends may differ slightly from the screen shots.*

6. Return to the Indicator list and open the **% Closed Incidents with Surveys / 7d running AVG** in the Analytics Hub.

7. Confirm that you have collected data for the past 3 months.



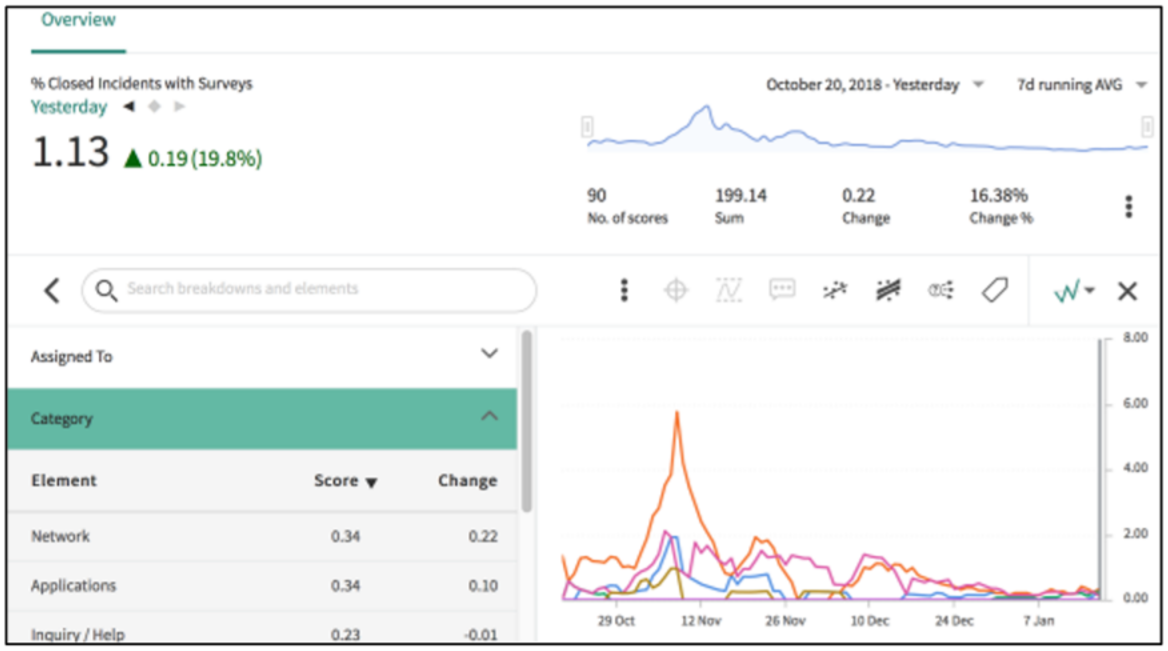
% Closed Incidents with Surveys
Yesterday
1.13 ▲ 0.19 (19.8%)

October 20, 2018 - Yesterday
7d running AVG

90	199.14	0.22	16.38%
No. of scores	Sum	Change	Change %

29 Oct 2018
● % Closed Incidents with Surveys / 7d running AVG: 2.14

8. Open the **Breakdowns** tab and select the **Category** breakdown.



% Closed Incidents with Surveys
Yesterday
1.13 ▲ 0.19 (19.8%)

October 20, 2018 - Yesterday
7d running AVG

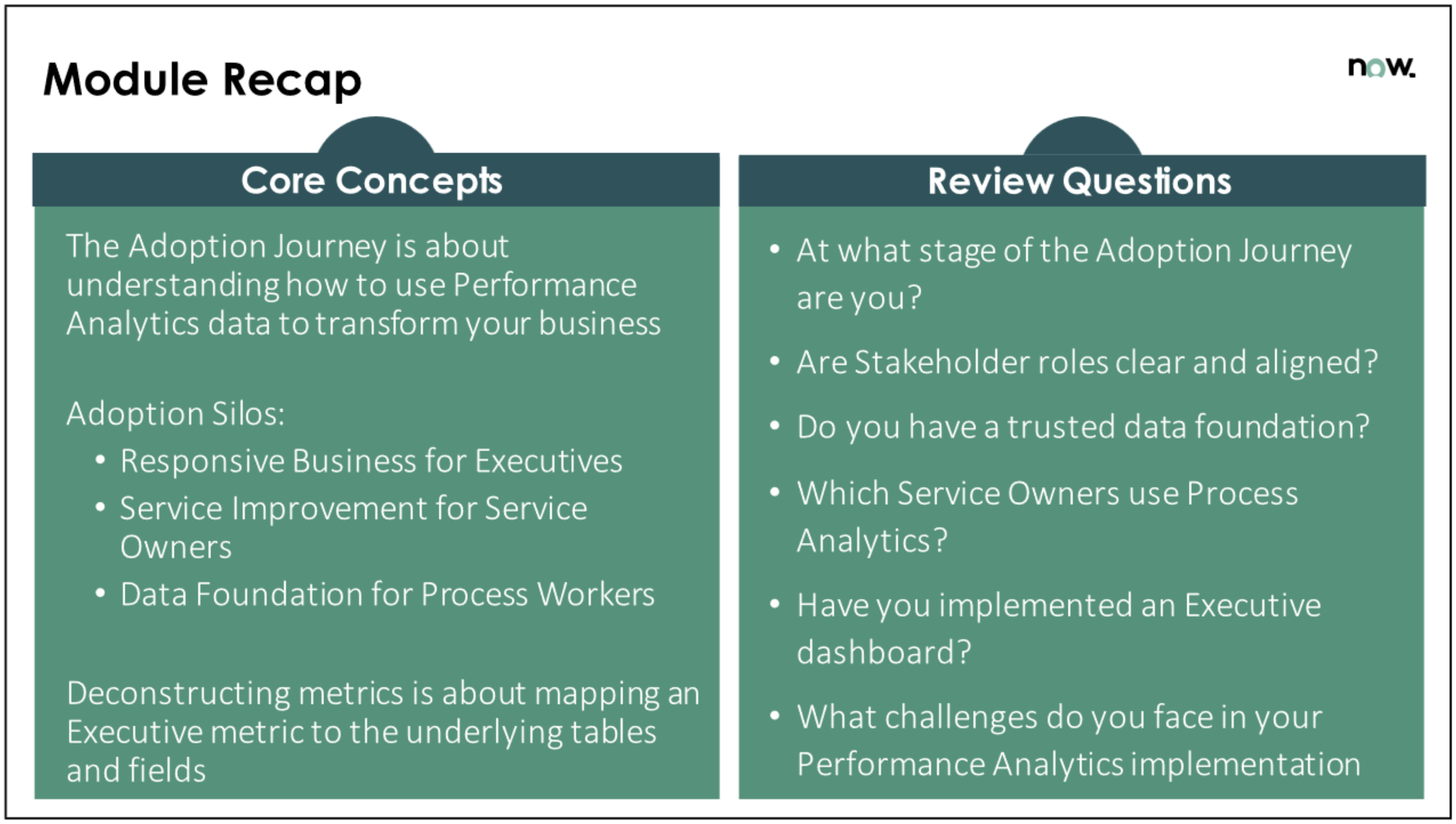
90	199.14	0.22	16.38%
No. of scores	Sum	Change	Change %

Assigned To
Category

Element	Score ▾	Change
Network	0.34	0.22
Applications	0.34	0.10
Inquiry / Help	0.23	-0.01

9. Confirm that scores have also been collected for the **Priority** and **Assigned To** Breakdowns.

You have now completed the Analytics for Custom Applications lab.



Module Recap

Core Concepts

The Adoption Journey is about understanding how to use Performance Analytics data to transform your business

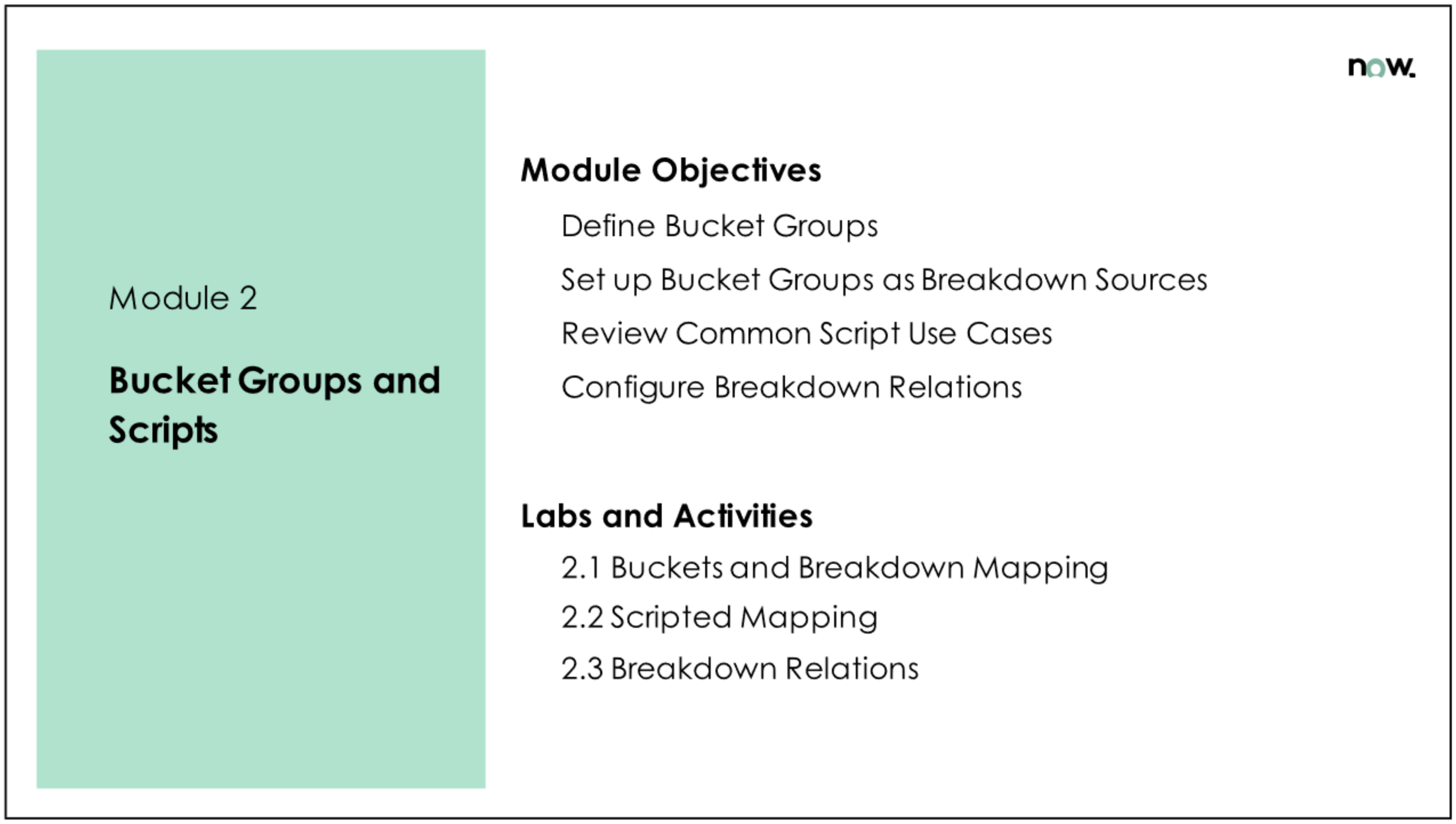
Adoption Silos:

- Responsive Business for Executives
- Service Improvement for Service Owners
- Data Foundation for Process Workers

Deconstructing metrics is about mapping an Executive metric to the underlying tables and fields

Review Questions

- At what stage of the Adoption Journey are you?
- Are Stakeholder roles clear and aligned?
- Do you have a trusted data foundation?
- Which Service Owners use Process Analytics?
- Have you implemented an Executive dashboard?
- What challenges do you face in your Performance Analytics implementation



Module 2

Bucket Groups and Scripts

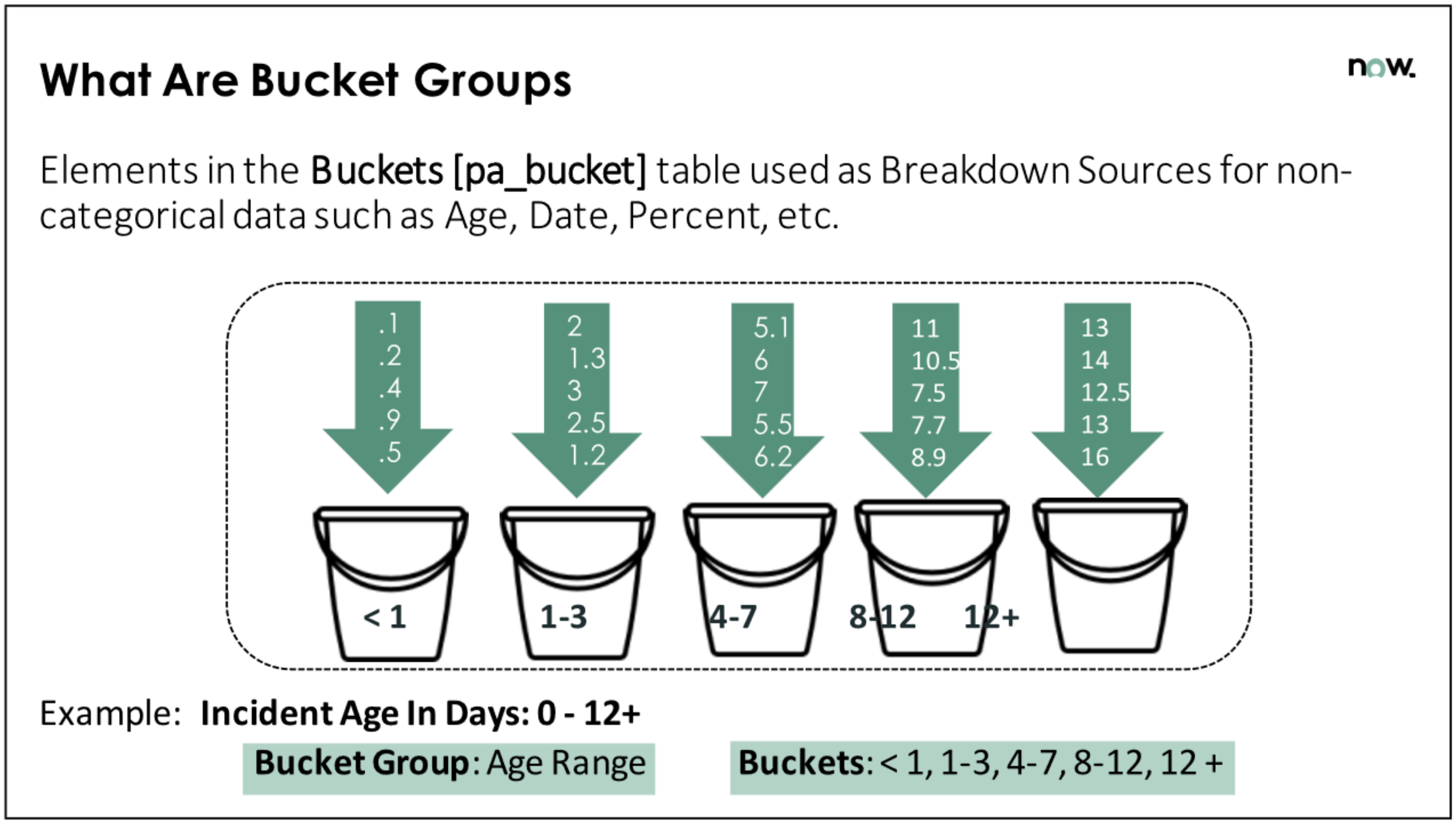
Module Objectives

- Define Bucket Groups
- Set up Bucket Groups as Breakdown Sources
- Review Common Script Use Cases
- Configure Breakdown Relations

Labs and Activities

- 2.1 Buckets and Breakdown Mapping
- 2.2 Scripted Mapping
- 2.3 Breakdown Relations

This module explores an alternative implementation of the Breakdown Source introduced in the previous chapter – the Bucket Groups. In addition, advanced breakdown mapping techniques are reviewed. Last, the concept of creating a relationships between unrelated breakdowns is explored.



What Are Bucket Groups

Elements in the **Buckets [pa_bucket]** table used as Breakdown Sources for non-categorical data such as Age, Date, Percent, etc.

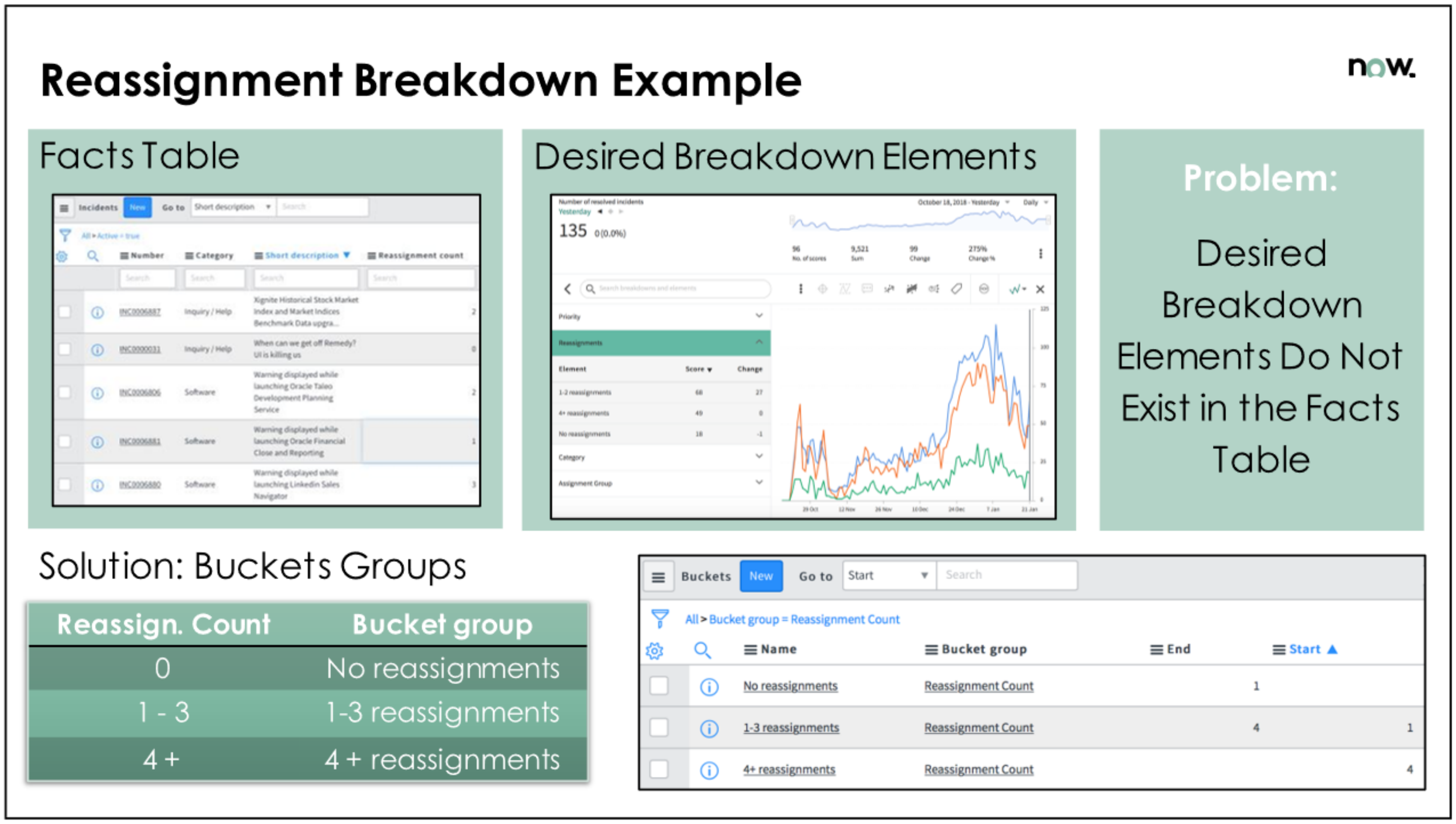
Example: **Incident Age In Days: 0 - 12+**

Bucket Group: Age Range **Buckets**: < 1, 1-3, 4-7, 8-12, 12 +

Attributes such as Priority, State, and Assignment Group have a finite set of values and are referred to as **Categorical**. Attributes such as Age, Cost, and Percent Complete are **Continuous** as their values can be any number within a given range.

Continuous attributes cannot be used as breakdowns. Can you imagine using Percent Complete as a breakdown? You could potentially have 100+ breakdown elements.

The solution to this problem is to use **Bucket groups**. A Bucket group organizes non-categorical data (continuous data) in categories. In the example here, the attribute to break by is Age. Age could potentially have infinite number of breakdown elements. Therefore we create a breakdown called Age Range which is based off of these Buckets representing age ranges: less than 1, 1-3, 4-7, 8-12, 12+ where the numbers describe the incident age in days.



Reassignment Breakdown Example

Facts Table

Incidents New Go to Short description Search

All > Active = true

Number	Category	Short description ▼	Reassignment count
INC0006887	Inquiry / Help	Xignite Historical Stock Market Index and Market Indices Benchmark Data upgra...	2
INC0000031	Inquiry / Help	When can we get off Remedy? UI is killing us	0
INC0006806	Software	Warning displayed while launching Oracle Taleo Development Planning Service	2
INC0006881	Software	Warning displayed while launching Oracle Financial Close and Reporting	1
INC0006880	Software	Warning displayed while launching Linkedin Sales Navigator	3

Desired Breakdown Elements

Number of resolved incidents

Yesterday

135 0 (0.0%)

October 18, 2018 - Yesterday Daily

96 No. of scores | 9,521 Sum | 99 Change | 275% Change %

Priority

Reassignments

Element	Score ▼	Change
1-2 reassignments	68	27
4+ reassignments	49	0
No reassignments	18	-1

Category

Assignment Group

Problem:

Desired Breakdown Elements Do Not Exist in the Facts Table

Solution: Buckets Groups

Reassign. Count	Bucket group
0	No reassignments
1 - 3	1-3 reassignments
4 +	4 + reassignments

Buckets New Go to Start Search

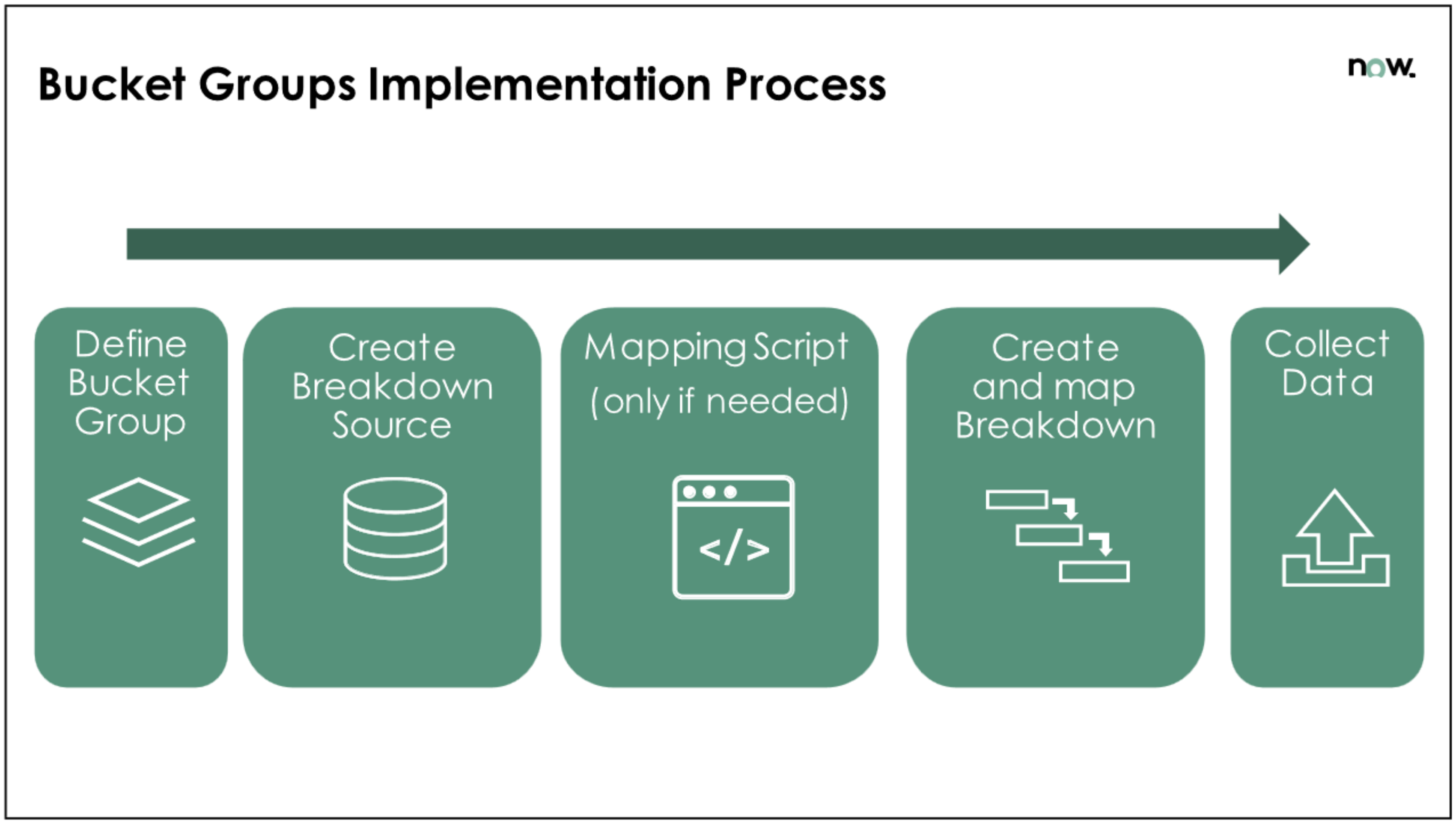
All > Bucket group = Reassignment Count

Name	Bucket group	End	Start ▲
No reassignments	Reassignment Count	1	
1-3 reassignments	Reassignment Count	4	1
4+ reassignments	Reassignment Count		4

In this example you have a requirement to navigate the Incident scores by Reassignment. Since the Reassignment count can be 0 or any positive number, the requirement is that you classify reassignment counts into 0, 1-3, and 4+ and create the corresponding breakdown structures.

The issue you are facing is that no source table (facts table) currently contains this summary representation of the reassignments.

The solution is to create breakdown elements as a bucket group where each bucket element corresponds to a Breakdown element. Bucket groups and elements are stored in the Buckets table which will later be used as a facts table for a Breakdown source.



Bucket Groups Implementation Process

now.

Define Bucket Group

Create Breakdown Source

Mapping Script (only if needed)

Create and map Breakdown

Collect Data

In addition to defining a breakdown with a Breakdown Source and Breakdown Mapping, a breakdown can also be described by a **Bucket Group** (acting as a source) and Breakdown mapping. Follow these steps to configure and use a Bucket group as a breakdown source:

- Create a Bucket group and define Buckets (categories)
- Create a Breakdown source using the Bucket group as the Facts table
- If necessary, create a script to perform the mapping
- Create a Breakdown if one doesn't exist already and map the Breakdown
- Do not forget to collect data

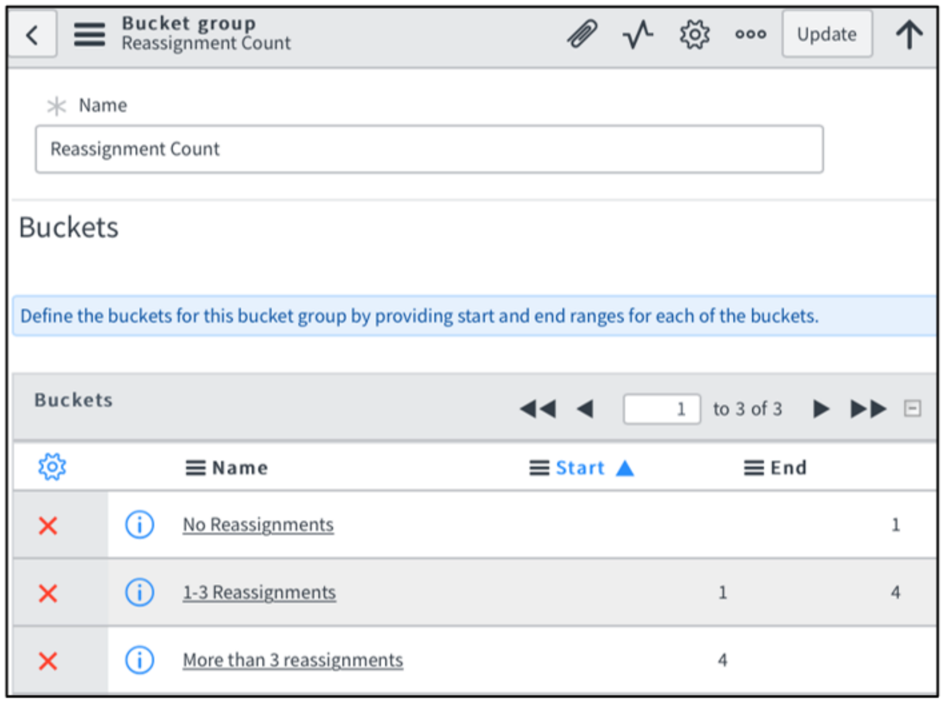
Bucket Group Configuration

- Bucket Name – used in the Breakdown Source
- Buckets Definitions
  - Bucket Name: Used as the breakdown / categorization element
  - Bucket Range:

    Start – Starting at this number

    End – Less than this number

  There should be no gaps!



Bucket group
Reassignment Count

Update

Name: Reassignment Count

Buckets

Define the buckets for this bucket group by providing start and end ranges for each of the buckets.

Buckets — 1 to 3 of 3

Name	Start	End
No Reassignments		1
1-3 Reassignments	1	4
More than 3 reassignments	4	

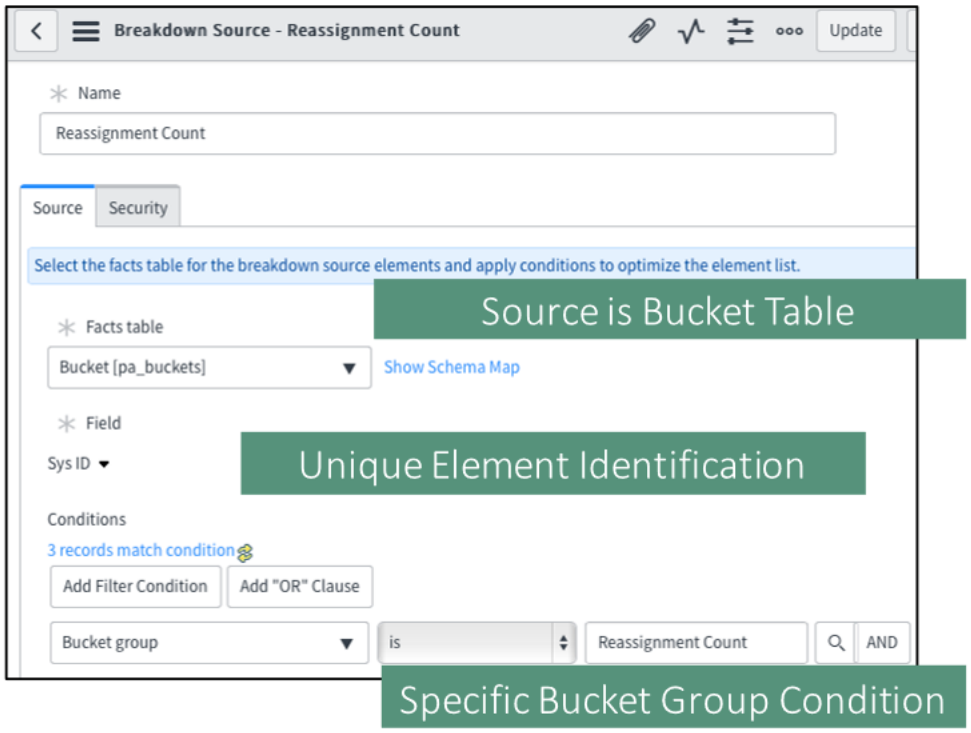
The Bucket group configuration steps are as follows:

- Navigate to **Performance Analytics > Breakdowns > Bucket Groups**
- Click **New**
- Provide a descriptive Name
- Define any number of Buckets:
  - Type in Bucket Name in the Buckets related list
  - Enter a Start and End value

Important: Bucket ranges are **inclusive** of the Start and **exclusive** of the End value.

Bucket Group as Breakdown Source

- Use when the source of the breakdown is contained in Buckets table
- Set the Facts table to **Bucket [pa_bucket]**
- Create a filter condition to identify the desired Bucket group by name



Breakdown Source - Reassignment Count

Name: Reassignment Count

Select the facts table for the breakdown source elements and apply conditions to optimize the element list.

Facts table: Bucket [pa_buckets]

Field: Sys ID

Conditions: 3 records match condition

Bucket group | is | Reassignment Count

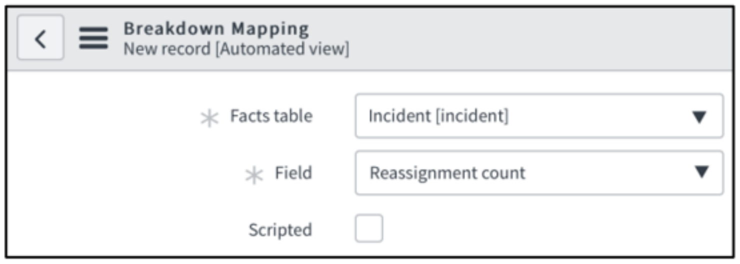
In this example, the breakdown source is not contained in any process or choice table. Rather, the breakdown source is a bucket group, stored in the pa_buckets table.

Therefore, the configuration of the breakdown source involves specifying the **Bucket** table (pa_buckets) as the Facts table.

The Condition **Bucket group is Reassignment Count** specifies the exact Bucket Group.

Breakdown based on Bucket Group

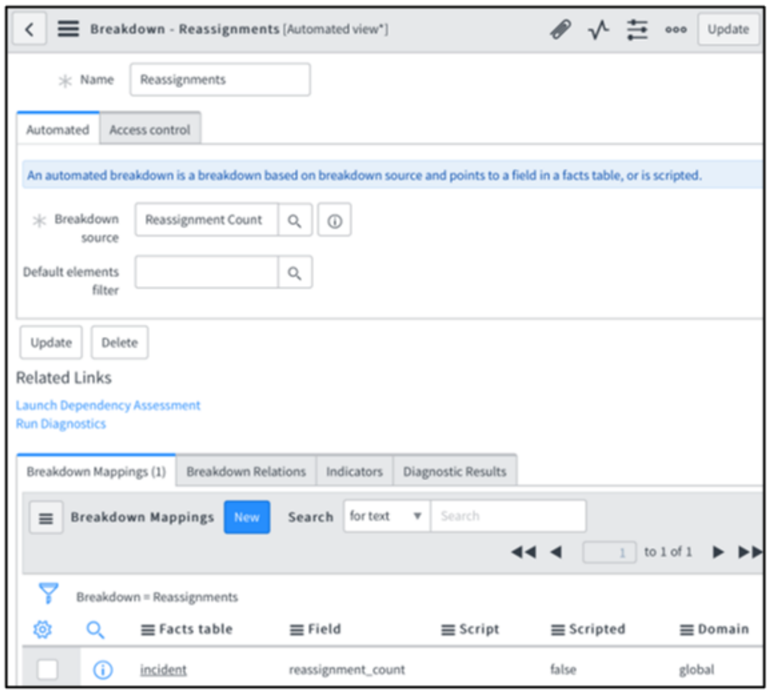
Breakdown using a Bucket-based breakdown source is configured identically to the non-bucket based breakdown



Breakdown Mapping
New record [Automated view]

* Facts table	Incident [incident]
* Field	Reassignment count
Scripted	□

The Bucket Group is only referenced by the Breakdown Source



Breakdown - Reassignments [Automated view*]

* Name: Reassignments

An automated breakdown is a breakdown based on breakdown source and points to a field in a facts table, or is scripted.

* Breakdown source: Reassignment Count

Default elements filter:

Related Links

Breakdown = Reassignments

		Facts table	Field	Script	Scripted	Domain
□	ⓘ	incident	reassignment_count		false	global

To complete the example, the corresponding breakdown uses the Reassignment Count breakdown source defined earlier. The fact that this is a bucket-based breakdown source is irrelevant here.

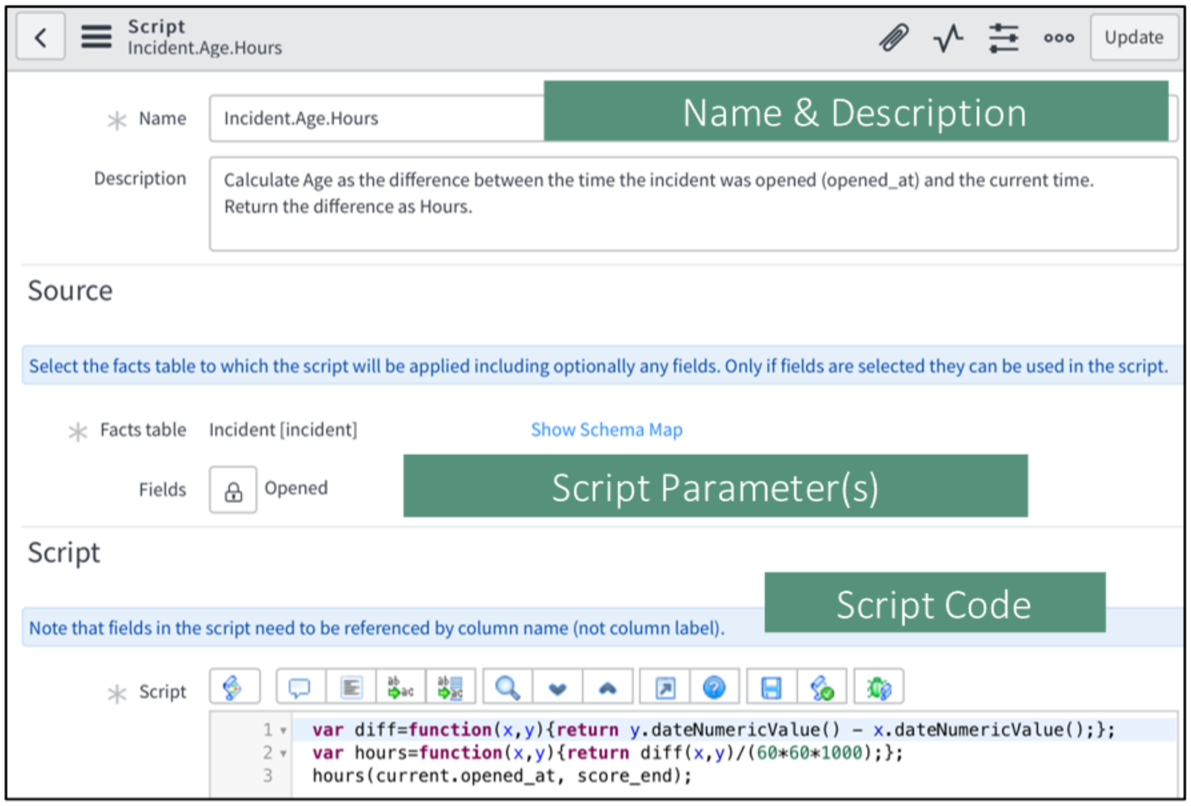
The breakdown mapping simply maps the value of the bucket group to the value of the reassignment count field of the incident table.

Bucket Groups review:

- Bucket groups create categories or classes based on numeric values that are not available in a table in ServiceNow.
- Buckets/elements are manually defined
- Bucket group is used as the Facts table for a breakdown source

Performance Analytics Scripts

- JavaScript code
- **Facts table**: Only indicators using the same Facts table can use the script
- **Fields**: Table fields that are passed to the script
- Additional avilable arguments:
  - score_start
  - score_end



Name: Incident.Age.Hours

Description: Calculate Age as the difference between the time the incident was opened (opened_at) and the current time. Return the difference as Hours.

Source

Select the facts table to which the script will be applied including optionally any fields. Only if fields are selected they can be used in the script.

Facts table: Incident [incident] Show Schema Map

Fields: Opened

Script

Note that fields in the script need to be referenced by column name (not column label).

Script:

```
var diff=function(x,y){return y.dateNumericValue() - x.dateNumericValue();};
var hours=function(x,y){return diff(x,y)/(60*60*1000);};
hours(current.opened_at, score_end);
```

Performance Analytics scripts are run during collection. The script executes for every row in the Indicator source set of records. The scripts use JavaScript syntax and the standard ServiceNow API.

You can pass table fields into the script when you include them in the **Fields** multi select field. In addition, these two variables are available for use in Performance Analytics scripts and Formula Indicator scripts:

- **score_start**: start of the collection period
- **score_end**: end of the collection period

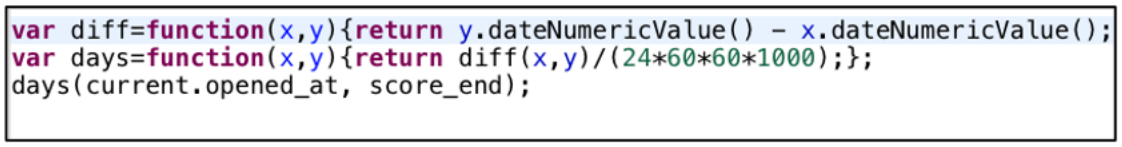
The values of these variables are determined by the frequency of the corresponding Indicator. For example, if a monthly indicator is being collected, score_start is the beginning of the month and score_end is the end of the month.

Note that script errors are reported in the Collection Jon and System logs.

Scripted Breakdown Mapping Use Case

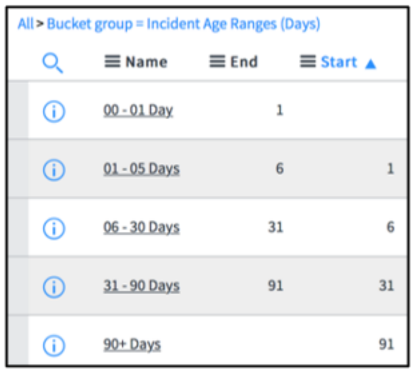
When a Breakdown element is not found in the Facts table, a script is needed to calculate the Facts table field to map to

The script calculates Incident Age as the difference between today and opened_at in days



```
var diff=function(x,y){return y.dateNumericValue() - x.dateNumericValue();
var days=function(x,y){return diff(x,y)/(24*60*60*1000);};
days(current.opened_at, score_end);
```

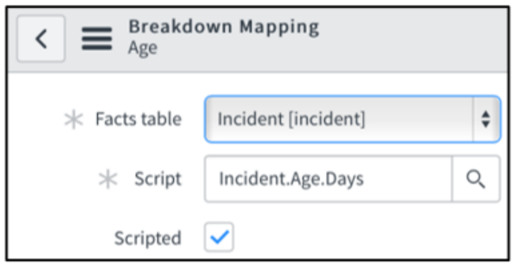
The Breakdown Elements are ranges of days: 0-1, 1-5, 6-30, etc.



All > Bucket group = Incident Age Ranges (Days)

Name	End	Start ▲
00 - 01 Day	1	
01 - 05 Days	6	1
06 - 30 Days	31	6
31 - 90 Days	91	31
90+ Days		91

The scripted Mapping maps the script output to a Bucket Group



Breakdown Mapping
Age

* Facts table: Incident [incident]

* Script: Incident.Age.Days

Scripted ☑

A scripted Mapping is needed when the Breakdown Elements are not available in a column of the Facts table. For instance, there is no field in the incident table which stores the age of the incident. But you need an Age Breakdown so you can analyze incident trends by Age.

Follow these steps to create a Breakdown on the Incident Age:

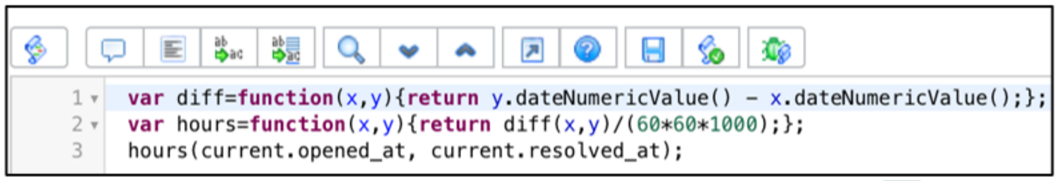
1. Create an Incident Age Ranges Bucket group with the following buckets of Ages: 0-1 days, 2-5 days, 6-30 days, 31-90 days, 90+ days.
2. Create an Age Breakdown and a Breakdown source based on the Age Ranges Bucket group.
3. Create the Incident.Age.Days script to calculate the incident's age as a difference of today's date and the incidents's opened_at field. (The script in this example converts the difference from milliseconds to days.)
4. Create a scripted Mapping which maps a Bucket group (e.g. 0-1 days) to the output of the Incident.Age.Days script.

Note that the calculated Age is not stored in a column in the Facts table of the indicator. Instead, the script adds a virtual column to each row in the data set. This additional virtual attribute is used to classify scores into Buckets groups.

Scripted Indicator Aggregate Use Case

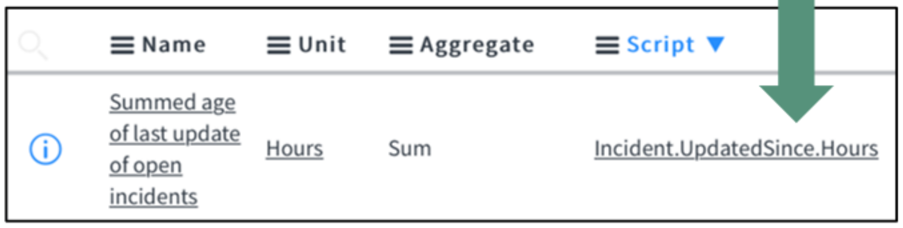
When your Indicator calculates an aggregation using a calculated field that does not exist in your Facts table

The script calculates the Age of Last update in hours



```
var diff=function(x,y){return y.dateNumericValue() - x.dateNumericValue();};
var hours=function(x,y){return diff(x,y)/(60*60*1000);};
hours(current.opened_at, current.resolved_at);
```

The Indicator returns the SUM of the Script output



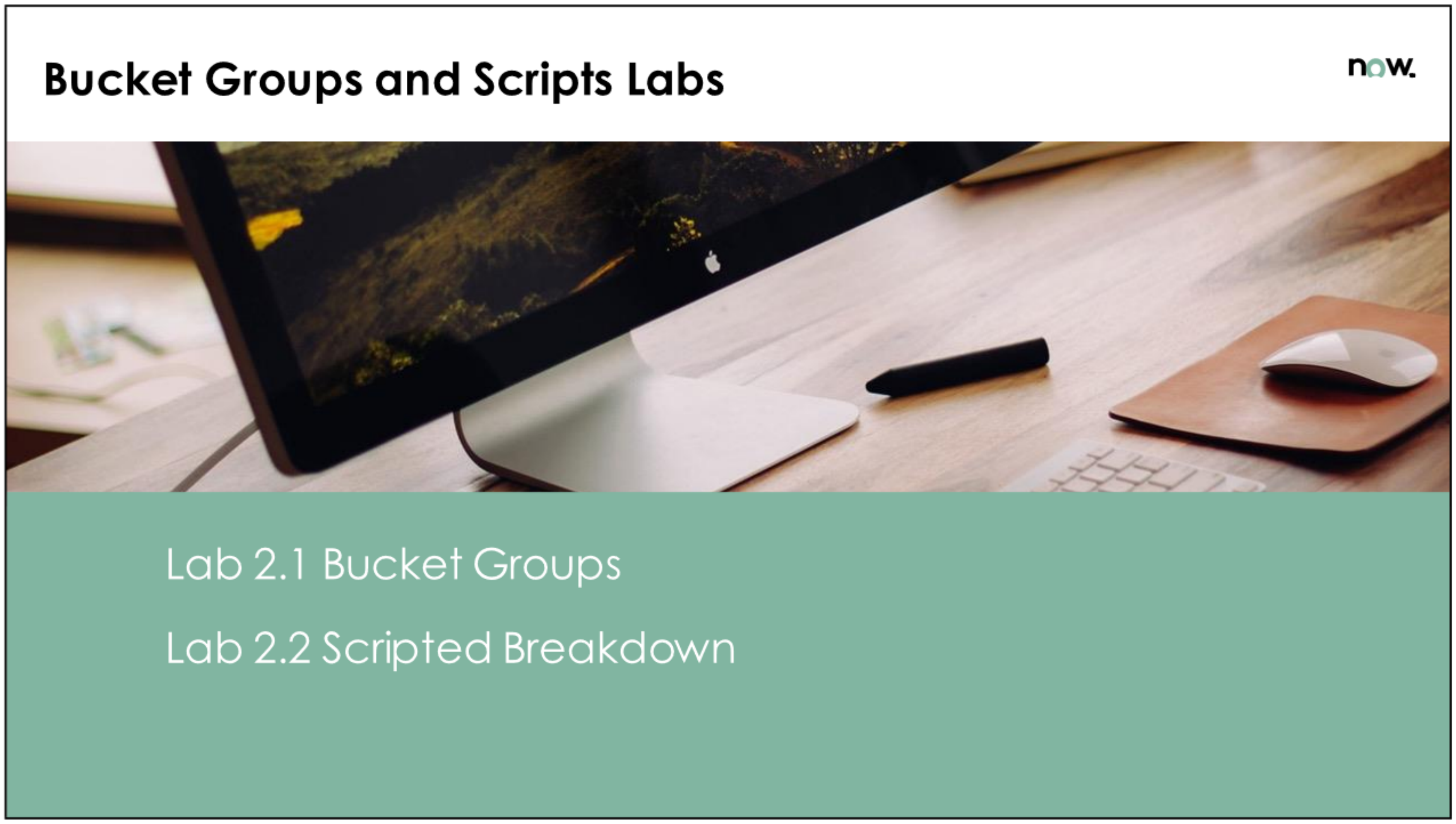
Name	Unit	Aggregate	Script ▼
Summed age of last update of open incidents	Hours	Sum	Incident.UpdatedSince.Hours

Another common use case for using a script is when having an Indicator with a scripted Aggregate. In this situation, what you want to aggregate is not a straightforward field, but rather a "calculation" derived using script logic.

In the above example, the **Summed age of last update of open incidents** Indicator returns the Sum of the Incident's last update. The last update is calculated and converted to hours using a script for all incidents in the data set of the indicator source. The script output is then summed up and returned as the Indicator score.

As a best practice, always check the out-of-box Performance Analytics to leverage existing code and use it as a template for further development.

Note: JavaScript and the ServiceNow API can also be used when building Metrics definitions, the Formulas code, and to evaluate custom logic when running conditional Scheduled Jobs.



Bucket Groups and Scripts Labs

Lab 2.1 Bucket Groups

Lab 2.2 Scripted Breakdown

2.1 Bucket Groups Lab

- Create a bucket group
- Create a breakdown and a breakdown source
- Collect data and verify indicator scorecard

2.2 Scripted Breakdown Lab

- Create a Bucket Group for the Closed Incidents using a Script
- Create a breakdown source and a scripted mapping

Bucket Groups

Lab 2.1

⌛15m

Lab Objectives

The Cloud Dimensions Incident process owner suspects that the number of incident reassignments is too high at times. Reports are showing that incident reassignments range from 0 to 10. A more meaningful categorization is needed to classify reassignments into "no reassignments", "one to three", and "more than four" which can be achieved using a Bucket group. In this lab, you perform the following:

- Create a Bucket group
- Create a Breakdown source using the Bucket group
- Create a corresponding Breakdown and associate with Indicators
- Run collection with Breakdown exclusions

A. Define Bucket Group

The reassignment count attribute of an Incident record contains the number of times an incident has been reassigned. The number is too granular for meaningful analysis. Therefore, you are going to represent as categories or "buckets", as per the logic below:

Bucket	Reassignment Count Start	Reassignment Count End
No reassignments		1
1-3 reassignments	1	4
4+ reassignments	4	

> ***Note:*** *The Bucket includes the Start field and excludes the End field. If an Incident is reassigned 4 times (reassignment_count = 4), it will be included in the* ***4+ reassignments*** *Bucket.*

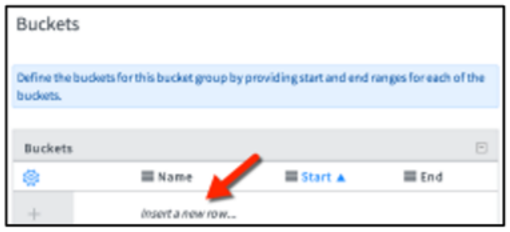
B. Implement Bucket Group

1. Navigate to **Performance Analytics > Breakdowns > Bucket Groups**.

2. Click **New**.

3. Set the Name to **Reassignment Count**.

4. In the **Buckets** section, double-click in the **Insert a new row...** field to add a new row.



Buckets

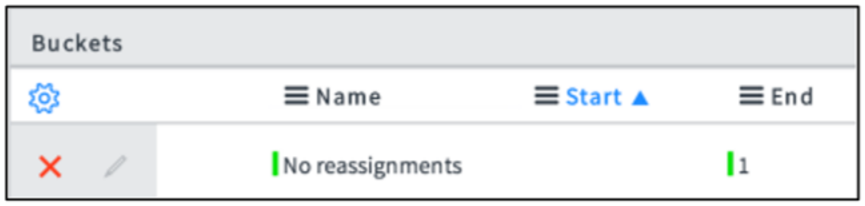
Define the buckets for this bucket group by providing start and end ranges for each of the buckets.

Buckets

	Name	Start ▲	End
+	*Insert a new row...*		

5. Make the following entries:

   - Name: **No reassignments**
   - Start: leave empty
   - End: **1**

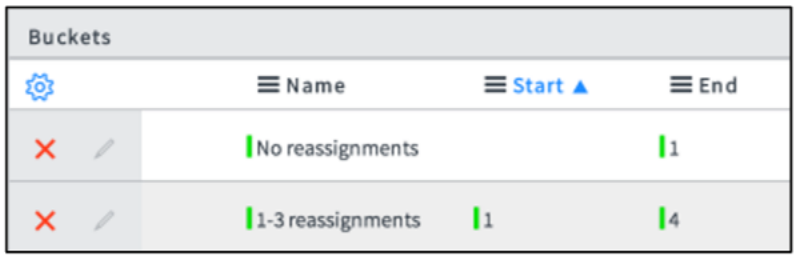


Buckets

	Name	Start ▲	End
✗	No reassignments		1

6. Double-click in the **Insert a new row...** field to insert one more new row. Edit as follows:

   - Name: **1-3 reassignments**
   - Start: **1**
   - End: **4**

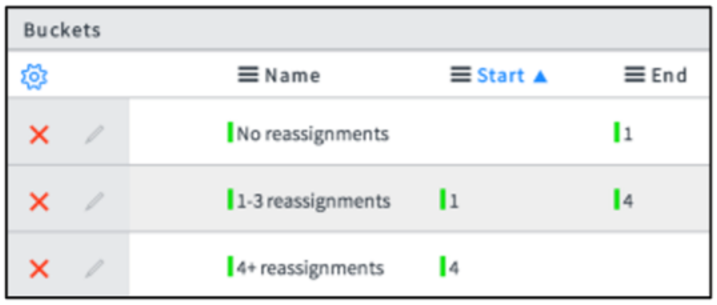


Buckets

	Name	Start ▲	End
✗	No reassignments		1
✗	1-3 reassignments	1	4

7. Double-click in the **Insert a new row...** field to insert one last new row. Edit as follows:

   - Name: **4+ reassignments**
   - Start: **4**
   - End: leave empty



Buckets

	Name	Start ▲	End
✗	No reassignments		1
✗	1-3 reassignments	1	4
✗	4+ reassignments	4	

8. Click **Submit**.

C. Create Breakdown Source based on Bucket Group

A Bucket group on its own is not enough. A Breakdown Source needs to be created that uses that Bucket Group as its Facts table.

1. Navigate to **Performance Analytics** > **Sources** > **Breakdown Sources**.

2. Click **New**.

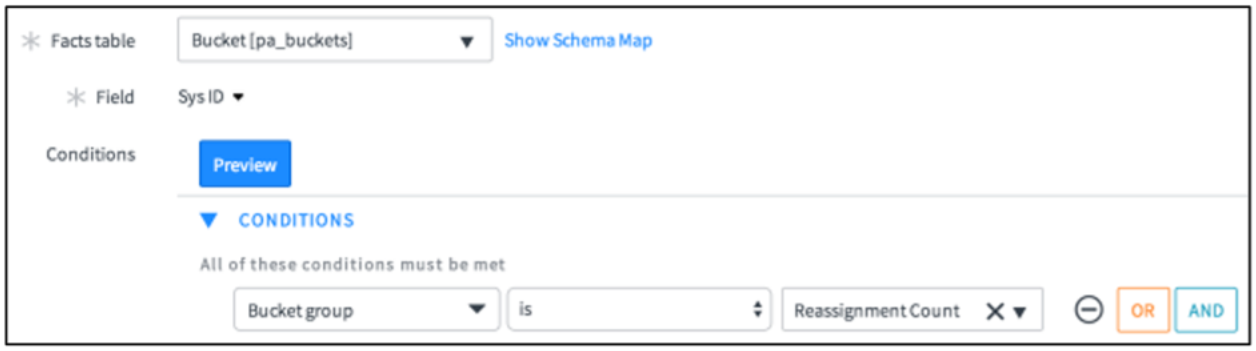
3. Set the name to **Reassignment Count**.

4. Complete the **Source** tab as shown:

   Facts table: **Bucket [pa_buckets]**

   Field: **Sys ID**

   Conditions: **Bucket group | is | Reassignment Count**



* Facts table: Bucket [pa_buckets]

* Field: Sys ID

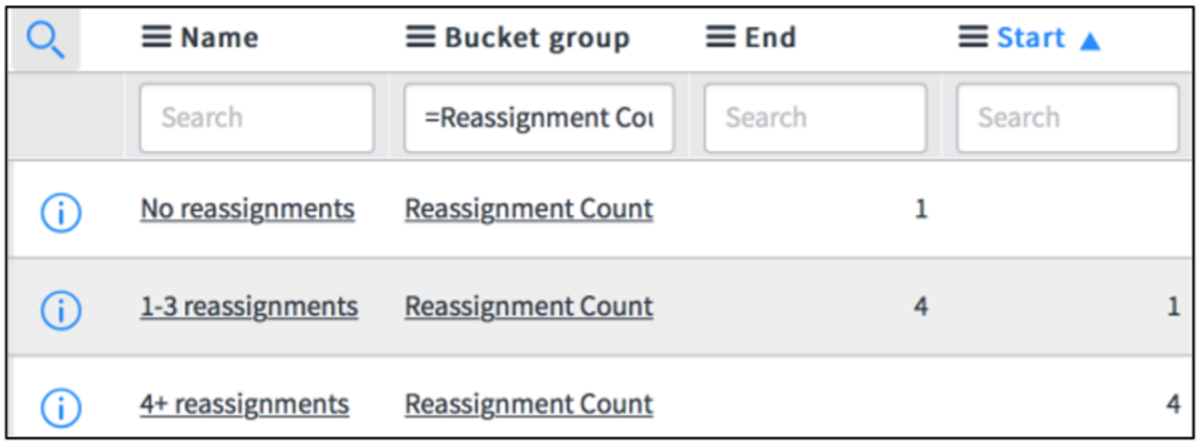
Conditions

All of these conditions must be met

Bucket group | is | Reassignment Count

5. Click the **Preview** button.

6. Click the **3 records match condition** link to confirm that it contains all defined buckets. This action opens a list in a new tab.



	Name	Bucket group	End	Start ▲
	Search	=Reassignment Coı	Search	Search
ⓘ	No reassignments	Reassignment Count	1	
ⓘ	1-3 reassignments	Reassignment Count	4	1
ⓘ	4+ reassignments	Reassignment Count		4

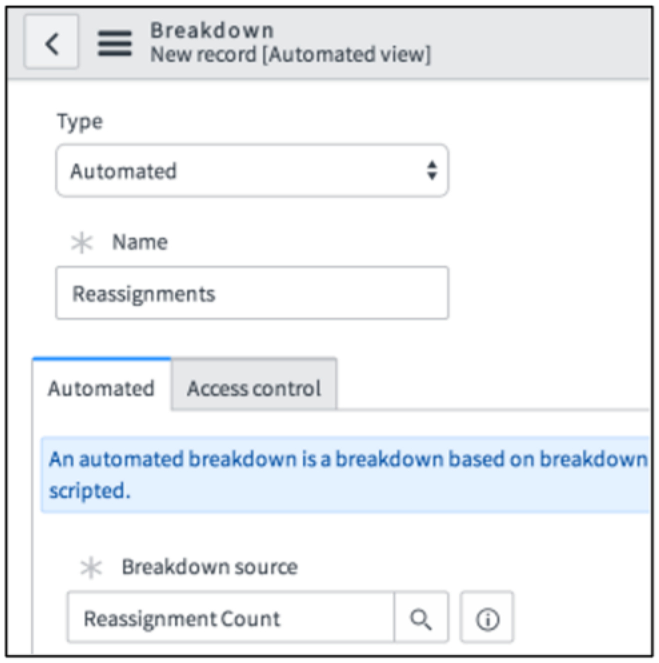
7. Close the new browser tab to return to your Breakdown Source.

8. Click **Submit**.

D. Automated Breakdown

Here you create a breakdown which uses the breakdown source just defined in order to categorize the incident reassignment count into one of the 3 reassignment count classes.

1. Navigate to **Performance Analytics** > **Breakdowns** > **Automated Breakdowns**.
2. Click **New** to begin creating a new breakdown.
3. Set the **Name** to **Reassignments**.
4. Set the **Breakdown source** field to **Reassignment Count**.



Breakdown
New record [Automated view]

Type: Automated

Name: Reassignments

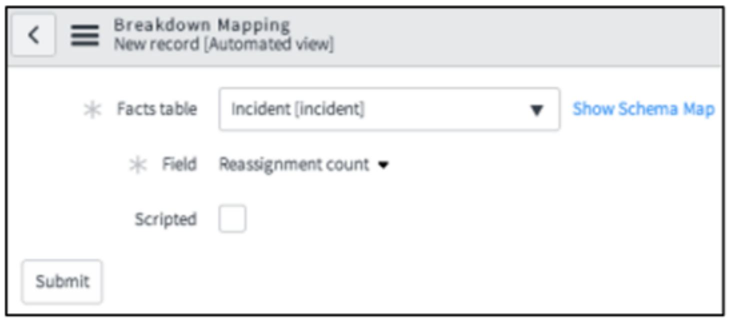
An automated breakdown is a breakdown based on breakdown
scripted.

Breakdown source: Reassignment Count

5. **Save** the Breakdown.
6. Navigate to the **Breakdown Mappings** Related List and click **New**.
7. Complete the form as shown:

Facts table: **Incident [incident]**

Field: **Reassignment count**



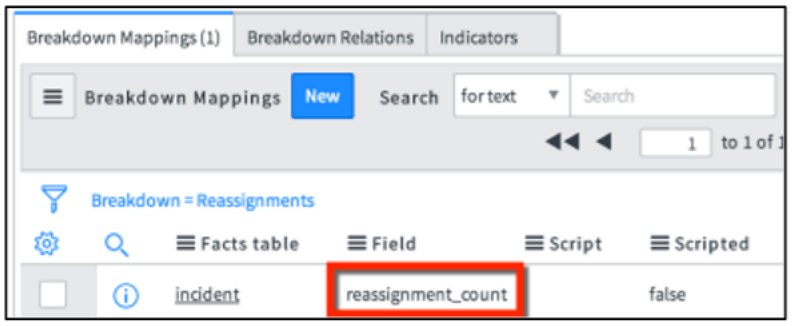
Breakdown Mapping
New record [Automated view]

Facts table: Incident [incident]

Field: Reassignment count

Scripted: □

8. Click **Submit**.
9. Confirm that you have correctly set the Breakdown Mapping field.

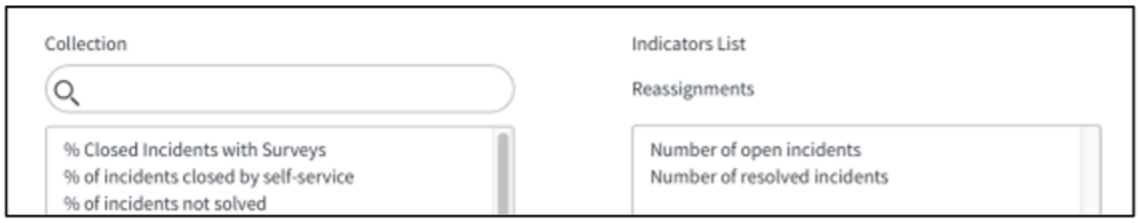


Breakdown = Reassignments

Facts table	Field	Script	Scripted
incident	reassignment_count		false

10. Navigate to the **Indicators** Related List and click **Edit**....

11. Add **Number of open incidents** and **Number of resolved incidents.**



Collection

% Closed Incidents with Surveys
% of incidents closed by self-service
% of incidents not solved

Indicators List

Reassignments

Number of open incidents
Number of resolved incidents

12. Click **Save** to return to the Breakdown form.

E. One-Off Historic Data Collector Job

Now that the Bucket Group, Breakdown Source, and Breakdown are defined, you need to collect historical data for the indicators that use them. Because you do not want to overwrite any existing history, you collect data only for the **Reassignments** Breakdown with Breakdown Exclusions.

1. Navigate to **Performance Analytics** > **Data Collector** > **Jobs**.

2. Click **New** and complete the form as follows:

   Name: **Reassignment Count Historic Collection**
   Relative start: **3**
   Relative start interval: **months ago**
   Active: **[unchecked]**
   Run: **On Demand**

3. Select **Save** from the form context menu to save the Job configuration.

4. Click **Edit**... in the **Indicators** Related List.

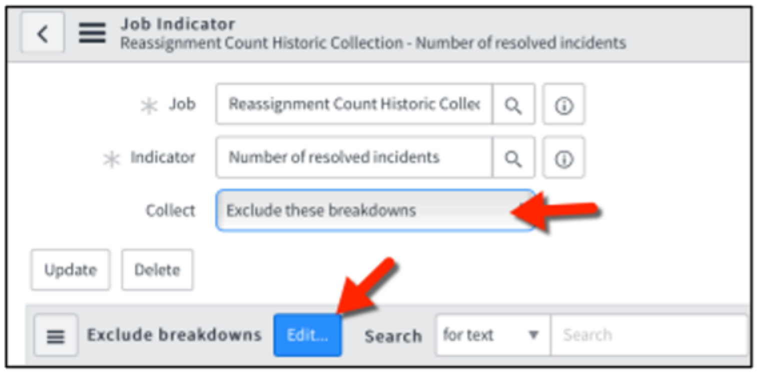
5. Add **Number of open incidents** and **Number of resolved incidents**.

6. Click **Save** to return to the job.

F. Breakdown Exclusions

By default, a Historic Data Collection job collects scores for *all* Breakdowns. If you only want to collect for the **Reassignments** breakdown, you need to exclude collecting the Indicator Scores as well as all other breakdowns.

1. While still on the historical job, navigate to the **Indicators** Related List.
2. Click to open **Number of resolved incidents** Job Indicator.
3. Uncheck the **Collect indicator** field.
4. In the **Collect** field, select **Exclude these breakdowns**.
5. Select **Save** from the form context menu.
6. Click **Edit**… in the **Exclude breakdowns** Related List.

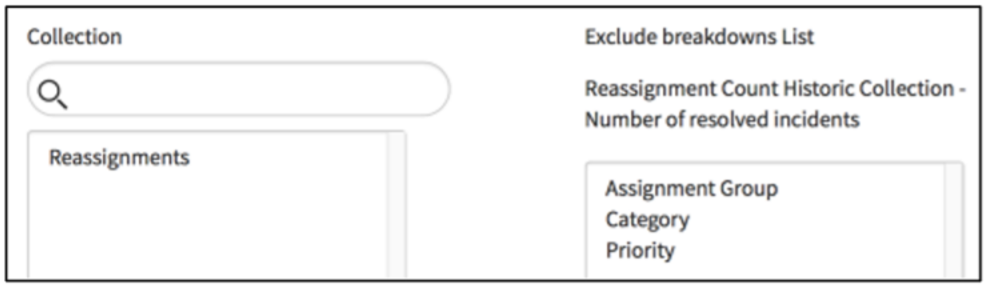

Job Indicator
Reassignment Count Historic Collection - Number of resolved incidents

Job: Reassignment Count Historic Collec

Indicator: Number of resolved incidents

Collect: Exclude these breakdowns

7. Add the **Assignment Group**, **Category**, and **Priority** Breakdowns to the **Exclude breakdowns** list:


Collection

Reassignments

Exclude breakdowns List

Reassignment Count Historic Collection - Number of resolved incidents

Assignment Group
Category
Priority

8. Make sure that **Reassignments** is the only breakdown that is not excluded and **Save**.
9. Click **Update** twice to return to the Job form.
10. Open the **Number of open incidents** Job Indicator.
11. Uncheck the **Collect indicator** field.
12. In the **Collect** field, select **Exclude these breakdowns**.

13. Select **Save** from the form context menu.

14. On the **Exclude breakdowns** tab, click **Edit…**.

15. Add the **Age, Assignment Group, Category, State,** and **Priority** Breakdowns to the **Exclude breakdowns** list.

16. Click **Save**, then click **Update** twice to return to the Job form.

17. Click **Execute Now.** Allow 1 -2 minutes for the Collection job to complete.

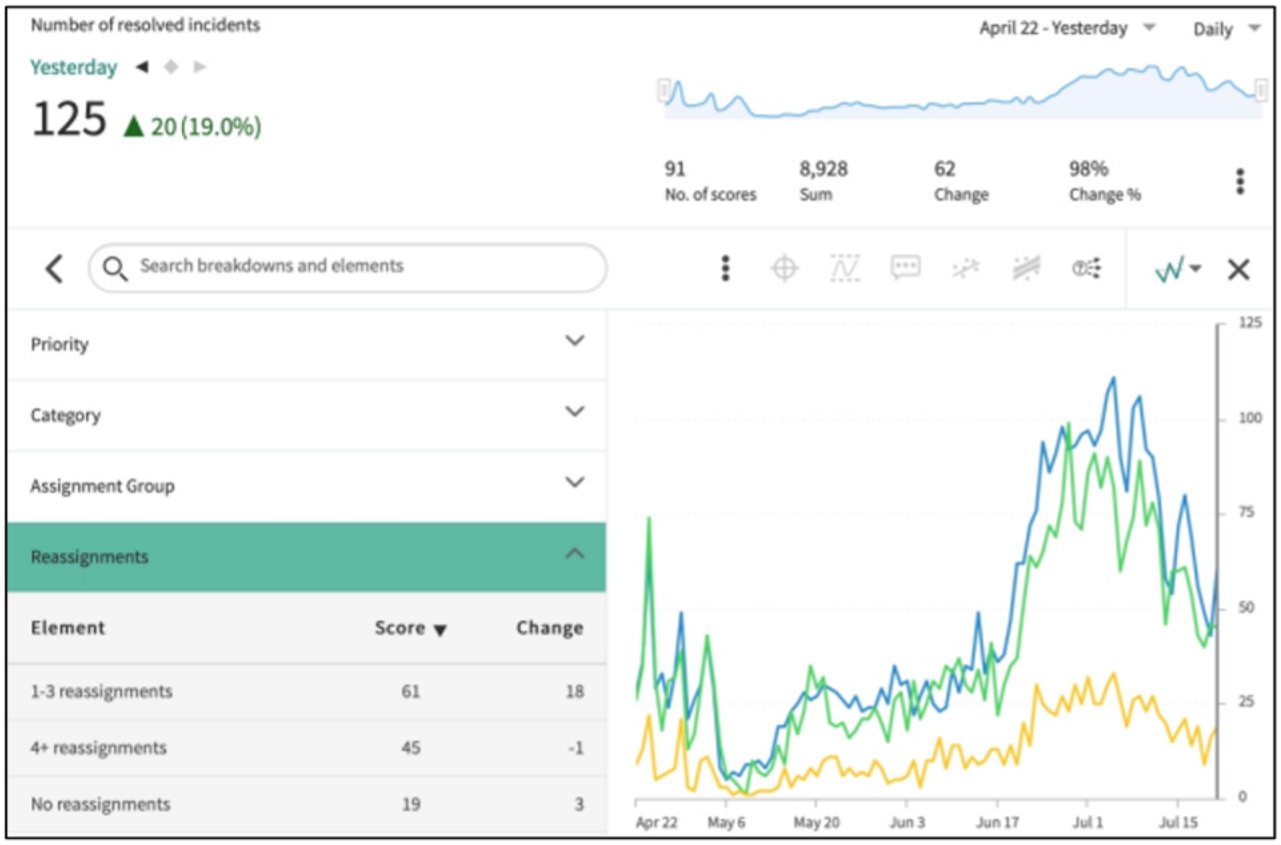
G. Breakdown Data Verification

1. Once collection has finished, navigate to **Performance Analytics > Analytics Hub**.

2. Open the **Number of resolved incidents** indicator.

3. Select the **Reassignments** Breakdown to review the results.

4. Confirm that you have the correct **Reassignment** breakdown elements *with data*.



Element	Score ▼	Change
1-3 reassignments	61	18
4+ reassignments	45	-1
No reassignments	19	3

5. Perform the same Breakdown data check for the **Number of open incidents** indicator.

You have now completed the Bucket Groups Lab.

Scripted Breakdown Mapping

Lab 2.2

⌛15m

Lab Objectives

The Cloud Dimensions Incident Process owner wants to simplify the categorization of incident resolution codes from the current seven classes to just three classes. Bucket groups are required along with a script to categorize the resolution codes of the incident records. In this lab, you create a new Bucket and a script for the Resolution codes.

A. Bucket Group

The Resolution Code breakdown needs to show scores in classes of Resolution Codes. These classes do not exist yet and need to be created as a new bucket group.

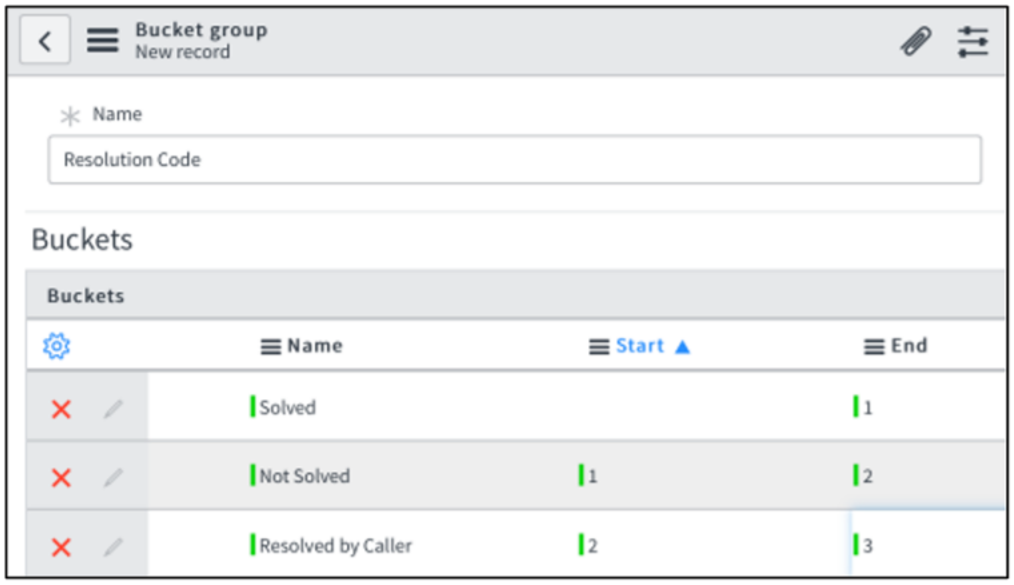
1. Navigate to **Performance Analytics** > **Breakdowns** > **Bucket Groups**

2. Click the **New** button to begin defining a new Bucket group.

3. Set the **Name** to **Resolution Code**.

4. Manually type in the following buckets:

Name: **Solved**		End: **1**
Name: **Not Solved**	Start: **1**	End: **2**
Name: **Resolved by Caller**	Start: **2**	End: **3**



Bucket group
New record

* Name
Resolution Code

Buckets

Buckets

Name	Start ▲	End
Solved		1
Not Solved	1	2
Resolved by Caller	2	3

5. **Submit** the new bucket group definition.

> ***Note:*** *This Bucket Group categorizes the output of the scripted mapping.*

B. Create a Breakdown Source for Closure Code

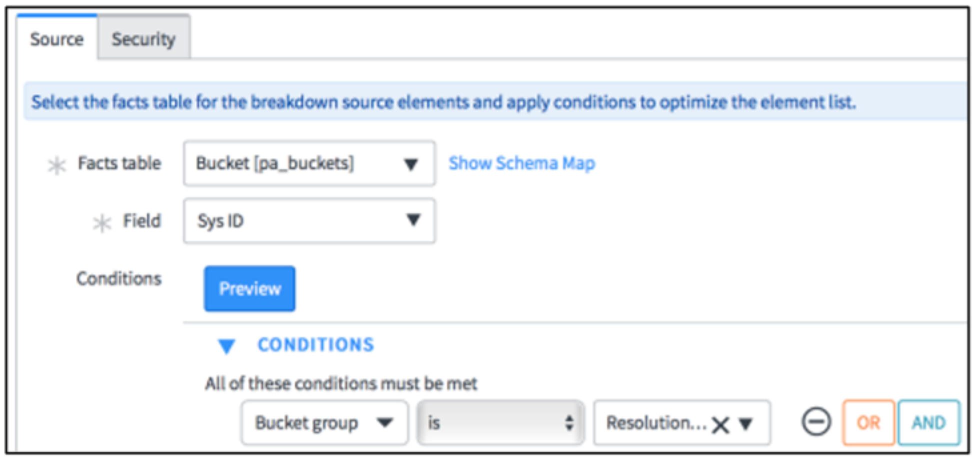
1. Navigate to **Performance Analytics** > **Sources** > **Breakdown Sources**.

2. Click the **New** button to begin defining a new breakdown source.

3. Set the name to **Resolution Code Classes**.

   Complete the **Source** tab as shown:

   Facts table: **Bucket [pa_buckets]**
   Field: **Sys ID**
   Conditions: **Bucket group | is | Resolution code**



Select the facts table for the breakdown source elements and apply conditions to optimize the element list.

Facts table: Bucket [pa_buckets]
Field: Sys ID
Conditions
All of these conditions must be met
Bucket group | is | Resolution...

4. Click the **Preview** button to view how many records match the condition.

5. Click the **3 records match conditions** link to view the actual records in a new tab.

6. Close the Buckets browser tab and **Submit** the new breakdown source.

C. Breakdown Mapping Script

Here you create a script that looks at the Incident closure code and categorizes it into one of three Bucket groups.

1. Navigate to **Performance Analytics > Automation > Scripts**.

2. Click **New**.

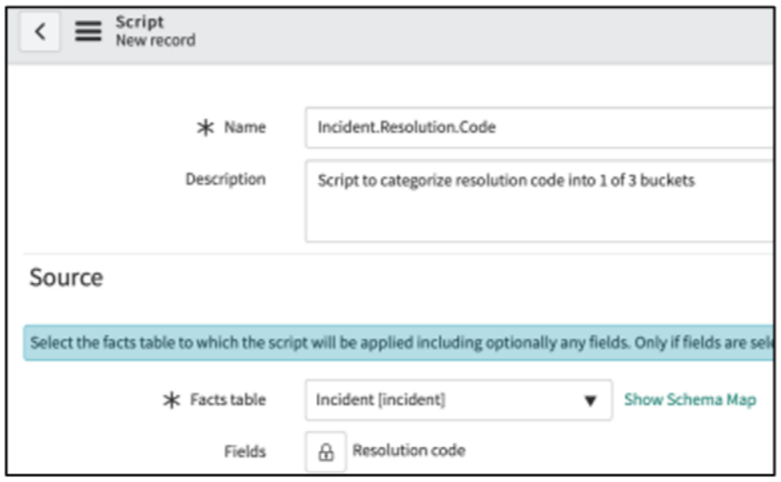
3. Create a new script as follows:

Name: **Incident.Resolution.Code**

Description: **Script to categorize resolution code into 1 of 3 buckets**

Facts table: **Incident [incident]**

Fields: **Resolution code**



Script
New record

✱ Name: Incident.Resolution.Code

Description: Script to categorize resolution code into 1 of 3 buckets

Source

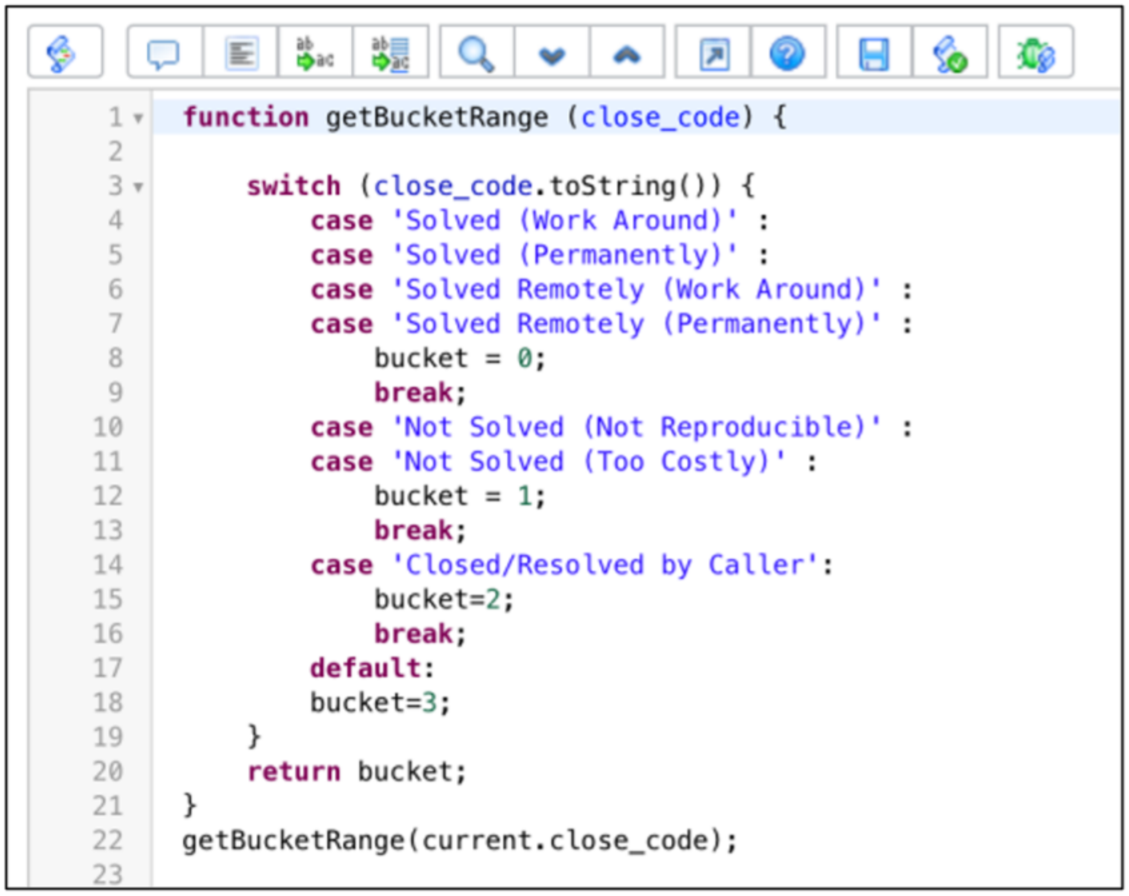
Select the facts table to which the script will be applied including optionally any fields. Only if fields are sel

✱ Facts table: Incident [incident] ▼ Show Schema Map

Fields: 🔒 Resolution code

4. Paste the Breakdown Mapping code provided by your instructor in the **Script** section.

> ***Tip:*** *The script can also be downloaded from the class Knowledge Base.*



```
function getBucketRange (close_code) {

    switch (close_code.toString()) {
        case 'Solved (Work Around)' :
        case 'Solved (Permanently)' :
        case 'Solved Remotely (Work Around)' :
        case 'Solved Remotely (Permanently)' :
            bucket = 0;
            break;
        case 'Not Solved (Not Reproducible)' :
        case 'Not Solved (Too Costly)' :
            bucket = 1;
            break;
        case 'Closed/Resolved by Caller':
            bucket=2;
            break;
        default:
        bucket=3;
    }
    return bucket;
}
getBucketRange(current.close_code);
```

5. **Submit** the new mapping script.

D. Automated Breakdown for Resolution Code

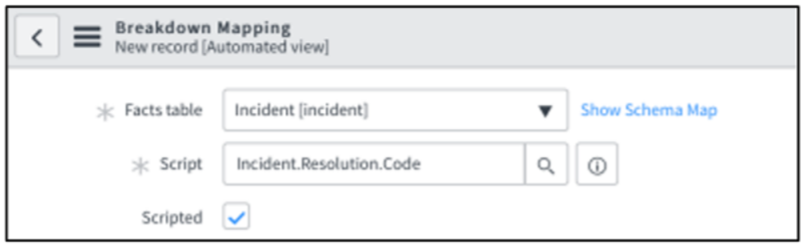
1. Navigate to **Performance Analytics** > **Breakdowns** > **Automated Breakdowns**

2. Click **New** and create the breakdown as follows:

- Name: **Resolution Code**
- Breakdown source: **Resolution Code Classes**

3. **Save** the breakdown to stay on the form.
4. Navigate to the **Breakdown Mappings** list and click **New**.
5. Define a scripted mapping to the incident tables as follows:

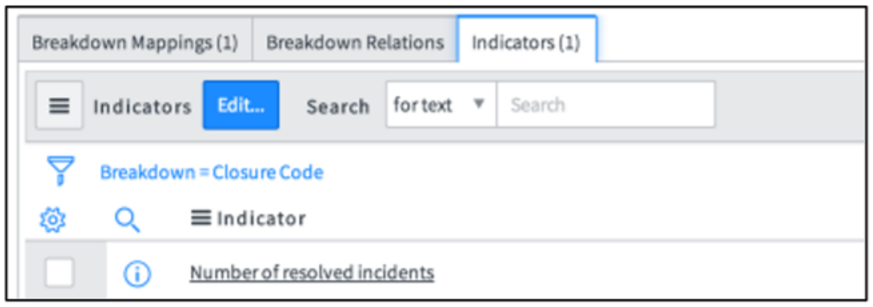
   Facts table: **Incident**
   Scripted: **Checked**
   Script:
   Incident.Resolution.Code



Breakdown Mapping
New record [Automated view]

Facts table	Incident [incident]
Script	Incident.Resolution.Code
Scripted	✓

6. **Submit** the new Mapping.
7. Navigate to the **Indicators** related list and click **Edit…**
8. Add the **Number of resolved incidents** indicator and click **Save**.



Breakdown = Closure Code

Indicator
Number of resolved incidents

E. Ad Hoc Historic Scores Collection

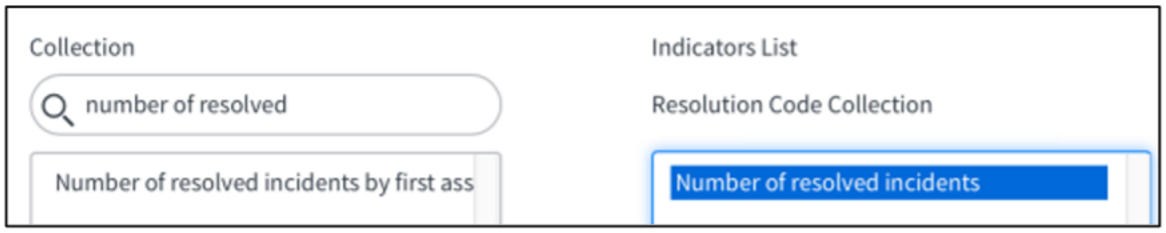
1. Navigate to the **Performance Analytics** > **Data Collector** > **Jobs**.
2. Click **New** and complete the form as follows:

   Name: **Resolution Code Collection**
   Description: **Temp collection for Resolution Code Breakdown**
   Relative start: **3**
   Relative start interval: **months ago**
   Active: [**unchecked**]
   Run: **On Demand**

3. Select **Save** from the form context menu.

4. Navigate to the **Indicators** Related List and click **Edit…**.

5. Add **Number of Resolved Incidents** to the Indicators List and click **Save**.



Collection	Indicators List
number of resolved	Resolution Code Collection
Number of resolved incidents by first ass	Number of resolved incidents

Tip: As a best practice, consider Breakdown exclusions so no historical data is overwritten.

6. Click **Execute Now** to execute the collection job immediately.

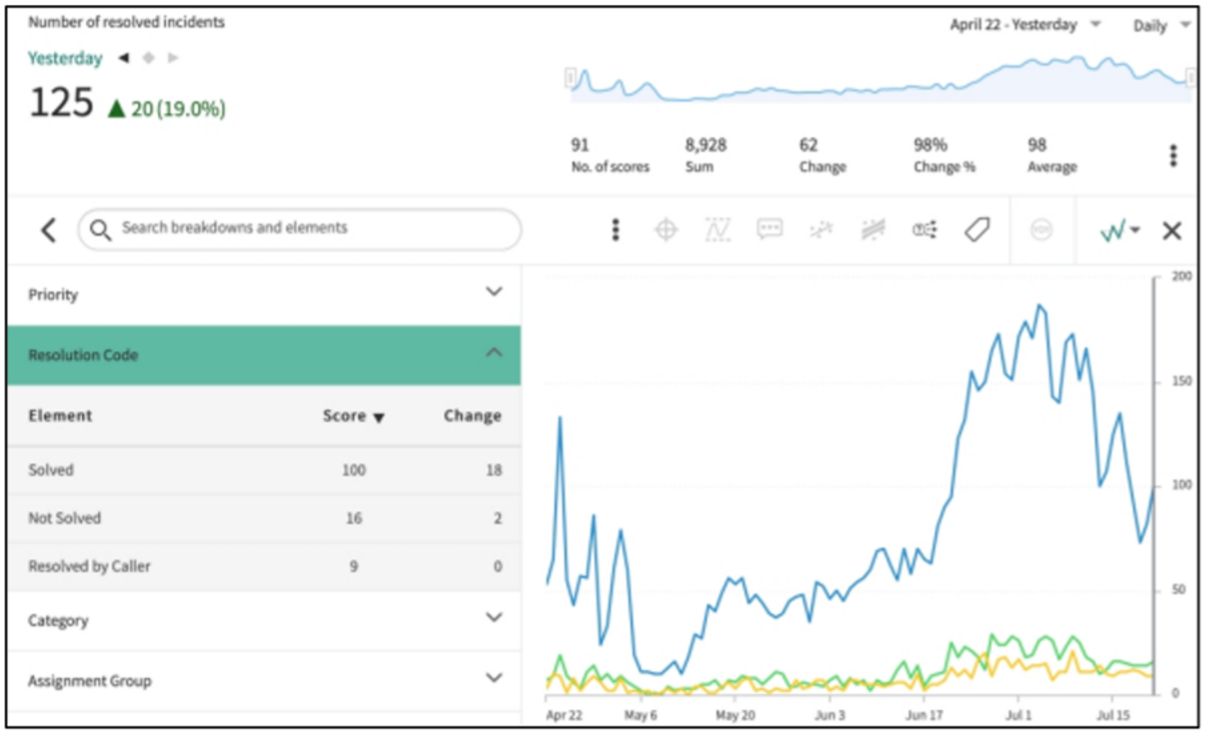
F. Breakdown Data Verification

1. Navigate to the **Job Logs** Related List.

2. Once the job completes, open the job log and make sure there are no script errors.

3. Navigate to the **Analytics Hub** and view the **Number of resolved incidents** indicator.

4. View the data by the **Resolution Code** breakdown.



Number of resolved incidents

April 22 - Yesterday · Daily

Yesterday

125 ▲ 20 (19.0%)

91 No. of scores · 8,928 Sum · 62 Change · 98% Change % · 98 Average

Search breakdowns and elements

Priority

Resolution Code

Element	Score ▼	Change
Solved	100	18
Not Solved	16	2
Resolved by Caller	9	0

Category

Assignment Group

Tip: If no data, check the Job and System Logs for errors and troubleshoot the Mapping Script.

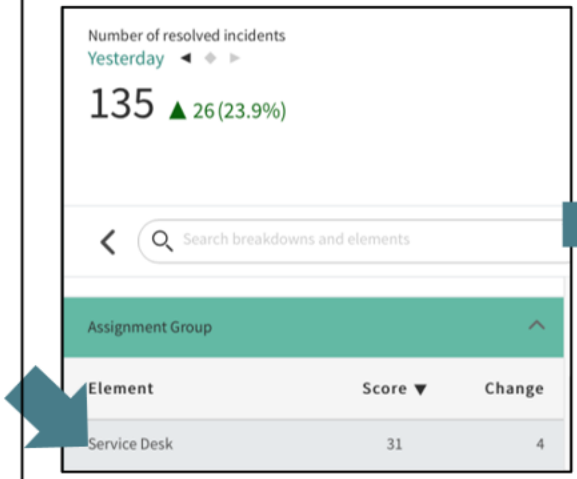
You have now completed the Scripted Breakdown Mapping Lab.

Breakdown Relation Defined

Navigational Link between Two Breakdowns

Example: Navigate from a Group to its Child Groups

#1 1st Breakdown: **Assignment Group**



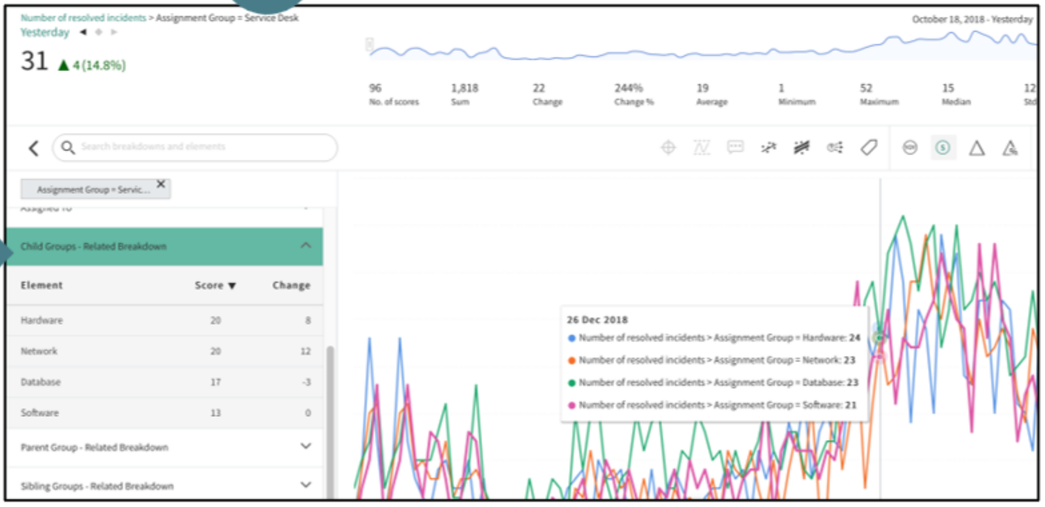
Number of resolved incidents
Yesterday ◄ ◆ ►
135 ▲ 26 (23.9%)

Search breakdowns and elements

Assignment Group

Element	Score ▼	Change
Service Desk	31	4

#2 Related Breakdown: **Child Groups**



Number of resolved incidents > Assignment Group = Service Desk
Yesterday ◄ ◆ ►
31 ▲ 4 (14.8%)

October 18, 2018 - Yesterday

96	1,818	22	244%	19	1	52	15	12
No. of scores	Sum	Change	Change %	Average	Minimum	Maximum	Median	Std

Search breakdowns and elements

Assignment Group = Servic... ✕

Child Groups - Related Breakdown

Element	Score ▼	Change
Hardware	20	8
Network	20	12
Database	17	-3
Software	13	0

Parent Group - Related Breakdown

Sibling Groups - Related Breakdown

26 Dec 2018
- Number of resolved incidents > Assignment Group = Hardware: 24
- Number of resolved incidents > Assignment Group = Network: 23
- Number of resolved incidents > Assignment Group = Database: 23
- Number of resolved incidents > Assignment Group = Software: 21

Breakdown relations define how two Breakdowns are related to each other and provide an additional way to navigate within Dashboards and Scorecards. With Breakdown relations, you can navigate from one breakdown element to another breakdown element.

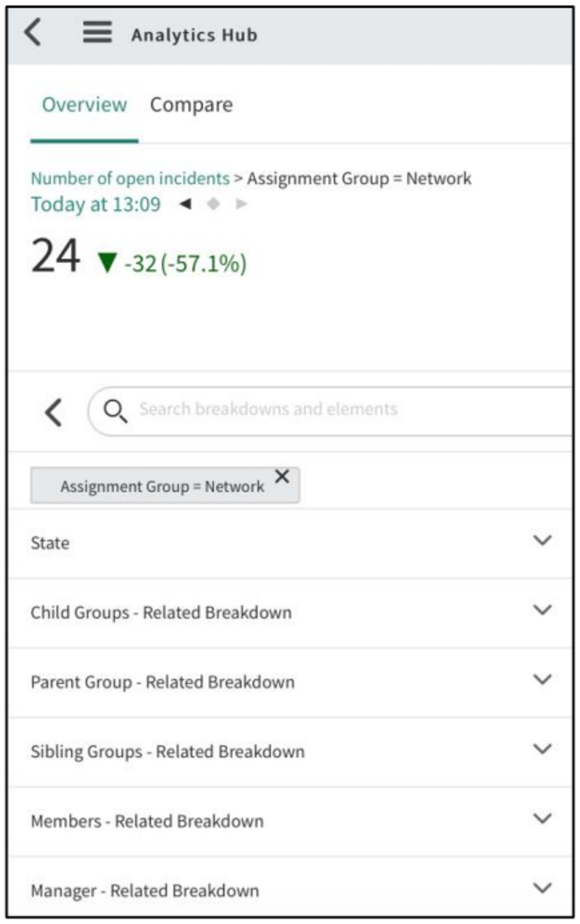
The example above creates a navigational path between two different breakdowns as follows:

- Open the Number of resolved incidents scorecard
- View incidents assigned to the **Service Desk** assignment group
- The following **Related breakdowns** become available to select as a 2nd level breakdown:
  - Child Groups – Related Breakdown
  - Parent Group– Related Breakdown
  - Sibling Groups – Related Breakdown
- Selecting the Child Groups breakdown allows you to see Incidents resolved by the child groups of Service Desk – **Hardware**, **Software**, **Network**, and **Database.**

Breakdown Relation Features

- Allow navigation beyond the 1st breakdown level using a relationship
- Enable multi-level drilldowns
- No limit to levels
- Rely on a defined data relationship
- Use current data hierarchy:
  - Location
  - Group Member
  - Business Services

Related Breakdowns display under Breakdowns and have similar drilldown behavior



Analytics Hub

Overview Compare

Number of open incidents > Assignment Group = Network

Today at 13:09

24 ▼ -32 (-57.1%)

Search breakdowns and elements

Assignment Group = Network ✕

State

Child Groups - Related Breakdown

Parent Group - Related Breakdown

Sibling Groups - Related Breakdown

Members - Related Breakdown

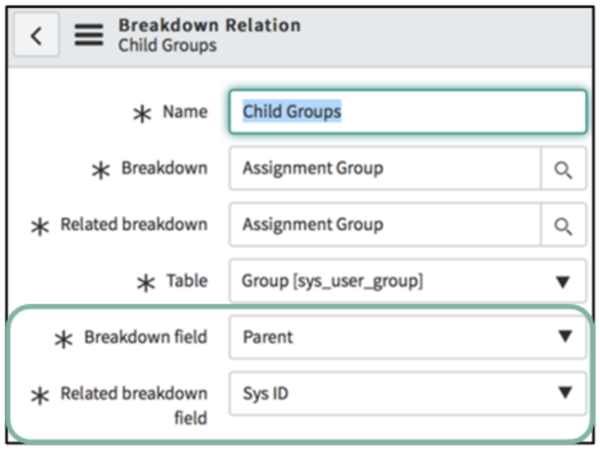
Manager - Related Breakdown

Breakdown Relations enrich Element navigation in Scorecards and Widgets. Once you select a 1st level breakdown element, applicable "Related" Breakdowns become available, if defined. This allows to jump from one Breakdown to another Breakdown emulating a multi-level drill-down.

Parent Child Breakdown Relation

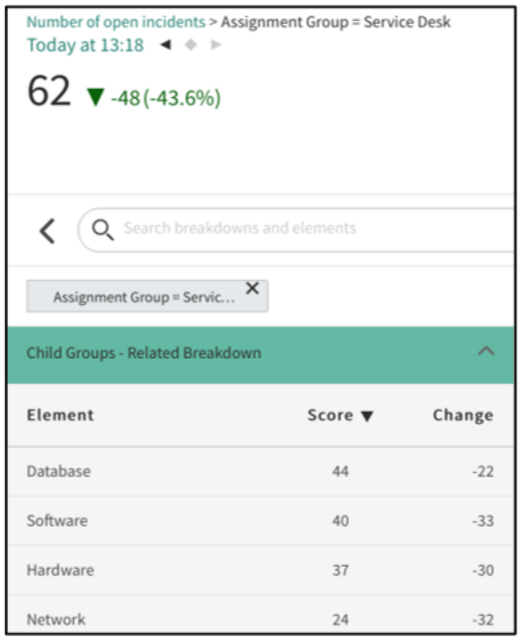
Select an initial breakdown then analyze by its CHILD records

There MUST be a Parent - Child relationship between the Breakdown and the Related Breakdown field



Breakdown Relation
Child Groups

* Name	Child Groups
* Breakdown	Assignment Group
* Related breakdown	Assignment Group
* Table	Group [sys_user_group]
* Breakdown field	Parent
* Related breakdown field	Sys ID



Number of open incidents > Assignment Group = Service Desk
Today at 13:18

62 ▼ -48 (-43.6%)

Search breakdowns and elements

Assignment Group = Servic... ✕

Child Groups - Related Breakdown

Element	Score ▼	Change
Database	44	-22
Software	40	-33
Hardware	37	-30
Network	24	-32

A Parent-Child Breakdown Relation specifies the same Breakdown object for the Breakdown and related Breakdown field.

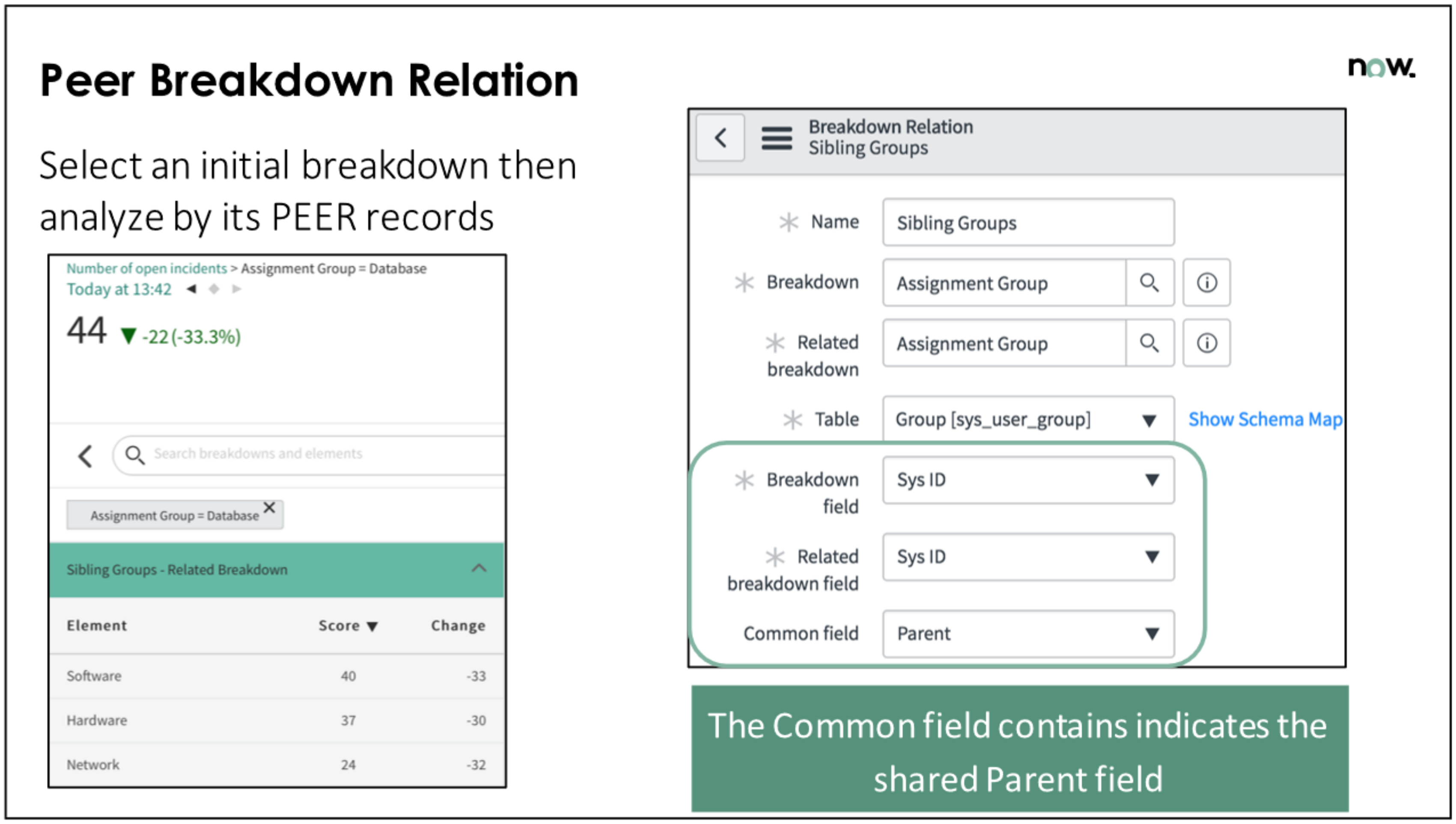
Here are some common Parent-Child Examples:

Location Structure, Management Chain, Category / Sub category, Nested Groups, Nested CMDB Configuration Items (CIs)

The Parent – Child Breakdown relation has these characteristics:

- Breakdown and Related Breakdown reference the same Breakdown
- The Breakdown field is the Parent Breakdown record
- The Related Breakdown is the Sys ID of the child Breakdown

To create a relationship that allows navigation in the opposite direction, from Child to Parent, you need to create another Breakdown Relation with the Breakdown and Related Breakdown fields swapped.



Peer Breakdown Relation

Select an initial breakdown then analyze by its PEER records

Number of open incidents > Assignment Group = Database
Today at 13:42

44 ▼ -22 (-33.3%)

Search breakdowns and elements

Assignment Group = Database ✕

Sibling Groups - Related Breakdown

Element	Score ▼	Change
Software	40	-33
Hardware	37	-30
Network	24	-32

Breakdown Relation
Sibling Groups

* Name	Sibling Groups
* Breakdown	Assignment Group
* Related breakdown	Assignment Group
* Table	Group [sys_user_group]
* Breakdown field	Sys ID
* Related breakdown field	Sys ID
Common field	Parent

The Common field contains indicates the shared Parent field

Here are the key characteristics of the Peer Breakdown Relation:

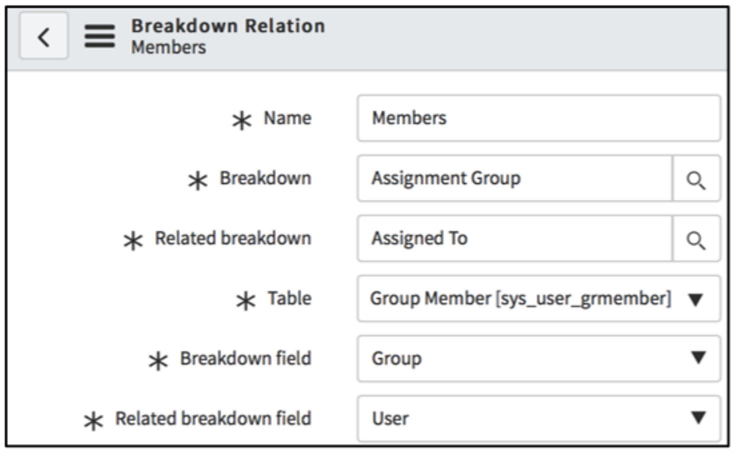
- The Common field is the field that makes the elements Peers, e.g. both Assignment Group Breakdown elements have the same Parent.
- The Original element will be excluded from the list of Sibling elements
- The Breakdown and Related Breakdown contain the same Breakdown
- The Breakdown field and the Related breakdown field are both Sys ID

Some Peer examples are:

- Other Locations in Region
- Peers working for the same Manager
- Colleagues working in the same Location
- Peer Groups

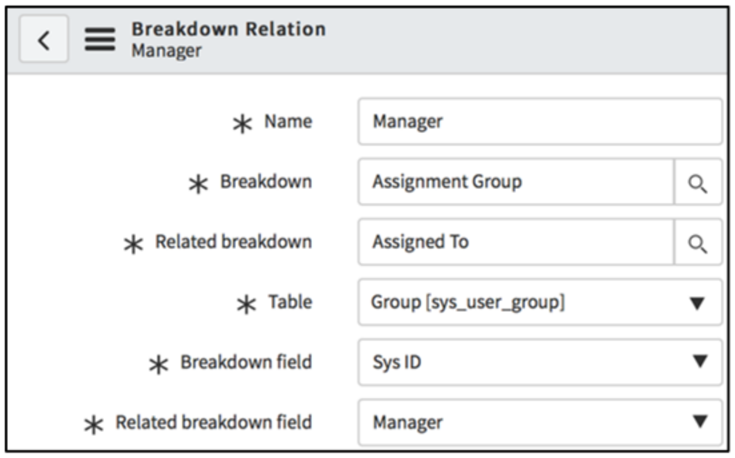
Member Breakdown Relation

- Relates *two different* breakdowns
- Uses the table of one of the breakdowns or a M2M table
- Breakdown field and Related breakdown are the referenced fields in a M2M
- Examples:
  - – Users in a Group
  - – Groups that a User belongs to
  - – Users in a Location
  - – Category / Sub Category
  - – Any choice list with a dependent field



Breakdown Relation
Members

* Name	Members
* Breakdown	Assignment Group
* Related breakdown	Assigned To
* Table	Group Member [sys_user_grmember]
* Breakdown field	Group
* Related breakdown field	User

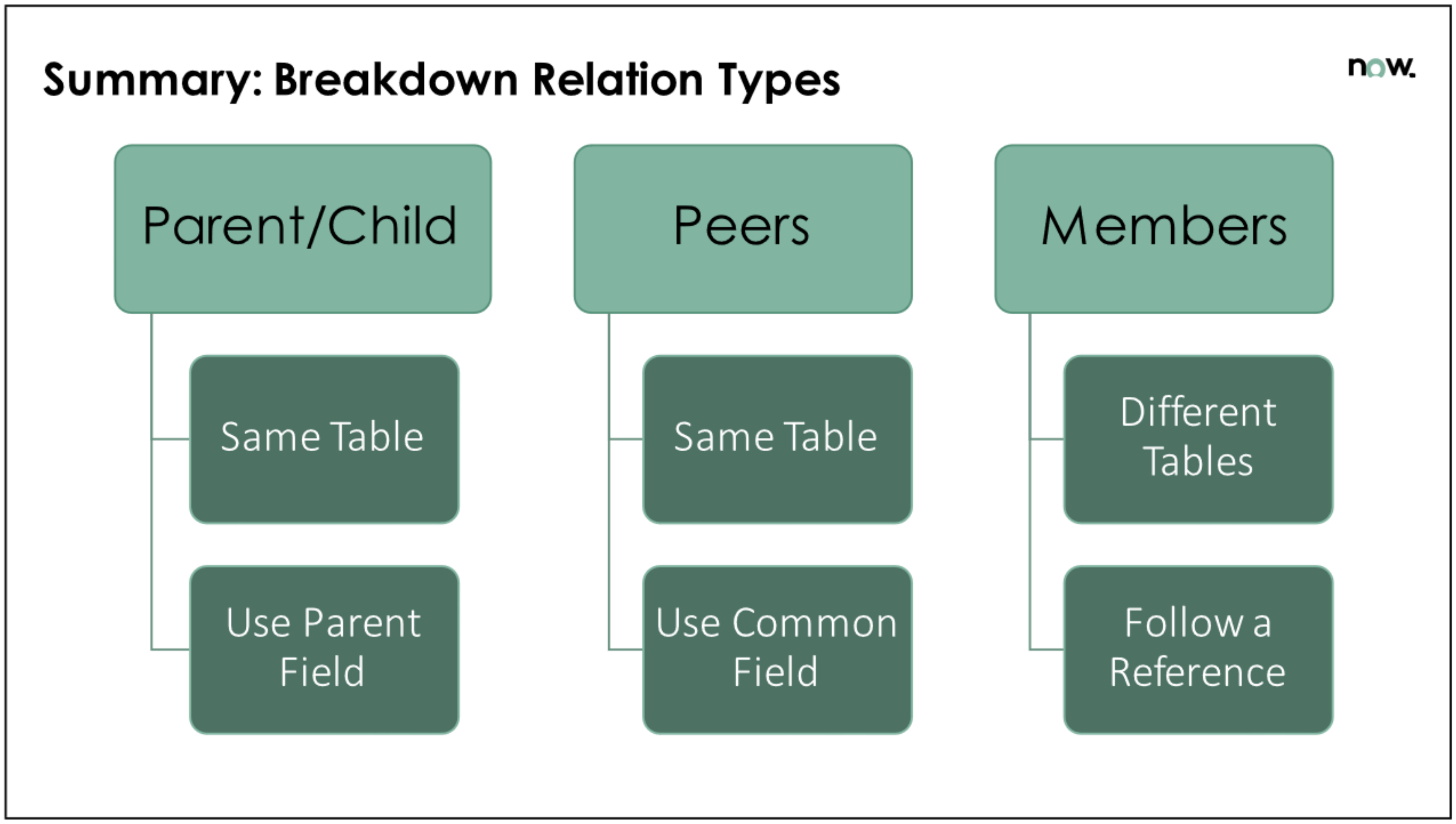


Breakdown Relation
Manager

* Name	Manager
* Breakdown	Assignment Group
* Related breakdown	Assigned To
* Table	Group [sys_user_group]
* Breakdown field	Sys ID
* Related breakdown field	Manager

The Member Breakdown relation differs from the Parent Child and Peer Relation in that it tells you how to navigate from one table to another table by specifying:

- Breakdown field – the starting point, Assignment Group in both examples shown here
- Related breakdown – the end point, Assigned To for both Manager and Members relations
- The Table that holds the relationship (the M2M table)
  - For Members, the Group Member table is used to determine all members of a group
  - For Manager, the Group table is used to determine the group's manager



Summary: Breakdown Relation Types

Parent/Child
- Same Table
- Use Parent Field

Peers
- Same Table
- Use Common Field

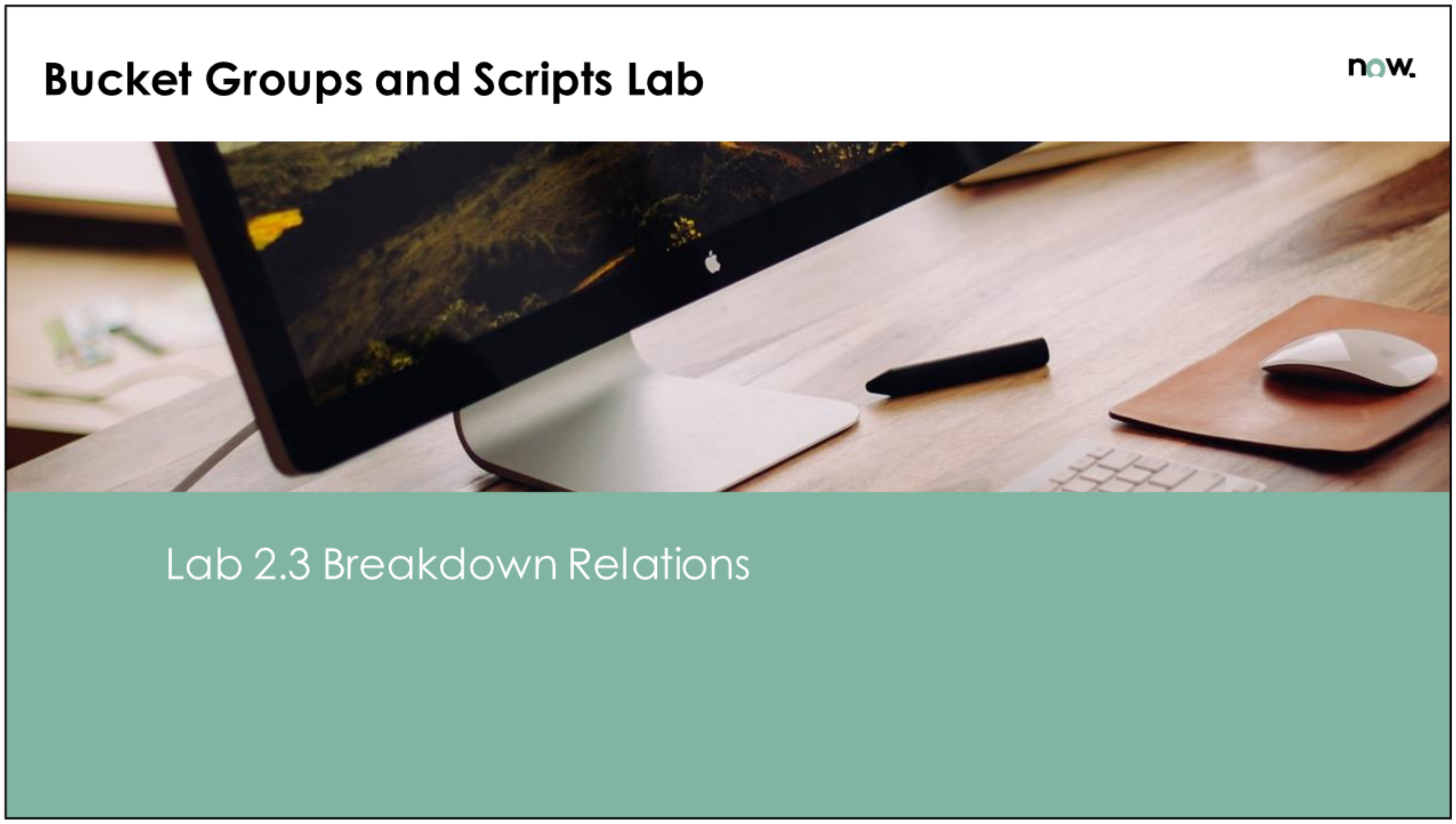
Members
- Different Tables
- Follow a Reference

Depending on the Breakdown element structure, the following navigation modes are possible:

- Parent Child: Navigate from a Parent to a Child in a hierarchy or from a Child to a Parent.
- Peers: Navigate from one element in the hierarchy to its peers (other elements at the same level)

In the above two cases, the elements are Groups, or Locations, or CIs stored in the same table. There is usually a Parent element that denotes the parent-child relationship.

- Members: When the elements of the hierarchy are not stored in the same table, there is a relationship or a reference field that can be followed. For instance, to navigate from the Assignment group to all Assignees that are its members, there has to be logic that specifies the table or the reference field that contains this relationship.



Bucket Groups and Scripts Lab

Lab 2.3 Breakdown Relations

2.3 Breakdown Relations Lab

- Create an Assignee to Assignment Group Breakdown Relation
- Create an Assignment Group to Assignee Breakdown Relation

Breakdown Relations

Lab

2.3

⌛15m

Lab Objectives

The Cloud Dimensions Incident process owner would like to navigate from the individual Assignees' scores to the scores for Assignment Groups that they are members of. To do this, you perform the following:

- Create a Breakdown Relation to navigate from Assignee to Assignment Group

A. Breakdown Source and Breakdown

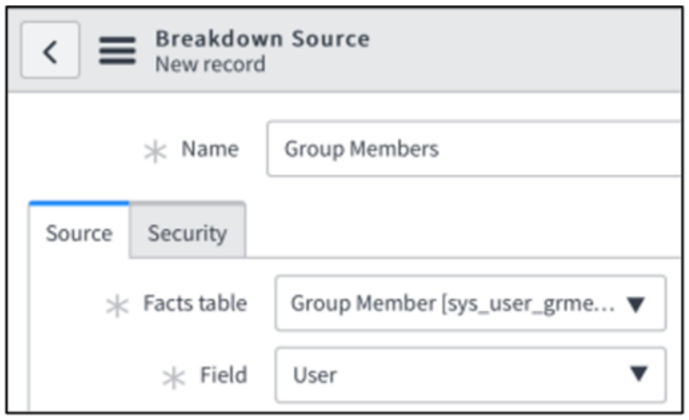
In this step, you create a breakdown source to represent all users that are members of one or more assignment groups.

1. Navigate to **Performance Analytics > Sources > Breakdown Sources**.
2. Click **New** to create a new breakdown source as follows:

   Name: **Group Members**

   Facts table: **Group Member [sys_user_grmember]**

   Field: **User**


Breakdown Source
New record

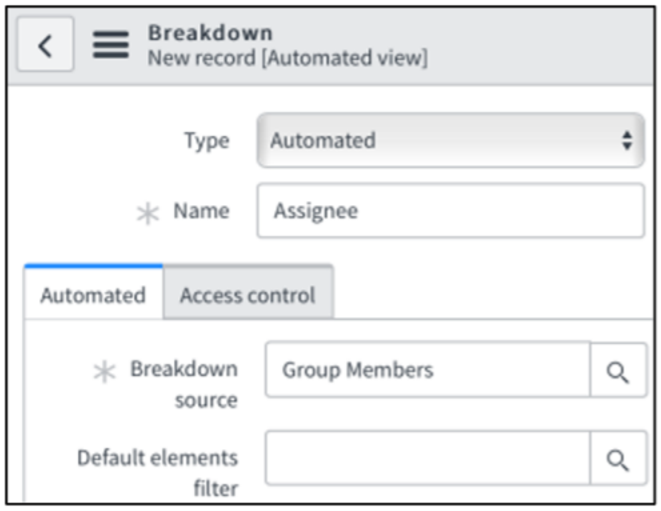
✱ Name: Group Members

✱ Facts table: Group Member [sys_user_grme... ▼

✱ Field: User ▼

3. Click **Submit**.
4. Navigate to **Performance Analytics > Breakdowns > Automated Breakdowns**.
5. Click **New** and define the new breakdown as follows:

Name: **Assignee**
Breakdown Source: **Group.Members**



Type: Automated

* Name: Assignee

* Breakdown source: Group Members

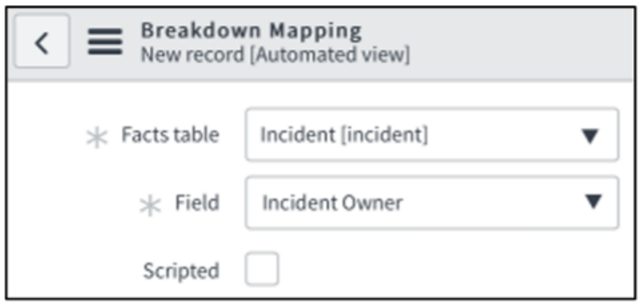
Default elements filter:

6. **Save** the new breakdown to stay on the form.

7. Navigate to the **Breakdown Mappings** Related list.

8. Click **New** and define the new mapping as follows:

Facts table: **Incident [incident]**
Field: **Incident Owner**

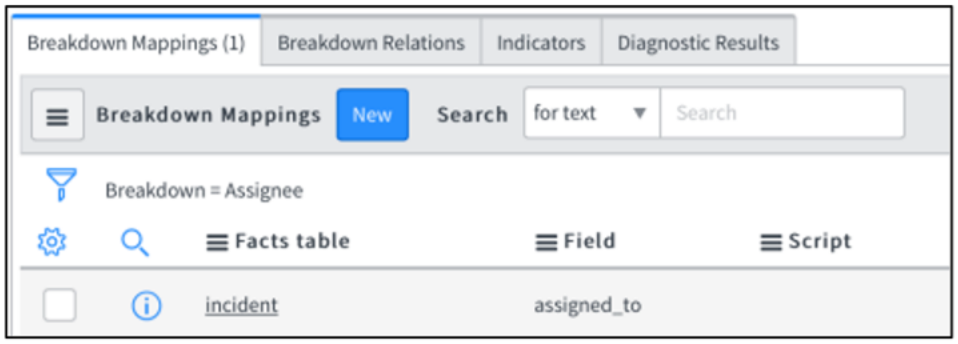


* Facts table: Incident [incident]

* Field: Incident Owner

Scripted: □

9. Click **Submit** to return to the Breakdown form and validate correct mapping.



Breakdown = Assignee

Facts table	Field	Script
incident	assigned_to	

Note: Incident Owner is the Column label. The Column name is assigned_to.

10. Navigate to the **Indicators** Related list and click **Edit...**

11. Add the **Number of resolved incidents** to the Indicators List.

12. Click **Save** to return and stay on the **Assignee** Breakdown form.

B. Assignee to Assignment Group Breakdown Relation

In order to navigate from the **Assignee** breakdown to its **Assignment Groups**, you need a breakdown relation on the **Assignment Group** breakdown.

1. In the **Assignee** Breakdown form, navigate to the **Breakdown Relations** related list.

2. Click **New** to begin creating a relation.

3. Define the Breakdown Relation as follows:

   Name: **Assignment Group(s)**

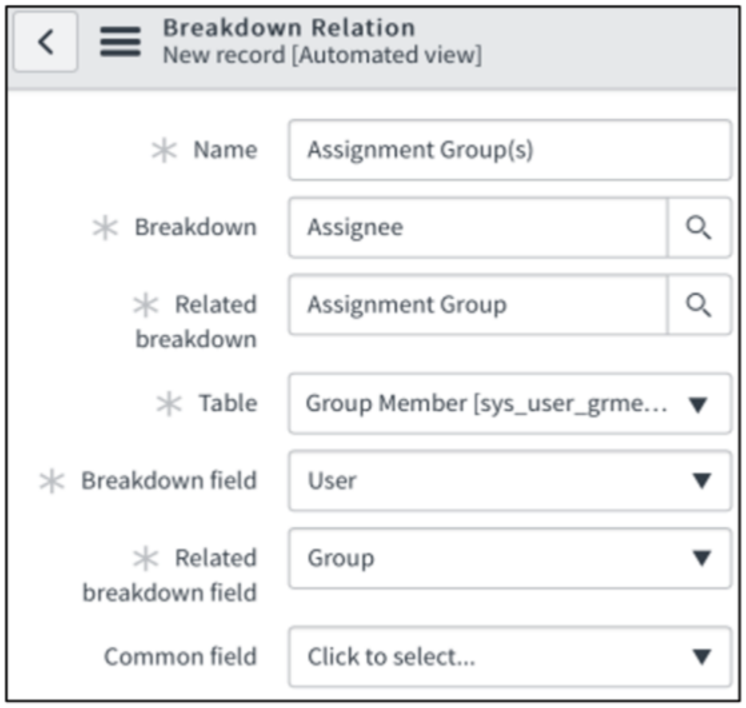
   Breakdown: **Assignee**

   Related Breakdown: **Assignment Group**

   Table: **Group Member** (sys_user_grmember)

   Breakdown field: **User**

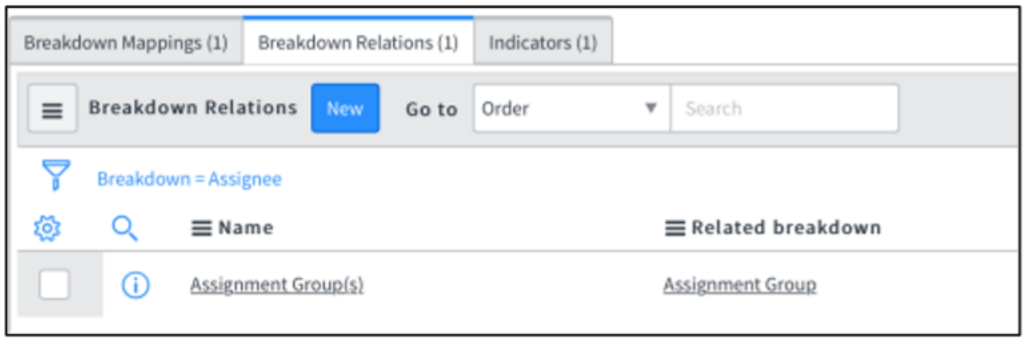
   Related Breakdown field: **Group**



Breakdown Relation
New record [Automated view]

* Name	Assignment Group(s)
* Breakdown	Assignee
* Related breakdown	Assignment Group
* Table	Group Member [sys_user_grme...
* Breakdown field	User
* Related breakdown field	Group
Common field	Click to select...

4. Click **Submit** and confirm that a new Breakdown Relation is created.



Breakdown Mappings (1) | Breakdown Relations (1) | Indicators (1)

Breakdown Relations New Go to Order Search

Breakdown = Assignee

Name	Related breakdown
Assignment Group(s)	Assignment Group

C. Collect New Breakdown Data

In this step, you run data collection to populate the new **Assignee** breakdown.

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Click **New** to create the following job:

   Name: **One Time Assignee Breakdown Collection**
   Relative start: **3**
   Relative start interval: **months ago**
   Active: **unchecked**
   Run: **On Demand**

3. **Save** the job and navigate to the **Indicators** related list.

4. Click **Edit…** and add the **Number of resolved incidents** indicator.

5. Click **Save**.

   Tip: *Consider adding Breakdown exclusions so that historical data is not overwritten.*

6. Click **Execute Now**. Allow about a minute for the collection to complete.

D. Follow Breakdown Relations in the Analytics Hub

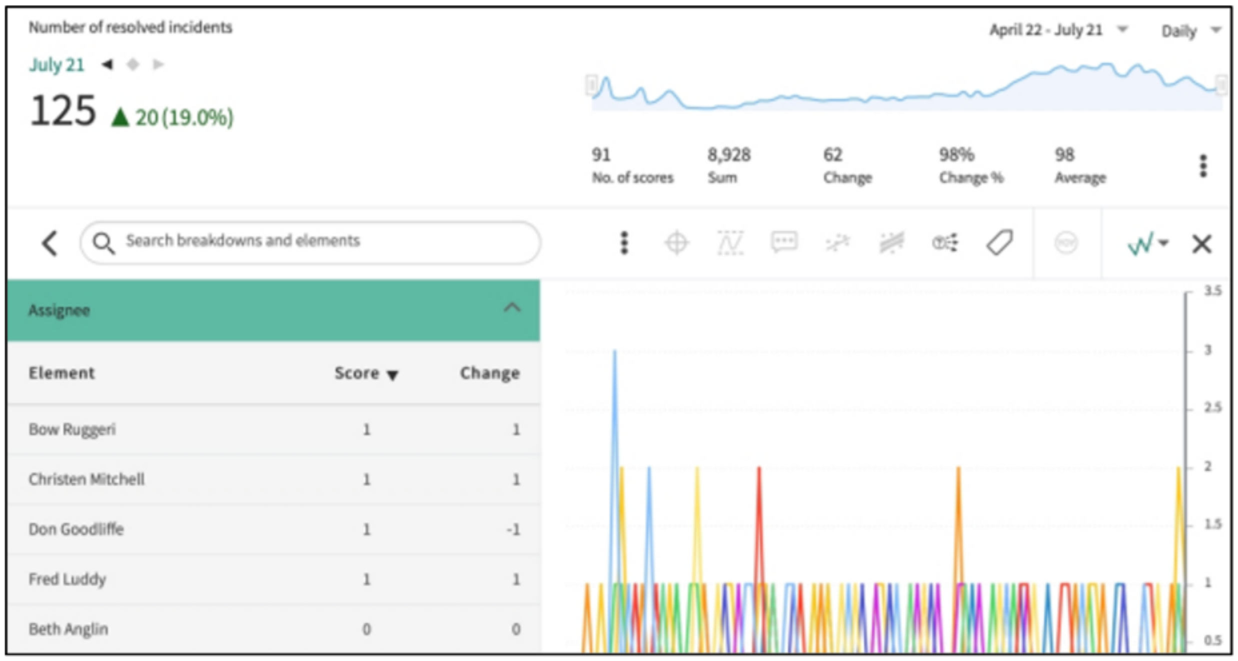
1. Check the **Job Logs** related list to make sure that the job has completed.

2. Navigate to **Performance Analytics > Analytics Hub**.

3. Click the **Favorites** (star) icon to view the recently accessed Indicators.

4. Open the **Number of resolved incidents** Indicator.

5. Expand the **Assignee** breakdown.



Number of resolved incidents

July 21

125 ▲ 20 (19.0%)

April 22 - July 21 Daily

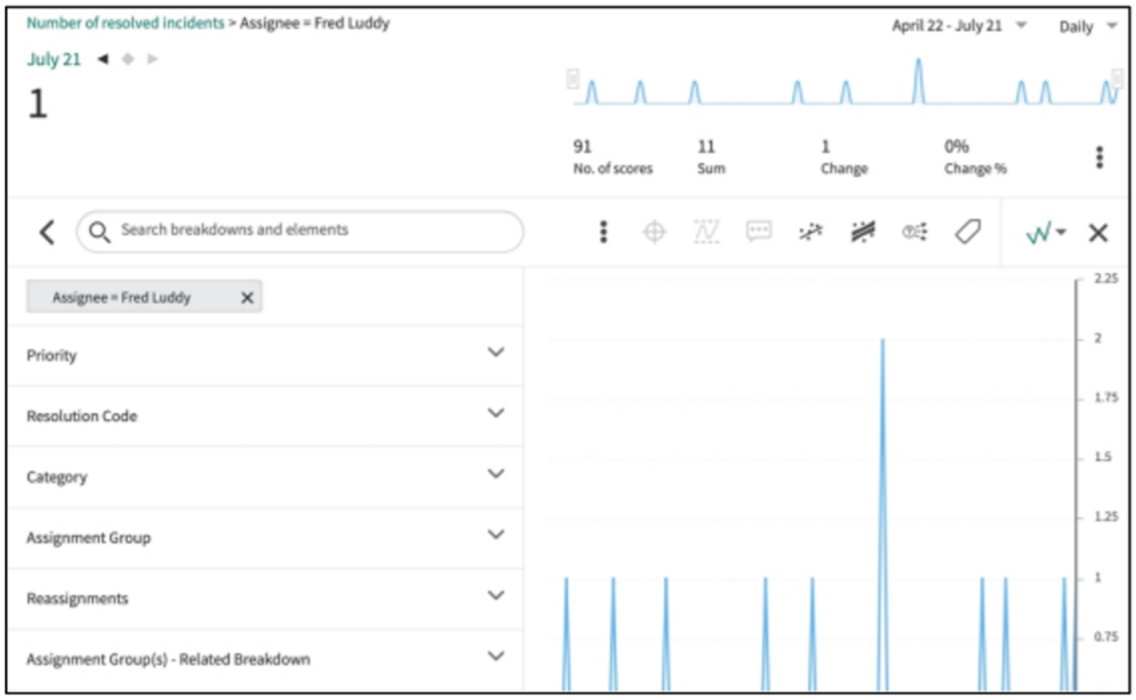
91 No. of scores | 8,928 Sum | 62 Change | 98% Change % | 98 Average

Search breakdowns and elements

Assignee

Element	Score ▼	Change
Bow Ruggeri	1	1
Christen Mitchell	1	1
Don Goodliffe	1	-1
Fred Luddy	1	1
Beth Anglin	0	0

6. Select the **Fred Luddy** Breakdown element to view only incidents assigned to Fred Luddy.



Number of resolved incidents > Assignee = Fred Luddy

July 21

1

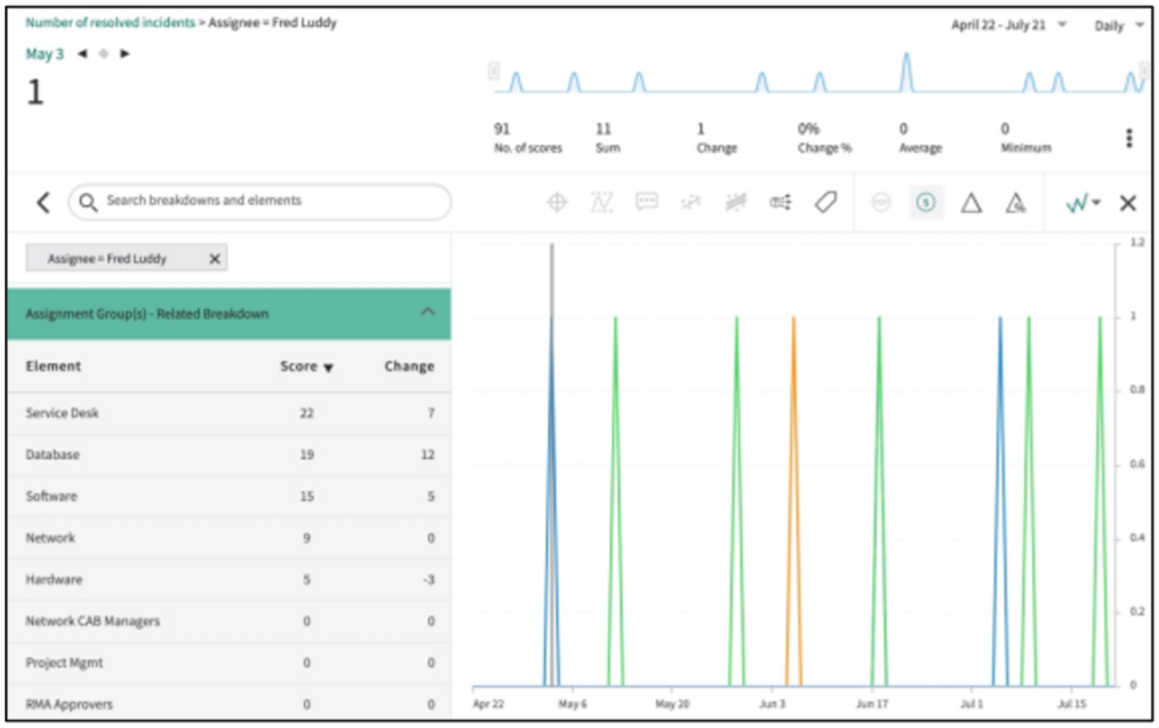
91	11	1	0%
No. of scores	Sum	Change	Change %

Search breakdowns and elements

Assignee = Fred Luddy

- Priority
- Resolution Code
- Category
- Assignment Group
- Reassignments
- Assignment Group(s) - Related Breakdown

7. Expand the **Assignment Group(s)** - **Related Breakdown** to view incidents assigned to Fred Luddy as well as groups that Fred belongs to.



Number of resolved incidents > Assignee = Fred Luddy

May 3

1

91	11	1	0%	0	0
No. of scores	Sum	Change	Change %	Average	Minimum

Search breakdowns and elements

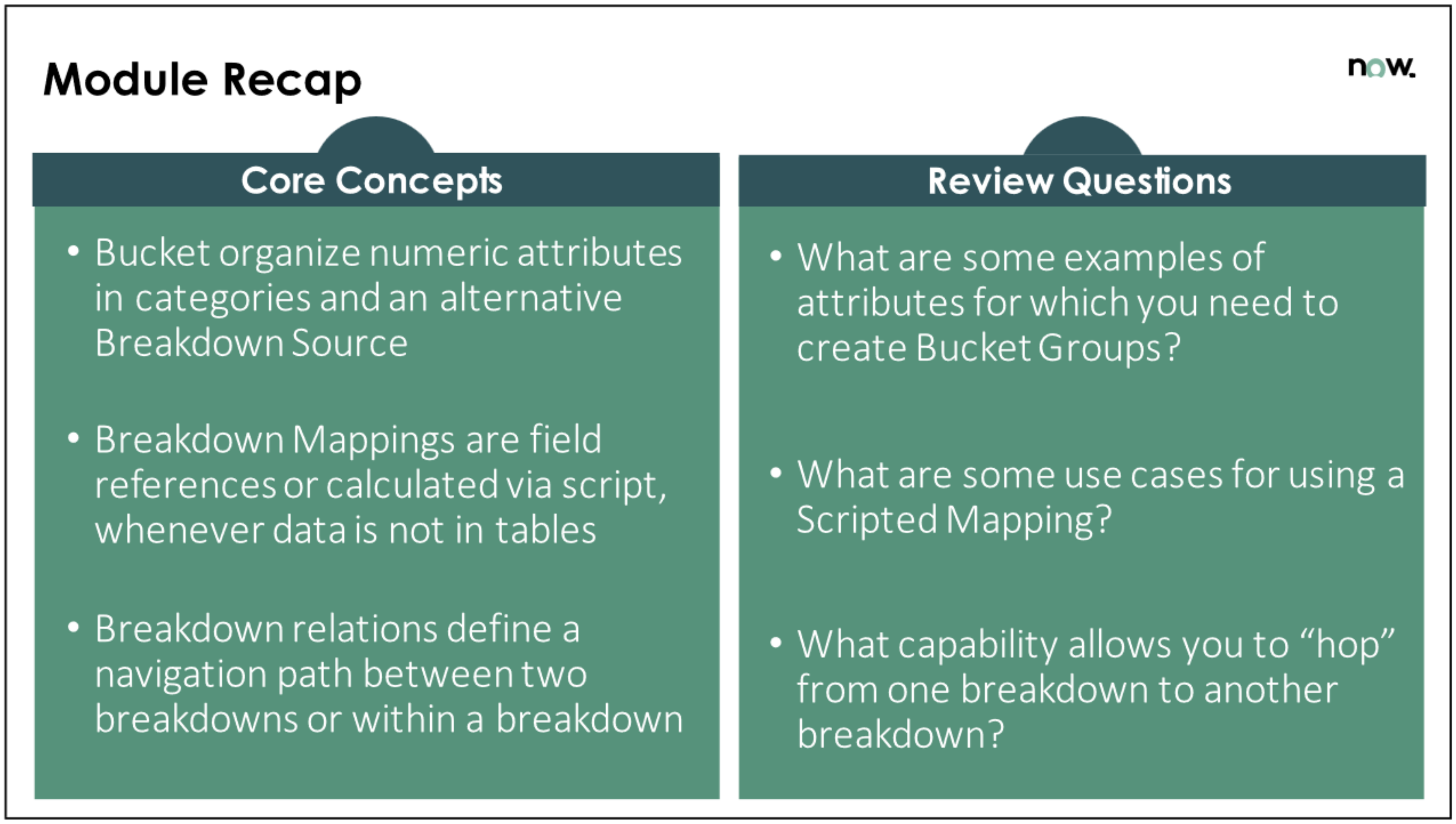
Assignee = Fred Luddy

Assignment Group(s) - Related Breakdown

Element	Score ▾	Change
Service Desk	22	7
Database	19	12
Software	15	5
Network	9	0
Hardware	5	-3
Network CAB Managers	0	0
Project Mgmt	0	0
RMA Approvers	0	0

> ***Notes**: The Assignment groups listed under the Assignment Group(s) - Related Breakdowns are ONLY the groups to which Fred Luddy belongs.*
>
> *Remember that the related breakdown does not drill into a 2nd level of breakdown but rather "hops" into another "dimension" of the data.*

You have now completed the Breakdown Relations Lab.



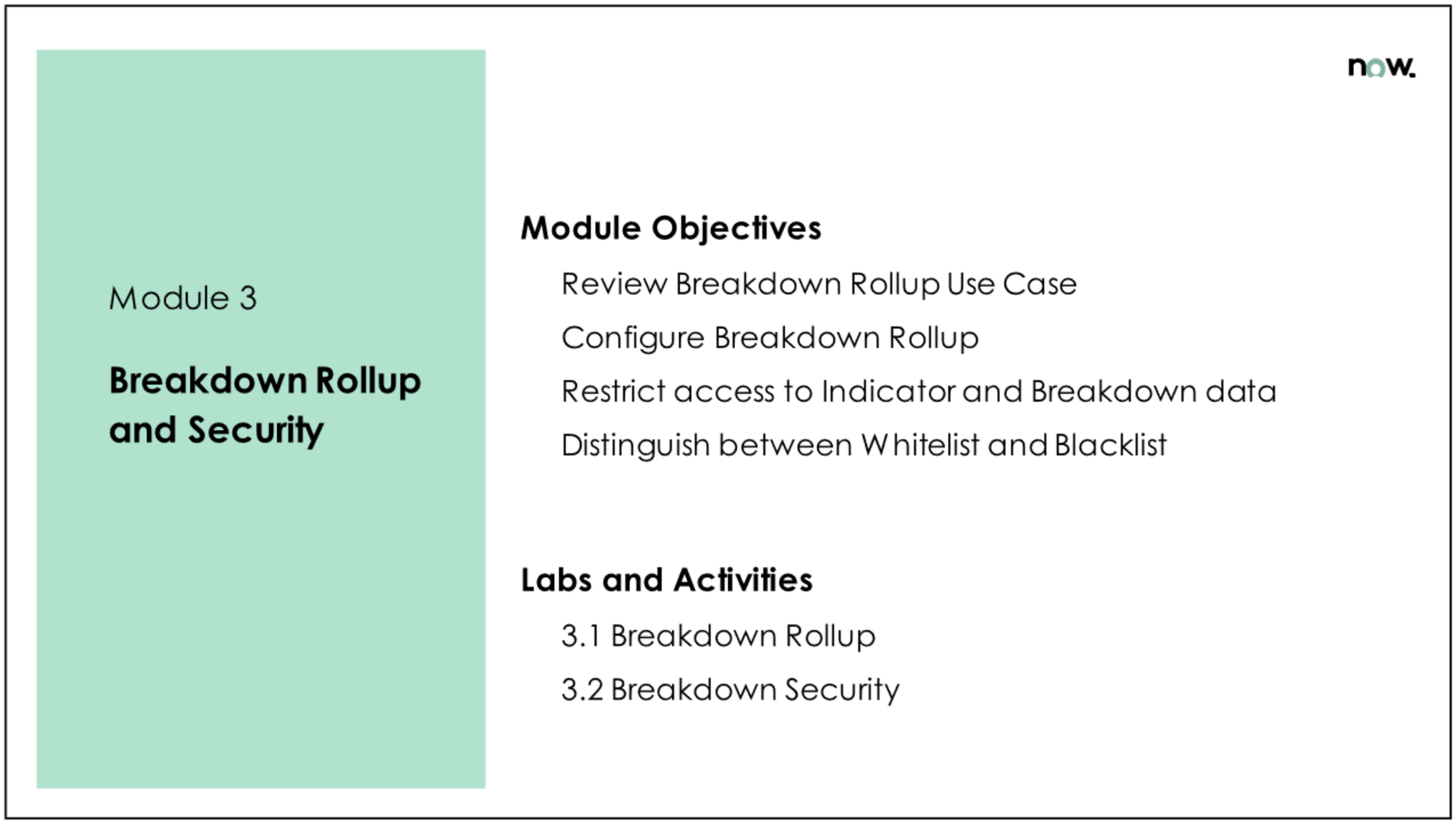
Module Recap

Core Concepts

- Bucket organize numeric attributes in categories and an alternative Breakdown Source
- Breakdown Mappings are field references or calculated via script, whenever data is not in tables
- Breakdown relations define a navigation path between two breakdowns or within a breakdown

Review Questions

- What are some examples of attributes for which you need to create Bucket Groups?
- What are some use cases for using a Scripted Mapping?
- What capability allows you to “hop” from one breakdown to another breakdown?



Module 3

Breakdown Rollup and Security

Module Objectives

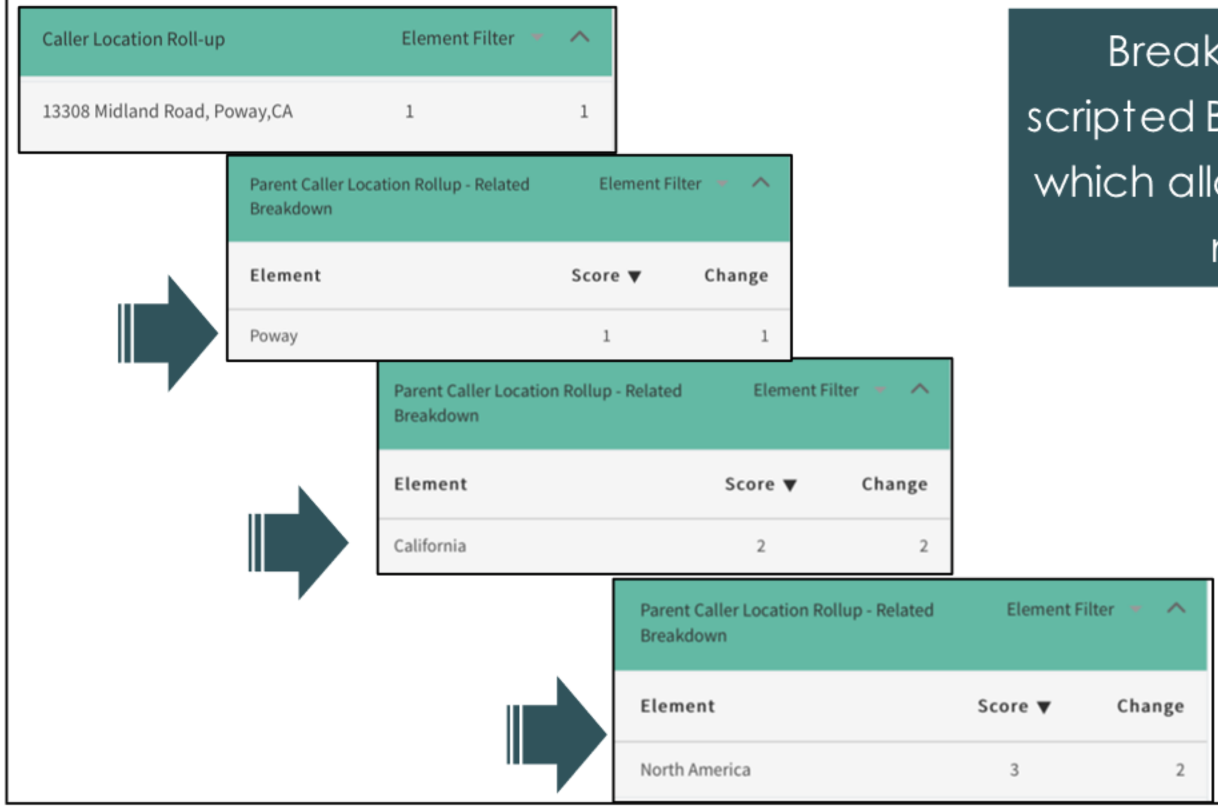
- Review Breakdown Rollup Use Case
- Configure Breakdown Rollup
- Restrict access to Indicator and Breakdown data
- Distinguish between Whitelist and Blacklist

Labs and Activities

- 3.1 Breakdown Rollup
- 3.2 Breakdown Security

This module explores an alternative implementation of the Breakdown Source introduced in the previous chapter – the Bucket Groups. In addition, advanced breakdown mapping techniques are reviewed. Last, the concept of creating a relationships between unrelated breakdowns is explored.

Breakdown Rollup Defined



Caller Location Roll-up — Element Filter

13308 Midland Road, Poway,CA	1	1

Parent Caller Location Rollup - Related Breakdown — Element Filter

Element	Score ▼	Change
Poway	1	1

Parent Caller Location Rollup - Related Breakdown — Element Filter

Element	Score ▼	Change
California	2	2

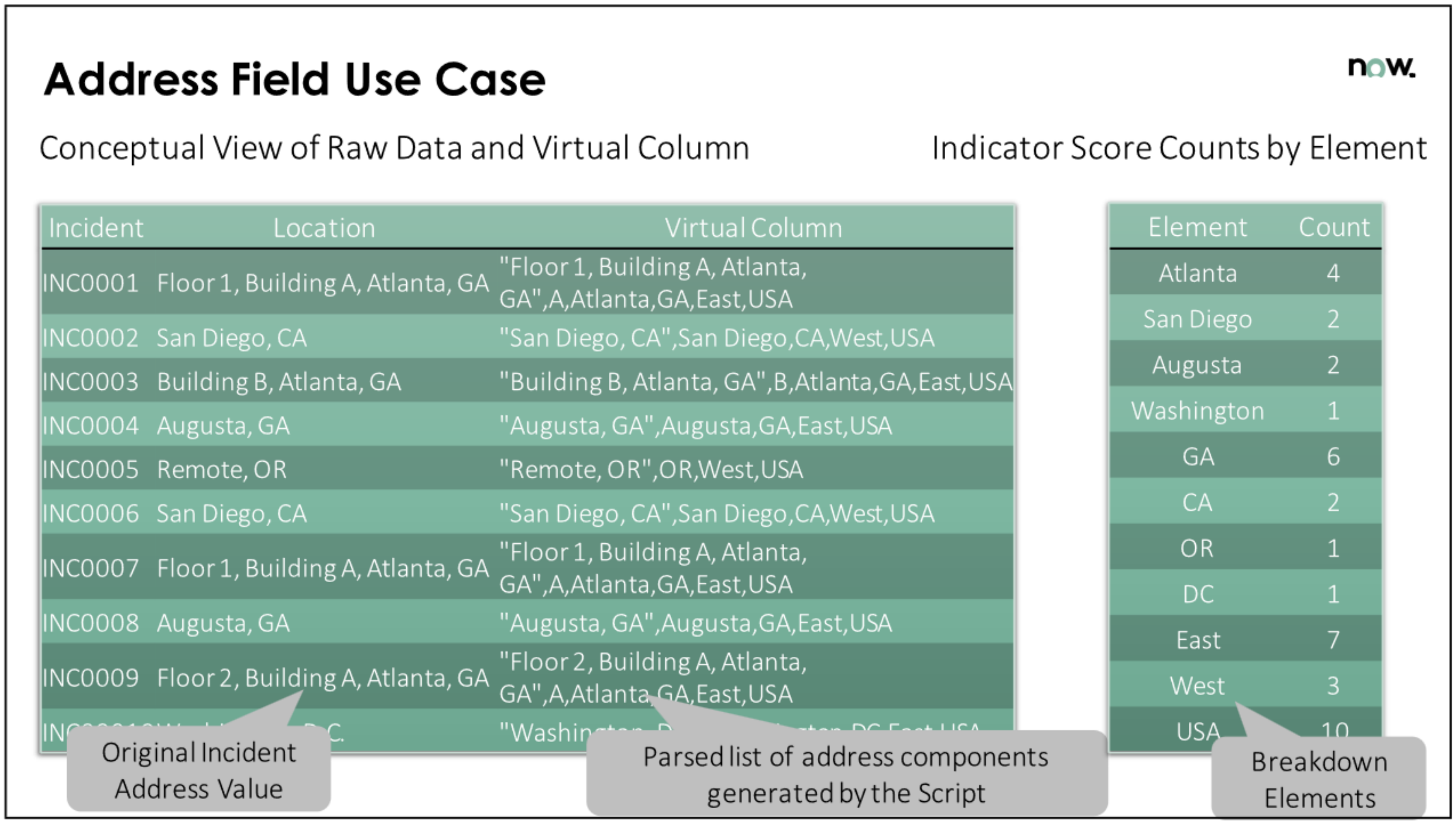
Parent Caller Location Rollup - Related Breakdown — Element Filter

Element	Score ▼	Change
North America	3	2

Breakdown rollup is a scripted Breakdown Relation which allows for hierarchical navigation

The Breakdown rollup is a special kind of Breakdown Relation. It differs from the Breakdown relations you have seen so far in that it uses a script to create an array of hierarchical breakdown elements. As a result, each drilldown into a Breakdown Element with a Breakdown rollup allows to navigate to the Parent breakdown element.

In the example shown here, you are able to drill into Incidents by Caller located at a specific Poway, CA address. From there, you can navigate to all incidents by callers from Poway. And from there, you are able to navigate to all incidents by callers from California, then North America, America, etc.



Address Field Use Case

Conceptual View of Raw Data and Virtual Column

Incident	Location	Virtual Column
INC0001	Floor 1, Building A, Atlanta, GA	"Floor 1, Building A, Atlanta, GA",A,Atlanta,GA,East,USA
INC0002	San Diego, CA	"San Diego, CA",San Diego,CA,West,USA
INC0003	Building B, Atlanta, GA	"Building B, Atlanta, GA",B,Atlanta,GA,East,USA
INC0004	Augusta, GA	"Augusta, GA",Augusta,GA,East,USA
INC0005	Remote, OR	"Remote, OR",OR,West,USA
INC0006	San Diego, CA	"San Diego, CA",San Diego,CA,West,USA
INC0007	Floor 1, Building A, Atlanta, GA	"Floor 1, Building A, Atlanta, GA",A,Atlanta,GA,East,USA
INC0008	Augusta, GA	"Augusta, GA",Augusta,GA,East,USA
INC0009	Floor 2, Building A, Atlanta, GA	"Floor 2, Building A, Atlanta, GA",A,Atlanta,GA,East,USA
INC0010	Washington, D.C.	"Washington, D.C.",Washington,DC,East,USA

Indicator Score Counts by Element

Element	Count
Atlanta	4
San Diego	2
Augusta	2
Washington	1
GA	6
CA	2
OR	1
DC	1
East	7
West	3
USA	10

Original Incident Address Value

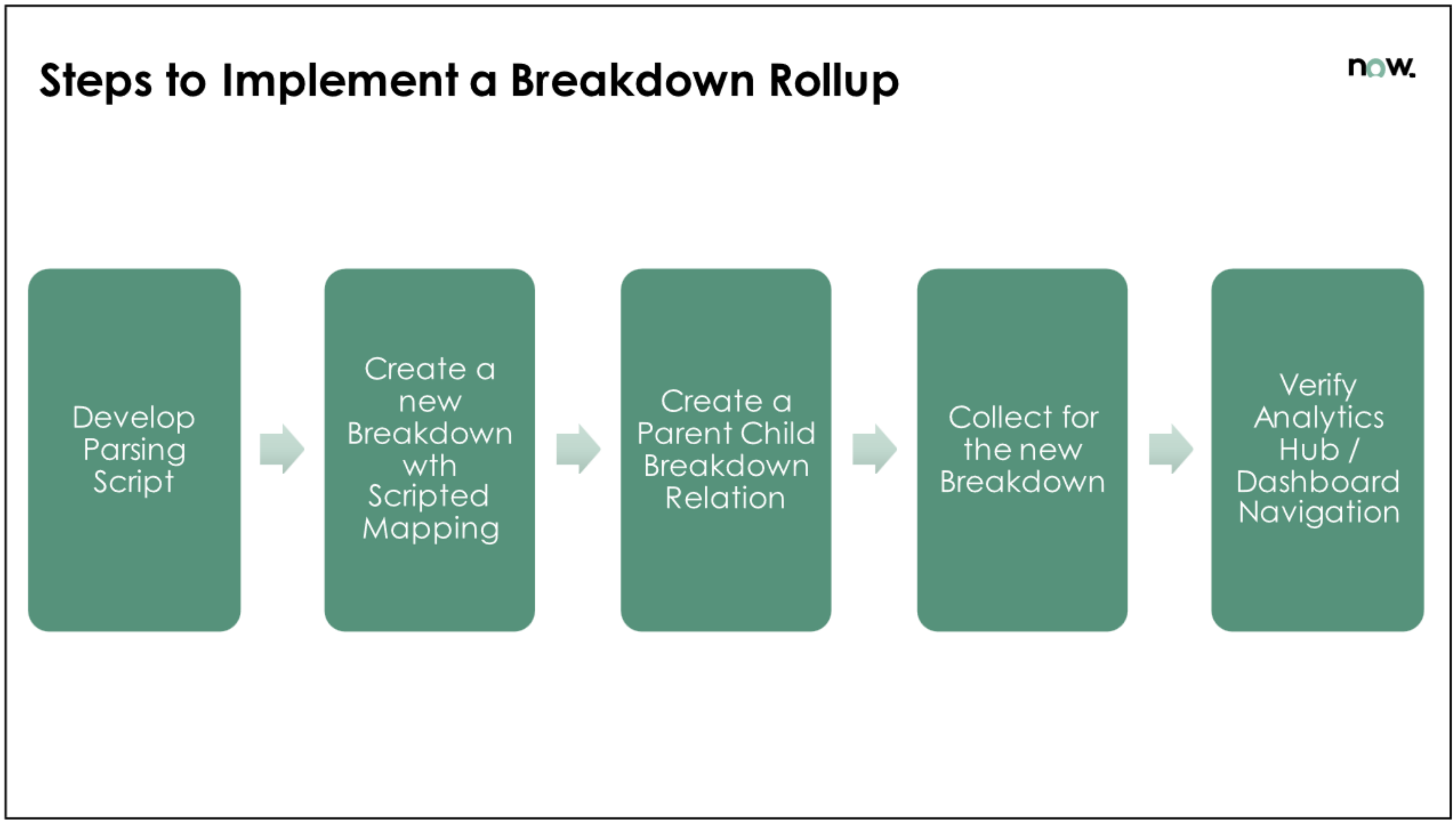
Parsed list of address components generated by the Script

Breakdown Elements

A common application of Breakdown Rollup is when processing hierarchical data such as addresses.

Shown here is a simplified view of the collected data for Incidents. With the addition of a Script, you are able to parse the address string into street, town, country, continent, etc. components. As a result, each address component is treated as a Breakdown element. So you can navigate the address by City, State, Country, etc.

Note the addition of a virtual column that contains the parsed out version of the Location field. Also note that the text is shown for illustration purposes only. In reality, the Virtual Column is a list of Sys IDs returned from the script added in the map/reduce step of data collection. Indicator score counts are calculated for each element of the Virtual column.



Steps to Implement a Breakdown Rollup

now.

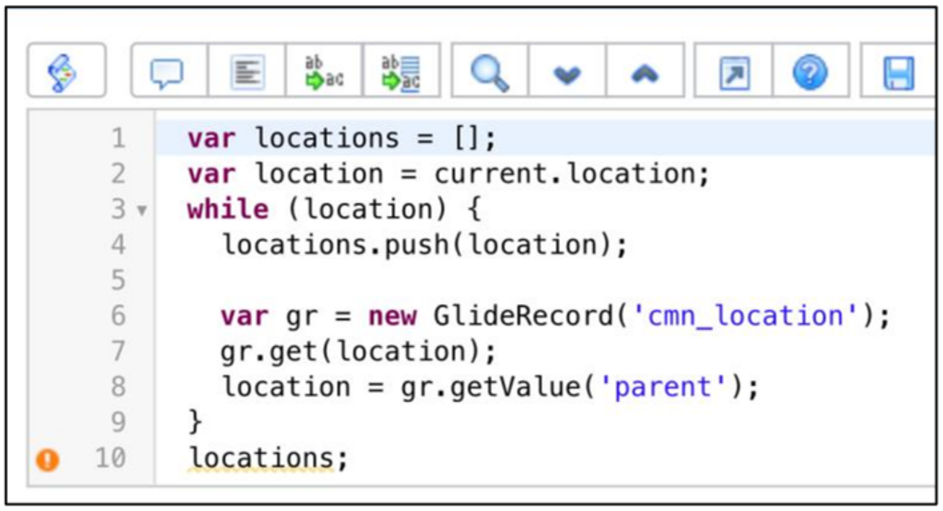
Develop Parsing Script → Create a new Breakdown wth Scripted Mapping → Create a Parent Child Breakdown Relation → Collect for the new Breakdown → Verify Analytics Hub / Dashboard Navigation

Here are the steps to implement a Breakdown rollup. The key element is the parsing script which is going to break up a hierarchical structure such as an address into multiple Breakdown Elements.

Step 1: Parsing / Hierarchy Lookup Script

1. Define an array to hold all elements of the Location hierarchy
2. Read in the Location of the Current Record
3. Repeat:
   4. Add the Location to the Location Array
   6. Create a Glide Record of Type Location
   7. Get the current location Sys Id
   8. Get the Parent of the current Sys Id
   9. Repeat to collect the Parent's Parent unless the current Parent is empty (meaning you are at the top of the hierarchy)

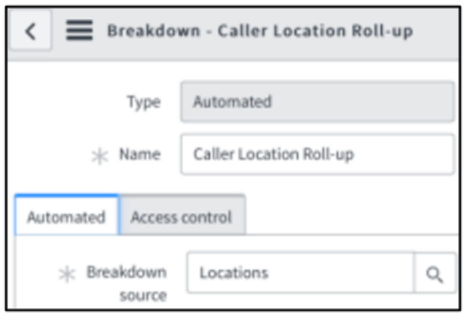
10. Return the array of parent elements



```
 1  var locations = [];
 2  var location = current.location;
 3  while (location) {
 4    locations.push(location);
 5
 6    var gr = new GlideRecord('cmn_location');
 7    gr.get(location);
 8    location = gr.getValue('parent');
 9  }
10  locations;
```

The first step in implementing a Breakdown rollup is to develop the parsing script which returns an ordered array of parent elements. If not a scripting expert, use the above explanation of code by line to get a better understanding of what the script does. All Performance Analytics baseline scripts can be found by navigating to **Performance Analytics > Automation > Scripts**. This script shown in the slide is the **Task.Location** script also included in the baseline.

Step 2: New Breakdown and Breakdown


Breakdown - Caller Location Roll-up

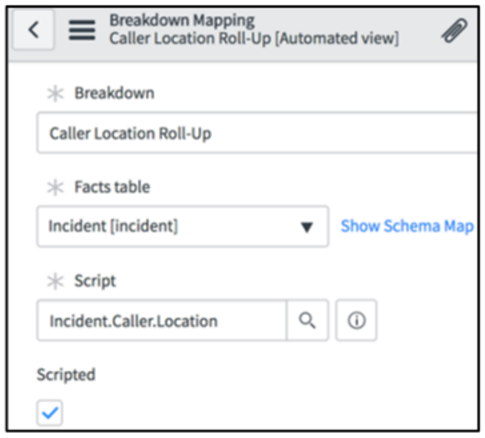
Type: Automated

Name: Caller Location Roll-up

Automated | Access control

Breakdown source: Locations

New Breakdown based on the source containing the element hierarchy


Breakdown Mapping
Caller Location Roll-Up [Automated view]

Breakdown: Caller Location Roll-Up

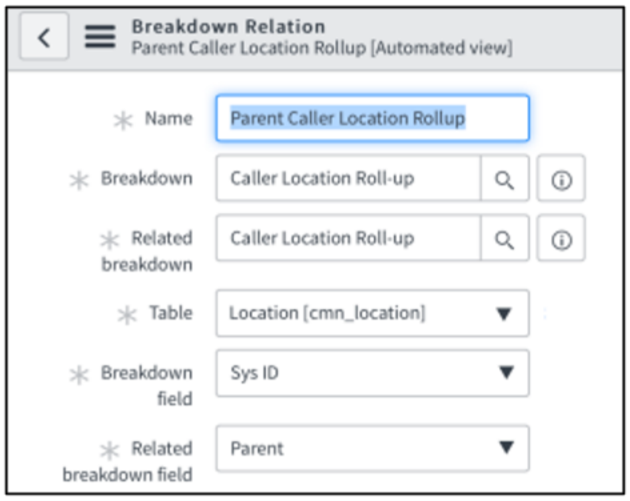
Facts table: Incident [incident] Show Schema Map

Script: Incident.Caller.Location

Scripted: ☑

Breakdown Mapping to create virtual column with parent elements

Step 3: Breakdown Relation


Breakdown Relation
Parent Caller Location Rollup [Automated view]

Name: Parent Caller Location Rollup

Breakdown: Caller Location Roll-up

Related breakdown: Caller Location Roll-up

Table: Location [cmn_location]

Breakdown field: Sys ID

Related breakdown field: Parent

Create Navigational Path from Location to its Parent

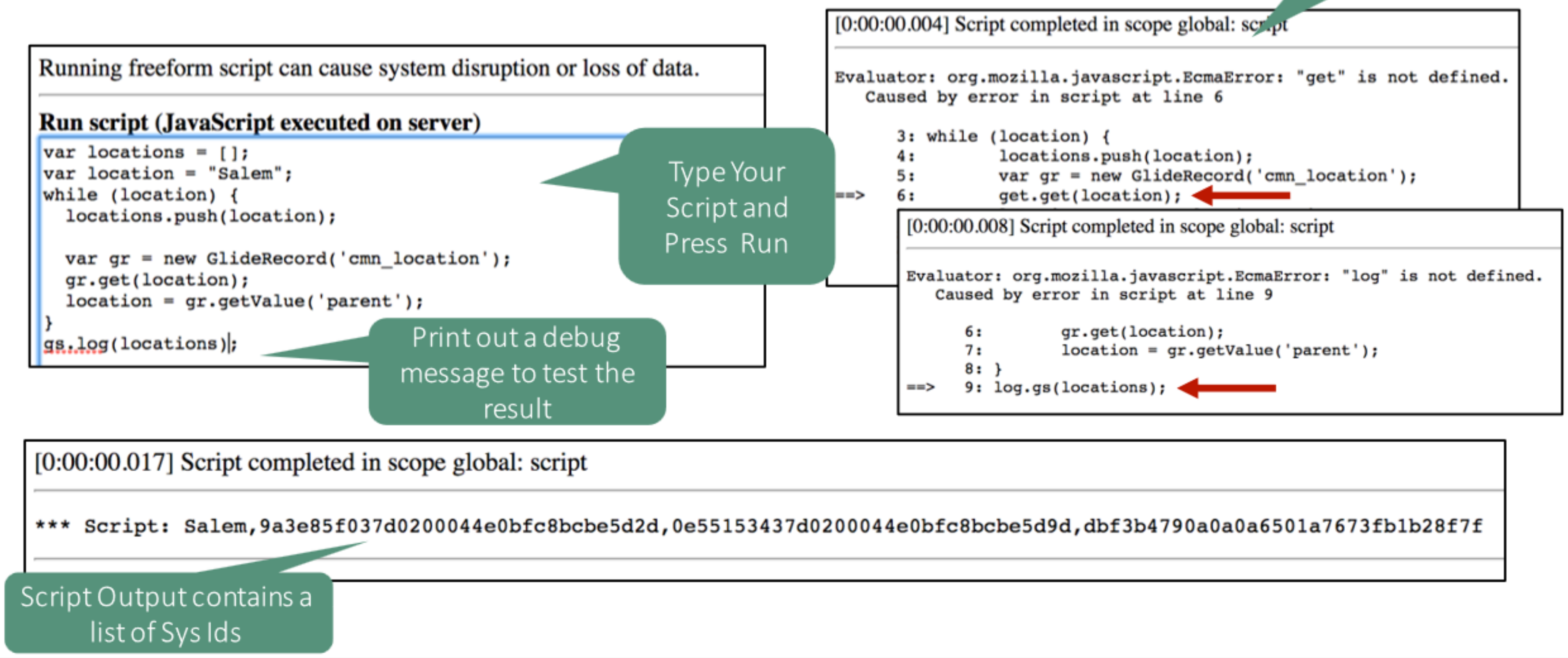
Here are the remaining steps in completing the Breakdown relation:

- Create a new Breakdown based on the source of the record array your parsing script is processing. In this example, the Caller Location Roll-up breakdown is based on the Locations breakdown source. The Location breakdown source contains location objects – regions, continents, countries, states, cities, street addresses, etc.
- Create a new scripted mapping that matches the Locations found in the Source to the location field of the Facts table records using the parsing script
- Create a Breakdown Relation which will allow to navigate from an element to its parent
- Last, add the Breakdown to desired Indicators

Best Practice: Always Test Your Scripts!

now.

Navigate to System Definition > Scripts – Background



Running freeform script can cause system disruption or loss of data.

Run script (JavaScript executed on server)

```
var locations = [];
var location = "Salem";
while (location) {
  locations.push(location);

  var gr = new GlideRecord('cmn_location');
  gr.get(location);
  location = gr.getValue('parent');
}
gs.log(locations);
```

Type Your Script and Press Run

Print out a debug message to test the result

Parser displays any script errors

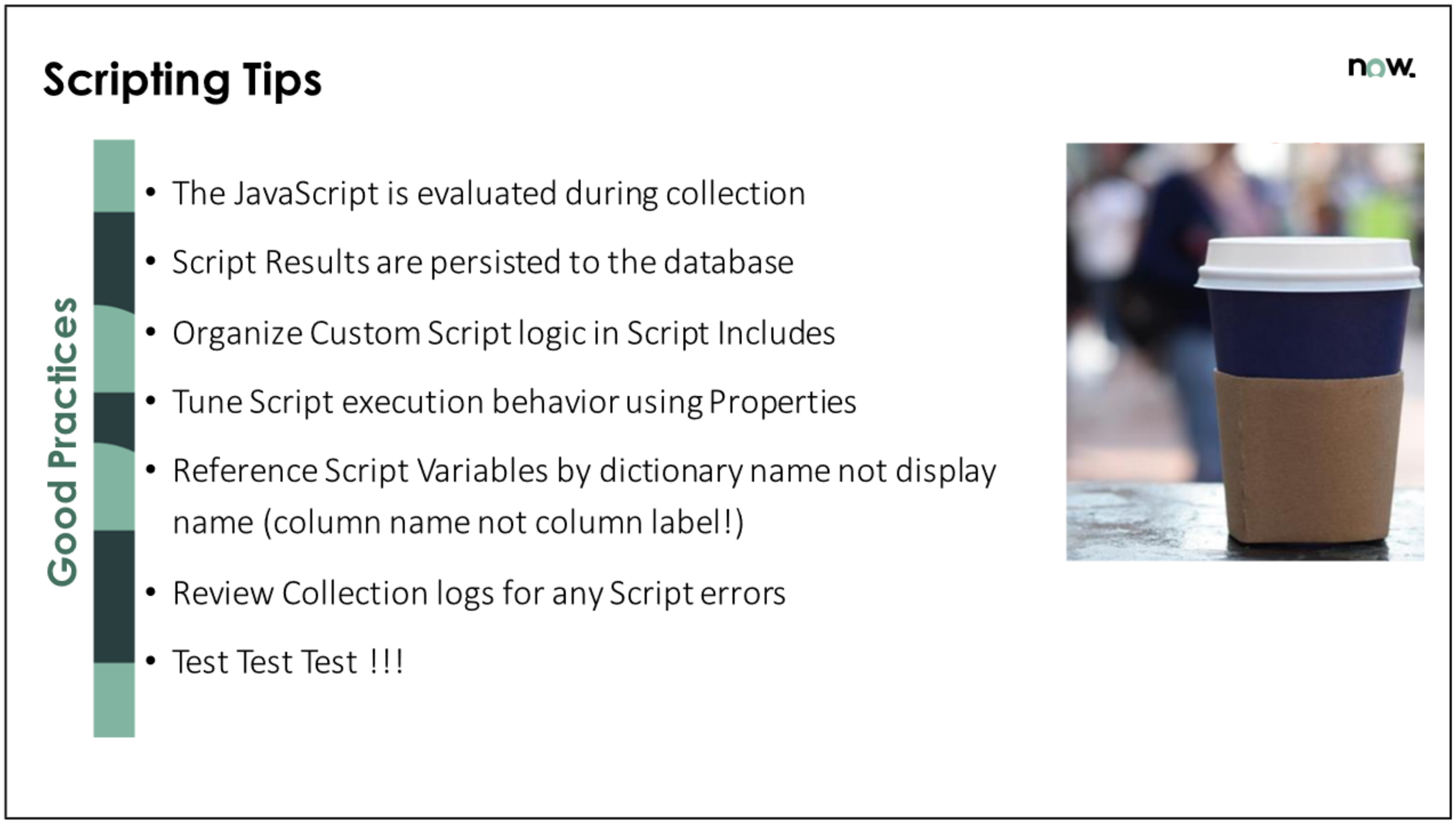
```
[0:00:00.004] Script completed in scope global: script

Evaluator: org.mozilla.javascript.EcmaError: "get" is not defined.
   Caused by error in script at line 6

      3: while (location) {
      4:         locations.push(location);
      5:         var gr = new GlideRecord('cmn_location');
==>   6:         get.get(location);
```

```
[0:00:00.008] Script completed in scope global: script

Evaluator: org.mozilla.javascript.EcmaError: "log" is not defined.
   Caused by error in script at line 9

      6:         gr.get(location);
      7:         location = gr.getValue('parent');
      8: }
==>   9: log.gs(locations);
```

```
[0:00:00.017] Script completed in scope global: script

*** Script: Salem,9a3e85f037d0200044e0bfc8bcbe5d2d,0e55153437d0200044e0bfc8bcbe5d9d,dbf3b4790a0a0a6501a7673fb1b28f7f
```

Script Output contains a list of Sys Ids

As a best practice, ALWAYS test your code. An easy way to test is by going to **System Definition > Scripts – Background** and typing in your logic to run and test.



Scripting Tips

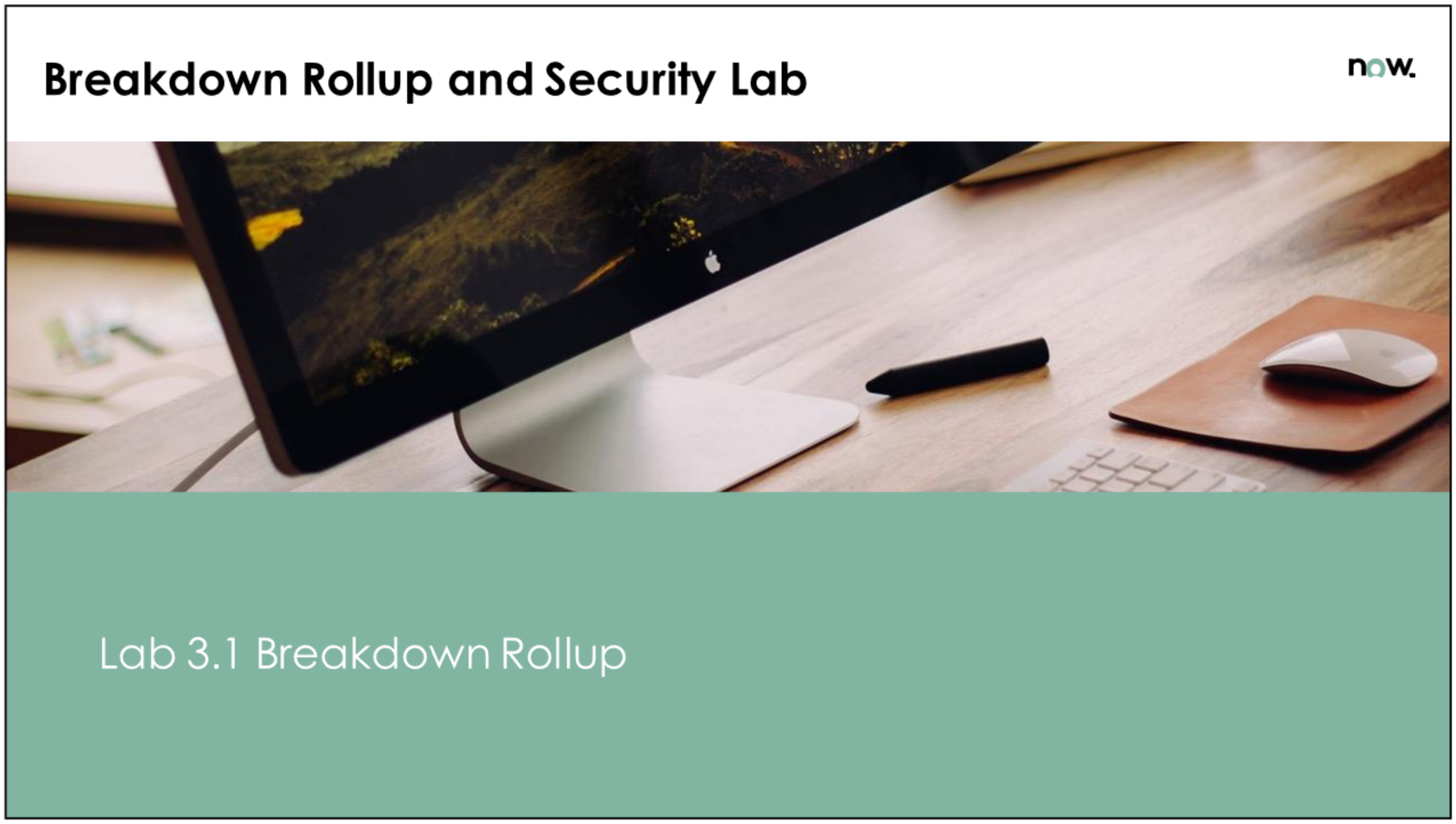
Good Practices

- The JavaScript is evaluated during collection
- Script Results are persisted to the database
- Organize Custom Script logic in Script Includes
- Tune Script execution behavior using Properties
- Reference Script Variables by dictionary name not display name (column name not column label!)
- Review Collection logs for any Script errors
- Test Test Test !!!

Remember to follow these tips when building Scripts in Performance Analytics.

To brush up on your Java script and ServiceNow APIs:

- Download the ServiceNow JavaScript Primer here: https://www.servicenow.com/content/dam/servicenow/other-documents/training/ServiceNow-JavaScript-Primer.pdf
- API Reference documentation: https://developer.servicenow.com/app.do#!/api_doc



Breakdown Rollup and Security Lab

Lab 3.1 Breakdown Rollup

Breakdown Rollup

Lab 3.1

⌛25m

Lab Objectives

Survey information contains the Caller address which typically consists of Street, City, State, Country, etc. As a Customer Success Advocate, you need to be able to aggregate scores information not just for the individual Street address but also for the parent City, State, Country, and Region. This lab will show you how to do the following.

- Create a Breakdown on a field containing hierarchical information
- Create a Breakdown mapping script to extract the hierarchy elements
- Perform hierarchical drilldown in the Analytics Hub

Location Breakdown Rollup

A. Breakdown and Breakdown Mapping

Survey Indicators need to allow multi-level drilldown from Street Address to top-most Region and display score information for each level of the location hierarchy.

Breakdown Mapping Script Logic

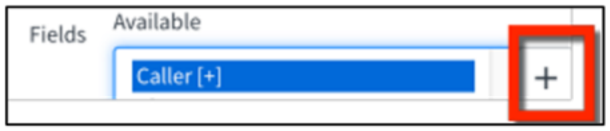
A special scripted breakdown mapping is required so that you can drill from a child to a parent element from within the Analytics Hub. The hierarchy here consists of the elements of the address field.

1. Navigate to **Performance Analytics > Automation > Scripts**.

2. Click **New** and create a new script as follows:

   - Name: **Survey.Caller.Location**
   - Facts table: **Survey Scores**
   - Fields: **Caller.Location**

Tip: Click the + icon to explain the Caller hierarchy.



3. Enter the following code in the **Scripts** section. This code is available in the Class Knowledge Base.

```
var locations = [];
var location = current.u_caller.location;
while (location) {
        locations.push(location);
        var gr = new
        GlideRecord('cmn_location');
        gr.get(location);
        location = gr.getValue('parent');
}
locations;
```

> ***Tip:*** *The above script is almost identical to the* ***Task.Location*** *script. As a shorthand, copy the* ***Task.Location*** *script and modify line 2 as well as the Facts table and the Fields attribute.*

4. **Submit** the script.

Automated Breakdown and Mapping

1. Navigate to **Performance Analytics > Breakdowns > Automated Breakdowns**.

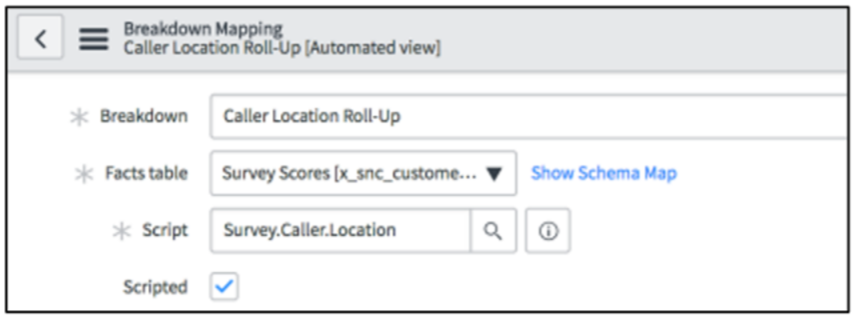
2. Click **New** and create a new Automated breakdown as follows:

   - Name: **Caller Location Roll-up**
   - Breakdown source: **Locations**

3. Save the Breakdown.

4. Add a Breakdown Mapping as follows:

   - Facts table: **Survey Scores**
   - Scripted: **checked**
   - Script: **Survey.Caller.Location**



Breakdown Mapping
Caller Location Roll-Up [Automated view]

Breakdown: Caller Location Roll-Up

Facts table: Survey Scores [x_snc_custome...

Script: Survey.Caller.Location

Scripted: ☑

5. Click **Submit**.

6. Add the following Indicators to the **Indicators** Related List:
   Number of CSAT Scores
   Average CSAT Score

7. **Save** to return to the Breakdown form.

Breakdown Relation

1. Navigate to the **Breakdown Relations** Related List.

2. Click **New** to add a new **Breakdown Relation** as follows:

   Name: **Parent Caller Location Rollup**

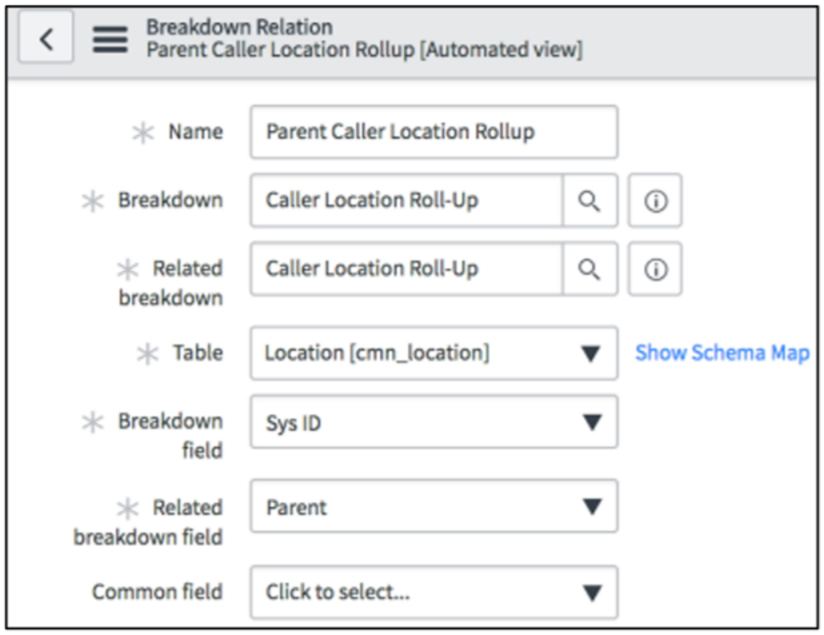
   Breakdown: **Caller Location Roll-up**

   Related breakdown: **Caller Location Rollup**

   Table: **Location [cmn_location]**

   Breakdown: **Sys ID**

   Related Breakdown field: **Parent**



Breakdown Relation
Parent Caller Location Rollup [Automated view]

✱ Name	Parent Caller Location Rollup
✱ Breakdown	Caller Location Roll-Up
✱ Related breakdown	Caller Location Roll-Up
✱ Table	Location [cmn_location]
✱ Breakdown field	Sys ID
✱ Related breakdown field	Parent
Common field	Click to select...

3. Click **Submit**.

B. Collect Data and Verify Analytics Hub Navigation

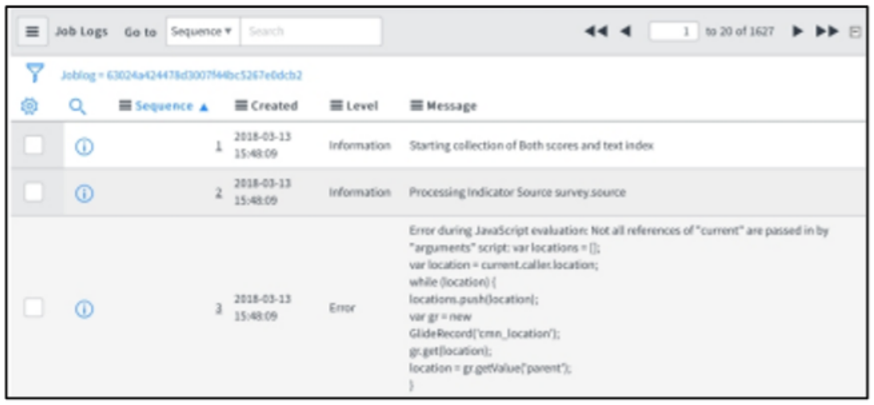
Collect Scores

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Open the **Survey Collection Historic** job.

3. Click **Execute Now** to re-run the job.

4. Check the **Job Logs** to make sure there are no JavaScript errors such as this one:



Job Logs Go to Sequence ▾ Search ◀◀ ◀ 1 to 20 of 1627 ▶ ▶▶

Joblog = 63024a424478d3007f44bc5267e0dcb2

Sequence ▲	Created	Level	Message
1	2018-03-13 15:48:09	Information	Starting collection of Both scores and text index
2	2018-03-13 15:48:09	Information	Processing Indicator Source survey.source

3	2018-03-13 15:48:09	Error	Error during JavaScript evaluation: Not all references of "current" are passed in by "arguments" script: var locations = [];
			var location = current.caller.location;
			while (location) {
			locations.push(location);
			var gr = new
			GlideRecord('cmn_location');
			gr.get(location);
			location = gr.getValue('parent');
			}

> ***Tip:*** *The above is a common error when working with Scripted Mappings. It indicates that the Script is missing the correct Fields value.*

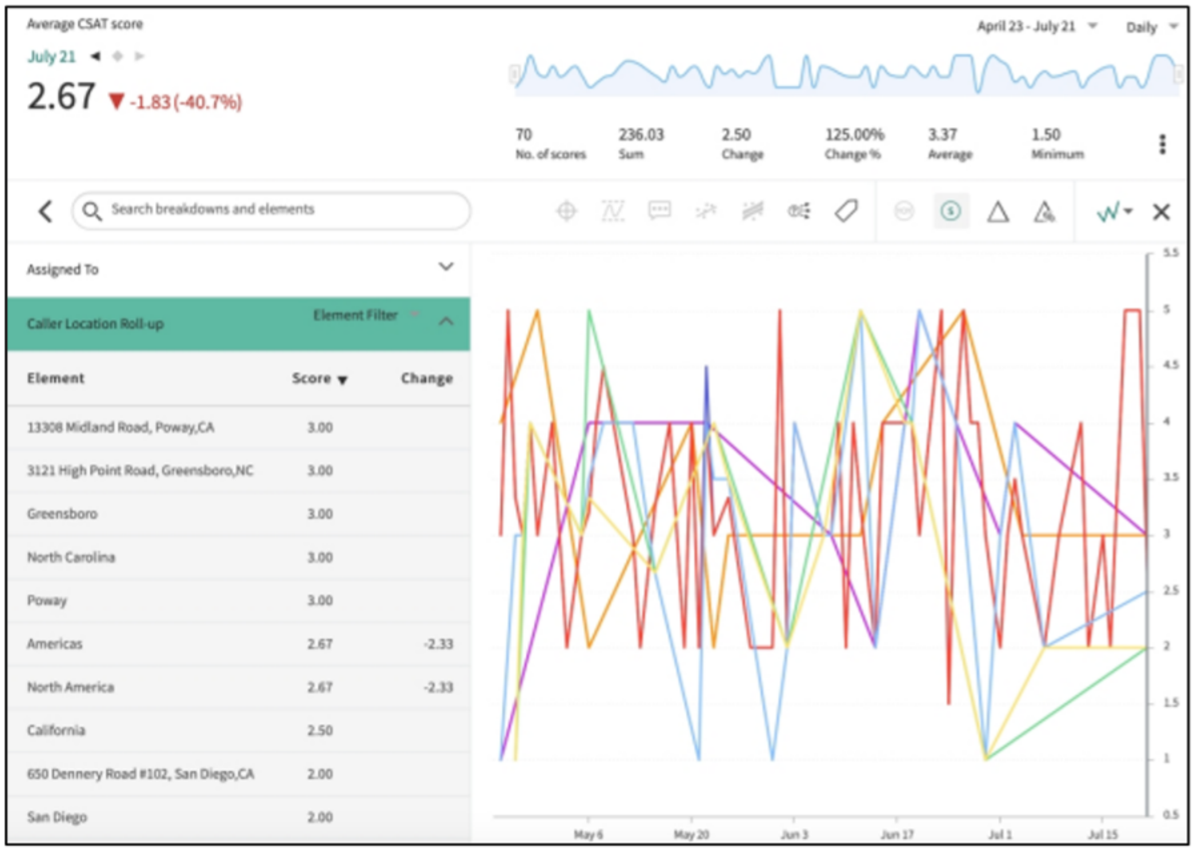
5. If needed, fix any reported errors and re-run the Collection Job.

Validate Breakdown Data

1. Navigate to the **Analytics Hub**.

2. Open the **Average CSAT Scores** indicator.

3. Expand the **Breakdowns** list and select **Caller Location Roll Up**.



Average CSAT score

July 21

2.67 ▼ -1.83 (-40.7%)

April 23 - July 21 Daily

70	236.03	2.50	125.00%	3.37	1.50
No. of scores	Sum	Change	Change %	Average	Minimum

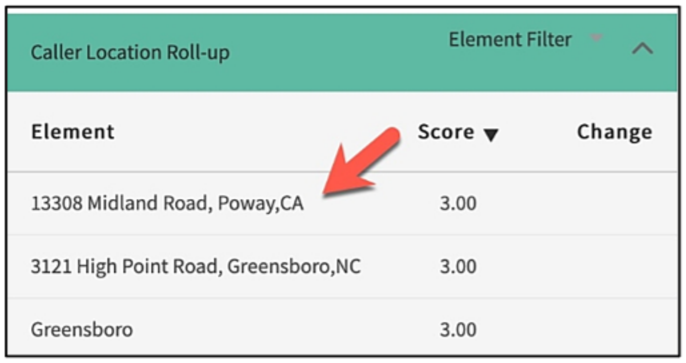
Search breakdowns and elements

Assigned To

Caller Location Roll-up Element Filter

Element	Score ▼	Change
13308 Midland Road, Poway,CA	3.00	
3121 High Point Road, Greensboro,NC	3.00	
Greensboro	3.00	
North Carolina	3.00	
Poway	3.00	
Americas	2.67	-2.33
North America	2.67	-2.33
California	2.50	
650 Dennery Road #102, San Diego,CA	2.00	
San Diego	2.00	

4. Select **13308 Midland Road, Poway CA** to view scores for this specific address.

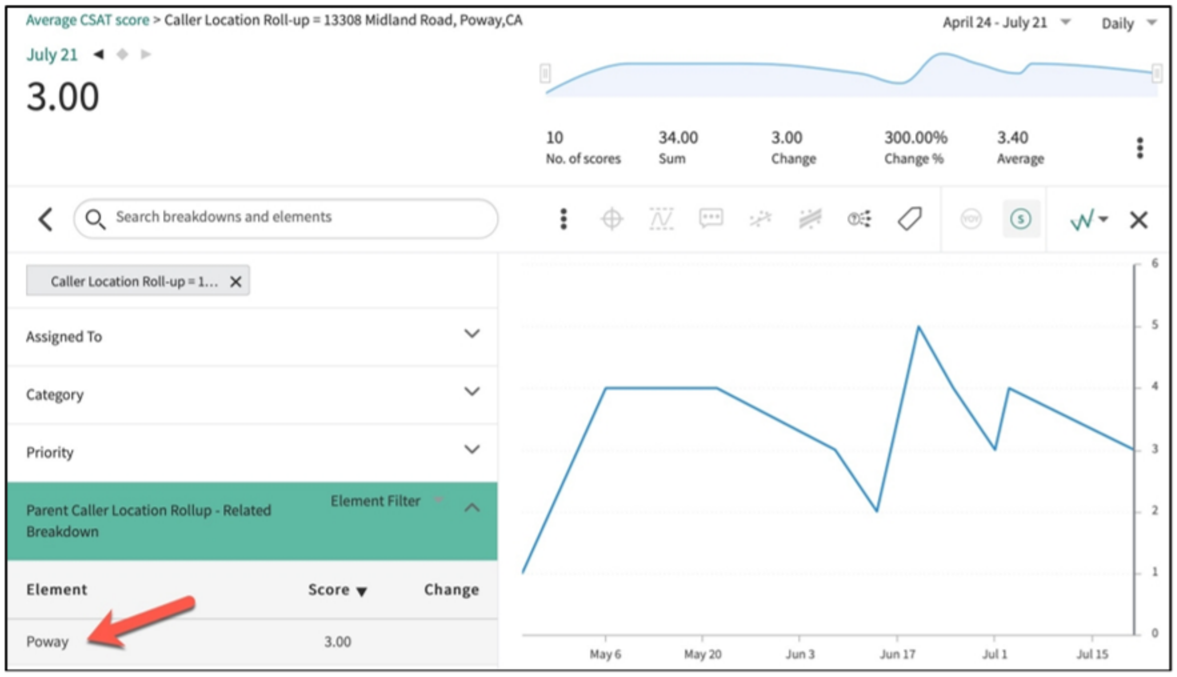


Caller Location Roll-up

Element Filter

Element	Score ▼	Change
13308 Midland Road, Poway,CA	3.00	
3121 High Point Road, Greensboro,NC	3.00	
Greensboro	3.00	

5. Select **Poway** in the **Parent Caller Location Rollup - Related Breakdown** dropdown.



Average CSAT score > Caller Location Roll-up = 13308 Midland Road, Poway,CA

April 24 - July 21 · Daily

July 21

3.00

10	34.00	3.00	300.00%	3.40
No. of scores	Sum	Change	Change %	Average

Search breakdowns and elements

Caller Location Roll-up = 1... ✕

Assigned To

Category

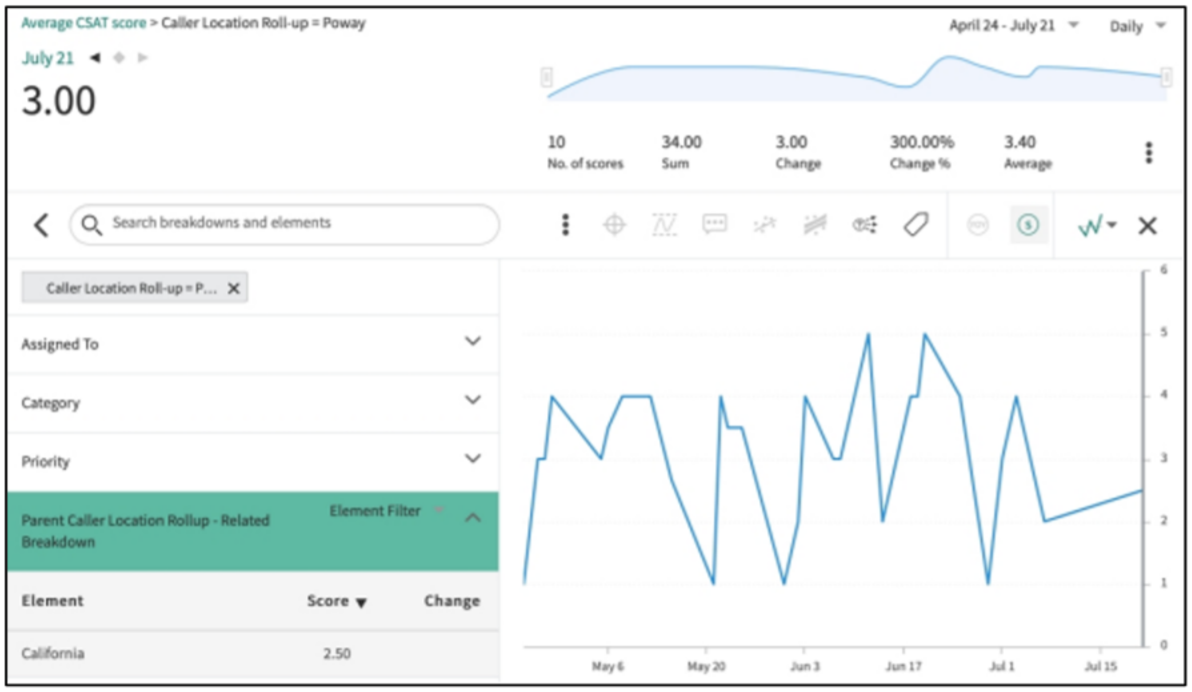
Priority

Parent Caller Location Rollup - Related Breakdown

Element Filter

Element	Score ▼	Change
Poway	3.00	

6. Select **California** in **Parent Caller Location Rollup - Related Breakdown**.



Average CSAT score > Caller Location Roll-up = Poway

April 24 - July 21 · Daily

July 21

3.00

10	34.00	3.00	300.00%	3.40
No. of scores	Sum	Change	Change %	Average

Search breakdowns and elements

Caller Location Roll-up = P... ✕

Assigned To

Category

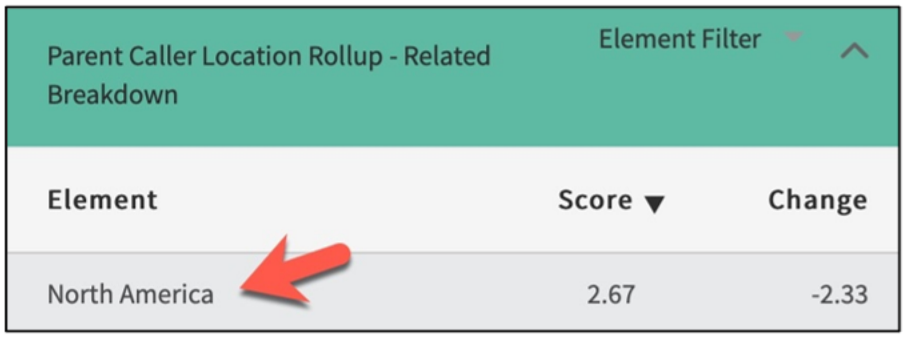
Priority

Parent Caller Location Rollup - Related Breakdown

Element Filter

Element	Score ▼	Change
California	2.50	

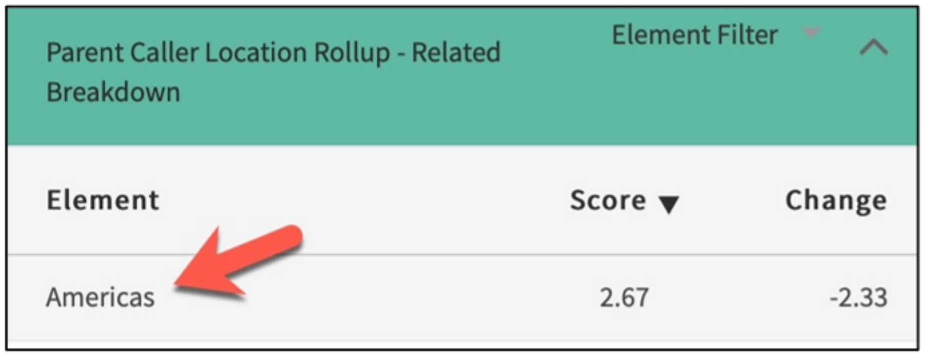
7. Select **North America** in **Parent Caller Location Rollup - Related Breakdown**.



Parent Caller Location Rollup - Related Breakdown

Element	Score ▼	Change
North America	2.67	-2.33

8. Select **Americas** in **Parent Caller Location Rollup - Related Breakdown**.



Parent Caller Location Rollup - Related Breakdown

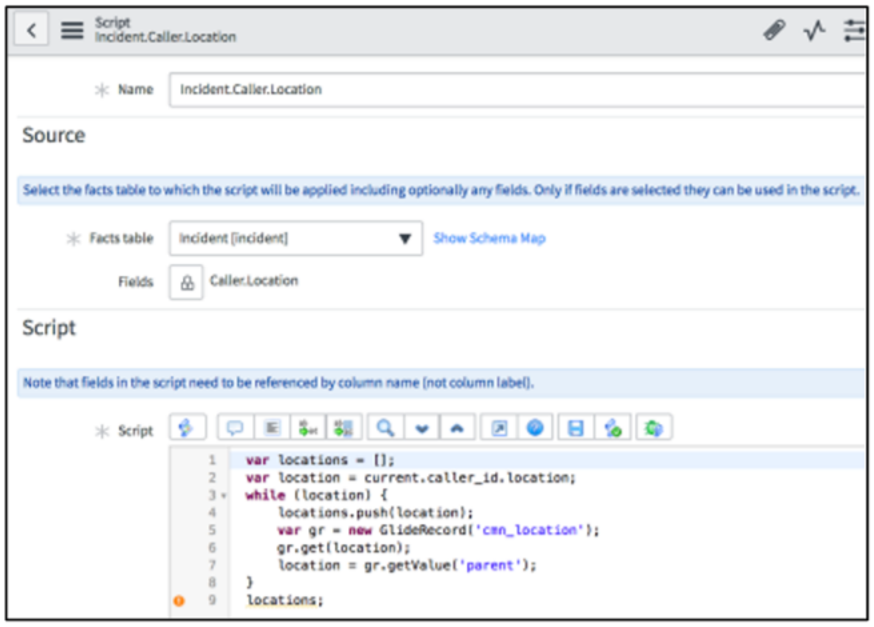
Element	Score ▼	Change
Americas	2.67	-2.33

***Question**: Do you see any more location related breakdowns?*

Optional Challenge Tasks:

1. Using the instructions for the **Survey Caller Location** roll-up, create a mapping for the **Incident Caller Location**.

   Hint: Create a script to parse and build a hierarchy of caller location components.



Script
Incident.Caller.Location

Name: Incident.Caller.Location

Source

Select the facts table to which the script will be applied including optionally any fields. Only if fields are selected they can be used in the script.

Facts table: Incident [incident]

Fields: Caller.Location

Script

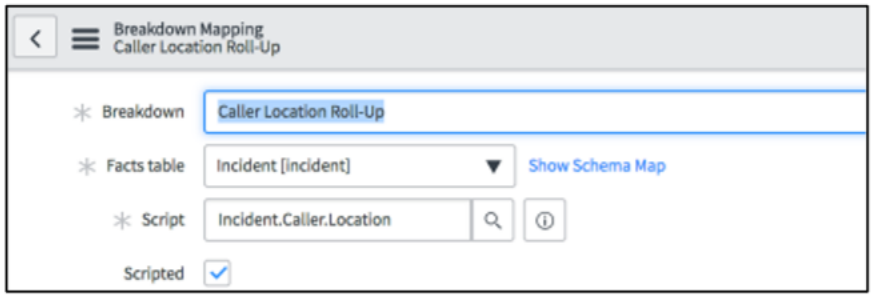
Note that fields in the script need to be referenced by column name (not column label).

Script:

```
var locations = [];
var location = current.caller_id.location;
while (location) {
    locations.push(location);
    var gr = new GlideRecord('cmn_location');
    gr.get(location);
    location = gr.getValue('parent');
}
locations;
```

2. Relate the mapping to the **Number of new incidents** indicator.

   Hint: You need a new Scripted mapping using the script created in step 1.



Breakdown Mapping
Caller Location Roll-Up

* Breakdown	Caller Location Roll-Up
* Facts table	Incident [incident]
* Script	Incident.Caller.Location
Scripted	☑

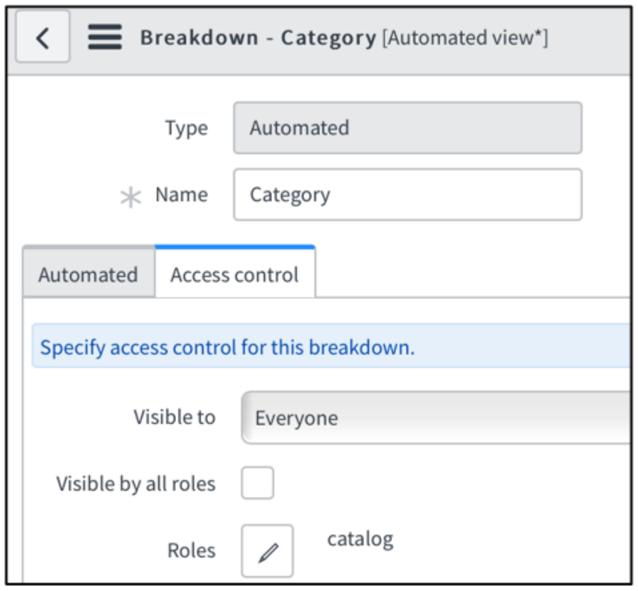
3. Collect data and test that you can navigate the **Number of new incidents** scorecard from the Caller's **Street** Location all the way to the Caller's **Region**.

You have now completed the Breakdown Rollup lab.

Breakdown and Indicator Security

- Indicator and Breakdown Access is controlled in the **Access control** configuration
  - Widgets are always visible, but drilldown to restricted Indicators in the Analytics Hub is disabled
  - Restricted Breakdowns are not visible in the Analytics Hub and cannot be applied in a Breakdown widget
- Visibility into Elements of a Breakdown List is controlled using **Security Lists**

Access Control settings limit visibility to Users, Groups and Roles


Breakdown - Category [Automated view*]

Type: Automated

Name: Category

Specify access control for this breakdown.

Visible to: Everyone

Visible by all roles: □

Roles: catalog

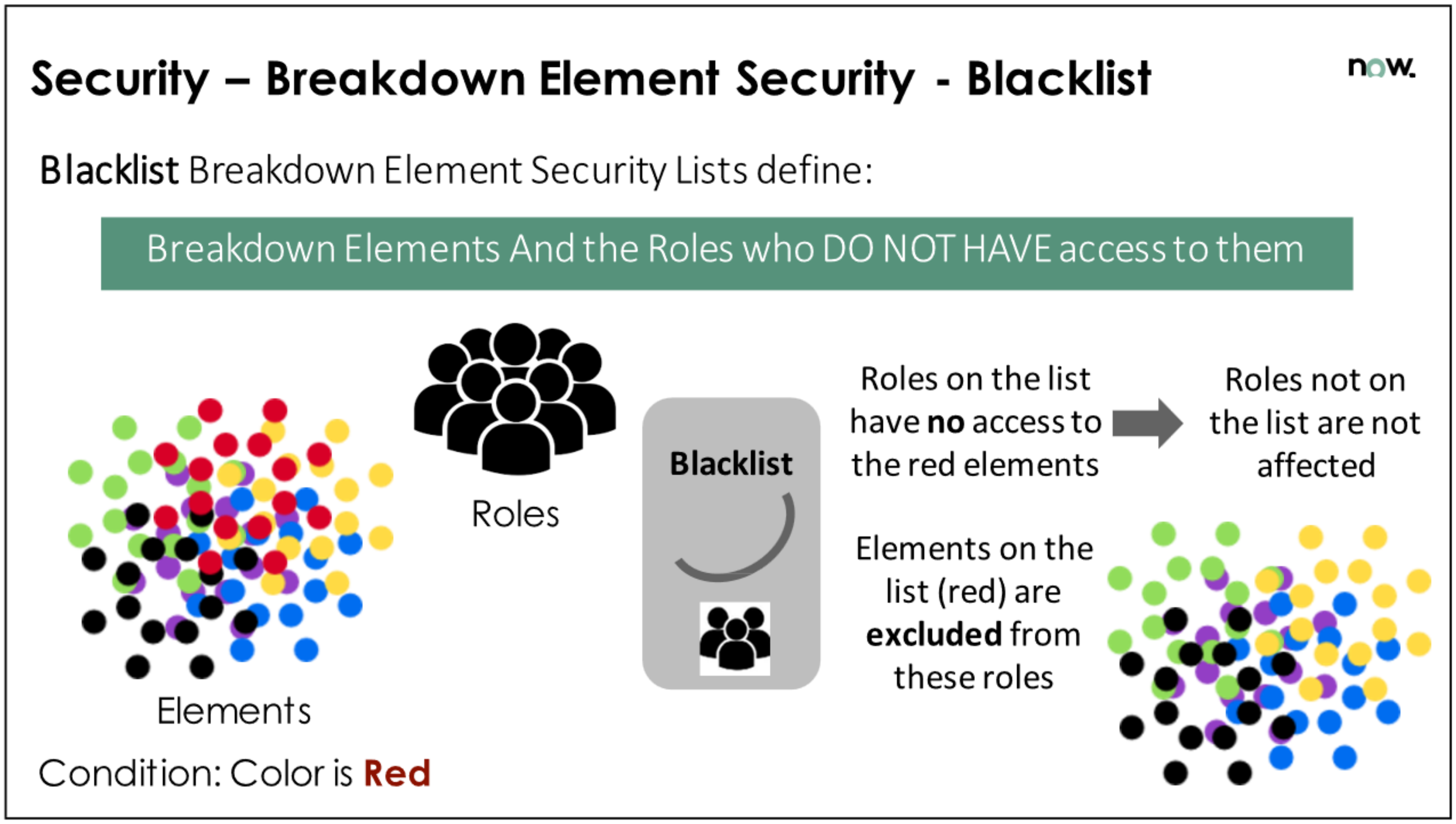
When users, groups, or roles are excluded in the Access Control configuration of an Indicator, they can still view widgets based on this indicator but will not be able to drill into the Indicator Analytics Hub. This error message will be displayed on the Dashboard indicating that access is denied: **Security rules prevent access to this indicator, breakdown, or element combination**

When users, groups, or roles are excluded in the Access Control configuration of a Breakdown, they will not be able to see that Breakdown in the Analytics Hub or apply that breakdown to a Breakdown widget. Time series widgets and Dashboard Breakdown Source filters will still show data from that Breakdown.

Access to the elements of a Breakdown list is controlled using **Security lists**:

- Blacklist (default) – specifies exclusions
- Whitelist – specifies inclusions

The Security list type is defined on the Breakdown source.



Security – Breakdown Element Security - Blacklist

Blacklist Breakdown Element Security Lists define:

Breakdown Elements And the Roles who DO NOT HAVE access to them

Roles

Blacklist

Roles on the list have **no** access to the red elements

Roles not on the list are not affected

Elements on the list (red) are **excluded** from these roles

Elements

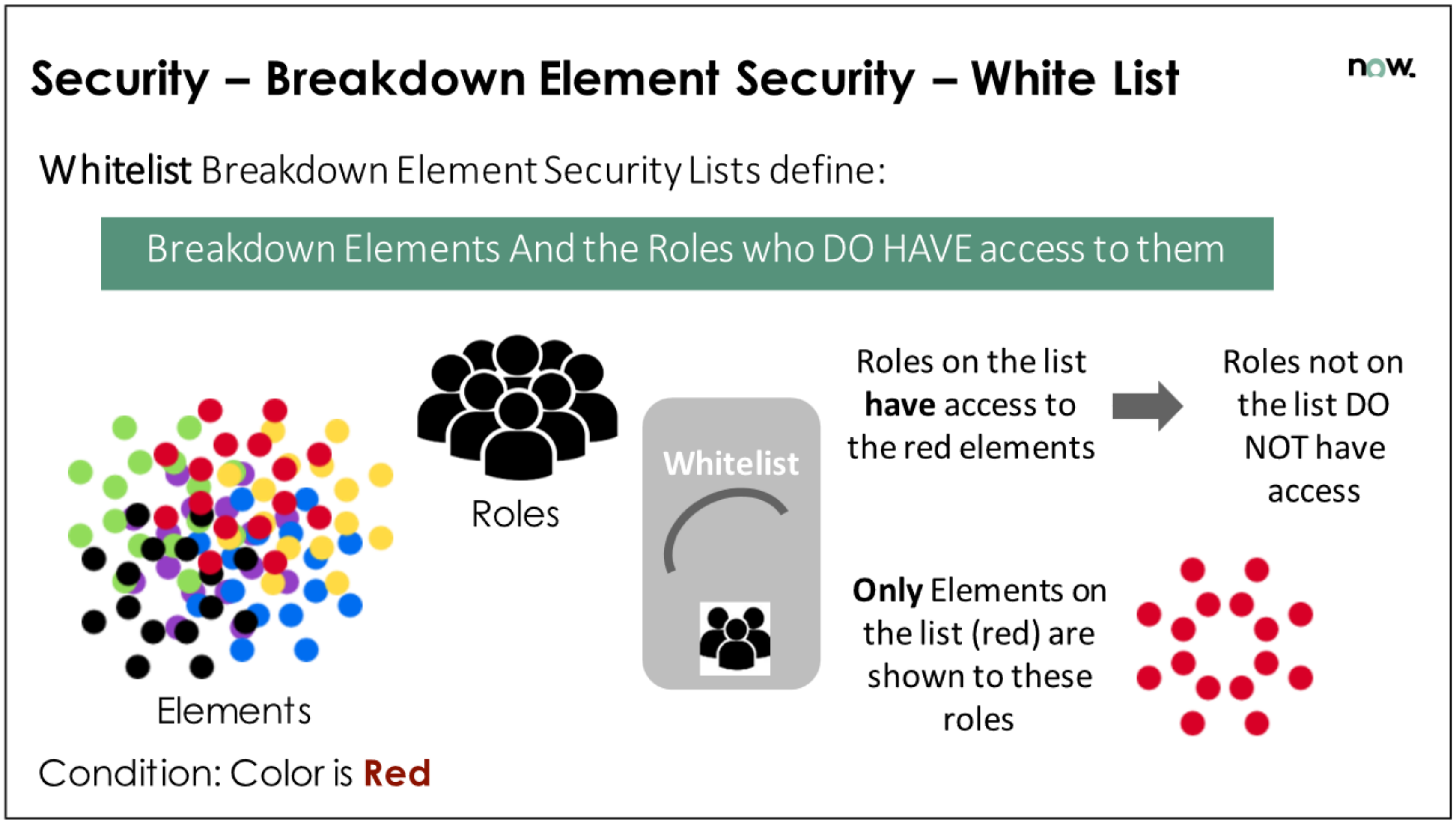
Condition: Color is **Red**

Limiting the visibility of Performance Analytics data can be achieved at the Dashboard and the Records level via:

- Roles – there are 8 out of the box PA roles that allows various level of access to the application components.
- Indicator-Level Security via the Access control tab which allows to select Everyone (the default) or specific Groups and Users

Note that if Indicator access is restricted, the indicator will not show in the scorecard list (widget). However the indicator will show when viewing that Indicator on a widget in a dashboard.

- Breakdown-Level Security – the most granular approach to data Security that allows per-Element specification of access
- A Blacklist Element Security specifies the Roles and the specific Breakdown Elements that these roles ARE NOT allowed to view.



Security – Breakdown Element Security – White List

Whitelist Breakdown Element Security Lists define:

Breakdown Elements And the Roles who DO HAVE access to them

Elements

Roles

Whitelist

Roles on the list **have** access to the red elements

Roles not on the list DO NOT have access

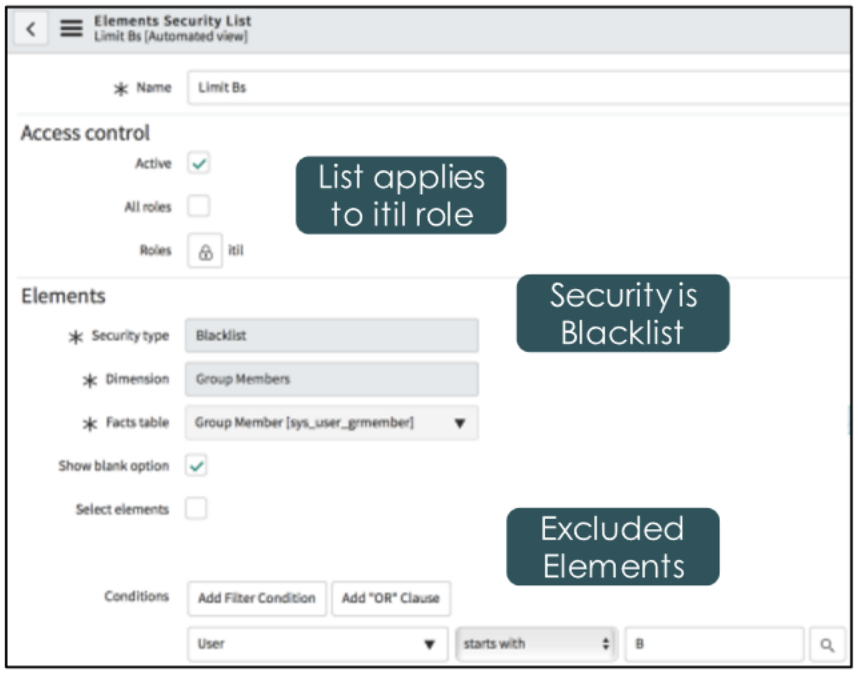
Only Elements on the list (red) are shown to these roles

Condition: Color is **Red**

A Whitelist Element Security specifies the Roles and the specific Breakdown Elements that these roles ARE allowed to view. The specified roles have access ONLY to the specified elements.

Blacklist Example

ITIL role cannot see B* Elements



Elements Security List
Limit Bs [Automated view]

* Name: Limit Bs

Access control

Active ☑

All roles □

Roles: itil

List applies to itil role

Elements

* Security type: Blacklist

* Dimension: Group Members

* Facts table: Group Member [sys_user_grmember]

Show blank option ☑

Select elements □

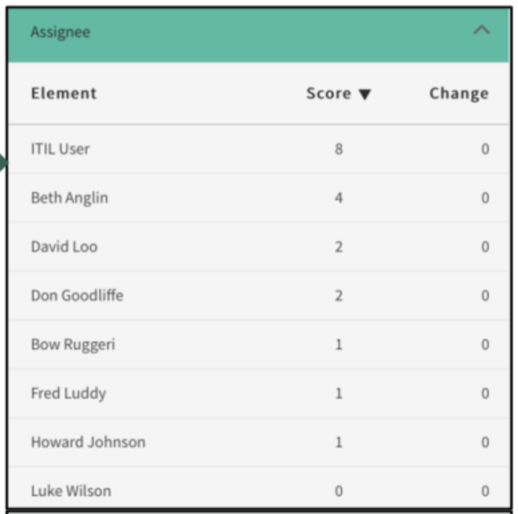
Security is Blacklist

Excluded Elements

Conditions: Add Filter Condition | Add "OR" Clause

User | starts with | B

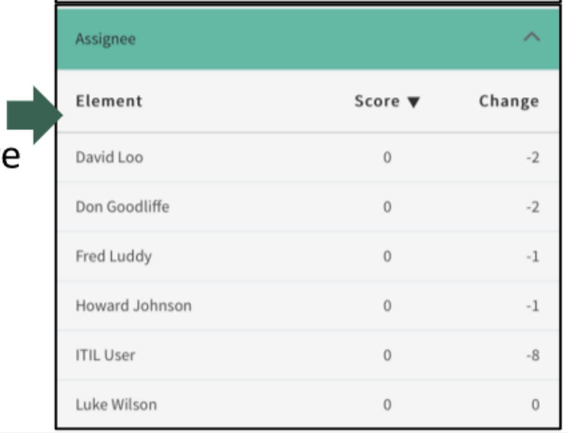
No Security List – all Assignees are visible



Assignee

Element	Score ▼	Change
ITIL User	8	0
Beth Anglin	4	0
David Loo	2	0
Don Goodliffe	2	0
Bow Ruggeri	1	0
Fred Luddy	1	0
Howard Johnson	1	0
Luke Wilson	0	0

Assignees with first name B* are missing



Assignee

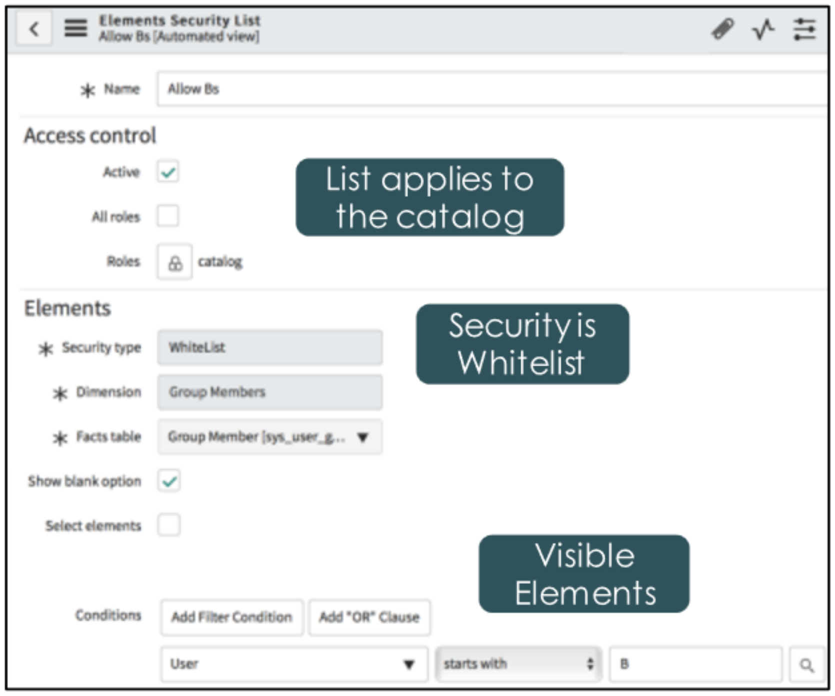
Element	Score ▼	Change
David Loo	0	-2
Don Goodliffe	0	-2
Fred Luddy	0	-1
Howard Johnson	0	-1
ITIL User	0	-8
Luke Wilson	0	0

The default security type for an Elements Security list is Blacklist. Here is how the Blacklist would work in the example above:

- Without an Elements Security list, anyone can see all Elements of the Assignee Breakdown
- With the Elements Security list shown above, the itil role cannot view Assignees whose name begins with B

Whitelist Example

Catalog role can see B* Elements only



Elements Security List
Allow Bs [Automated view]

* Name	Allow Bs

Access control

Active	☑
All roles	□
Roles	catalog

Elements

* Security type	WhiteList
* Dimension	Group Members
* Facts table	Group Member [sys_user_g...
Show blank option	☑
Select elements	□

Conditions: Add Filter Condition | Add "OR" Clause

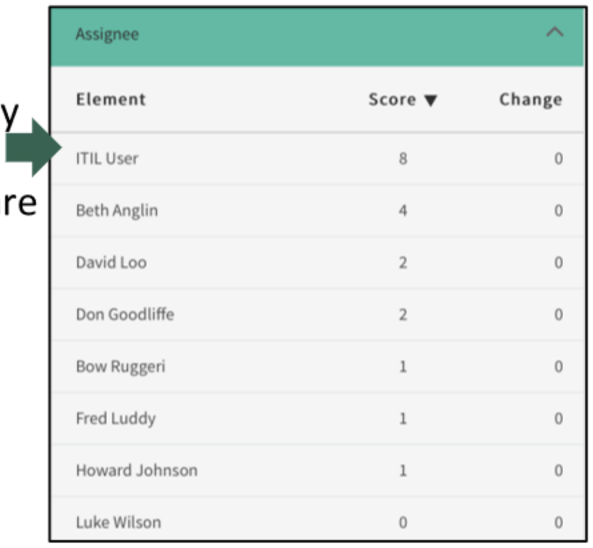
User | starts with | B

List applies to the catalog

Security is Whitelist

Visible Elements

Before Whitelist

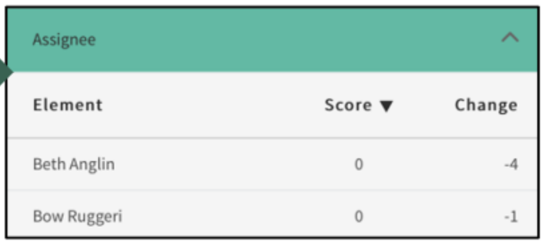


Assignee

Element	Score ▼	Change
ITIL User	8	0
Beth Anglin	4	0
David Loo	2	0
Don Goodliffe	2	0
Bow Ruggeri	1	0
Fred Luddy	1	0
Howard Johnson	1	0
Luke Wilson	0	0

No Security List – all Assignees are visible

After Whitelist



Assignee

Element	Score ▼	Change
Beth Anglin	0	-4
Bow Ruggeri	0	-1

Only Assignees with first name B* are shown

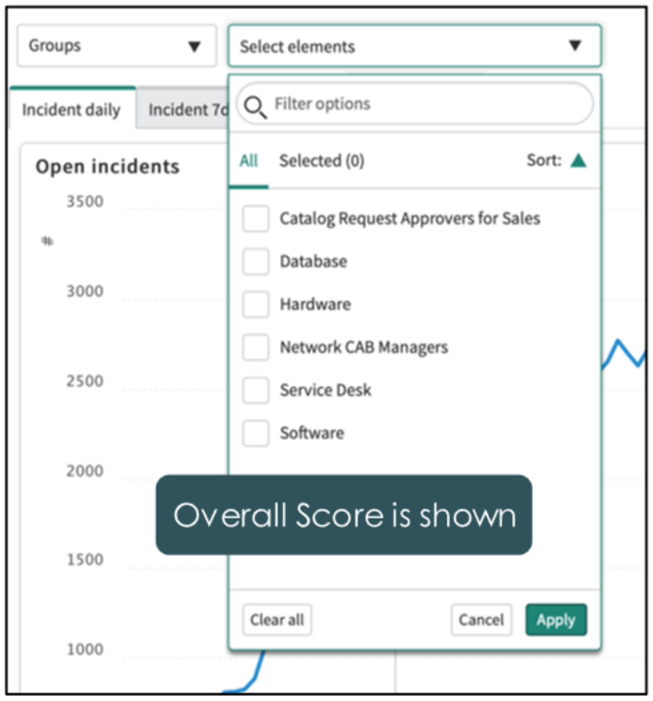
Here is how the Whitelist would work in the example above:

- Without an Elements Security list, anyone can see all Elements of the Assignee Breakdown
- With the Elements Security list above, the catalog role can only view Assigned To's whose name begins with B

Caution! Forgetting to add a role to a Whitelist definition will prevent all data from being shown on a dashboard!

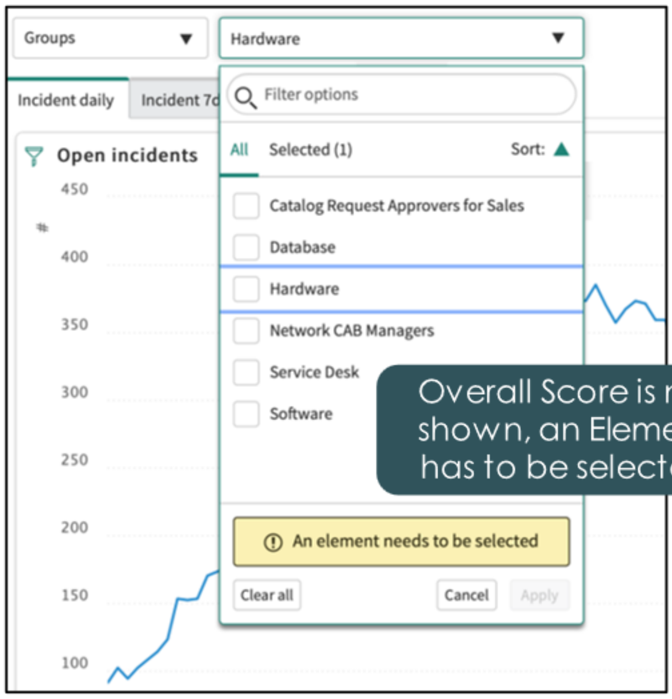
Show Blank Option

Show Blank Option is ON



Overall Score is shown

Show Blank Option is OFF



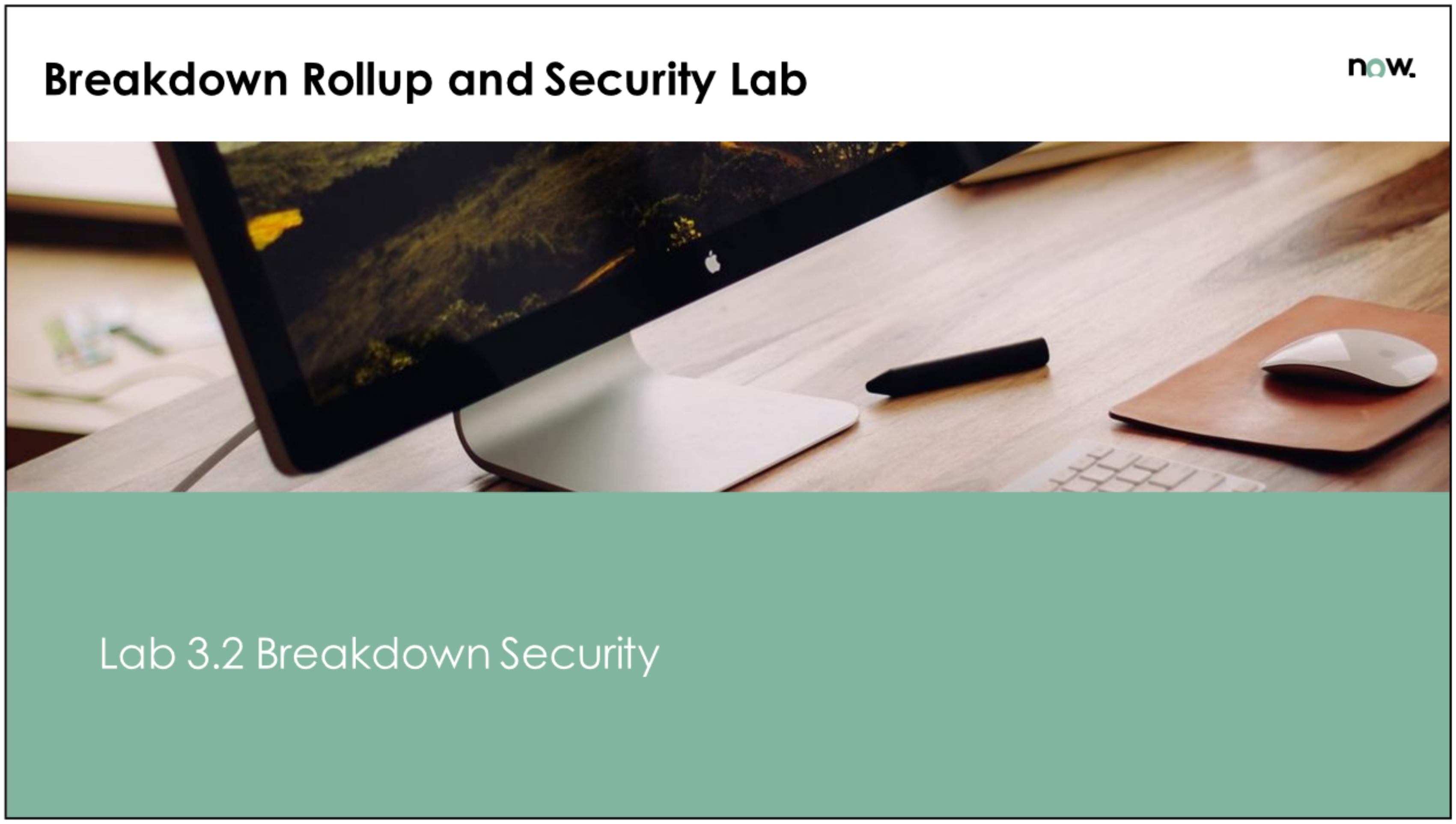
An element needs to be selected

Overall Score is not shown, an Element has to be selected

The **Show blank option** is a property on the Elements Security list which let you tailor the level of access to Breakdown elements as follows:

- Select the **Show blank option** checkbox to allow a user on a breakdown dashboard to see scores without any breakdown elements specified.
- Clear the **Show blank option** checkbox to allow a user on a breakdown dashboard to see only scores for the breakdown elements that are visible to their role.

Note that this setting affects only widgets that follow breakdown dashboard elements. Users with the admin role can always see unfiltered scores on breakdown dashboards.



Breakdown Rollup and Security Lab

Lab 3.2 Breakdown Security

Breakdown Security

Lab 3.2

⌛10m

Lab Objectives

The Incident Manager wants to limit visibility of Incidents assigned to specific groups or users to only the managers of those groups and users. To accomplish this, you perform the following:

- Create an Elements Security list on a Breakdown source
- Limit Group and Assignee breakdown visibility to only their respective managers

Breakdown Security for Groups

A. Groups Breakdown Source

Add New Security List

1. Navigate to **Performance Analytics > Sources > Breakdown Sources**.

2. Open the **Groups** Breakdown Source.

3. Navigate to the **Security** tab.

4. Set the **Security type** to **Whitelist**.

5. **Save** the Breakdown Source.

6. Scroll down to the **Elements Security List** Related List and click **New**.

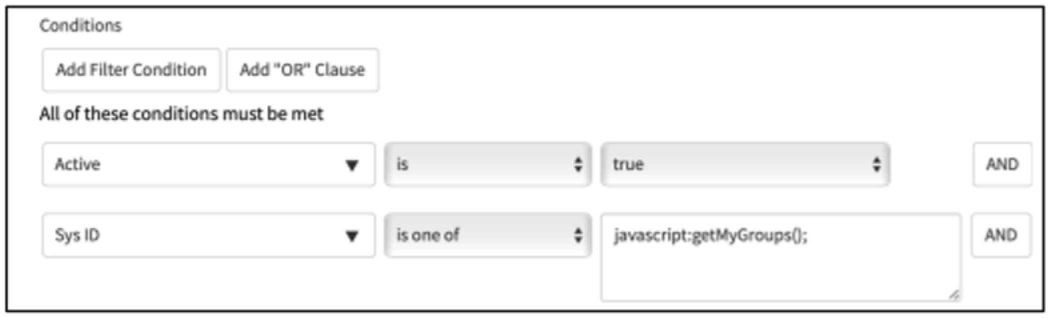
7. Create a new Elements Security List as follows:
   Name: **I am Member of**
   Active: **checked**
   All Roles: **unchecked**
   Role: **catalog**
   Show blank option: **unchecked**

*Tip: Disabling **Show blank option** prevents the viewer from seeing the overall Indicator score.*

Select elements: **unchecked**

Conditions:

Active	**is**	**true**	AND
Sys ID	**is one of**	**javascript:getMyGroups();**	


Conditions

All of these conditions must be met

Active is true AND

Sys ID is one of javascript:getMyGroups(); AND

***Note:** The JavaScript returns all groups managed by the logged in user.*

8. Click **Submit**.

B. Security List Validation

User and Role Configuration

1. Navigate to **System Security > Users and Groups > Users**.
2. Open the user **David Dan**.
3. Assign these two roles to **David Dan**: **pa_viewer** and **catalog**.
4. Confirm that David Dan is a member of these groups: **Hardware**, **Network**
5. Return to the **Users** list and open the user **Beth Anglin**.
6. Assign the **pa_viewer** role to **Beth Anglin**.
7. Review the six groups that Beth Anglin is a member of**.**

Security Validation

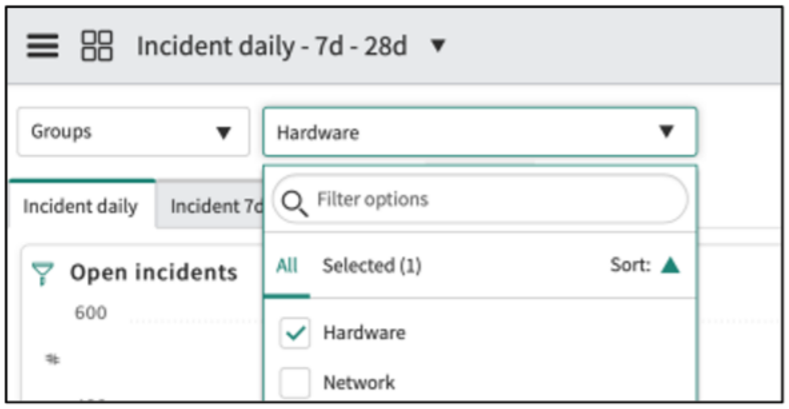
In this step, you validate that the Groups Breakdown source filters widgets as expected.

1. Navigate to **Performance Analytics > Dashboards**.
2. Open the **Incident daily - 7d – 28d** dashboard.
3. View the **Sharing** panel and confirm that the dashboard is shared with **pa_viewer**.

4. Impersonate the user **David Dan**.

5. Navigate to the **Incident daily - 7d – 28d** dashboard.

6. Confirm that David Dan can only view Incident scores for **Groups** he is a member of.



Question: *What happens when you try to clear all Breakdown elements and why?*

7. Impersonate **Beth Anglin.**

8. Navigate to the Dashboard **Incident daily - 7d – 28d**.

9. Confirm that Beth Anglin can only view scores for groups that she is a member of.

Make Overall Score Visible

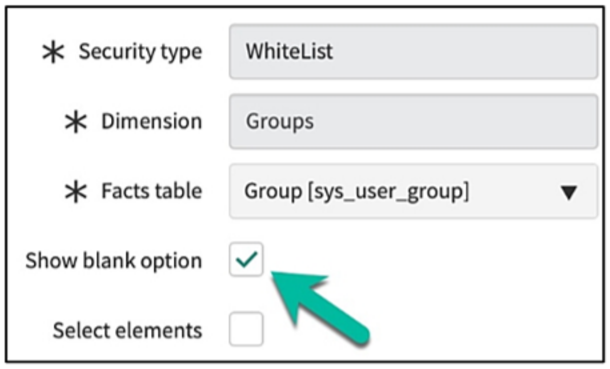
In this step, you update the **Groups** security to allow for the overall score to be shown.

1. Impersonate the **System Administrator**.

2. Open **Groups** Breakdown Source.

3. Open the **I am Member of** Elements Security List.

4. Enable the **Show blank option** and click **Update**.

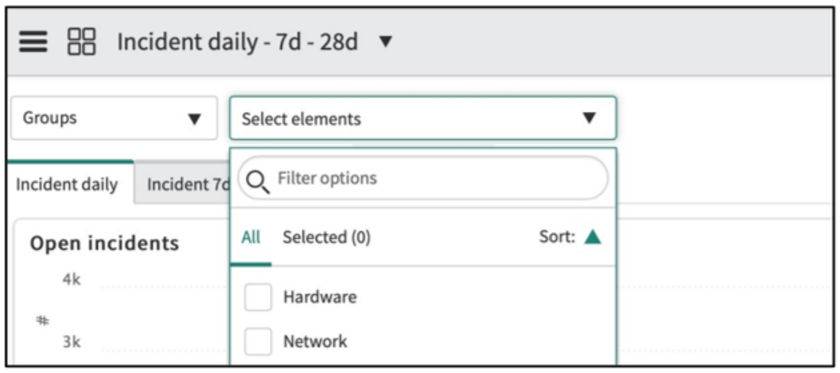


* Security type	WhiteList
* Dimension	Groups
* Facts table	Group [sys_user_group]
Show blank option	☑
Select elements	□

5. Impersonate either **David Dan** or **Beth Anglin** and open **Incident daily - 7d – 28d**.

6. Select the **Groups** Breakdown Source and **Clear all** Breakdown Element filters.

7. Confirm that the widgets display scores for the indicator overall.



Breakdown Security for Assignees

C. Users.Active Breakdown Source

Add New Security List

In this step, you add security to limit the visibility of the **Users.Active** Breakdown elements.

1. Impersonate the **System Administrator**.

2. Open the **Users.Active** Breakdown Source and view the **Security** tab.

3. Set the **Security type** to **Whitelist**.

4. **Save** the Breakdown Source.

5. Create a new **Elements Security List** as follows:

Name: **My team**

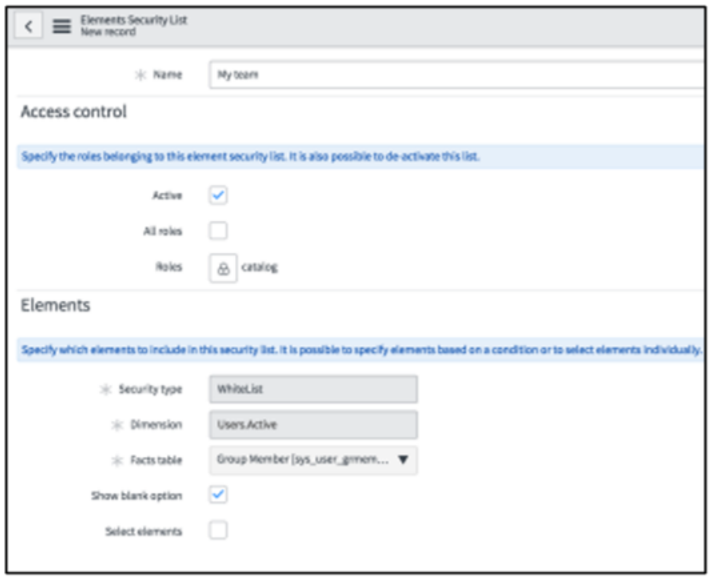
Active: **checked**

All Roles: **unchecked**

Role: **catalog**

Show blank option: **checked**

Select elements: **unchecked**


Elements Security List
New record

Name: My team

Access control

Specify the roles belonging to this element security list. It is also possible to de-activate this list.

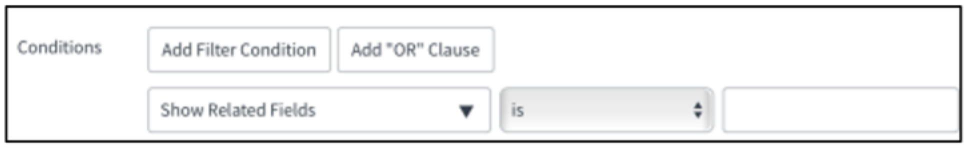
Active ☑
All roles □
Roles catalog

Elements

Specify which elements to include in this security list. It is possible to specify elements based on a condition or to select elements individually.

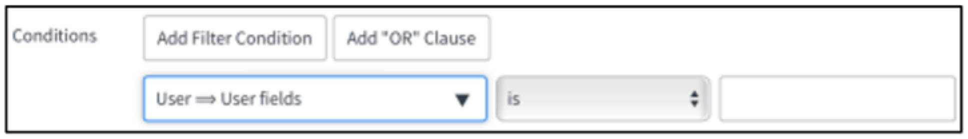
Security type: WhiteList
Dimension: Users.Active
Facts table: Group Member [sys_user_grmem... ▼
Show blank option ☑
Select elements □

6. Under **Conditions**, select **Show Related fields** (last in the dropdown) to enable the display of User related fields.



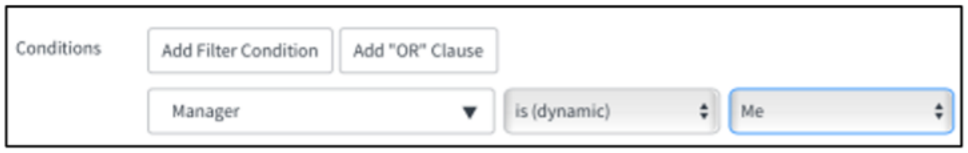
Conditions
Show Related Fields ▼ | is

7. Set the Condition to: **User -> User fields**



Conditions
User ⟹ User fields ▼ | is

8. Set the Condition to: **Manager is (dynamic) Me**



Conditions
Manager ▼ | is (dynamic) | Me

9. Click **Submit**.

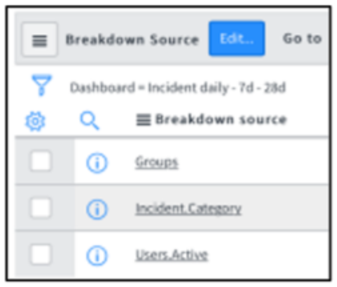
Security List Test

In this step, you add the **Users.Active** Breakdown source to the **Incident daily - 7d – 28d** dashboard and test to make sure that it filters User data correctly.

1. Open the **Incident daily - 7d – 28d** dashboard.

2. Click **Dashboard Controls** and select **Dashboard properties**.

3. Navigate to the **Breakdown Sources** Related List.



Dashboard = Incident daily - 7d - 28d

Breakdown source
Groups
Incident.Category
Users.Active

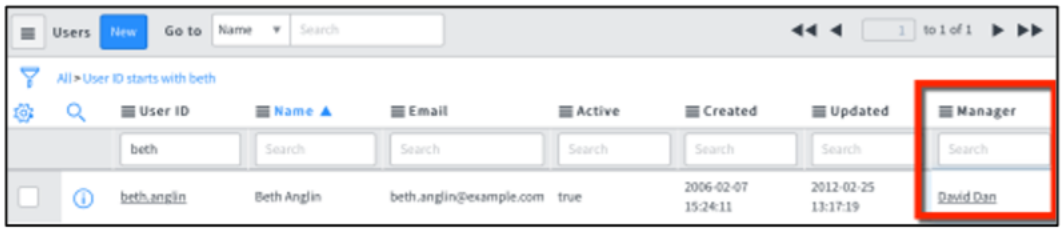
4. Add the **Users.Active** Breakdown Source.

5. Navigate to **System Security > Users and Groups > Users**.

6. Locate the **beth.anglin** user record.

7. Configure the Users list to show the Manager field.

8. Set the **Manager** field to **David Dan**.



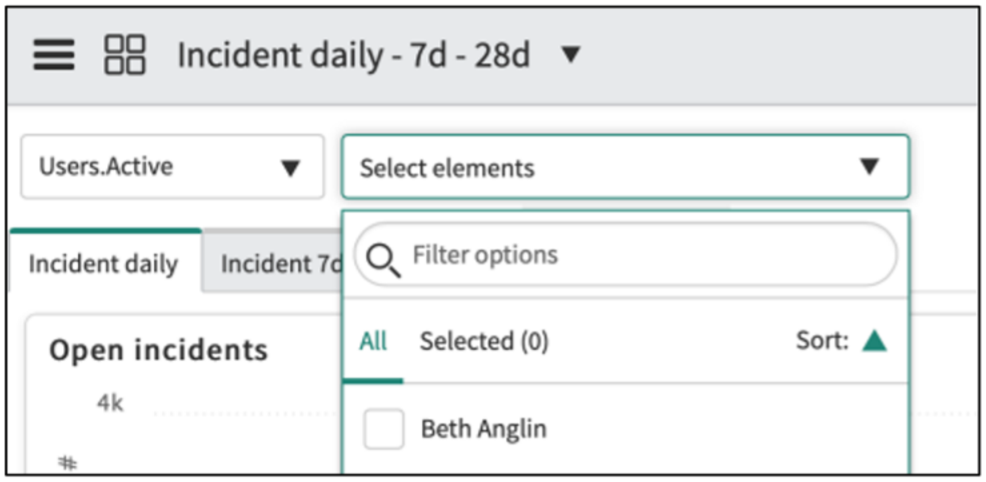
All > User ID starts with beth

		User ID	Name ▲	Email	Active	Created	Updated	Manager
		beth	Search	Search	Search	Search	Search	Search
□	ⓘ	beth.anglin	Beth Anglin	beth.anglin@example.com	true	2006-02-07 15:24:11	2012-02-25 13:17:19	David Dan

9. Impersonate **David Dan**.

10. Open the **Incident daily - 7d – 28d** dashboard.

11. Confirm that only **Beth Anglin** is selectable in the **Users.Active** dropdown.



Users.Active ▼

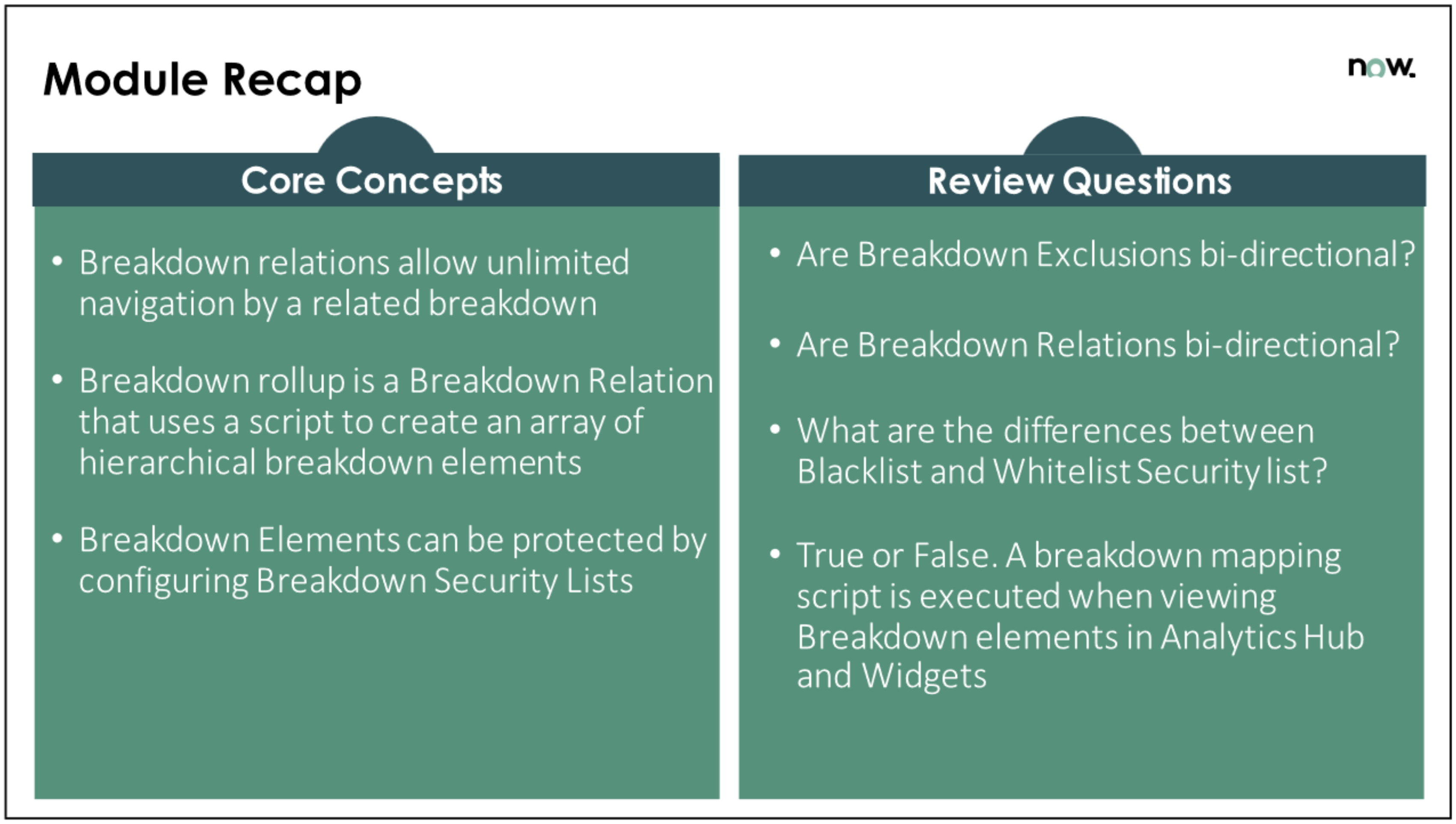
Select elements ▼

□ Beth Anglin

Tip: *To make sure that the above works for all Managers regardless of role, do not specify a role in the definition of Security Lists.*

12. Impersonate the **System Administrator** again.

You have now completed the Breakdown Security lab.



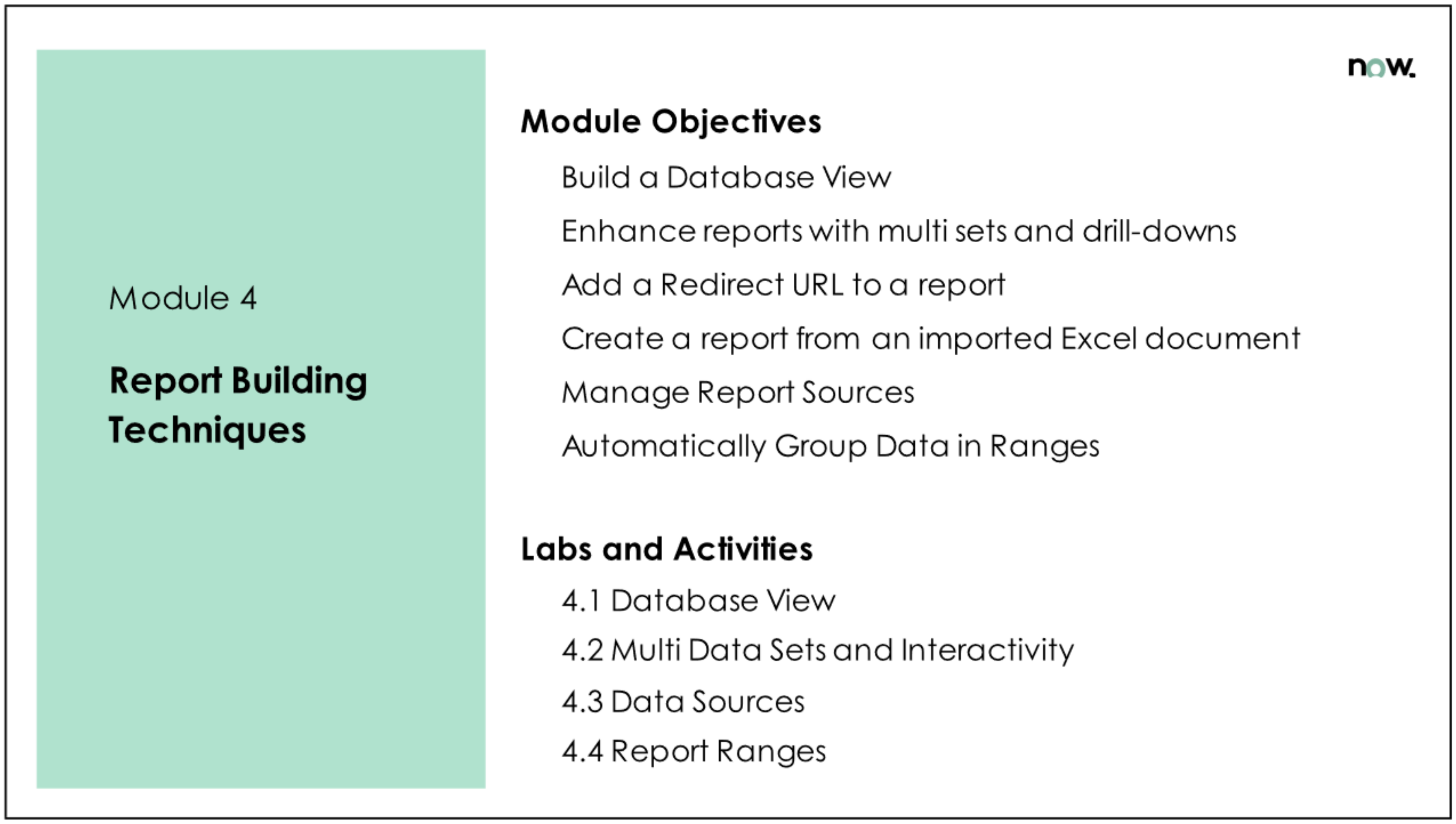
Module Recap

Core Concepts

- Breakdown relations allow unlimited navigation by a related breakdown
- Breakdown rollup is a Breakdown Relation that uses a script to create an array of hierarchical breakdown elements
- Breakdown Elements can be protected by configuring Breakdown Security Lists

Review Questions

- Are Breakdown Exclusions bi-directional?
- Are Breakdown Relations bi-directional?
- What are the differences between Blacklist and Whitelist Security list?
- True or False. A breakdown mapping script is executed when viewing Breakdown elements in Analytics Hub and Widgets



Module 4

Report Building Techniques

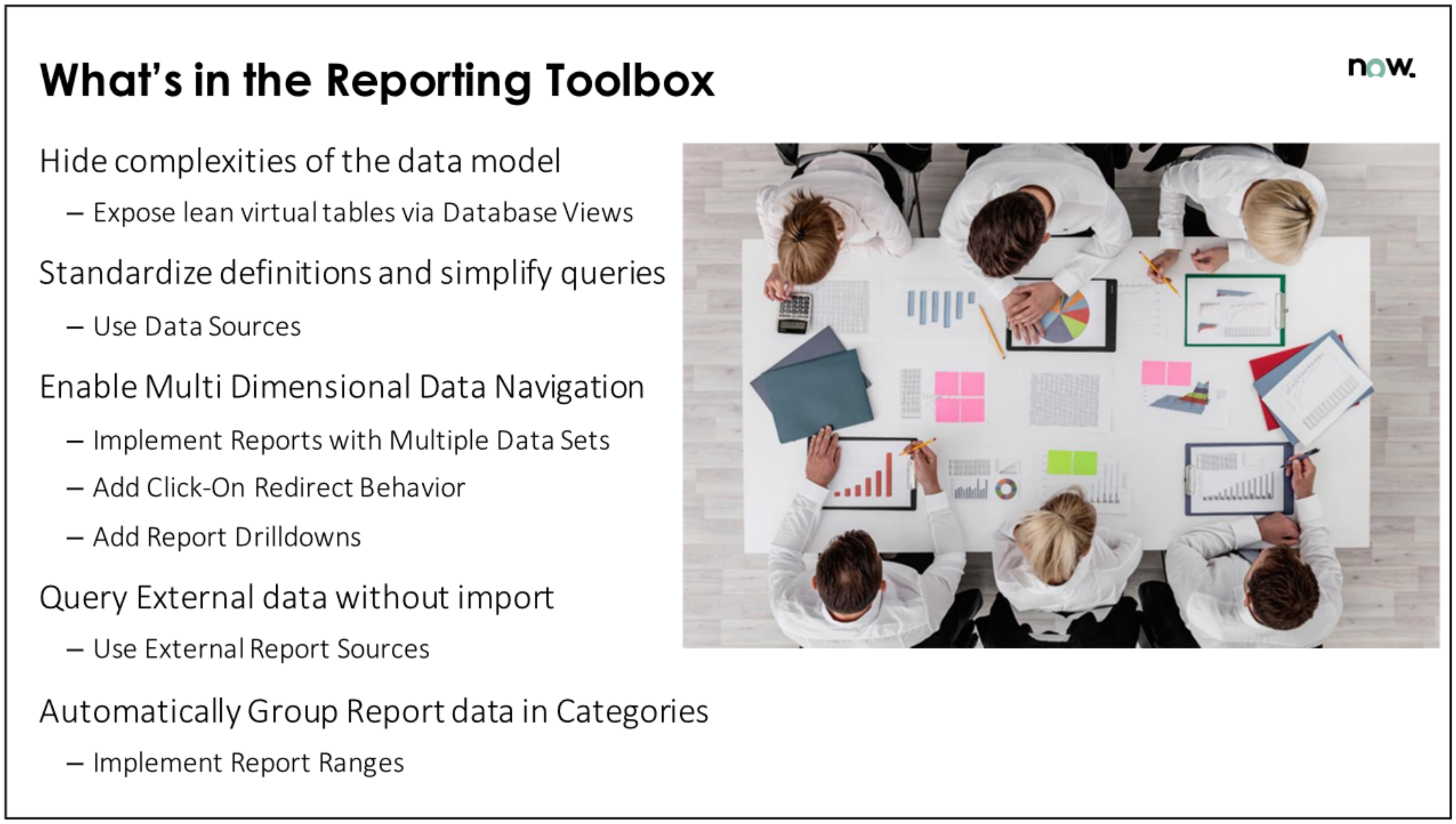
Module Objectives

- Build a Database View
- Enhance reports with multi sets and drill-downs
- Add a Redirect URL to a report
- Create a report from an imported Excel document
- Manage Report Sources
- Automatically Group Data in Ranges

Labs and Activities

- 4.1 Database View
- 4.2 Multi Data Sets and Interactivity
- 4.3 Data Sources
- 4.4 Report Ranges

This module discusses sophisticated reporting techniques which enable powerful data analysis and deeper process insights.



What’s in the Reporting Toolbox

Hide complexities of the data model

- Expose lean virtual tables via Database Views

Standardize definitions and simplify queries

- Use Data Sources

Enable Multi Dimensional Data Navigation

- Implement Reports with Multiple Data Sets
- Add Click-On Redirect Behavior
- Add Report Drilldowns

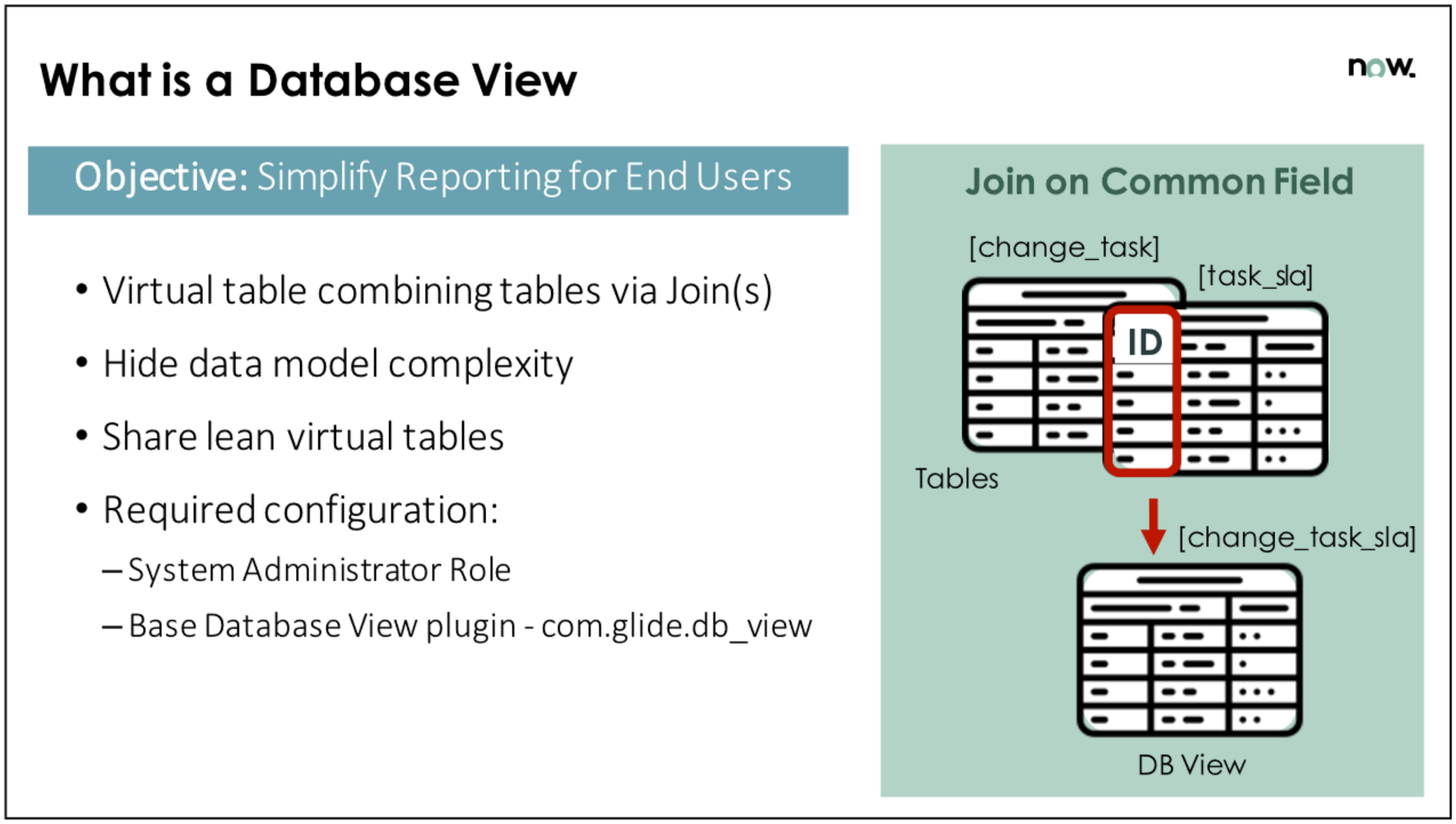
Query External data without import

- Use External Report Sources

Automatically Group Report data in Categories

- Implement Report Ranges

Use the above advanced configuration techniques to build interactive and efficient reports tailored to specific business requirements.



What is a Database View

now.

Objective: Simplify Reporting for End Users

- Virtual table combining tables via Join(s)
- Hide data model complexity
- Share lean virtual tables
- Required configuration:
  - System Administrator Role
  - Base Database View plugin - com.glide.db_view

Join on Common Field

A Database view is a technique that uses database joins to create read-only views that act like regular ServiceNow tables. The main purpose of a database view is to make reporting easier and more intuitive for Report designers. The Database View functionality is only available to System Administrators.

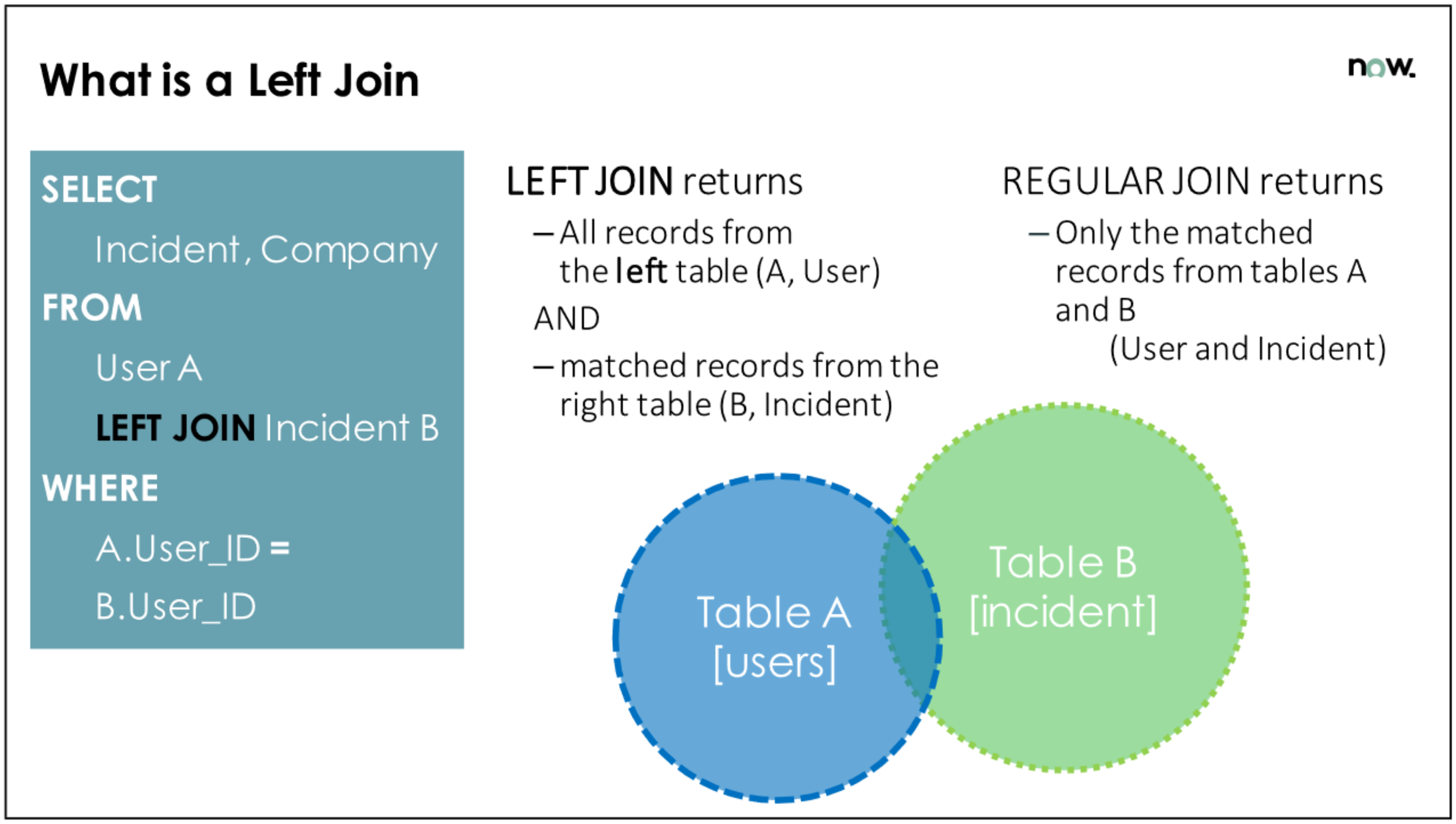
In addition to the base com.glide.db_view plugin, ITSM database views can be deployed by installing the Database Views for Service Management plugin – com.snc.service.db_views. The purpose of these views is to enable easier metric Reporting for ITSM tables.

It is not possible to edit data within a database view.

Here are some examples of using database views:

Join [incident], [metric_definition] and [metric_instance] – Allows you to report on any incident and metric details from either table

Join [change_task] and [task_sla] – Allows you to report on change tasks resolved by SLA



What is a Left Join

now.

```
SELECT
    Incident, Company
FROM
    User A
    LEFT JOIN Incident B
WHERE
    A.User_ID =
    B.User_ID
```

LEFT JOIN returns

- All records from the **left** table (A, User)

AND

- matched records from the right table (B, Incident)

REGULAR JOIN returns

- Only the matched records from tables A and B (User and Incident)

The default Join combines records from both joined tables, if there is a match. No records are returned if there is no match. ServiceNow Database Views may use a technique called **LEFT JOIN.** A **Left Join** returns all rows from the **left** table, even if there are no matches in the right table. The result is NULL from the right side table if there is no match

Example:

Select Incident, Company from User LEFT JOIN Incident

This query returns all User records even if they do not have a corresponding incident in the Incident table.

Incident Time Worked Example

Benefit:

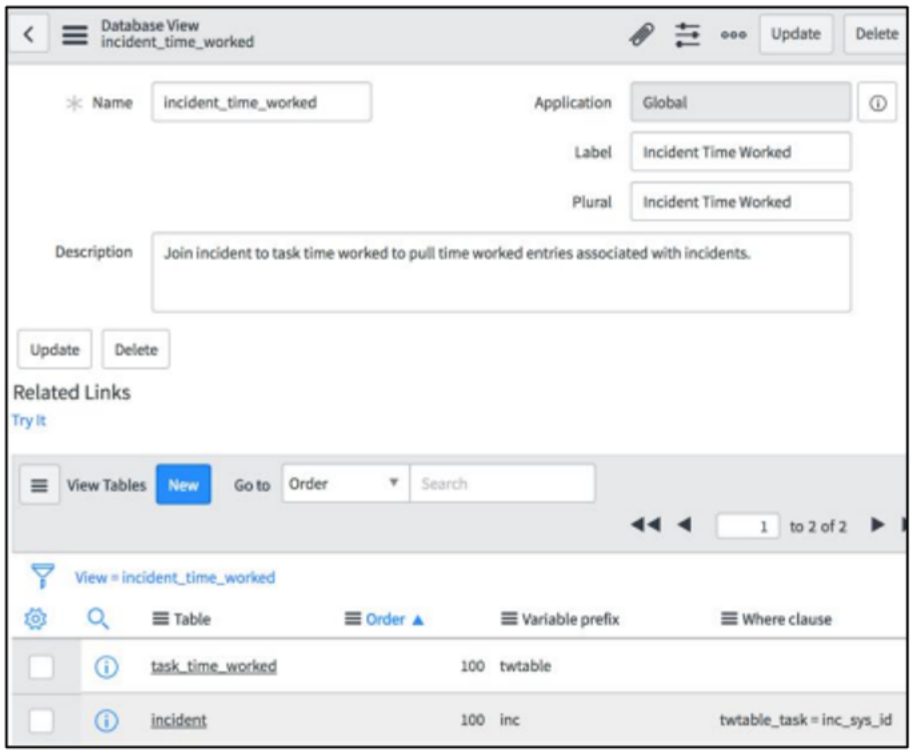
Simplifies reporting on Time Worked

Approach:

DB View joining [incident] and [task_time_worked]

Configuration Steps:

1. Create View
2. Add Tables
3. Define Where Clause
4. Add Fields
5. Test It!



Database View
incident_time_worked

Name: incident_time_worked

Application: Global

Label: Incident Time Worked

Plural: Incident Time Worked

Description: Join incident to task time worked to pull time worked entries associated with incidents.

Related Links
Try It

View = incident_time_worked

Table	Order ▲	Variable prefix	Where clause
task_time_worked	100	twtable	
incident	100	inc	twtable_task = inc_sys_id

Process managers are frequently interested in the time it takes to work respective process tasks. This information is obtainable from two tables – the process table and the [task_time_worked] table using dot-walking.

A System Administrator can build a virtual table for each process owner to simplify the report building and avoid excessive dot-walking. The System Administrator can also decide to restrict the virtual table to only select base table fields. A filter can also be applied so that only select records are shared. For instance, only the Active Tasks. This will minimize the size of the data set you report on and improve performance.

Where Clause

- Specifies how to combine tables
- Supports conditional operators:

  =, !=, <, <=, >, >=, &&, ||
- Separate multiple WHERE clauses with parenthesis {}
- WHERE clause fields must be listed as View Fields
- Add Left Join on the View Table with the WHERE clause, if needed

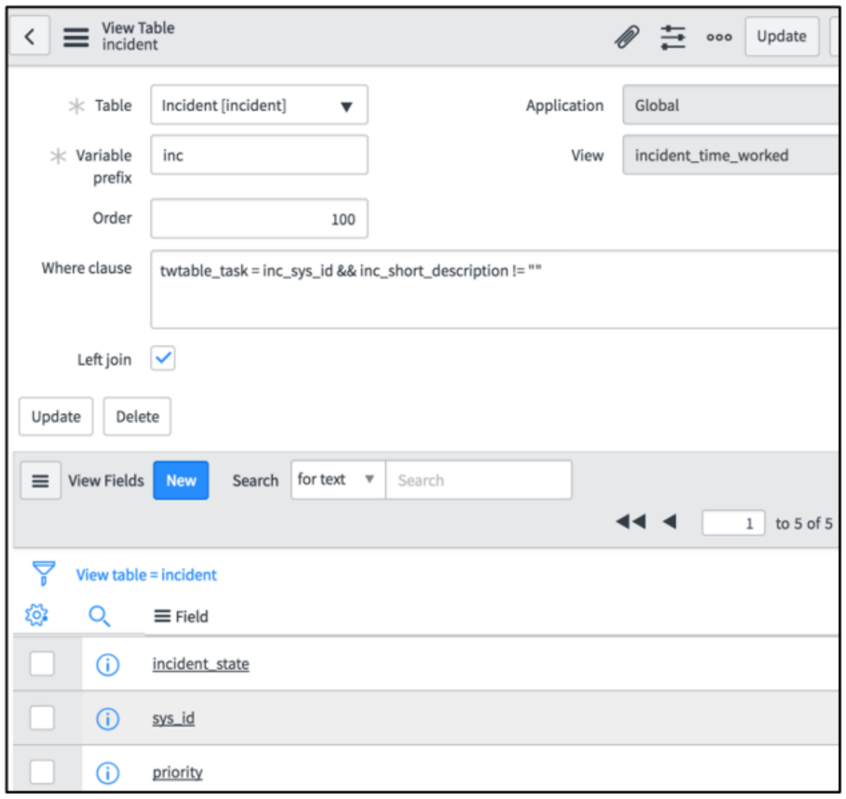


Table: Incident [incident]

Variable prefix: inc

Order: 100

Where clause: twtable_task = inc_sys_id && inc_short_description != ""

Left join: ☑

Application: Global

View: incident_time_worked

Search: for text

1 to 5 of 5

	Field
□	incident_state
□	sys_id
□	priority

To optimize the performance of the database view ensure that the WHERE clauses that are defined in the database view are based on indexed fields.

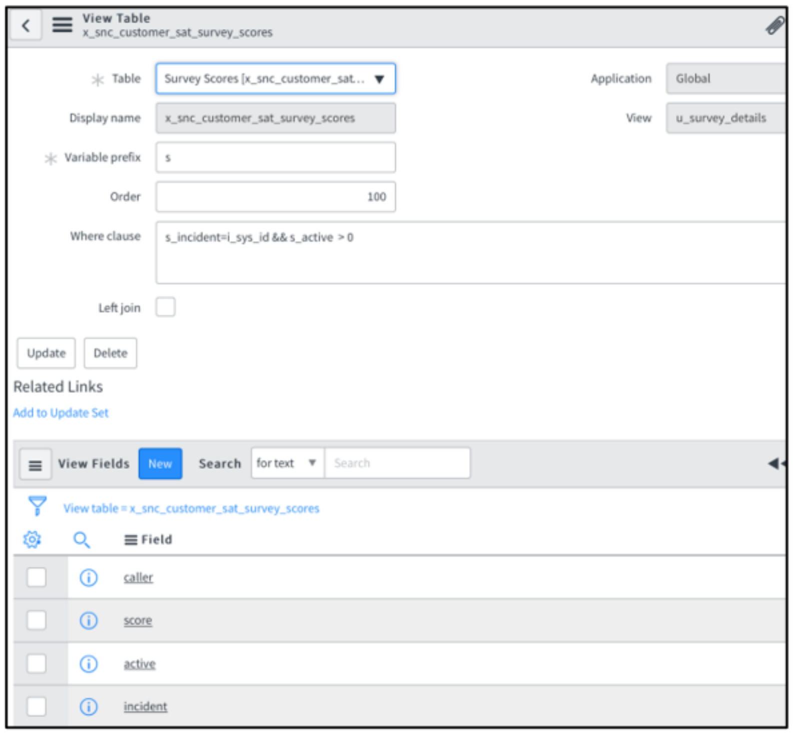
Selecting **Left join** causes the left-hand table in the database view to display all records, even if the join condition does not find a matching record on the right-hand table. Select this check box for view tables that specify a **Where clause**. Selecting **Left join** for view tables without a **Where clause** does not affect the query.

Joined tables are ordered left to right from lowest to highest **Order** values.

As a best practice, use parenthesis in multi-clause joins like this: (md_table = 'incident' || md_table = 'task') && mi_definition = md_sys_id && mi_id = inc_sys_id

Using DB Views as Tables

- Visible as tables in Report Builder
- Available to query in a Script
- Not editable in a list or by a Script
- Only specified View Fields visible in the Report Designer
- Use View Fields to Limit Data Set Size
- Not visible outside of the scoped app they are created in



View Table
x_snc_customer_sat_survey_scores

* Table	Survey Scores [x_snc_customer_sat… ▼	Application	Global
Display name	x_snc_customer_sat_survey_scores	View	u_survey_details
* Variable prefix	s		
Order	100		
Where clause	s_incident=i_sys_id && s_active > 0		
Left join	□		

Update Delete

Related Links

Add to Update Set

View Fields New Search for text ▼ Search

View table = x_snc_customer_sat_survey_scores

	Field
□	caller
□	score
□	active
□	incident

Here are some considerations when working with **View Fields**:

- When creating DB views, use the **View Field** form to specify a field that you want returned by the joined table. Note that if no fields are defined in the **View Fields** list, all fields are returned. If any fields are defined, then only those fields are returned.
- When you restrict the fields returned by creating **View Field** records, you must create a record for the join field from the **Where clause** in the parent record. If you omit a record for this field, it cannot be returned, and the join fails.
    - As an example, if your **Where clause** uses the sys_id field from the Incident table to establish the join, you must include a record for the sys_id field, for the join to succeed.

Advanced Reporting Techniques Lab

Lab 4.1: Database View

Database View

Lab 4.1

⌛20m

Lab Objectives

Managers reviewing surveys require a simple way of retrieving additional information about the Incident for which a survey has been submitted. A database view eliminates the complexities of joining the incident and Survey Scores tables every time additional Incident data is required for analysis.

This lab will show you how to do the following:

- Create a database view
- Troubleshoot the Join Condition in a database view

Database View

A. Identify Database View Components

View Tables

1. Navigate to **System Definition > Database Views**.
2. Click **New** and create a new database view as follows:
   - Name: **Survey Details**
   - Description: **Join Incident and Survey Score tables**
3. Save the view.
4. Navigate to the **View Tables** Related List and click **New**.
5. Add a new View Table as follows:
   - Table: **Incident [incident]**
   - Variable prefix: **i**
   - Where clause: **leave empty**

***Tip:** A Where clause only needs to be added to one of the tables.*

6. Click **Submit**.

7. Click **New** in the **View Tables** Related List to add another View Table as follows:

    - Table: **Survey Scores [u_x_snc_customer_sat_survey_scores]**
    - Variable prefix: **s**
    - Where clause: **s_u_incident = i_sys_id**

    > *This Where clause joins the Incident and Survey Scores tables using the incident sys_id*

8. Click **Submit**.

9. Click the **Try It** Related Link to test the view. Confirm that **250** rows are returned.

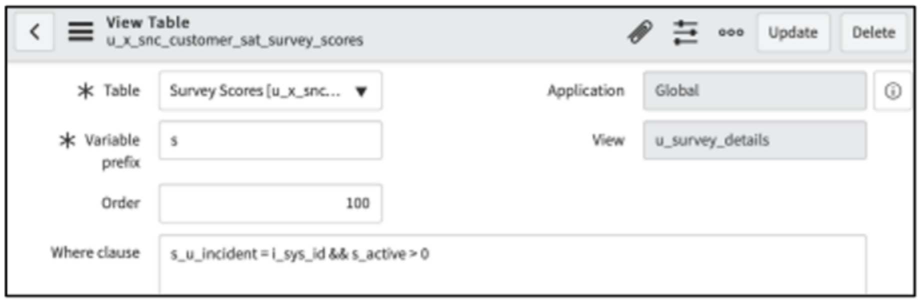
Modify Where Clause

1. Click the browser back button to go back to the **u_survey_details** view.

2. Open the **u_x_snc_customer_sat_survey_scores** View Table.

3. Modify the **Where clause** to return only Active Survey Score records as follows:

    Where clause: **s_u_incident = i_sys_id && s_active > 0**



View Table
u_x_snc_customer_sat_survey_scores

* Table: Survey Scores [u_x_snc...

Application: Global

* Variable prefix: s

View: u_survey_details

Order: 100

Where clause: s_u_incident = i_sys_id && s_active > 0

4. Click **Update** to return to the view.

5. Click the **Try It** Related Link to test the View again. The view should return **250** rows.

    > *Note: The same number of rows is returned as all records are currently Active.*

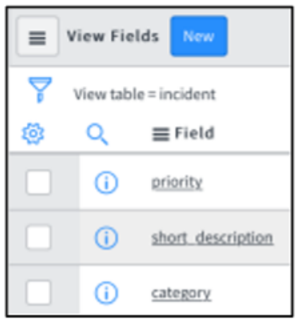
View Fields

1. Navigate back to the **u_survey_details** db view.

2. Open the **incident** View Table.

3. Using the **New** button, add the following View Fields:

   - priority
   - short_description
   - category



View table = incident

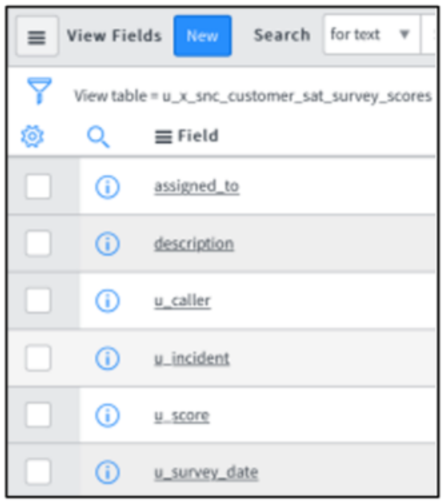
priority
short_description
category

4. Navigate back to the **u_survey_details** db view.

5. Open the **u_x_snc_customer_sat_survey_scores** View Table.

6. Add the following View Fields:

   - **Incident Owner**
   - **Satisfaction Comment**
   - **Caller**
   - **Score**
   - **Survey_Date**
   - **Incident**



View table = u_x_snc_customer_sat_survey_scores

assigned_to
description
u_caller
u_incident
u_score
u_survey_date

7. Return to the view and test it using the **Try It** action.

> ***Note:*** *The view fails to render as you need to add the* ***Active*** *field to the Survey Scores table.*

8. Open the **u_x_snc_customer_sat_survey_scores** View Table.

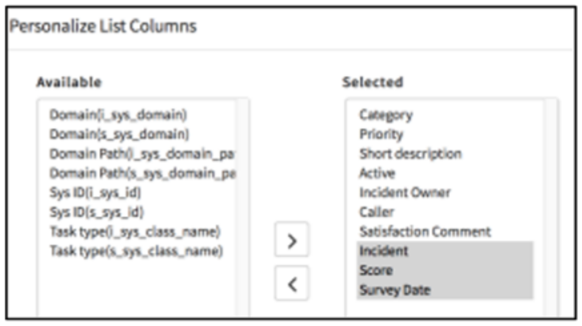
9. Add the **Active** View Field.

10. Test the View using the **Try It** action and confirm there are no errors.

11. Click **Personalize list** button and add the following columns:

Category
Priority
Short Description
Active
Incident Owner

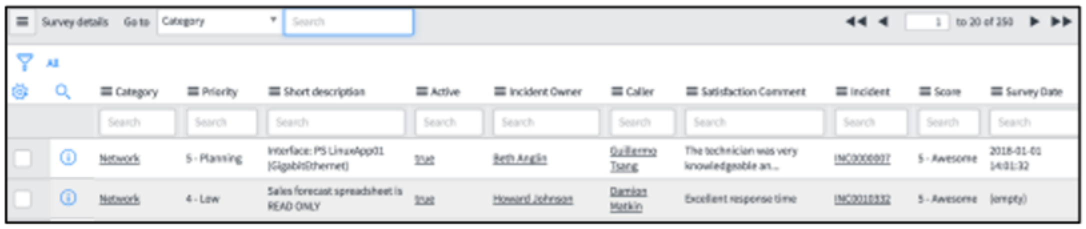
Caller
Satisfaction Comment
Incident
Score
Survey Date



Available:
Domain(i_sys_domain)
Domain(s_sys_domain)
Domain Path(i_sys_domain_pa
Domain Path(s_sys_domain_pa
Sys ID(i_sys_id)
Sys ID(s_sys_id)
Task type(i_sys_class_name)
Task type(s_sys_class_name)

Selected:
Category
Priority
Short description
Active
Incident Owner
Caller
Satisfaction Comment
Incident
Score
Survey Date

12. Verify that the Survey details list view looks like this:



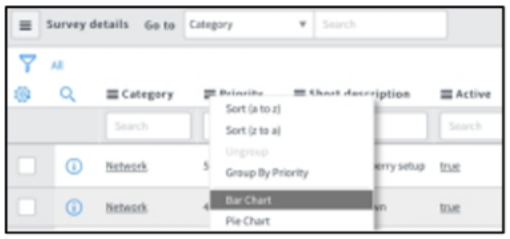
Category	Priority	Short description	Active	Incident Owner	Caller	Satisfaction Comment	Incident	Score	Survey Date
Network	5 - Planning	Interface: PS LinuxApp01 (GigabitEthernet)	true	Beth Anglin	Guillermo Tsang	The technician was very knowledgeable an...	INC0000007	5 - Awesome	2018-01-01 14:01:32
Network	4 - Low	Sales forecast spreadsheet is READ ONLY	true	Howard Johnson	Damian Matkin	Excellent response time	INC0010332	5 - Awesome	(empty)

Create View-Based Report

In this section, you create a report and share it with a non-Admin user so that view fields access can be tested.

1. Remain on the **Survey details** list of record you just viewed.

2. Right-click the **Priority** column menu and select **Bar Chart**.



3. Navigate to the **Data** section of the report builder

4. Change the **Name** to **Survey Details by Priority**.

5. **Save** the Report.

6. Share the report with **David Loo** and save the report again.

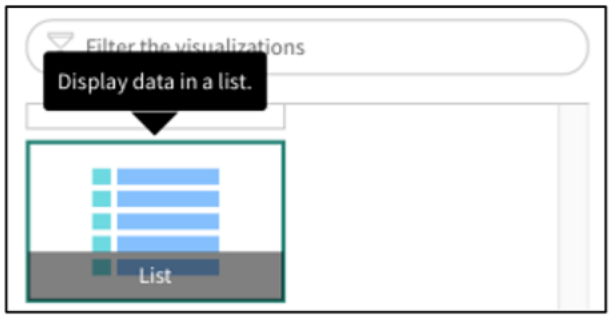
View Reports

1. Impersonate **David Loo**.

2. Navigate to **Reports > View / Run**.

3. Open the report **Survey Details by Priority** (the report can be found under the **Group** section).

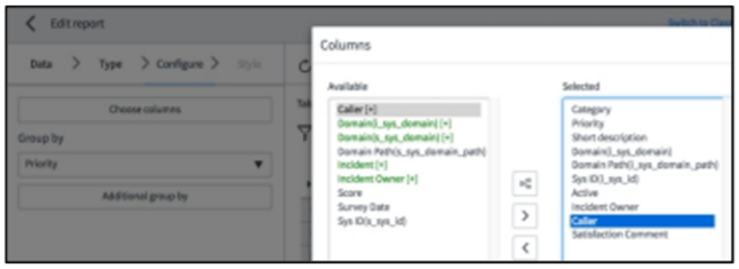
4. Change the report type to **List** and **Run** the report.



5. Navigate to the **Configure** section.

6. Click the **Choose columns** button.

7. Confirm that only shared view fields are available to be selected as report fields.

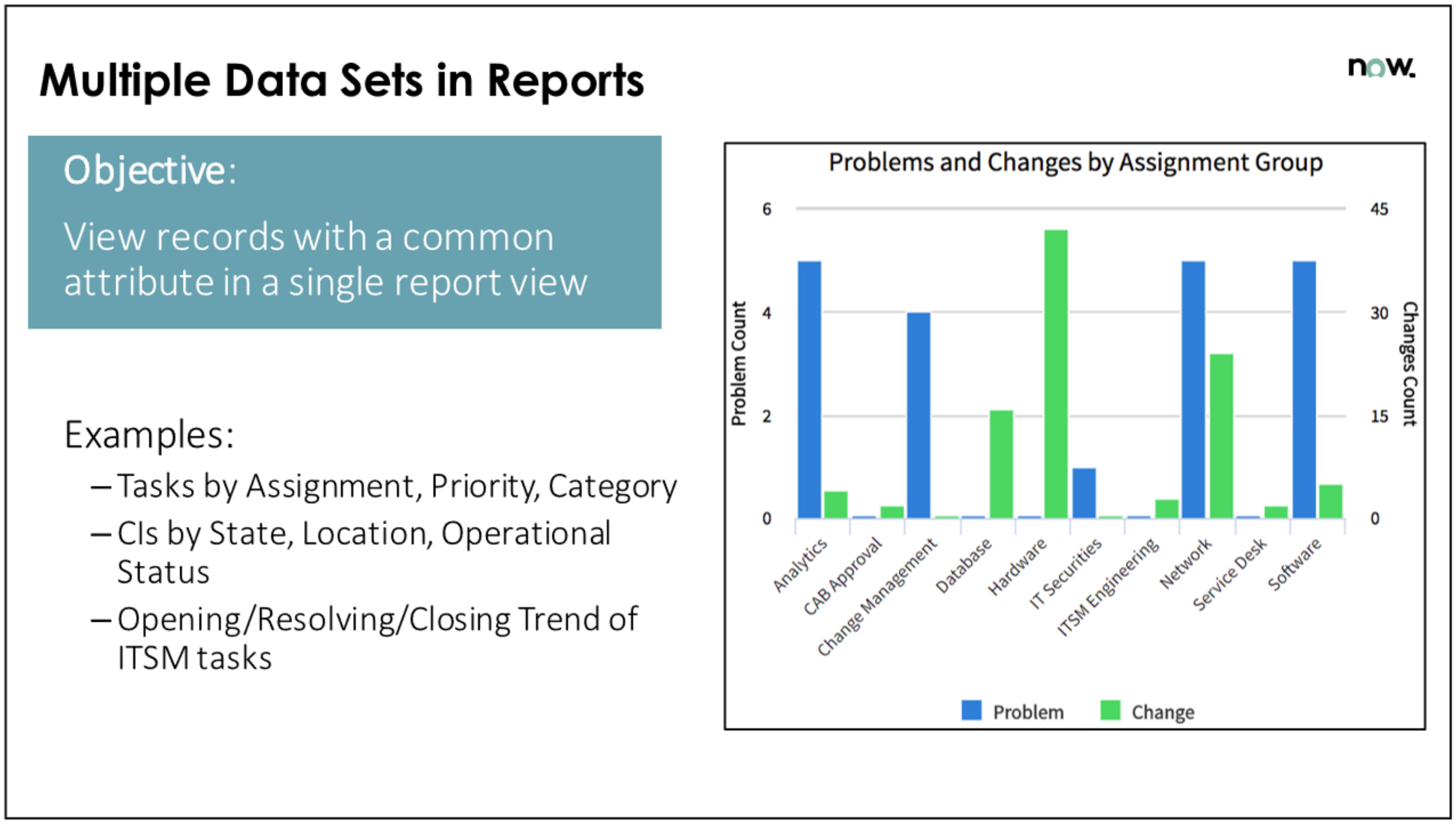


Note: *Only configured View Fields are available for viewing and reporting.*

8. Change the report type back to **Bar** and **Save** the report.

9. Impersonate the **System Administrator**.

You have now completed the Database View lab.



Multiple Data Sets in Reports

Objective:

View records with a common attribute in a single report view

Examples:

- Tasks by Assignment, Priority, Category
- CIs by State, Location, Operational Status
- Opening/Resolving/Closing Trend of ITSM tasks

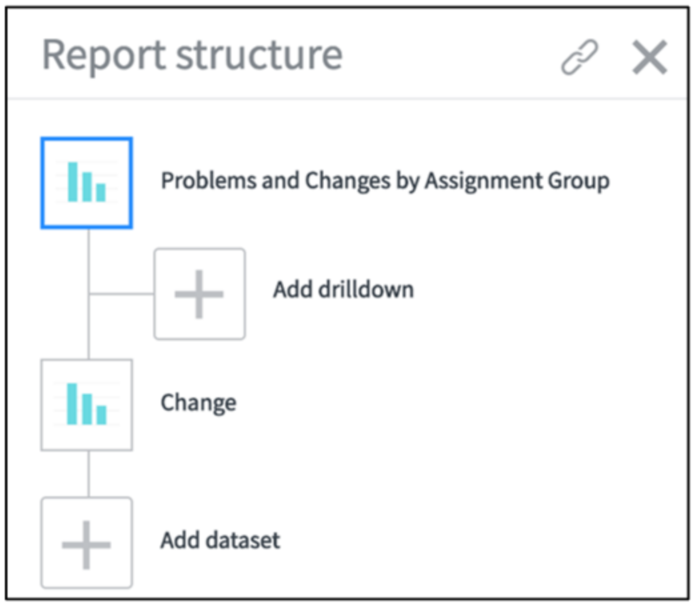
Multiple Data Sets in a report allow you to visualize records from different table sources in the same chart. All data sets must have the same report presentation type – bar chart, trend line, etc.

Note that multiple data sets are no longer a premium Performance Analytics feature.

Multiple Data Set Implementation

Restrictions:

- Common graph type
- Supported graph types:
  - Bar, Horizontal Bar, Line, Column, Area, Spline
- Same Group by/Per attribute for all datasets
- No Multiple Group by
- Only main data set can be saved as data source



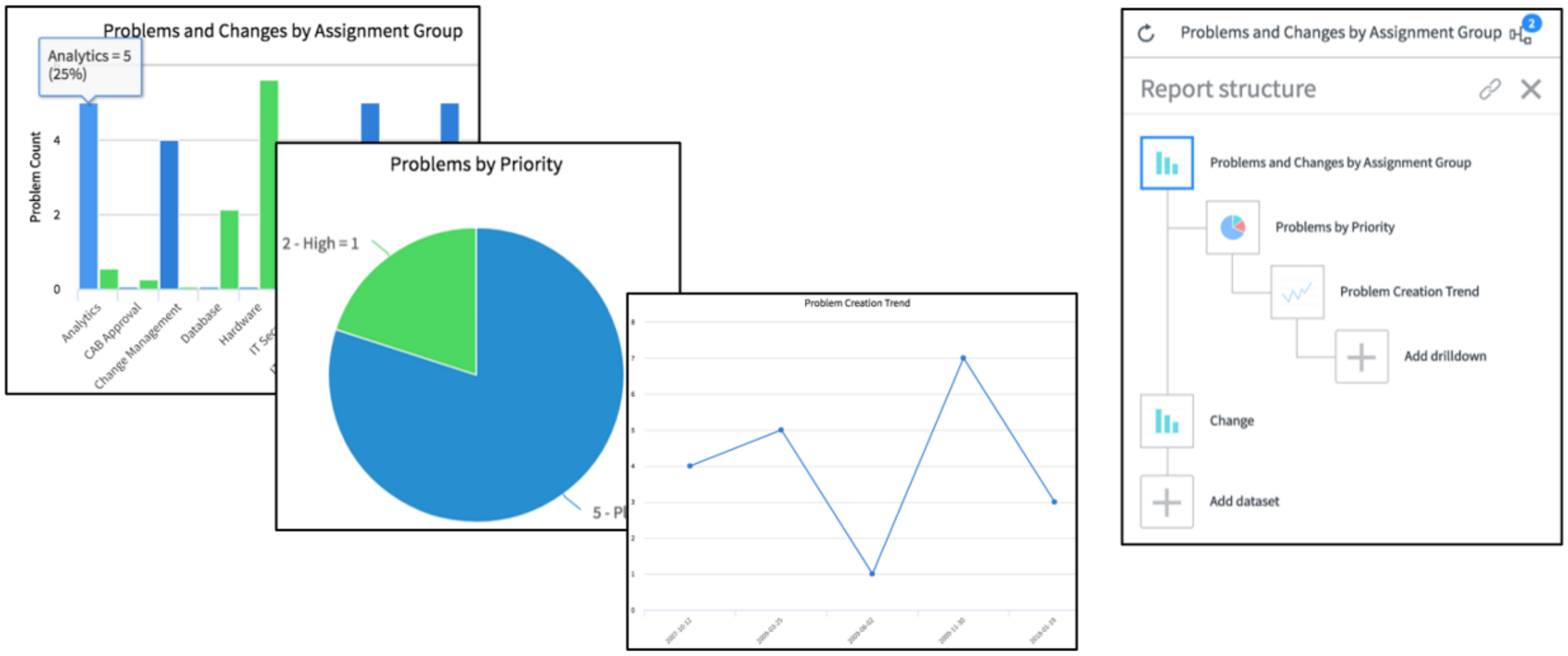
To implement Multiple Data sources and visualize data from multiple tables and sources in a single report, you need to modify the Report Structure as follows:

- Use the **Add dataset** UI Action in the Report Structure to define an additional data set
- If using Time series charts, all datasets must have the same **Per** field value to ensure that the frequency interval is the same for all data
- If using Bar or Horizontal Bar charts, all data must have the same **Group by** value. For example, the data must have the same reference table or the same column name

Note that when using multiple data sets, the report legend is always displayed so that all data sources are clearly visible.

Report Drilldowns

Objective: Customize and enhance widget navigation for Report Viewers



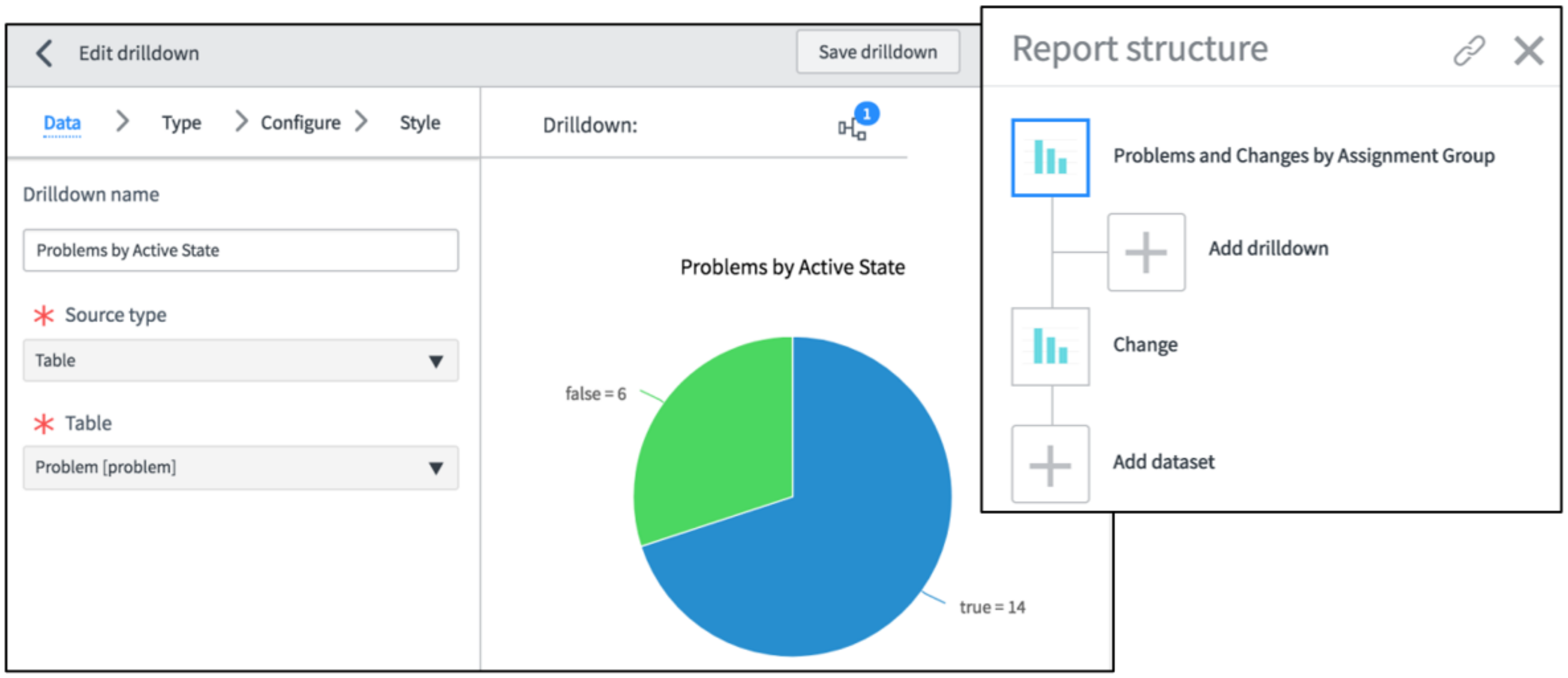
You can define a report drilldown to allow reporting users to view subsets of the report data. In this example, viewers navigate the Problems and Changes by Assignment Group report as follows:

- Drill into the Problem bar and open the Problems by Priority pie chart
- Drill into the pie chart and open the Problems by Creation date trend line

As you can see from the Report Structure, Drilldowns are only defined for the main data set – Problems.

Report Drilldowns

Objective: Customize and enhance widget navigation



Edit drilldown

Drilldown name: Problems by Active State

* Source type: Table

* Table: Problem [problem]

Configuration notes:

- Use **Add drilldown** and **Save drilldown** UI Actions in the Report Structure to configure additional data sets. Individual data set configuration is identical to the configuration of a report.
- The limitation is that when using multi data set, drilldowns are only created on the main set
- When you define a report drilldown, it applies only to the report for which you define it.
- The user can now drill down from the top-level report to the specified drilldown report visualizations.

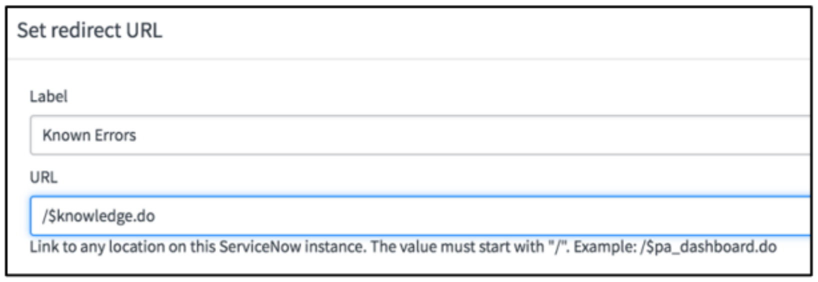
Note that all users can view report visualizations, such as pie charts and column reports. However, the last level of a drilldown is always a list.

Report Redirect URL / On Click Behavior

Objective:

Redirect Report Users to anywhere within the ServiceNow instance

- All drilldowns are disabled
- In the **Set redirect URL** dialog box, enter relative link in the instance
- The URL **Label** provides a helpful hint as a tool tip


Set redirect URL

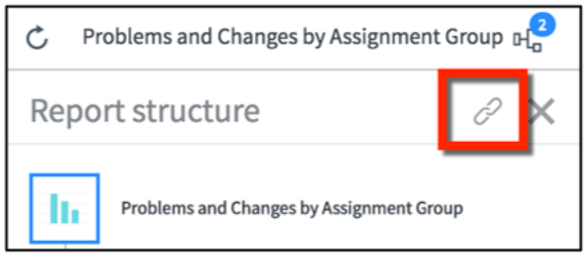
Label

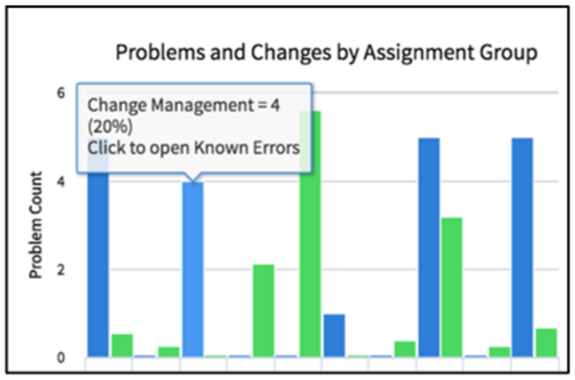
Known Errors

URL

/$knowledge.do

Link to any location on this ServiceNow instance. The value must start with "/". Example: /$pa_dashboard.do


Problems and Changes by Assignment Group
2
Report structure
Problems and Changes by Assignment Group


Problems and Changes by Assignment Group
Change Management = 4
(20%)
Click to open Known Errors
Problem Count
6
4
2
0

A Redirect URL can be configured in the Report Structure which allows the report viewer to browse away from the report to anywhere in the ServiceNow instance.

Note that when the user clicks the report, the redirect URL replaces any drilldown functionality.

Advanced Reporting Techniques Lab

now.

Lab 4.2: Multi Data Sets and Interactivity

Multi Data Sets and Interactivity

Lab
4.2
⌛20m

Lab Objectives

ITSM Managers need reports displaying multiple datasets from various task tables within a single chart. By adding Incidents and Surveys into a single visualization, viewers can immediately see patterns and trends across different processes.

In this lab, you perform the following:

- Create a Multi Data Set Report
- Add interactivity to the report with Drilldowns and Redirect URLs

Multi Data Set Report

A. Incident Data Set

Create Base Report in Report Designer

1. Impersonate the **System Administrator**.
2. Navigate to **Reports > Create New.**
3. Create a new Report as follows:
   - Report name: **Incidents and Surveys**
   - Source type: **Table**
   - Table: **Incident [incident]**
4. Navigate to the **Type** section.
5. Select the **Line** visualization type (under Time Series).

   > ***Tip:*** *When you consider the design, note that only the following report types support multiple data series: Bar / Horizontal Bar, Line, Column, Area, Spline.*

6. Navigate to the **Configure** panel and set the following properties as shown:

Group by: **None**
Trend by: **Opened**
Per: **Date**
Aggregation: **Count**

7. Navigate to the **Style** section and enable **Show marker** to emphasize each data point.

8. Click **Save** to preview the report.

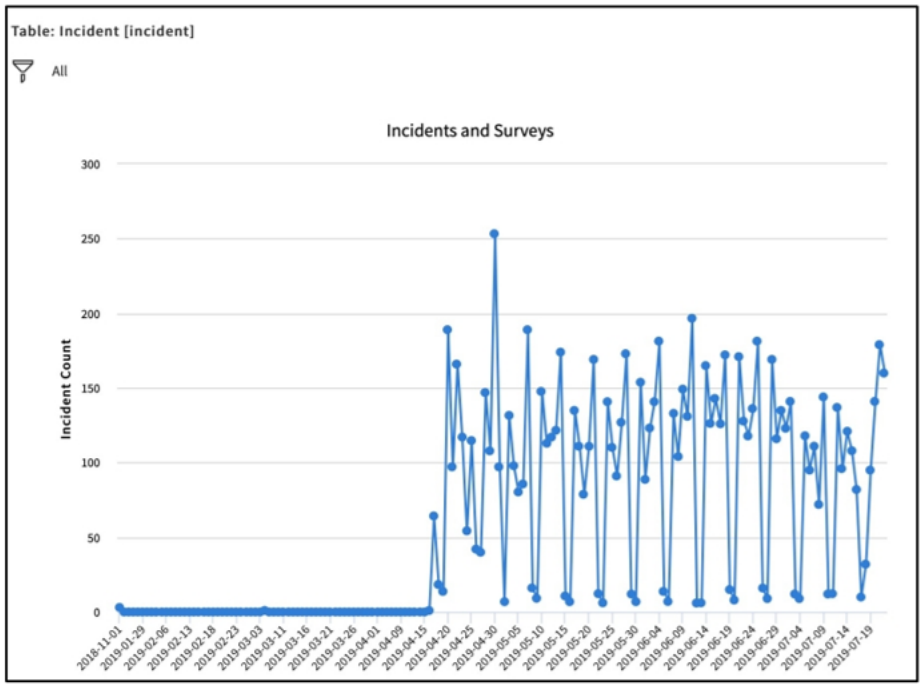


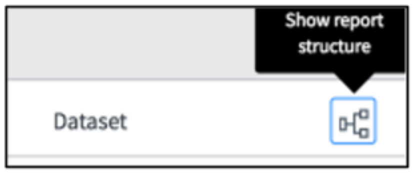
Table: Incident [incident]

All

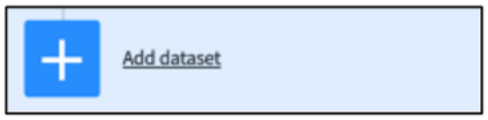
B. Survey Data Set

Modify Report Structure

1. Click the **Show Report Structure** option in the upper-right corner of the report panel.



2. Click the **Add dataset** option.



3. Enter **Surveys** as the Dataset name.

4. Set the table to **Survey Scores**.

5. Choose **Line** chart from the available chart types.

6. Set the following values in the Configure panel:

   Group by: **None**
   Trend by: **Survey Date**
   Aggregation: **Count**

   > ***Note:*** *Per is set to the same value as the main dataset and not editable.*

7. Modify the **Condition Builder** to only select Active Surveys.

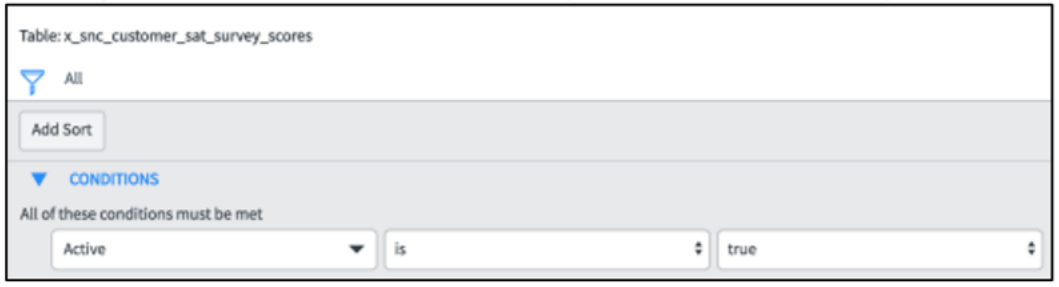


Table: x_snc_customer_sat_survey_scores

All

Add Sort

CONDITIONS

All of these conditions must be met

Active | is | true

8. Navigate to the **Style** section and optionally change the **Chart color**.

9. Select **Run** to preview the results and click the **Save dataset** button.

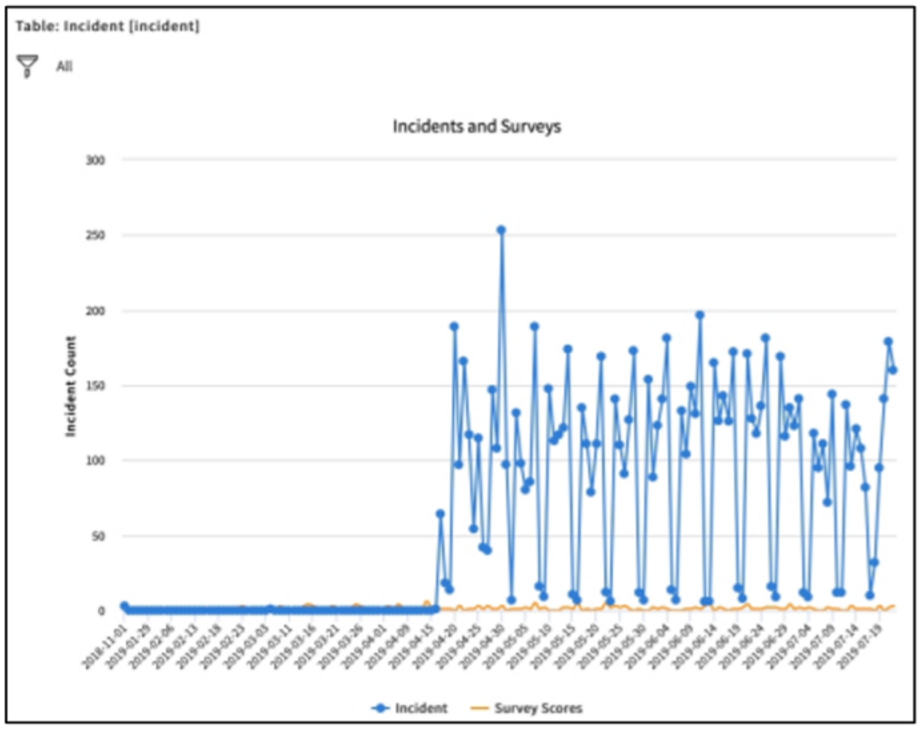


Table: Incident [incident]

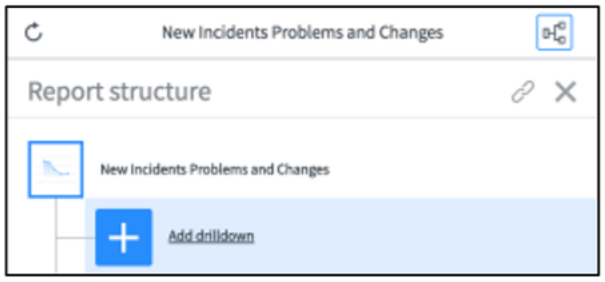
All

Report Interactivity

C. Add Drilldown

Modify Report Structure

1. Open the Report Structure and click **Add drilldown** as shown:


New Incidents Problems and Changes
Report structure
New Incidents Problems and Changes
Add drilldown

2. Configure the new drilldown as follows:

   Drilldown name: **Incidents by Category**
   Type: **Bar** chart
   Group by: **Category**
   Aggregation: **Count**

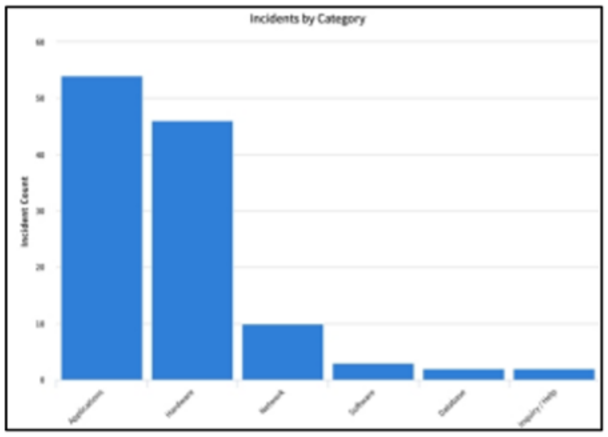
> ***Note:*** *Source type and Table are set to those of the base report and cannot be modified.*

3. Click the **Save drilldown** button in the banner.

4. Close the report structure.

5. Click into the main **Incident Time series** to drill to the next pre-defined view.

6. Confirm that you have drilled down to the **Incidents by Category** view.



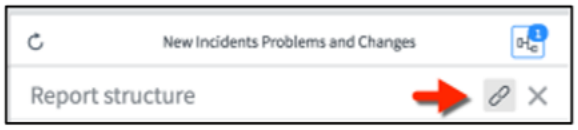
> *What happens when you attempt to drill down from the Incidents by Category?*
>
> *How can you display a different view than a list?*

D. Add Redirect URL

Modify Report Structure

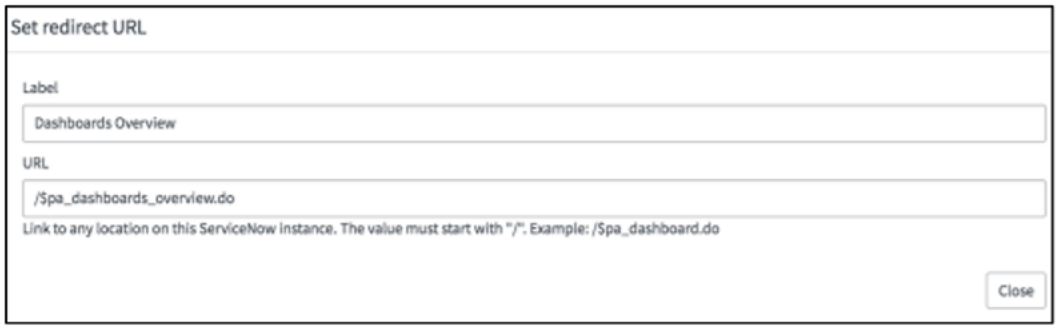
1. Click the **Show report structure** button.

2. Click the **Set redirect URL** button as shown.



3. Set the Label to **Dashboards Overview**.

4. Set the URL to the Dashboards overview page by typing **/$pa_dashboards_overview.do** in the **URL** text box.



Set redirect URL

Label

Dashboards Overview

URL

/$pa_dashboards_overview.do

Link to any location on this ServiceNow instance. The value must start with "/". Example: /$pa_dashboard.do

Close

> ***Note:*** *Adding a redirect URL disables all defined Report drilldowns.*

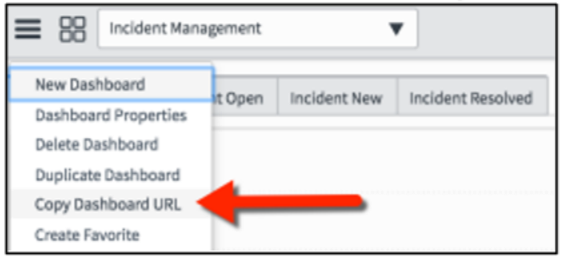
5. Close the report structure and **Save** the report.

6. Test the report by clicking into the Incidents trend.

7. Confirm that you are redirected to the **Dashboards Overview** page.

Optional Challenge Task - Modify Direct URL

The task is to modify the Redirect URL to navigate directly to the **Incident Management** dashboard using these hints:

> ***Hint:*** *Retrieve the dashboard ID from the Dashboard URL:*
>
> 
>

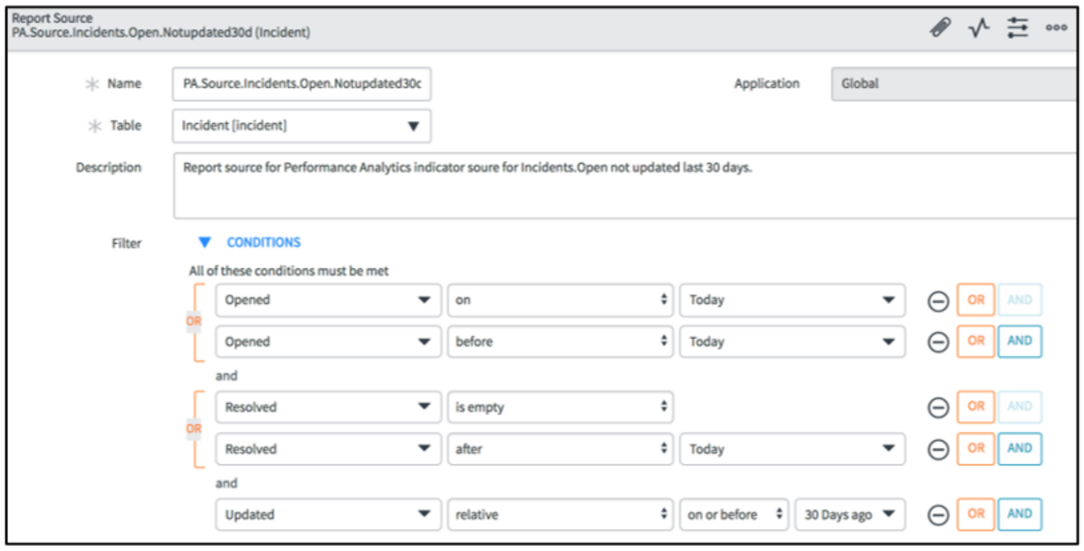
> *Redirect URL – Sample Syntax:*
>
> ***/$pa_dashboard.do?sysparm_dashboard=****a64b7031d7201100b96d45a3ce610335*

You have now completed the Multi Data Sets and Interactivity lab.

Report (Data) Sources: Predefined Subsets of Table Data

Objective: Avoid repetitively building the same report conditions

- Use the same definitions across the organization
- Enforce best practices for report writing
- Report on a custom set of data
- Simplify reporting on data from multiple tables



Report Source
PA.Source.Incidents.Open.Notupdated30d (Incident)

Name: PA.Source.Incidents.Open.Notupdated30c

Application: Global

Table: Incident [incident]

Description: Report source for Performance Analytics indicator soure for Incidents.Open not updated last 30 days.

Filter: CONDITIONS

All of these conditions must be met

Opened on Today
OR
Opened before Today

and

Resolved is empty
OR
Resolved after Today

and

Updated relative on or before 30 Days ago

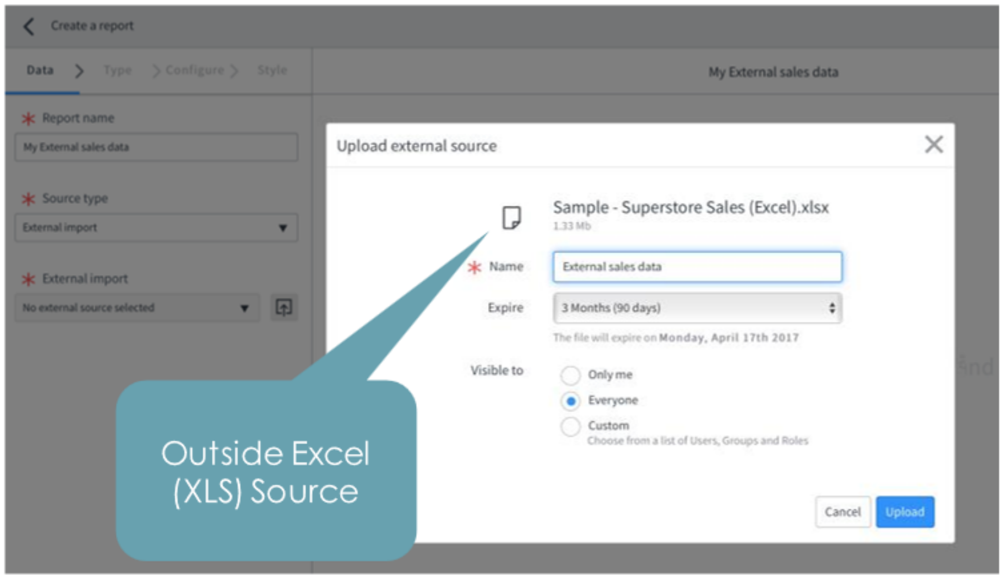
A **Report Source** consists of a table and any number of filter conditions. Some examples are: open incidents, closed problems, top-level assignment groups. Report Admins define Report Sources to encapsulate a set of commonly used report conditions and facilitate report creation while promoting shared definitions across the organization.

Note that In the Report Designer and Report Builder, report sources are called **Data Sources**.

Excel File Import in Report Designer

Objective:

Provide end users with an easy way of visualizing off-platform data in ServiceNow



Report name: My External sales data

Source type: External import

External import: No external source selected

My External sales data

Upload external source

Sample - Superstore Sales (Excel).xlsx
1.33 Mb

Name: External sales data

Expire: 3 Months (90 days)

The file will expire on Monday, April 17th 2017

Visible to:
○ Only me
● Everyone
○ Custom
Choose from a list of Users, Groups and Roles

Outside Excel (XLS) Source

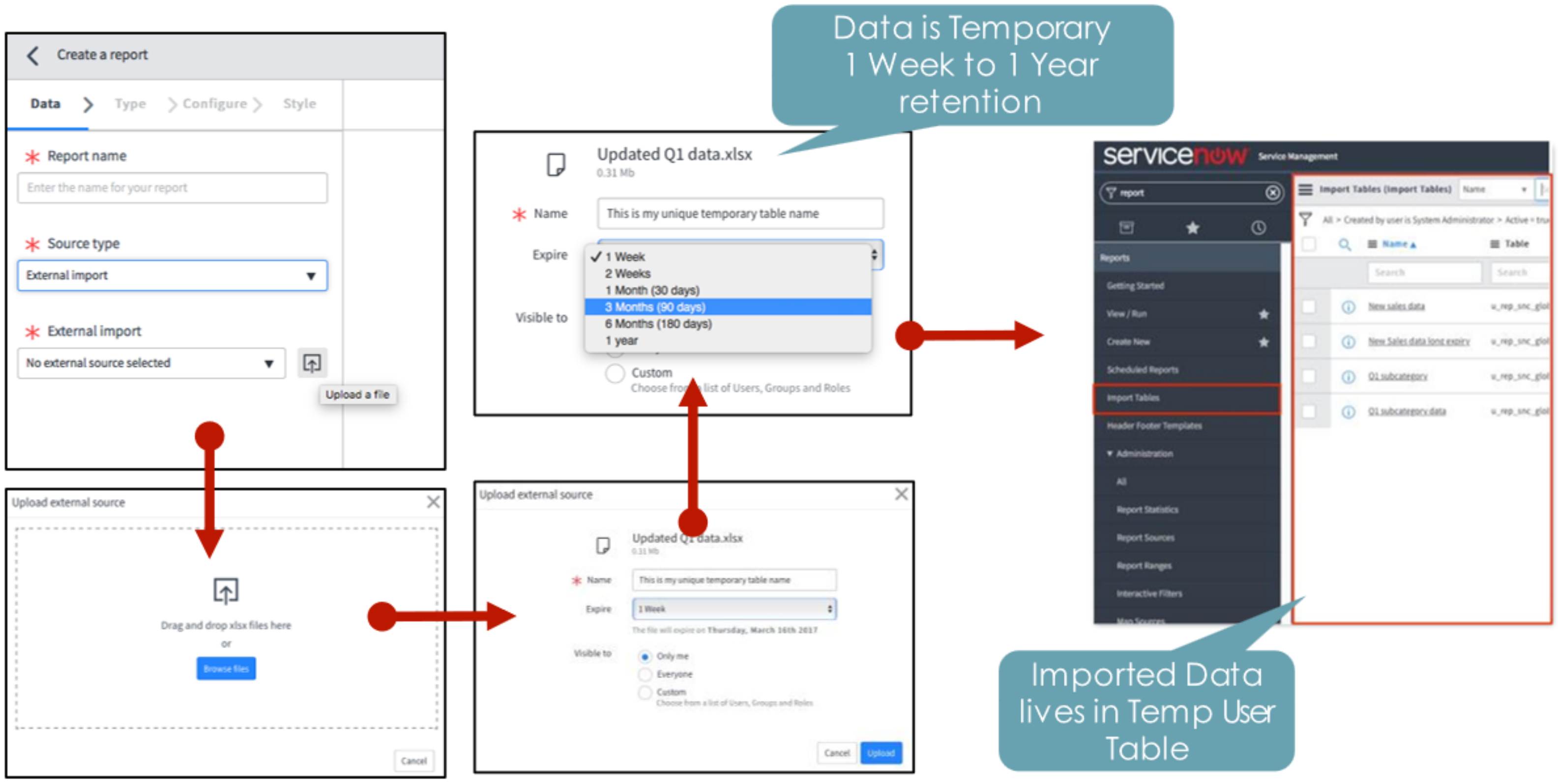
With the External source feature, users are able to select an off-platform data set (.xlsx) as the source for their report. Doing so allows to easily visualize external data using familiar in-platform reporting solution. Once created, the external report widget can be added to a dashboard and shared so that all information is visible in a one platform.

The External source feature requires the Premium Performance Analytics license.

Import Rules and Limitations:

- Required data format is similar to that of the platform Easy Import
- The first worksheet of the Excel document is imported only
- The first Excel document row defines column names
- The first row / column must have data to correctly import

Excel XLSx Import Details



Data is Temporary
1 Week to 1 Year
retention

Imported Data
lives in Temp User
Table

Create a report

Data > Type > Configure > Style

* Report name: Enter the name for your report

* Source type: External import

* External import: No external source selected

Upload a file

Upload external source

Drag and drop xlsx files here
or
Browse files

Cancel

Updated Q1 data.xlsx
0.31 Mb

* Name: This is my unique temporary table name

Expire: 1 Week

- ✓ 1 Week
- 2 Weeks
- 1 Month (30 days)
- 3 Months (90 days)
- 6 Months (180 days)
- 1 year

The file will expire on **Thursday, March 16th 2017**

Visible to:

- ● Only me
- ○ Everyone
- ○ Custom — Choose from a list of Users, Groups and Roles

Cancel | Upload

servicenow Service Management

report

Reports: Getting Started, View / Run, Create New, Scheduled Reports, Import Tables, Header Footer Templates, Administration, All, Report Statistics, Report Sources, Report Ranges, Interactive Filters, Map Sources

Import Tables (Import Tables)

All > Created by user is System Administrator > Active = tru

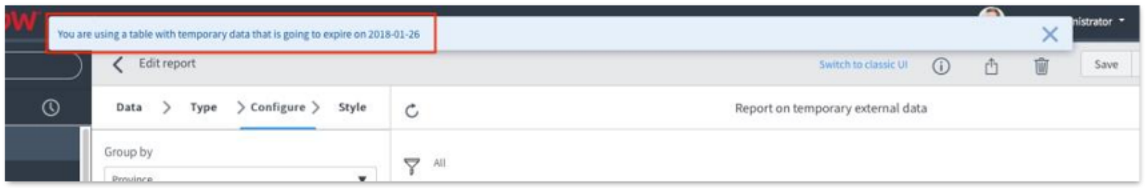
Name ▲	Table
New sales data	u_rep_snc_glol
New Sales data long expiry	u_rep_snc_glol
Q1 subcategory	u_rep_snc_glol
Q1 subcategory data	u_rep_snc_glol

External source configuration notes:

1. Select the third source option "External source" in the Data step of the guided workflow
2. Select Upload a File icon to upload a new Excel file. Or choose a previously imported file from the External import dropdown selector.
3. From here, a power user or admin can select the Excel file (xlsx), and set the temporary table name, expiry time, security (whom can see & report on the data).
4. Name your imported xlsx file, this will be the name of the temporary table. The system will do a check if the name is unique and will mention a warning otherwise.
5. Specify the visibility as follows:
   - Only Me (default); data is only visible to the user who is uploading the file
   - Everyone; any logged in user will be able to read and report on that imported table
   - Users / Groups / Roles; only the set users, groups & roles specified will be able to read and report on that imported table

Temporary Data Management

- External data tables and reports are auto-deleted after source expiry
- Existing widgets remain as empty placeholders
- Information messages when opening a report when the data will expire
- Report owner receives a notification mail when source is 1-2 days from expiry
- Import a new report source to prevent the deletion of reports



You are using a table with temporary data that is going to expire on 2018-01-26

Imported data is stored in temporary ServceNow tables. These are regular ServiceNow tables but the system controls the expiry date of the data and cleans the table after expiration.

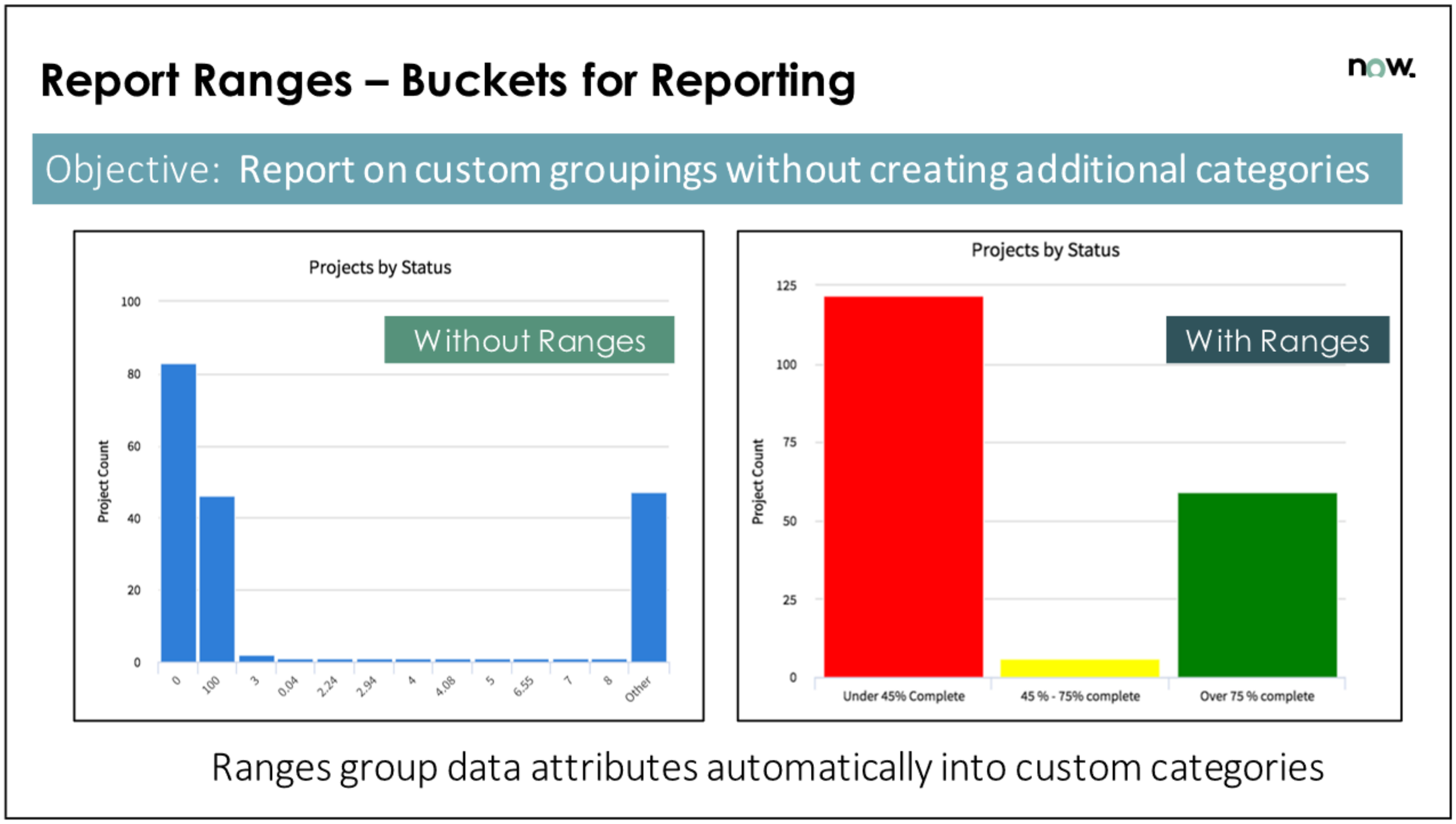
The system informs the users when they open an external data source report, that they are looking at a report based on an imported table with an expiry date.

Note that after the specified time expires, your temporary table will auto-delete.

External data that lives outside the platform is not real-time data and may become outdated soon after the import. The expected behavior is that users of this feature create a new temporary imported source when a new version of that external source becomes available.

Additional Import Limitations:

- Import limits are set on maximum file size, row & column count
  - Maximum file size: 2MB (MAX_IMPORT_TABLE_FILE_SIZE, "2")
  - Row count: 10000 (MAX_IMPORT_TABLE_FILE_ROWS, "10000")
  - Column count: 25 (MAX_IMPORT_TABLE_FILE_COLUMNS, "25")
- Appending data is not supported
- XLSX is the only supported format



Report Ranges – Buckets for Reporting

Objective: Report on custom groupings without creating additional categories

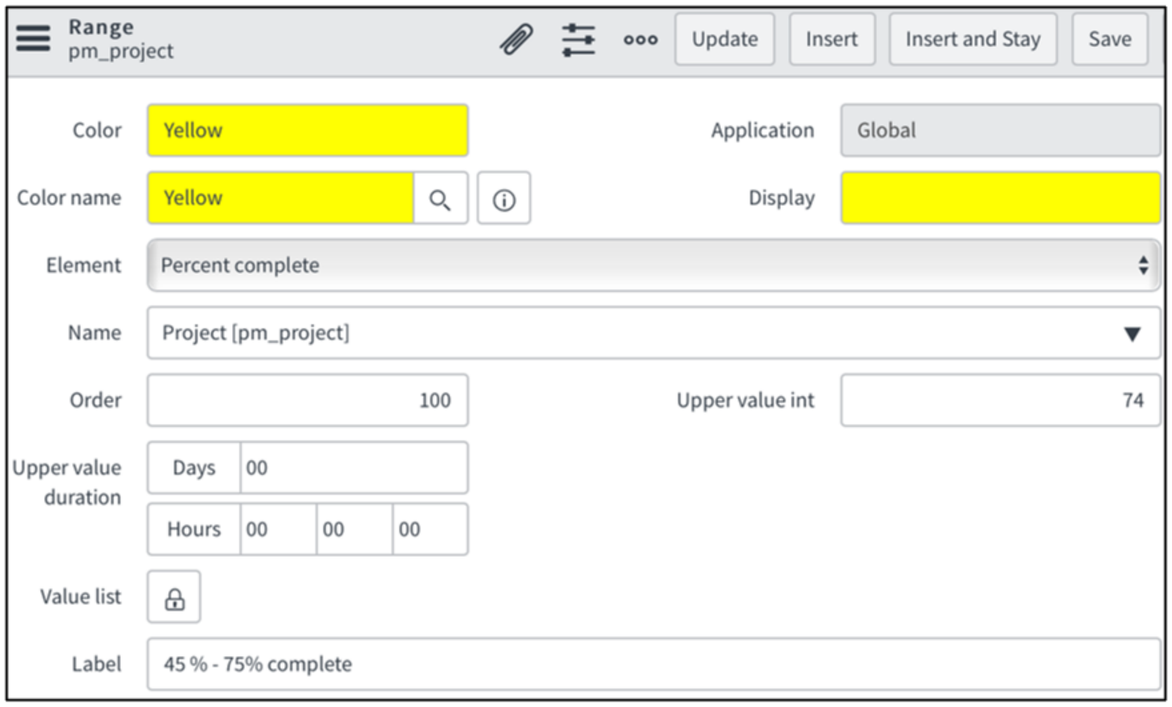
Ranges group data attributes automatically into custom categories

Report ranges are ideal when reporting on continuous data. For instance, consider the example where you need a report on transaction completion times. This data is obviously continuous and does not render itself well to charting. Consider creating ranges or buckets of transaction response times to organized and group the data by.

Report ranges work with elements that hold only dates, lists, or integers. Report ranges can be globally applied to all date type fields (date, due date, duration, date/time, date time), or you can limit report ranges to a specific table. Configuring a Report Range requires a **report_admin** role.

Creating Report Ranges

- Automatically apply to report output
- Can be defined per Table or globally
- Integer fields use **Upper value int**
- Date fields use **Upper value duration**
- Color is optional



Color	Yellow	Application	Global
Color name	Yellow	Display	🟨
Element	Percent complete		
Name	Project [pm_project]		
Order	100	Upper value int	74
Upper value duration	Days 00 Hours 00 00 00		
Value list			
Label	45 % - 75% complete		

When configuring report ranges, the following must be defined:

Name and **Element**: Table and the table field to draw the values from.

Label: The name for the report range that is displayed in reports.

Value list: For choice list elements, this field defines which values are within the range.

Upper value int: For integer-type elements, this field defines the upper limit of the range. If no range with a lower **Order** exists, the lower limit is zero.

Upper value duration: For duration-type elements (dates), this field defines the upper limit of the range.

The **Order** element is used to determine the order of applying ranges.

Advanced Reporting Techniques Labs

Lab 4.3: Data Sources

Lab 4.4: Report Ranges

Data Sources

Lab 4.3

⌛20m

Lab Objectives

Data sources promote the use of common query language for reporting and analysis purposes. As a Report Administrator, you need to create several report sources to simplify queries and ensure that reporting is performed in a consistent way across the organization.

Customer success managers at your company need the ability to report on information that is vital to tracking the customer experience but does not reside in ServiceNow. With that in mind, you need to incorporate the Outages spreadsheet as a temporary ServiceNow table.

This lab will show you how to do the following:

- Build a Report Source to encapsulate a definition for easier reporting
- Configure an external file for reporting

Using Report Sources

A. Report Source Creation

Creating Report Source in Report Designer

In this section, you access the Report Designer and configure a Report Source to display *Top Level Groups* only. Top level groups are defined as groups that do not have a parent group.

1. Navigate to **Reports > Create New**.

2. Create a new report as follows:

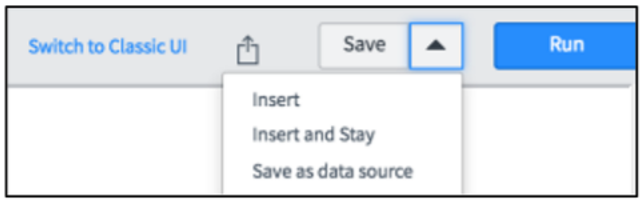
   Name: **Top Level Groups I Manage**
   Table: **Group [sys_user_group]**
   Type: **List**
   Report Columns: **Name**, **Description**, **Manager**, **Parent, Active**

3. Set the **Conditions** to:

   Parent is empty AND
   Active is true

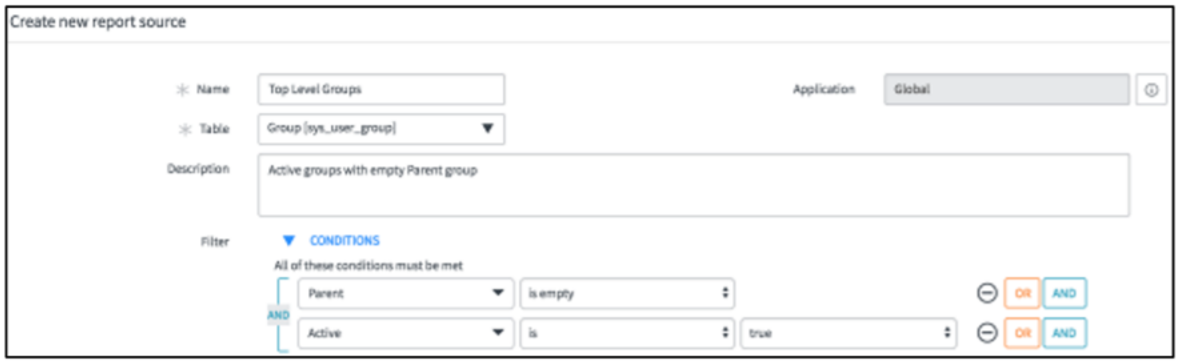
4. Run the report and confirm that it returns 25+ group records.

5. Click the **arrow** next to the **Save** menu and select **Save as data source**.



6. Set the Name of the new report source to **Top Level Groups**.

7. Set the description to **Active groups with empty Parent group**.


Create new report source

Name: Top Level Groups

Application: Global

Table: Group [sys_user_group]

Description: Active groups with empty Parent group

Filter: CONDITIONS

All of these conditions must be met

Parent is empty

AND Active is true

8. Click **Submit**.

Using Report Source

1. Navigate to **Reports > Create New** to begin creating a new report.

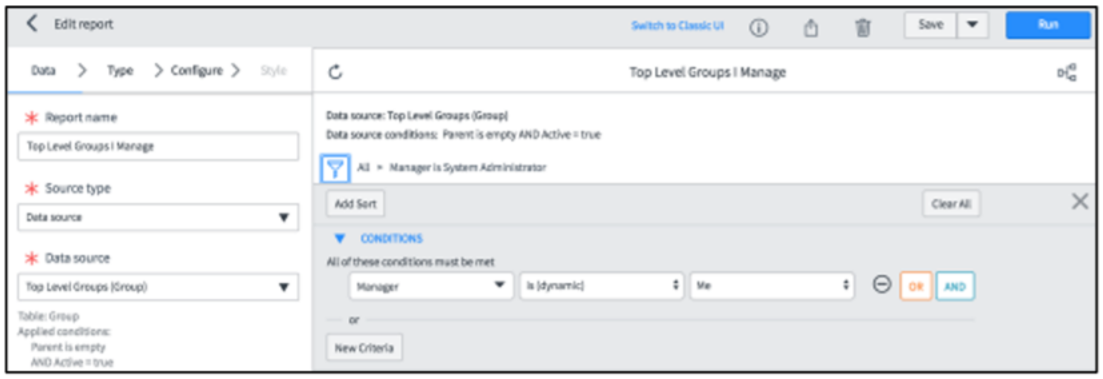
2. Set the Report name to **Top Level Groups I Manage**.

3. Set the Source type to **Data source**.

4. Set the Data source to **Top Level Groups (Group)**.

5. Add the following **Condition:**

   Manager is(dynamic) Me



Top Level Groups I Manage

Report name: Top Level Groups I Manage

Source type: Data source

Data source: Top Level Groups (Group)

Table: Group

Applied conditions:

Parent is empty

AND Active = true

Data source: Top Level Groups (Group)

Data source conditions: Parent is empty AND Active = true

All > Manager is System Administrator

CONDITIONS

All of these conditions must be met

Manager is (dynamic) Me

or

6. Navigate to **Configure** and click **Choose columns .**

7. Add these report columns: **Name**, **Manager**

8. Click the **Sharing** icon and share the report with **Don Goodliffe**.

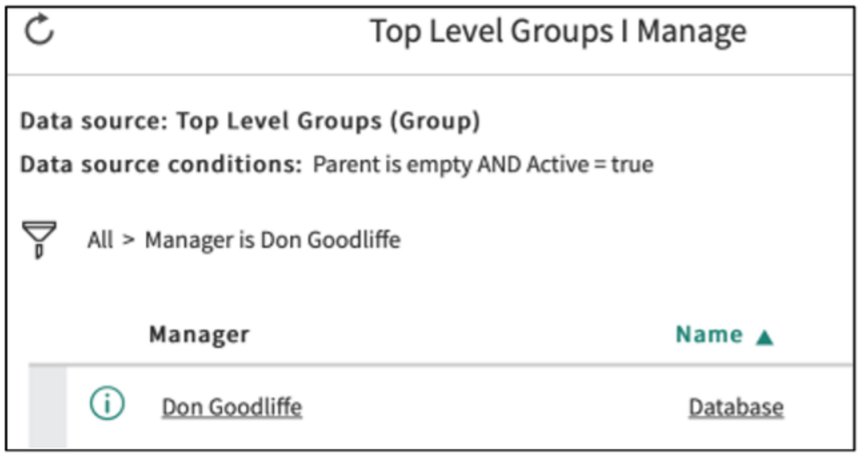
9. **Save** the report.

10. Impersonate **Don Goodliffe**.

11. Navigate to **Reports > View Run**.

12. Click the **Group** section where the **Top Level Groups I Manage** report can be found.

13. Run the **Top Level Groups I Manage** report and confirm that the report accurately returns the **Database** group that Don manages.



Top Level Groups I Manage

Data source: Top Level Groups (Group)

Data source conditions: Parent is empty AND Active = true

All > Manager is Don Goodliffe

Manager	Name ▲
Don Goodliffe	Database

Tip: *If you are unable to see the Top Level Groups I Manage report as Don Goodliffe, you need to impersonate the System Administrator, Share the report to Everyone once again, AND save it.*

Working with External Import

B. File Download and Review

MS Excel File

1. Download the **OutageLast3MoToDate_combined.xlsx** file from the Class KB – KB0010004 or the location provided by the instructor.

2. Open the file and validate the following:

    - The file has the *.xlsx extension

- All data is contained in the first worksheet
- The first row of the file contains all column names
- The file size is less than 2Mb
- There are fewer than 1000 rows
- There are less than 25 columns

3. Make any adjustments necessary to comply with the import requirements.

C. Report based on External Import

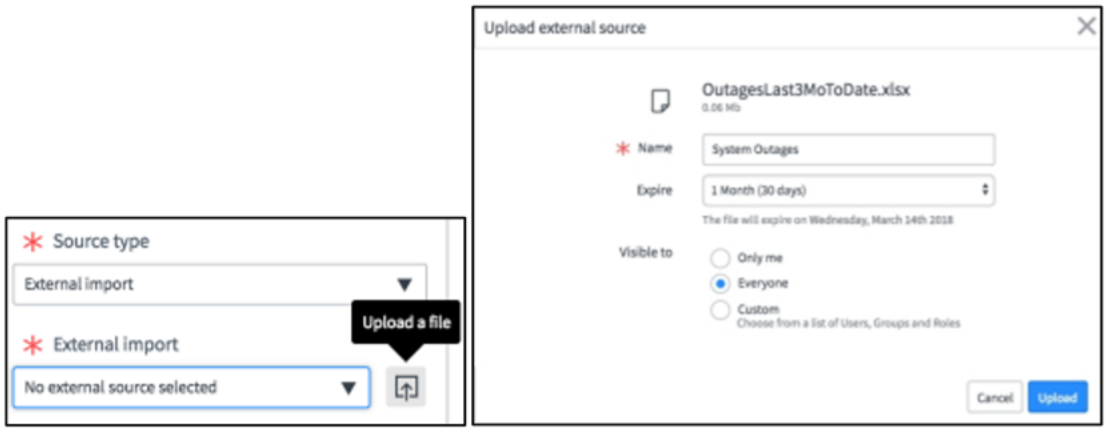
Report Designer

1. Navigate to **Reports > Create New**.
2. Configure the new report as follows:

   Report name: **Outages**
   Source type: **External import**

3. Click the **Upload a file** button and configure the External Import as follows:

   Name: **System Outages**
   Expire: **1 Month (30 days)**
   Visible to: **Everyone**



* Source type: External import
* External import: No external source selected

Upload external source

OutagesLast3MoToDate.xlsx
0.06 Mb

* Name: System Outages
Expire: 1 Month (30 days)
The file will expire on Wednesday, March 14th 2018
Visible to: ○ Only me ● Everyone ○ Custom
Choose from a list of Users, Groups and Roles

4. Click **Upload** to complete the file upload.

   Tip: *A successful import displays* ***Import Successful*** *message.*

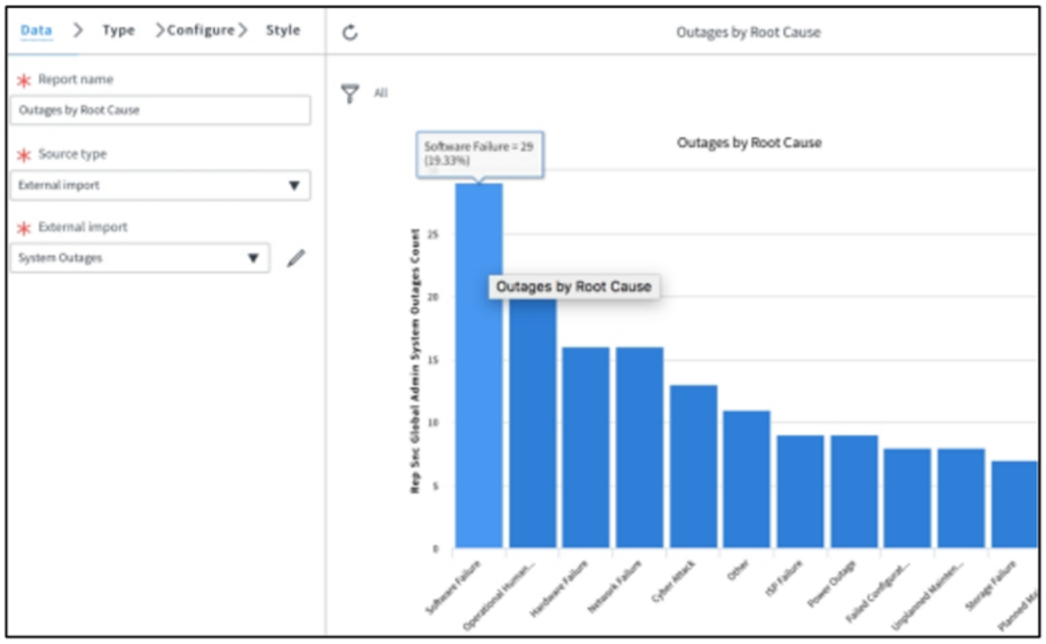
5. Continue to configure the report as follows:

Report type: **Bar** chart
Group by: **Root Cause**
Aggregation: **Count**
Change the name to **Outages by Root Cause**

6. Save the report.

> ***Note:*** *Upon saving, the application displays a reminder that this is a temporary report with a predetermined expiration date.*

7. Confirm that your report looks similar to this:



Report name: Outages by Root Cause
Source type: External import
External import: System Outages

> ***Question:*** *What would happen to existing report data if you decided to edit and reimport the external import source?*

You have now completed the Data Sources lab.

Report Ranges

Lab 4.4

⌛15m

Lab Objectives

The SLA Percent Completion ranges from 0 to 100%. The Incident Manager would like a higher level of grouping that allows to organize SLAs by the extent of SLA completion and not by the exact Percent completion number. To do this, you perform the following:

- Define data intervals (report ranges) for grouping
- Use report ranges for an easier interpretation of the business data

Report Ranges

A. View Incident SLAs without Report Ranges

Report Designer

1. Navigate to **Reports > Create New**.

2. Begin configuring a new report as follows:

   - Report name: **Incident SLA Completion Breakdown**
   - Source type: **Table**
   - Table: **incident_sla**

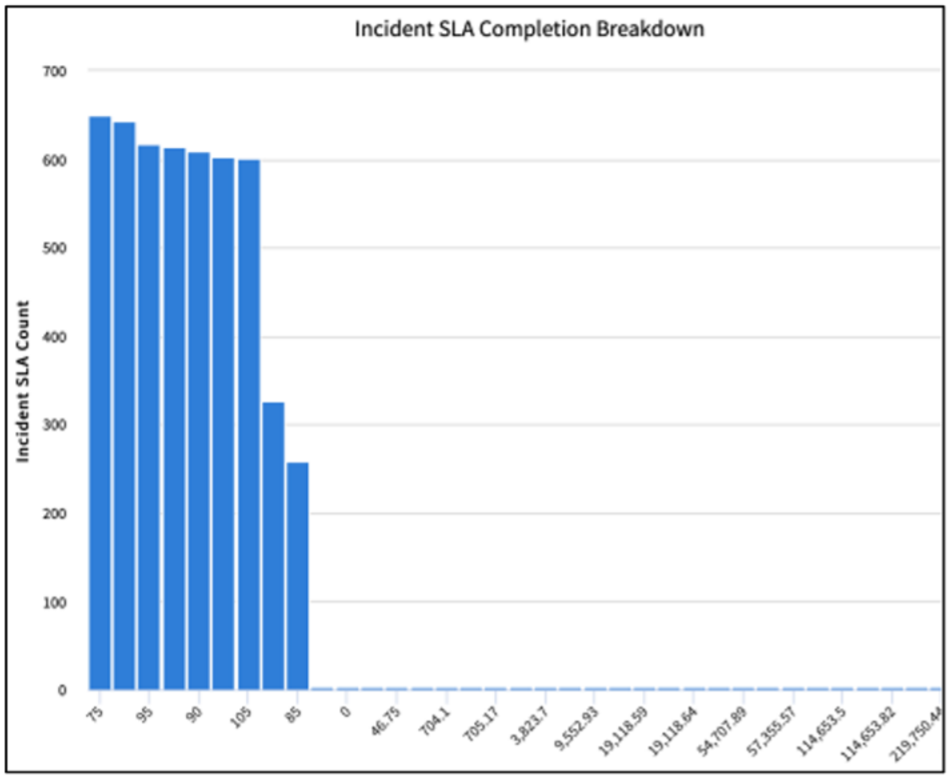
   Note: *The incident_sla table is a DB View that joins the Incident and the Task SLA tables*

   - Type: **Bar** chart
   - Group by: **Actual elapsed percentage**
   - Aggregation: **Count**
   - Max number of groups: **Show all**
   - Deselect **Show Other**

3. Set the **Conditions** to **Actual elapsed percentage is not empty**.

4. **Save** and preview the report.

> ***Question:*** *Does the information shown here present a clear view of the overall SLA compliance?*



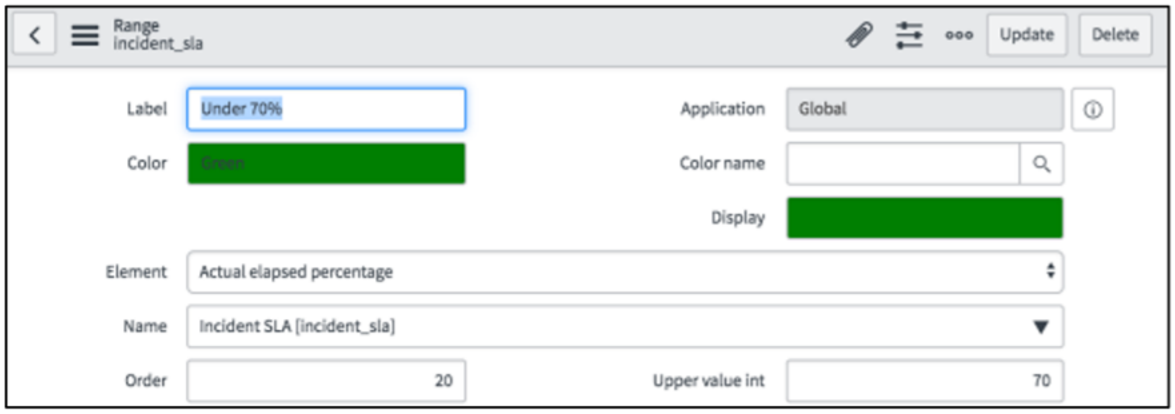
B. Create Report Ranges

In this section, you create Report Ranges to make the SLA completion info more intuitive.

1. Navigate to **Reports > Administration > Report Ranges**.
2. Define a New report ranges as follows:
   - Label: **Under 70%**
   - Color: **Green**
   - Name: **Incident SLA [incident_sla]**

   > ***Note****: The Table has to be specified first as the Name ahead of the Column (Element) attribute.*

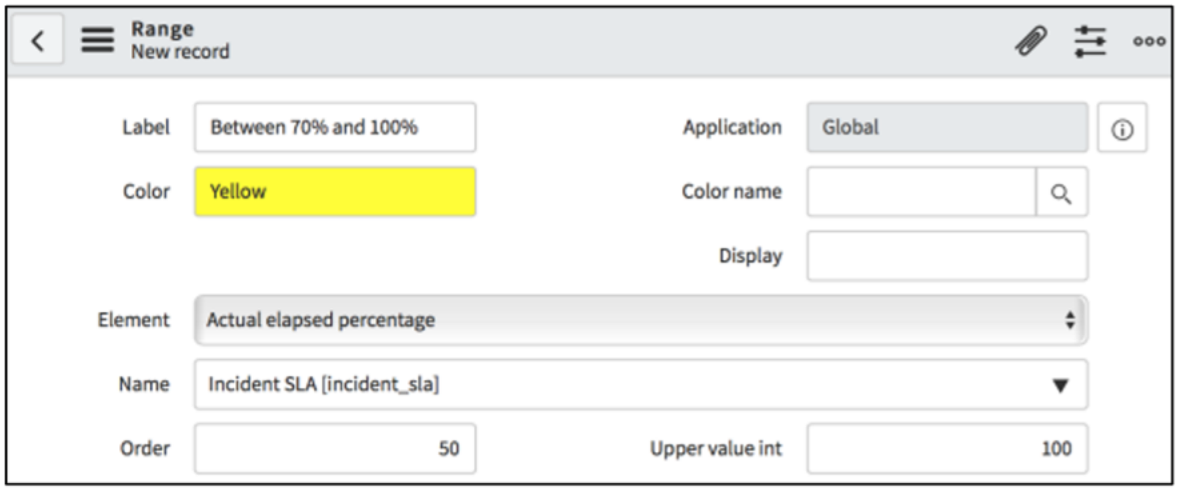
   - Element: **Actual elapsed percentage**
   - Order: **20**
   - Upper value int: **70**
3. Click **Submit**.
4. Verify that the Range was created as shown:



Label	Under 70%	Application	Global
Color	Green	Color name	
		Display	🟩
Element	Actual elapsed percentage		
Name	Incident SLA [incident_sla]		
Order	20	Upper value int	70

5. Create another new **Report Range** record as follows:

    - Label: **Between 70% and 100%**
    - Color: **Yellow**
    - Name: **Incident SLA [incident_sla]**
    - Element: **Actual elapsed percentage**
    - Order: **50**
    - Upper value int: **100**



Label	Between 70% and 100%	Application	Global
Color	Yellow	Color name	
		Display	
Element	Actual elapsed percentage		
Name	Incident SLA [incident_sla]		
Order	50	Upper value int	100

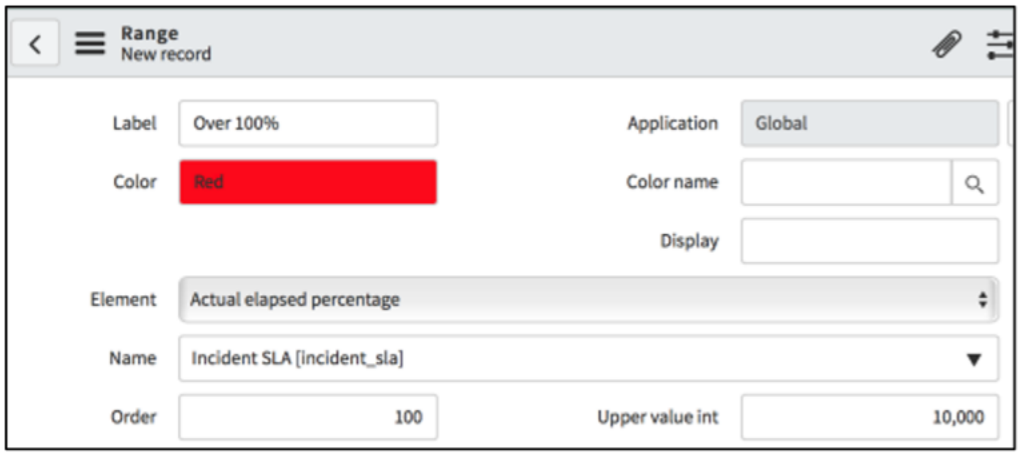
6. Click **Submit.**

7. Create the third and final new **Report Range** record as follows:

8. Enter the following values:

    - Label: **Over 100%**
    - Color: **Red**
    - Name: **Incident SLA[incident_sla]**
    - Element: **Actual elapsed percentage**

- Order: **100**
- Upper value int: **10,000**



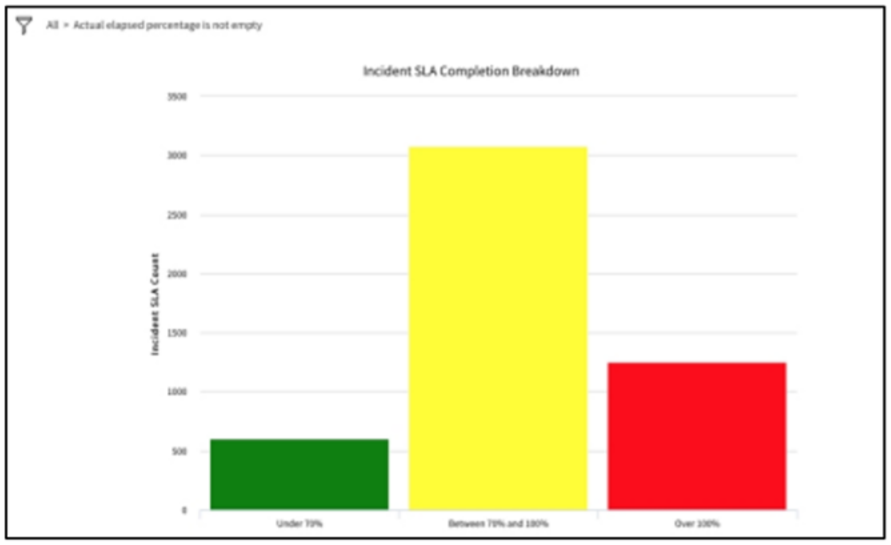
Range
New record

Label	Over 100%	Application	Global
Color	Red	Color name	
		Display	
Element	Actual elapsed percentage		
Name	Incident SLA [incident_sla]		
Order	100	Upper value int	10,000

9. Click **Submit**.

C. Create the Report using Report Ranges

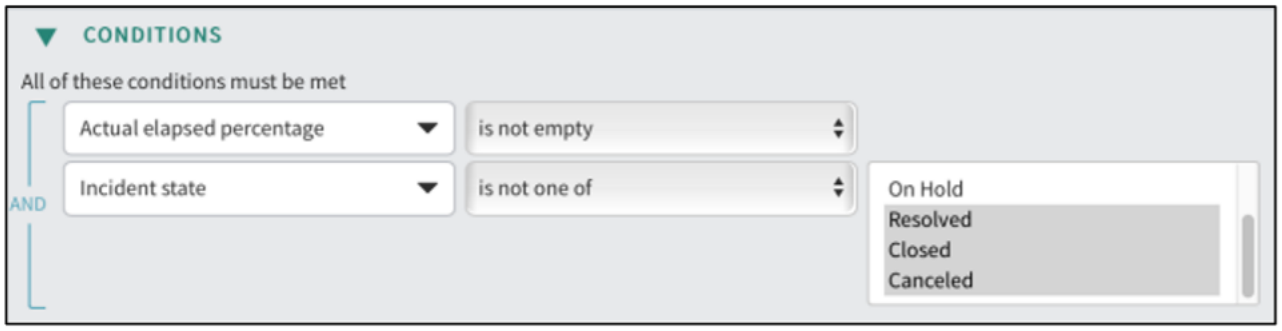
1. Navigate to the **Report > View/Run**.
2. Search for and open the **Incident SLA Completion Breakdown** report.
3. Confirm that the report groups Incident SLAs into the three predefined ranges:



Customize Report

1. Navigate to the **Configure** section.
2. Click **Add** and add this clause to the **CONDITIONS**:

Incident state is not one of Resolved / Closed / Cancelled



CONDITIONS

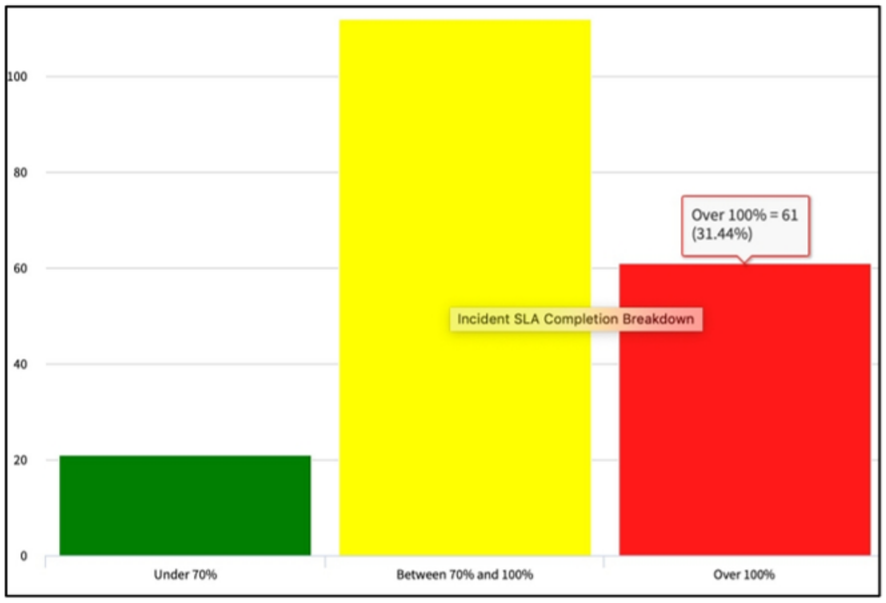
All of these conditions must be met

Actual elapsed percentage | is not empty

AND

Incident state | is not one of | On Hold, Resolved, Closed, Canceled

3. Click **Save** and preview the report.



Note: *These are active Incidents categorized by SLA completion percentage.*

D. Report Ranges in Action

Report Designer

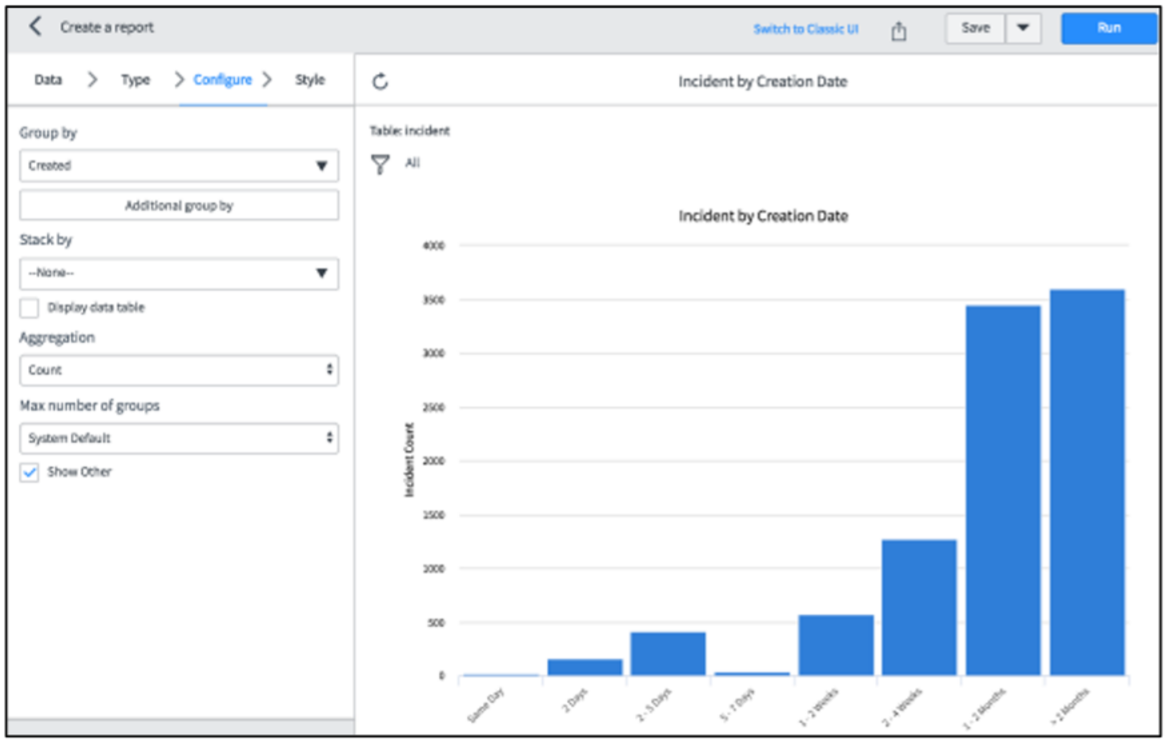
1. Navigate to **Reports > Create New** and create the following report:

   Name: **Incidents by Creation Date**
   Table: **Incident [incident]**
   Type: **Bar** chart

   Group by: **Created**
   Aggregation: **Count**

2. Save and preview the report.

Note*: The Incident Created attribute is automatically grouped into pre-defined ranges.*



Group by: Created

Stack by: --None--

□ Display data table

Aggregation: Count

Max number of groups: System Default

☑ Show Other

Table: incident

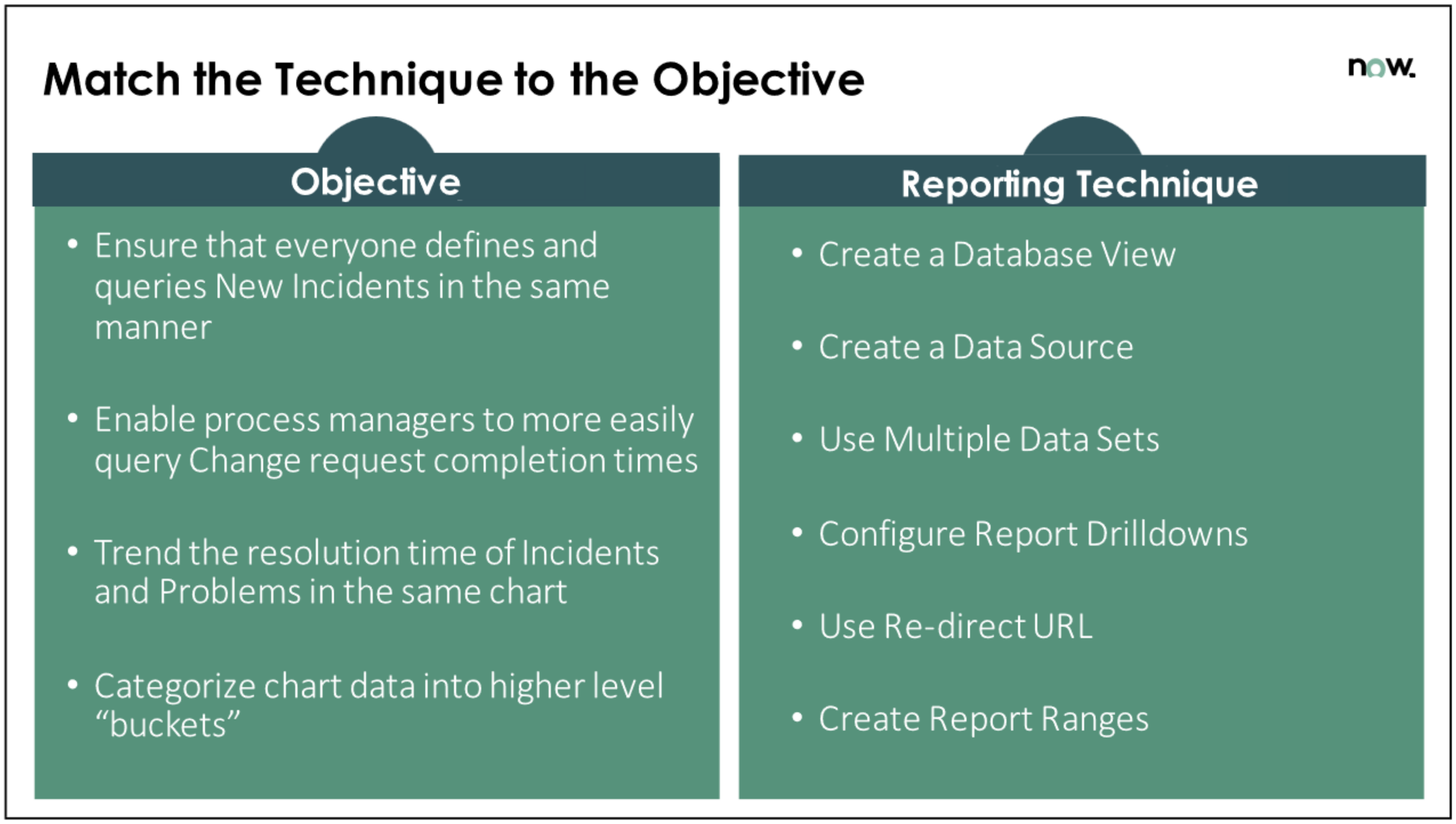
All

Questions:

Which Report Ranges are responsible for grouping the Created attribute in categories?

Which tables does this set of Report Ranges apply to and why?

You have now completed the Report Ranges lab.



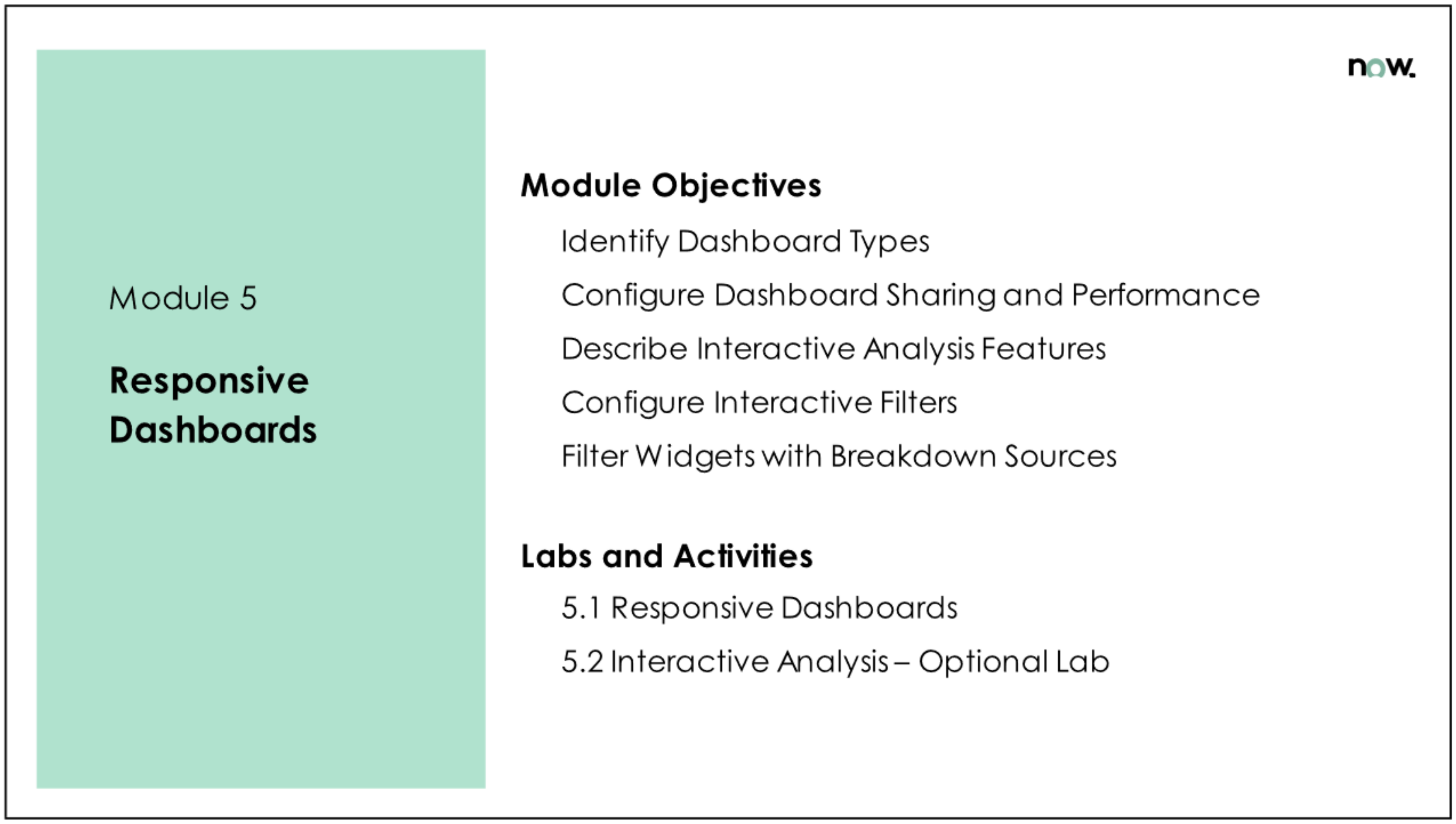
Match the Technique to the Objective

Objective

- Ensure that everyone defines and queries New Incidents in the same manner
- Enable process managers to more easily query Change request completion times
- Trend the resolution time of Incidents and Problems in the same chart
- Categorize chart data into higher level "buckets"

Reporting Technique

- Create a Database View
- Create a Data Source
- Use Multiple Data Sets
- Configure Report Drilldowns
- Use Re-direct URL
- Create Report Ranges



Module 5

Responsive Dashboards

Module Objectives

- Identify Dashboard Types
- Configure Dashboard Sharing and Performance
- Describe Interactive Analysis Features
- Configure Interactive Filters
- Filter Widgets with Breakdown Sources

Labs and Activities

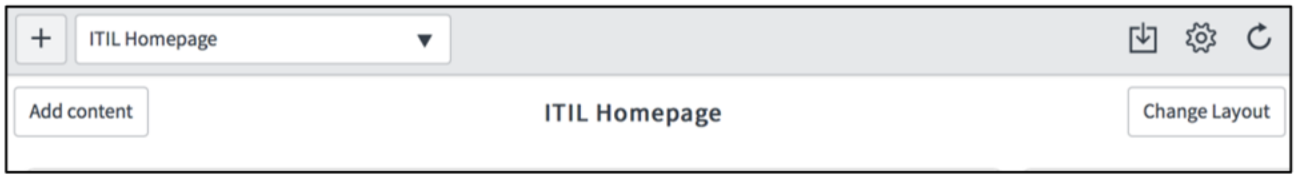
- 5.1 Responsive Dashboards
- 5.2 Interactive Analysis – Optional Lab

This module takes a closer look at the ServiceNow Responsive dashboards. Discussed here are the different dashboard types, the Interactive Analysis capability, and ways to filter widgets using Interactive Filters, Cascading Filters, and Breakdown Sources. Dashboard configuration properties and dashboard performance management techniques are also introduced.

Dashboards Types

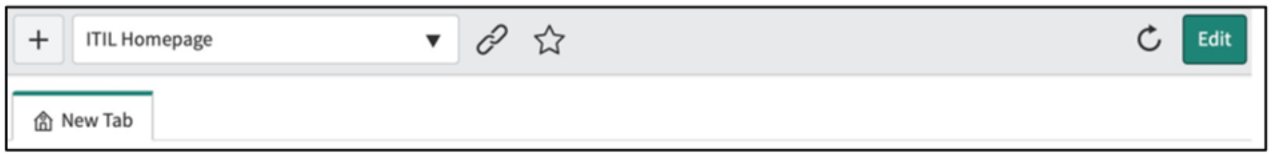
TIP:
The Dashboard Type is visible in the Header

Homepages are legacy containers for Widgets and Gauges

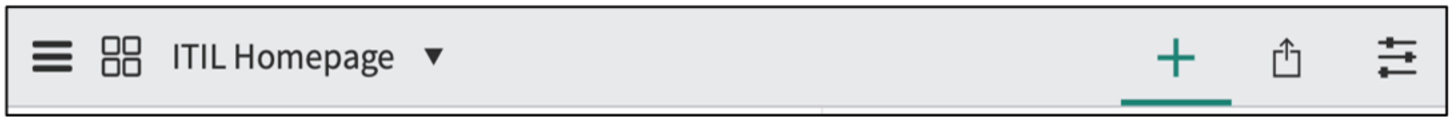


Dashboards are the modern implementation of Homepages

Non - Responsive dashboards support multiple tabs, improved editing, and sharing



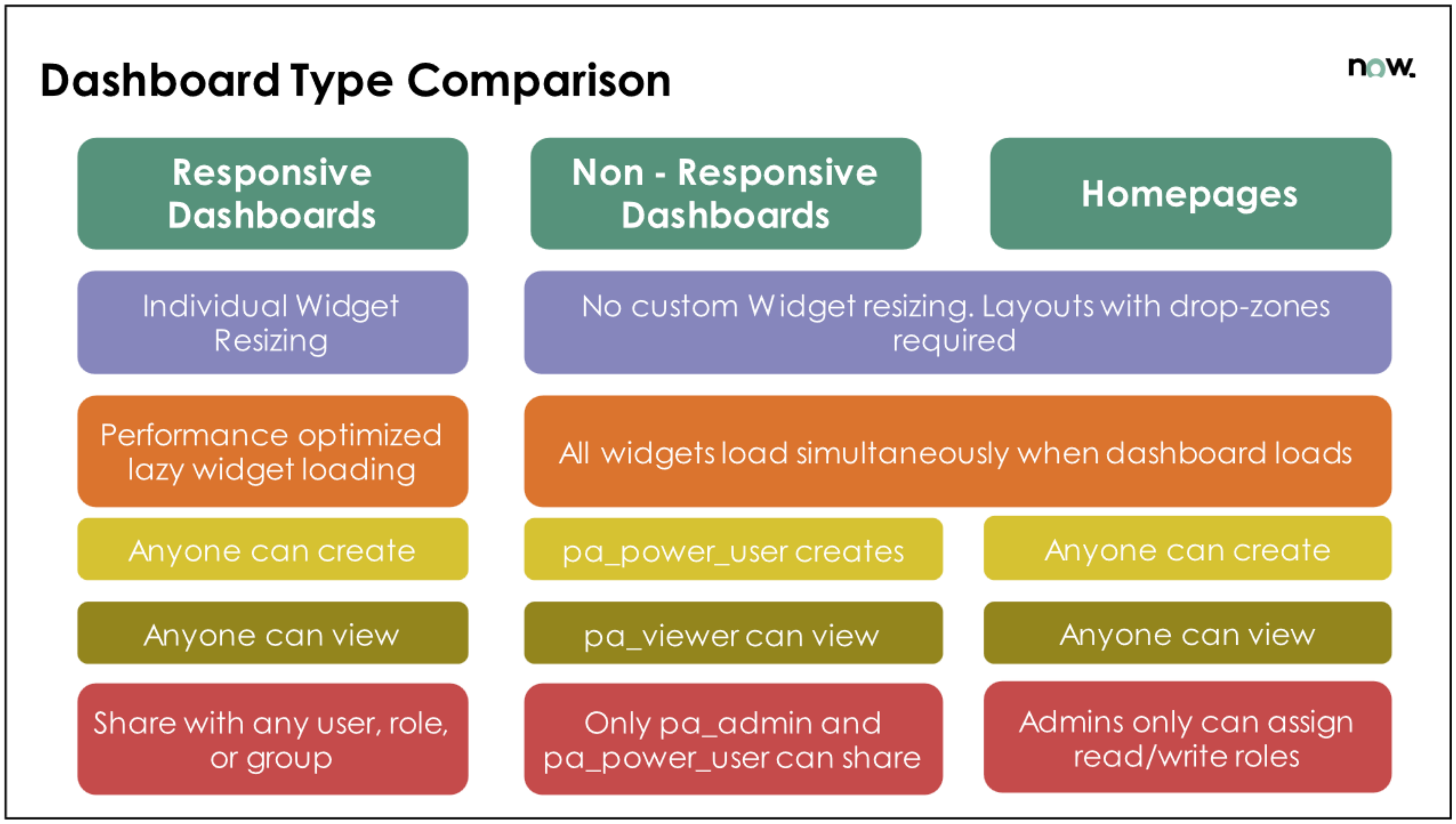
Responsive dashboards offer superior content management, sharing, and performance



Reports, Gauges, non-PA widgets, PA widgets, and content blocks are some commonly added objects to Homepages and Dashboards. Homepages are a legacy feature and, even though still supported, are being superseded by Dashboards. Dashboards can be configured to be **Responsive** or **Non-Responsive**. Their type can be identified by the *header*. Responsive dashboards have the Add Widget, Sharing, and Configuration buttons visible for dashboard owners and editors.

Responsive dashboards have a more intuitive interface and are easier to share and edit than non-responsive dashboards. To unlock the full power of Performance Analytics, consider *always* working with Responsive Dashboards. Here are some additional notes on Responsive dashboard enablement:

- Responsive dashboards are enabled by default on new and upgraded instances
- When you enable responsive dashboards, all existing dashboards are converted to responsive dashboards
- Set the system property **glide.cms.enable.responsive_grid_layout** to enable or disable responsive dashboards
- Any user can convert their existing Homepages to a modern dashboard



Dashboard Type Comparison

Responsive Dashboards	Non - Responsive Dashboards	Homepages
Individual Widget Resizing	No custom Widget resizing. Layouts with drop-zones required	
Performance optimized lazy widget loading	All widgets load simultaneously when dashboard loads	
Anyone can create	pa_power_user creates	Anyone can create
Anyone can view	pa_viewer can view	Anyone can view
Share with any user, role, or group	Only pa_admin and pa_power_user can share	Admins only can assign read/write roles

Here are some additional notes on the different types of Dashboards:

Responsive dashboards:

- Created and shared by any user with any role
- Flexible editing, content creation, and sharing capabilities
- Dashboard sharing can be tailored by an Admin
- Only visible widgets load when opening a dashboard. More widgets load as the user scrolls down

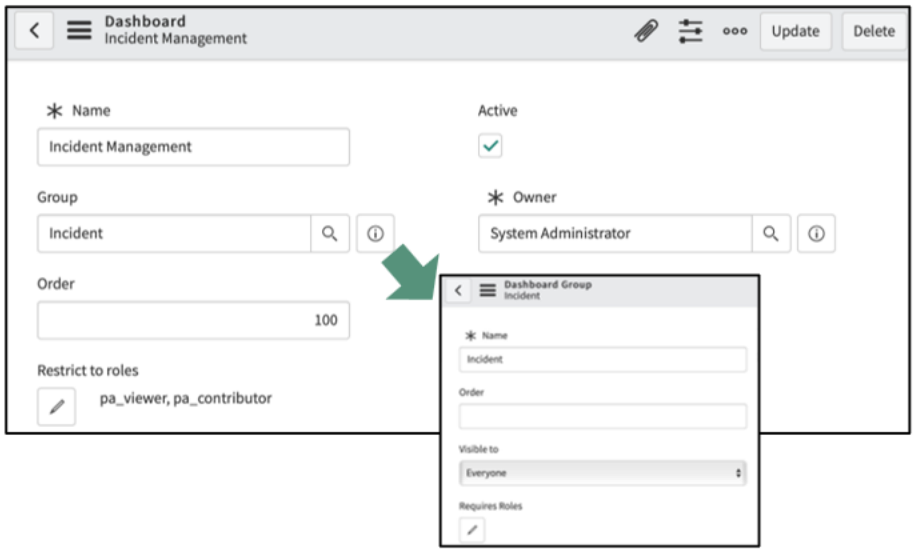
Non-Responsive and Home Pages:

- All widgets load when the dashboard is opened which results in slower performance.
- No custom resizing of widgets, legacy layouts with drop zones are used
- Less intuitive interface for adding content and sharing

For more detailed information, see https://docs.servicenow.com/bundle/newyork-performance-analytics-and-reporting/page/use/dashboards/concept/differences-between-respsonsive-and-non-responsive-dashboards.html

Responsive Dashboard Permissions

- Add Dashboard to Groups for better organization
- Dashboard Permissions override Group permissions
- Use **Restrict to roles** to limit dashboard access



Name: Incident Management

Active: ☑

Group: Incident

Owner: System Administrator

Order: 100

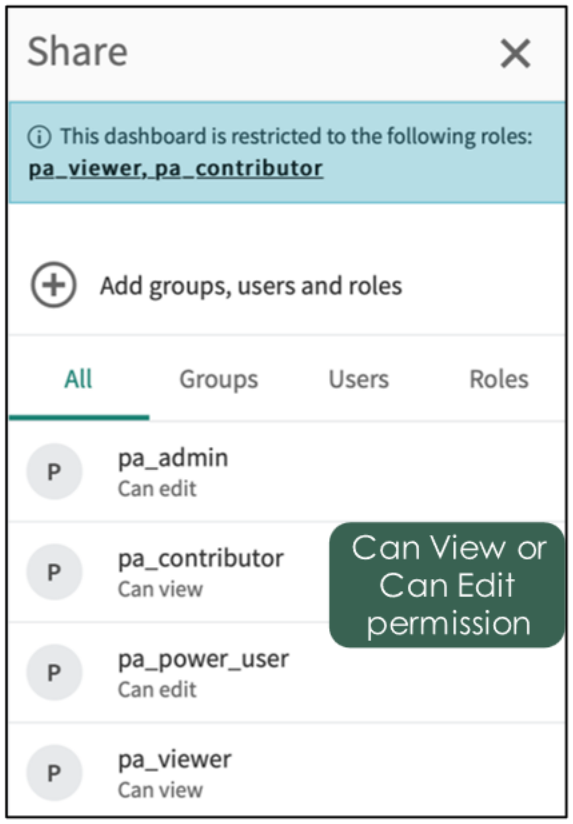
Restrict to roles: pa_viewer, pa_contributor

Name: Incident

Order:

Visible to: Everyone

Requires Roles:



This dashboard is restricted to the following roles: pa_viewer, pa_contributor

Add groups, users and roles

pa_admin	Can edit
pa_contributor	Can view
pa_power_user	Can edit
pa_viewer	Can view

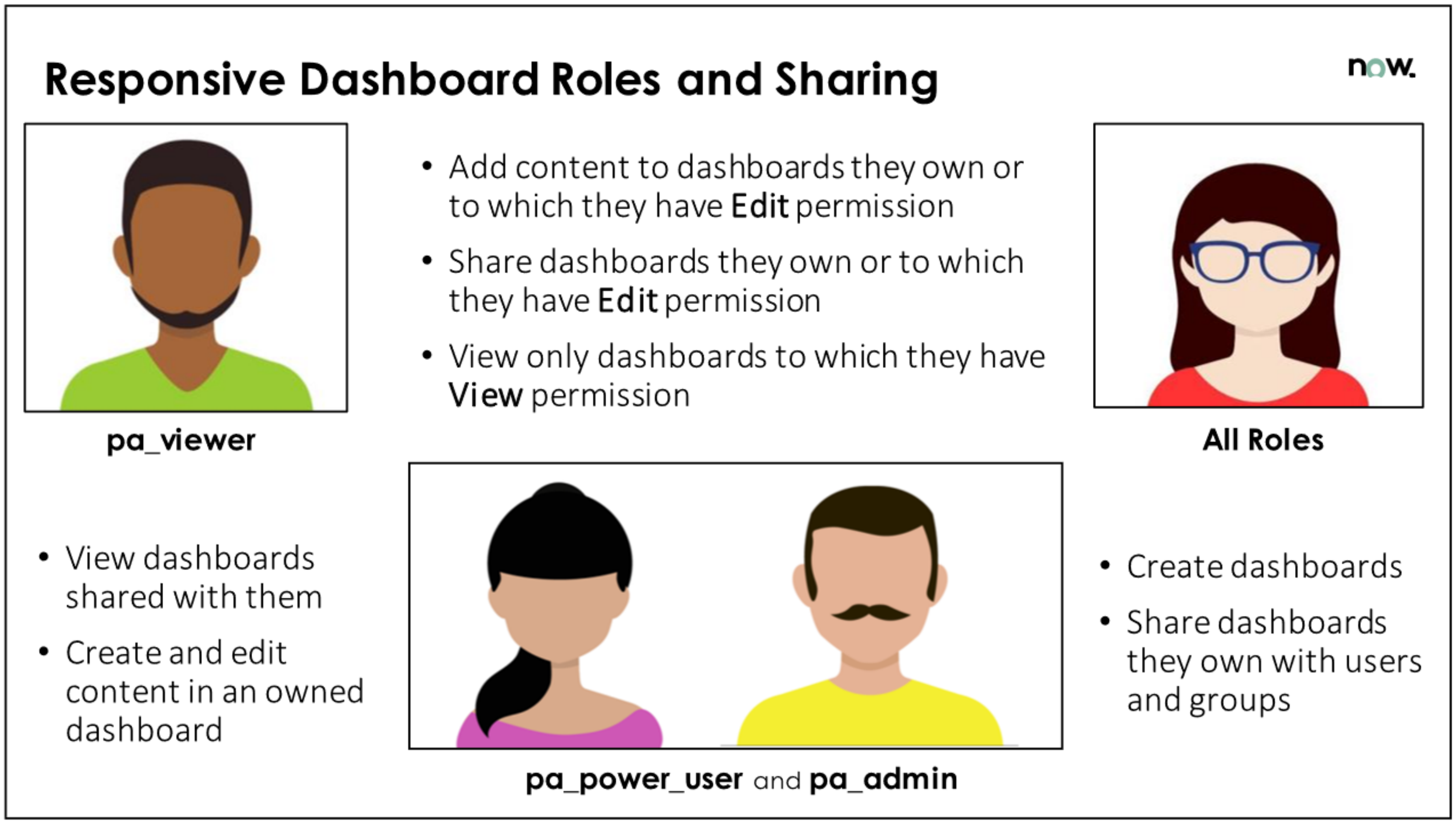
Can View or Can Edit permission

Responsive dashboards have special granular view and edit permissions that are managed from the Sharing panel. When a dashboard is shared with a user or a group, they receive a **Can view** or **Can edit** access to that dashboard.

Any role can restrict access by role to any dashboard that they have created. To do that, specify additional roles in the **Restrict to roles** field. Only users who the dashboard has been shared with and who have one of the specified roles are able to access the dashboard.

Dashboards can be assigned to groups so that users can find the dashboards they want more easily. Dashboard groups determine how dashboards appear in the dashboard picker when you navigate to **Self-Service** > **Dashboards**. You can also add view permissions to dashboard groups. The following is true about Dashboard group and Dashboard permissions:

- View permissions on an individual dashboard override the permissions set at the dashboard group level
- Editing permissions on a dashboard does not affect group permissions
- Dashboard group permissions do not appear in the dashboard Sharing panel



Responsive Dashboard Roles and Sharing

pa_viewer

- View dashboards shared with them
- Create and edit content in an owned dashboard

pa_power_user and **pa_admin**

- Add content to dashboards they own or to which they have **Edit** permission
- Share dashboards they own or to which they have **Edit** permission
- View only dashboards to which they have **View** permission

All Roles

- Create dashboards
- Share dashboards they own with users and groups

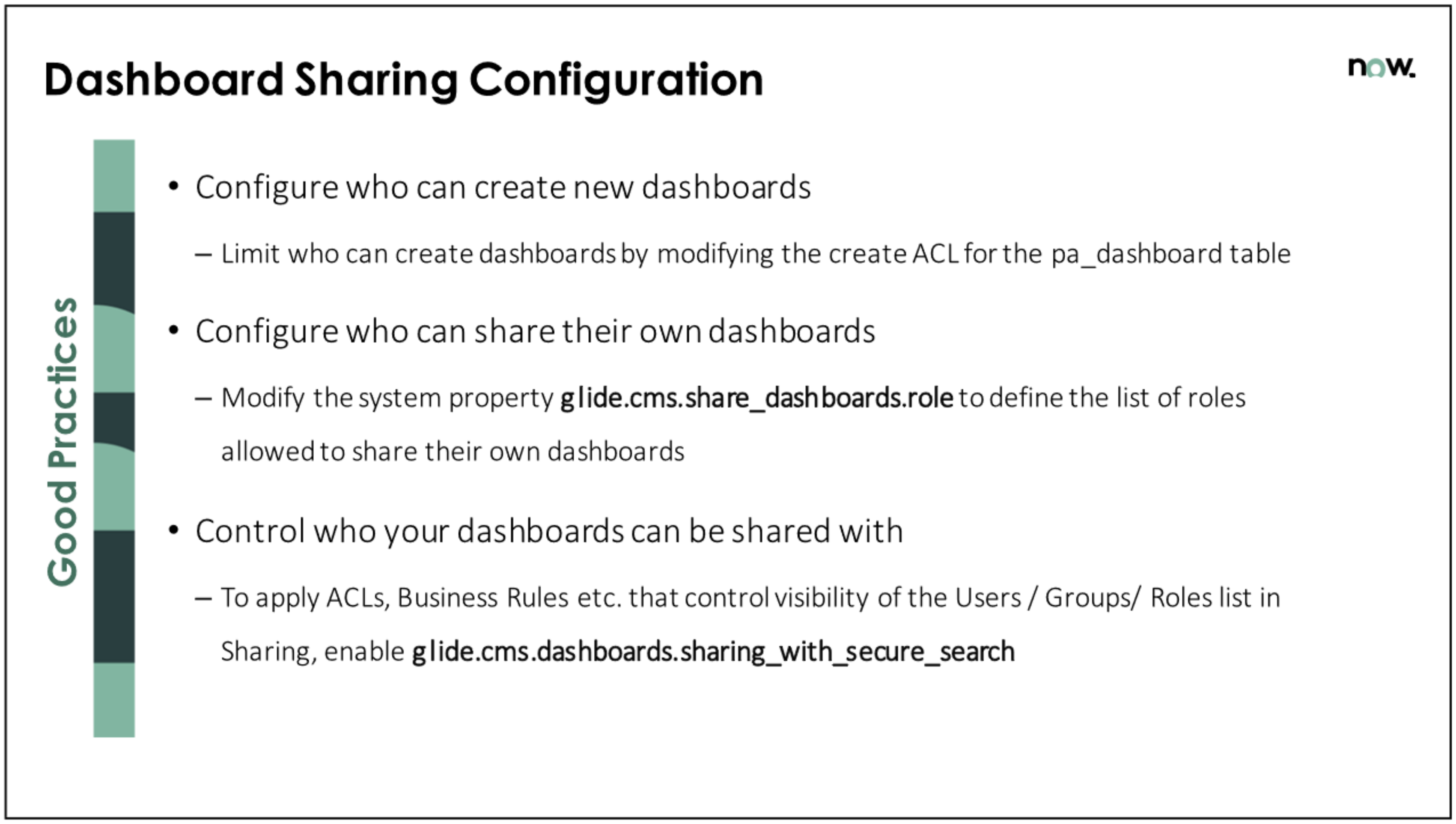
Here is some additional details on dashboard sharing:

- Any role can create and share dashboards that they own with users and groups, and edit dashboards if they have been given **Can edit** permission
- Any role can restrict access by role to any dashboard that they have created.
- Users without a role can view dashboards that have been shared with them, but cannot create or edit dashboards
- The ability of users to share responsive dashboards may be limited by the administrator by setting the **glide.cms.share_dashboards.role** property

Note: To create and edit Performance Analytics widgets, a user needs to have a pa_admin or a pa_power_user role.

Notes on **ACL**s:

- ACLs are not applied to Performance Analytics widgets that are added to dashboards. Any user who can view a dashboard can view all its Performance Analytics widgets.
- ACLs apply when a Report is added to dashboards. If a user can view a dashboard but does not have ACLs to view a Report, an empty widget placeholder is displayed. ACLs do not apply to pie or bar reports. ACLs always apply to list data that is displayed in widgets.



Dashboard Sharing Configuration

Good Practices

- Configure who can create new dashboards
  - – Limit who can create dashboards by modifying the create ACL for the pa_dashboard table
- Configure who can share their own dashboards
  - – Modify the system property **glide.cms.share_dashboards.role** to define the list of roles allowed to share their own dashboards
- Control who your dashboards can be shared with
  - – To apply ACLs, Business Rules etc. that control visibility of the Users / Groups/ Roles list in Sharing, enable **glide.cms.dashboards.sharing_with_secure_search**

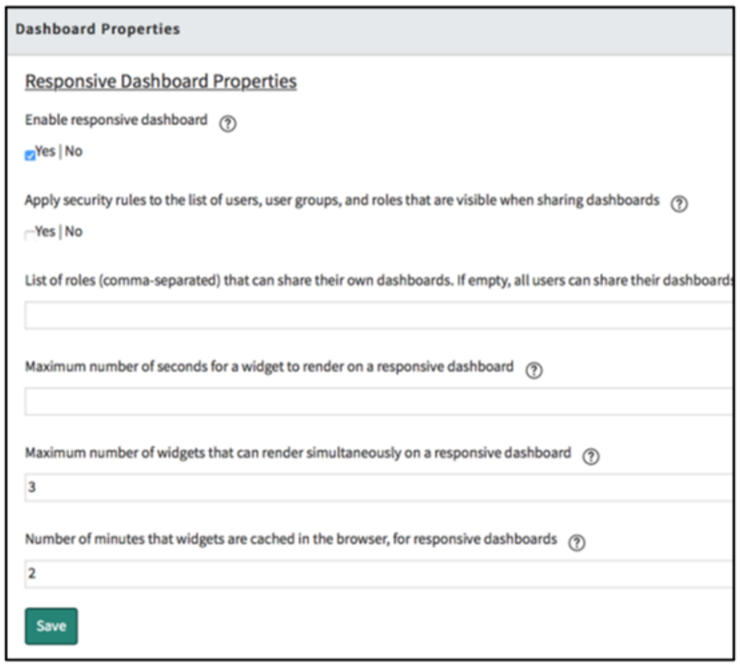
The default behavior of Responsive dashboards can be tailored to support your specific organizational needs. The following capabilities can be modified:

- Adjust who can create new dashboards
  - By default any user that has roles can create new dashboard. If this functionality is not desired by customer they can change the logic by adjusting create ACL for pa_dashboard table to set required logic. It is recommended to create new ACL and disable the existing one.
- Adjust who can share their own dashboards
  - By default all users can share dashboards that they created. To limit this behavior, a System administrator can modify the system property **glide.cms.share_dashboards.role** to define list of roles that should be allowed share own dashboards. This configuration may result in some performance overhead.
- Control who your dashboards can be shared with
  - By default, the list of users/groups/roles that is available in the Sharing panel is not restricted by security rules. To apply security rules (ACLs, Business Rules etc.), the System Admin can enable system property **glide.cms.dashboards.sharing_with_secure_search**

Dashboard Performance Configuration

Good Practices

- Widget Rendering Max Time:
  - Widgets exceeding the allowed time are not rendered
- Concurrent Widget Loading:
  - How many widgets to load with a single request from the server
- Widget Caching
  - Minutes to cache widgets in the browser
- Apply Security rules to Sharing panel lists
  - Results in a performance penalty



Dashboard Properties

Responsive Dashboard Properties

Enable responsive dashboard ⓘ

☑ Yes | No

Apply security rules to the list of users, user groups, and roles that are visible when sharing dashboards ⓘ

☐ Yes | No

List of roles (comma-separated) that can share their own dashboards. If empty, all users can share their dashboards

Maximum number of seconds for a widget to render on a responsive dashboard ⓘ

Maximum number of widgets that can render simultaneously on a responsive dashboard ⓘ

3

Number of minutes that widgets are cached in the browser, for responsive dashboards ⓘ

2

Navigate to **System Properties** > **Dashboard Properties** to configure these properties and achieve optimal performance:

- **glide.canvas.grid.widget_performance_threshold** – the maximum number of seconds for a widget to render on a responsive dashboard. The default value is none or unlimited load time.
- **glide.canvas.grid.widget_render_concurrent_max –** the maximum number of widgets that can render simultaneously on a responsive dashboard. The default value is 3 widgets.
- **glide.canvas.grid.widget_cache_ttl – this is the** Number of minutes that responsive dashboard widgets are cached in the browser. The Default value is 2 minutes.
- **glide.cms.share_dashboards.role** – the list of roles that should be allowed to share own dashboards. Using this configuration may result in some performance overhead.

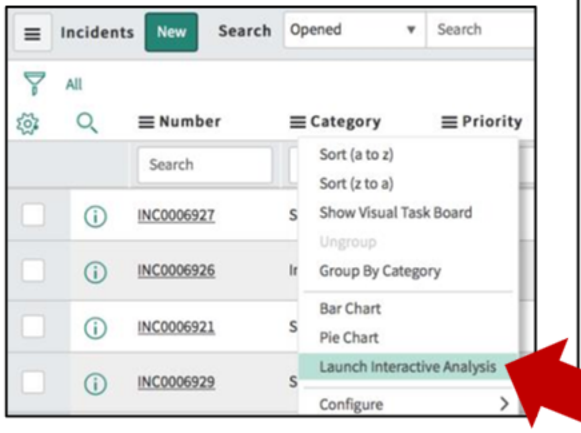
To enable Responsive dashboards in an older instance, you perform these two steps:

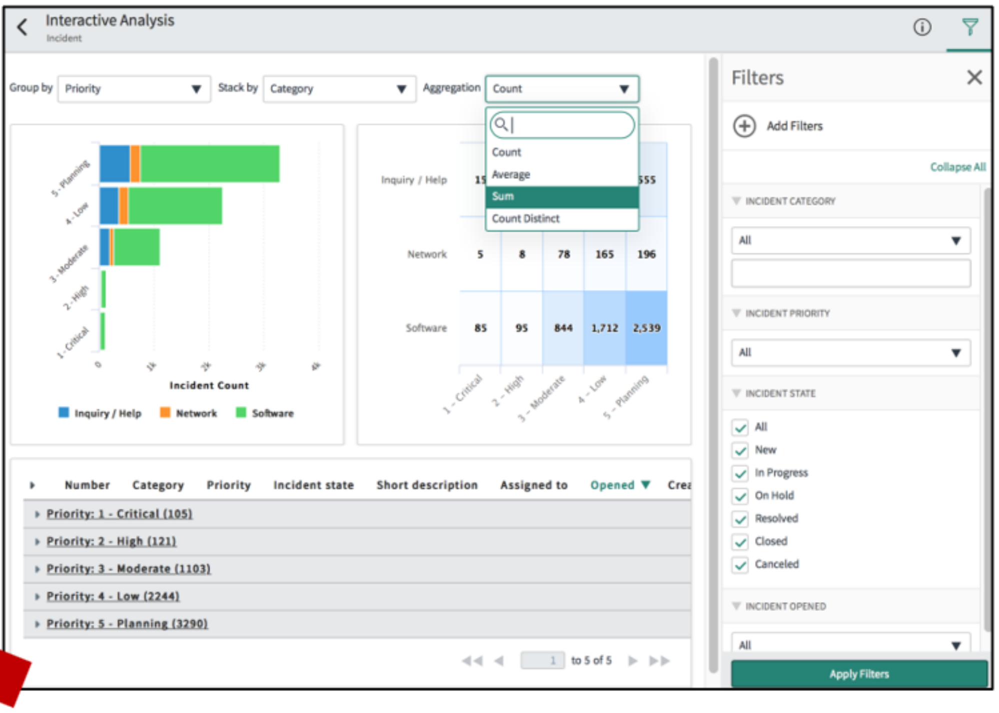
- Activate the responsive dashboards plugin - *com.glideapp.dashboards*
- Set **glide.cms.enable.responsive_grid_layout** to true

Interactive Analysis Dashboard

On-Demand Self Service Analysis on *any* list in the platform

- Auto-generated responsive dashboard
- Flexible aggregation options with pre-built reports and filters





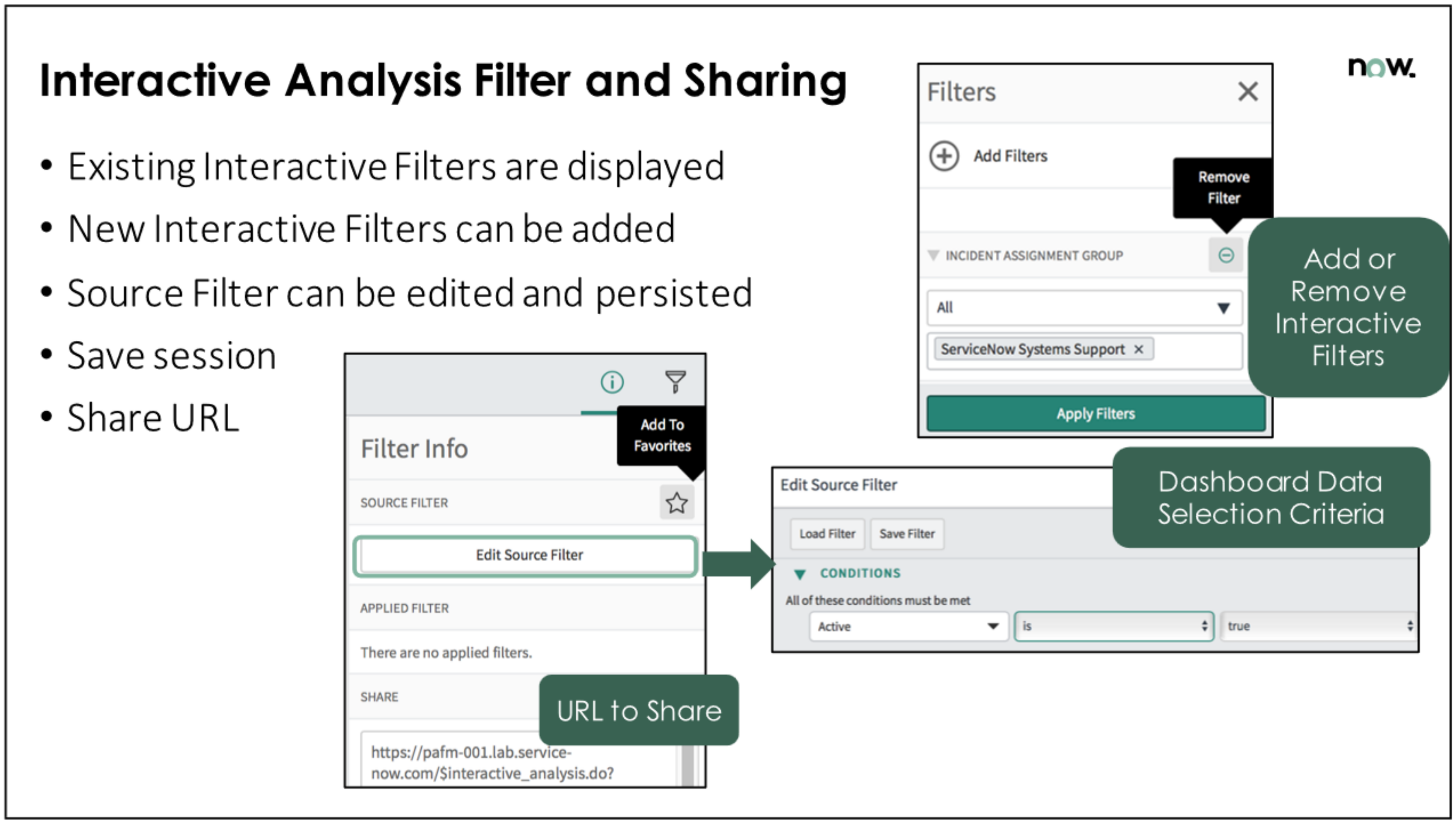
The Interactive Analysis capability is a *Premium* Reporting solution that lets you launch a detailed Analysis dashboard from any list of records in the platform. It offers ad-hoc self service analysis capabilities to task records of any type. This feature provides out of the box summary charting and visualizations by attribute with the ability to quickly apply multiple aggregations and filters. Using the Interactive Analysis eliminates the need to build custom reports and context switch to the Reports app.

The Interactive Analysis dashboard analyzes data by the attributes specified in the **Group by** and **Stack by** fields using the specified **Aggregation.** The dashboard has these charts:

- Horizontal Stacked Bar of the attribute in the Group by field
- Pivot Table of the Stack by and Group by attributes
- List grouped by the Group by attribute

The aggregation field can be of type duration (Date/Time), Integer, or Floating Point. All the visualizations except the List at the bottom of the dashboard are refreshed when a new Aggregation type and field are selected.

To use Interactive Analysis, enable the *com.glideapp.interactive_analysis* plugin.



Interactive Analysis Filter and Sharing

- Existing Interactive Filters are displayed
- New Interactive Filters can be added
- Source Filter can be edited and persisted
- Save session
- Share URL

Add or Remove Interactive Filters

Dashboard Data Selection Criteria

URL to Share

Interactive analysis lets you add any available Interactive filters created on the base table. Follow these steps to manage Interactive filters:

- Click the **Add Filters** button to pick an existing filter
- Select a filter from the list or type the filter name to begin a search. Only existing Interactive Filters for the record set currently displayed are listed.
- To remove a filter, hover over the filter until you see a delete icon. An option is presented to also remove the corresponding Filter Elements from the Group by and Stack by drop-downs.
- When the filter is deleted, the Interactive Analysis dashboard is refreshed.

Favorites and Sharing

- The **Add to Favorites** icon in the Filter Info panel allows a user to bookmark an Interactive Analysis session. The **Remove from Favorites** icon can be used to remove a bookmark.
- The SHARE section of the Filter Info panel contains the Dashboard URL that can be copied and shared to promote collaboration and visibility.
- The **Filter Info** Panel displays the **Source Filter** and additional Applied Filter Information. **Edit Source Filter** allows to modify the base data set from the current analysis session. Perform the desired filtering and click the **Save Changes** button. The Interactive analysis dashboard refreshes with the new filter and the Filter panel is updated with the new source filter details.

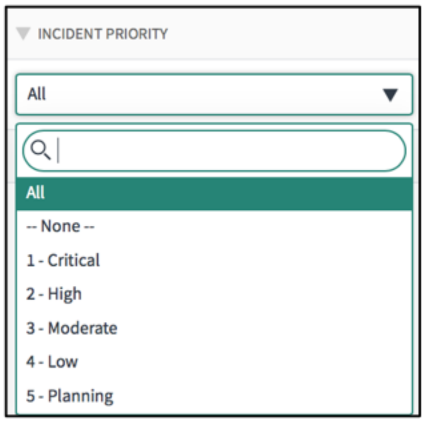
Note on persistence: When bookmarking and sharing, the system preserves the selections on the filters as well as the Group by, Stack by, Aggregation and Filter order.

Interactive Filter Configuration

Interactive Filters are Dashboard Report Filtering Widgets

Available UI control type:

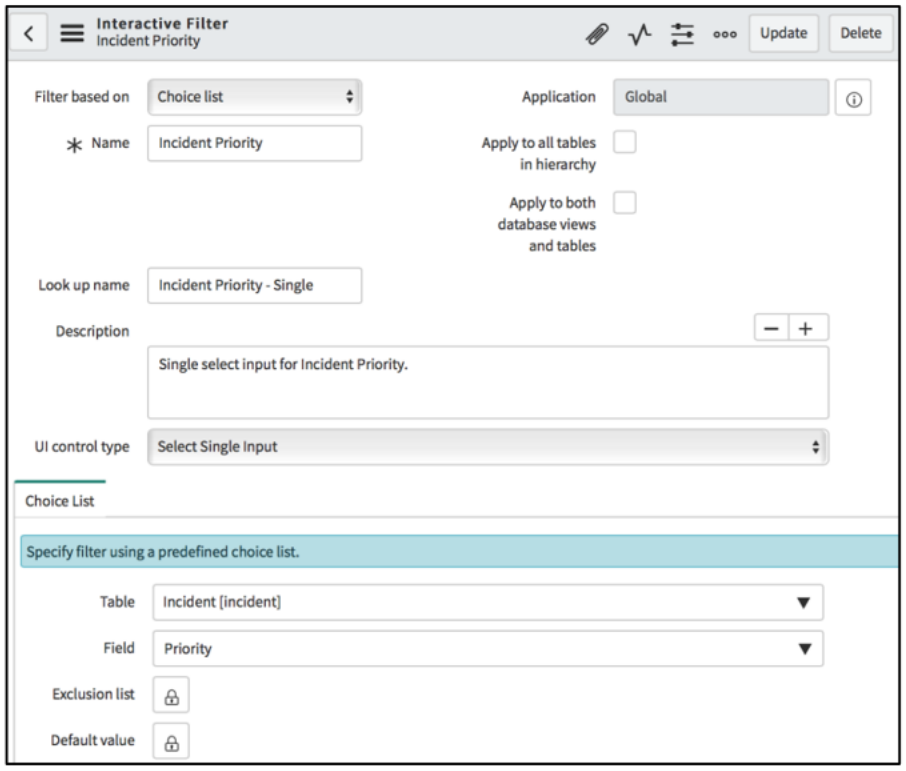
Radio Buttons / Checkboxes / Single Select Dropdown / Multi Select Dropdown



INCIDENT PRIORITY

All ▼

- All
- -- None --
- 1 - Critical
- 2 - High
- 3 - Moderate
- 4 - Low
- 5 - Planning



Interactive Filter
Incident Priority

Filter based on	Choice list	Application	Global
* Name	Incident Priority	Apply to all tables in hierarchy	□
		Apply to both database views and tables	□
Look up name	Incident Priority - Single		
Description	Single select input for Incident Priority.		
UI control type	Select Single Input		

Choice List

Specify filter using a predefined choice list.

Table	Incident [incident]
Field	Priority
Exclusion list	
Default value	

Interactive Filters are widgets that filter the contents of widgets and reports to only display a subset of the data. The Interactive Filters are a *Premium* feature.

Most commonly, the Interactive Filters are based on:

- Reference
- Choice List
- Date
- Boolean (such as the Active field)

Additional supported types are:

- Cascading Filters
- Interactive Filter Groups. With Filter Groups, a user creates a group of existing filters, and selects which one to apply on the Dashboard from a drop-down.

Report Administrators can create Interactive Filters to add to Interactive Analysis and regular dashboards.

By default, Interactive Filters apply to the specified table only. To apply the filter to extending tables as well as database views using the main table select one or both of these options:

- Apply to all tables in hierarchy
- Apply to both database views and tables

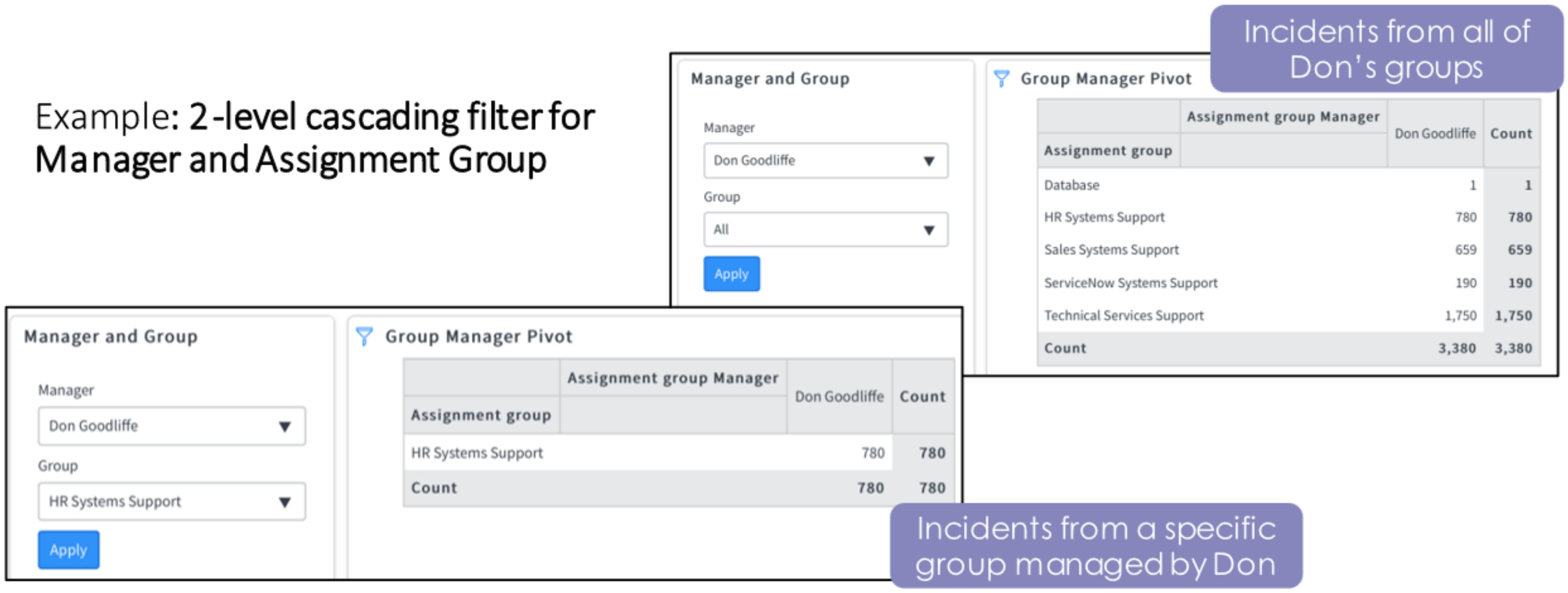
Note that working with Interactive Filters requires the **report_admin** role.

Cascading Filters for Reports

Cascading Filters: Hierarchical filter chain for Dashboard Reports

Cascading filters use an existing relationship in the base report table

Example: **2-level cascading filter for Manager and Assignment Group**



Incidents from all of Don's groups

Manager and Group

Manager: Don Goodliffe

Group: All

Group Manager Pivot

	Assignment group Manager	
Assignment group	Don Goodliffe	Count
Database	1	1
HR Systems Support	780	780
Sales Systems Support	659	659
ServiceNow Systems Support	190	190
Technical Services Support	1,750	1,750
Count	3,380	3,380

Manager and Group

Manager: Don Goodliffe

Group: HR Systems Support

Group Manager Pivot

	Assignment group Manager	
Assignment group	Don Goodliffe	Count
HR Systems Support	780	780
Count	780	780

Incidents from a specific group managed by Don

Cascading Filters are a special type Interactive Filters that filter the contents of reports to display hierarchical relationships.

In the example here, the Cascading Filter is used to display incidents as follows:

- All Managers and their Assignments groups
- A specific Manager and all the groups he or she manages
- Specific Manager and specific Assignment Group that he or she manages

A Cascading filter is meant to be used only when there is an established structure, for example:

- A manager can manage one or more groups
- A district can have one or more territories

Note that working with Cascading Filters requires the **report_admin** role.

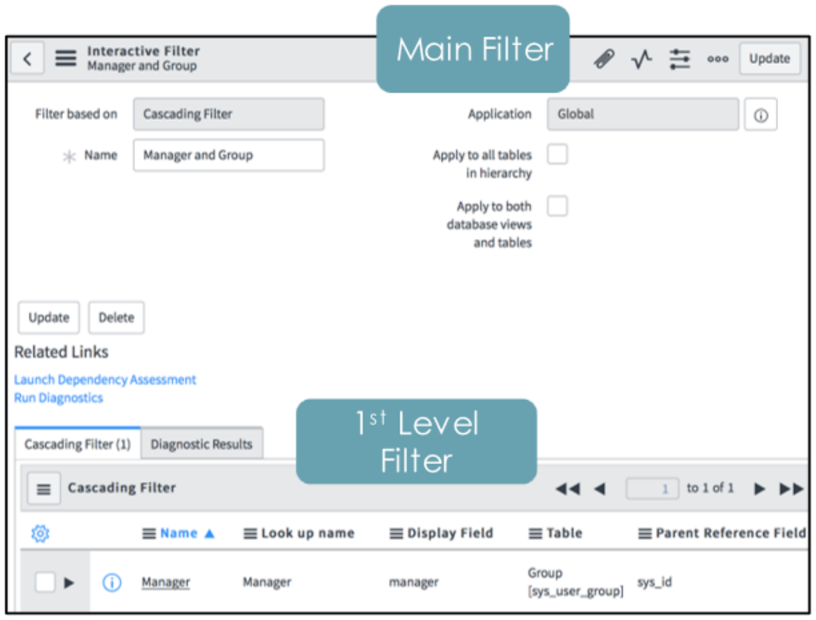
Cascading Filter Creation

Create Main Filter Definition

- Specify **Filter based on Cascading Filter**

For every additional Filter Level define:

- Filter Selection
- Target Mapping



Main Filter

Interactive Filter
Manager and Group

Update

Filter based on: Cascading Filter

Application: Global

* Name: Manager and Group

Apply to all tables in hierarchy ☐

Apply to both database views and tables ☐

Update | Delete

Related Links

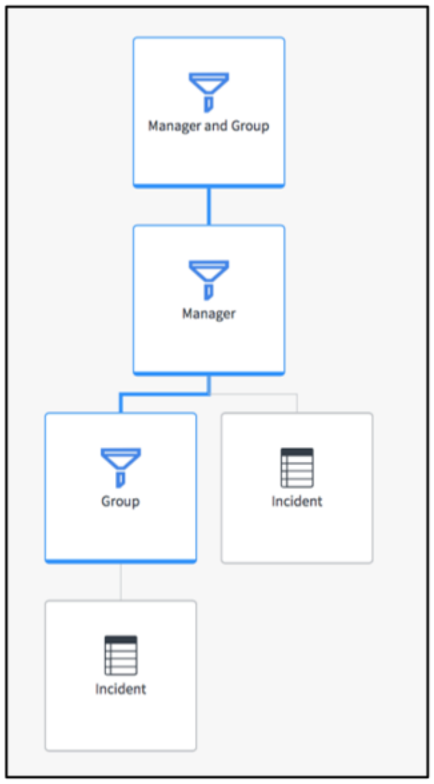
Launch Dependency Assessment
Run Diagnostics

1st Level Filter

Cascading Filter (1) | Diagnostic Results

Cascading Filter — 1 to 1 of 1

Name ▲	Look up name	Display Field	Table	Parent Reference Field
Manager	Manager	manager	Group [sys_user_group]	sys_id



Here are the steps to create a new Cascading Filter:

- Create a new Interactive Filter and set **Filter based on** to **Cascading Filter**

For every additional Cascading filter level, perform the following steps:

- Add new Cascading Filter in the related list to represent the first filter level. There can only be one Cascading filter at each level.
- Define the Display filter and the table that contains the data for this filter
- Specify the Target table to which the filter needs to map to

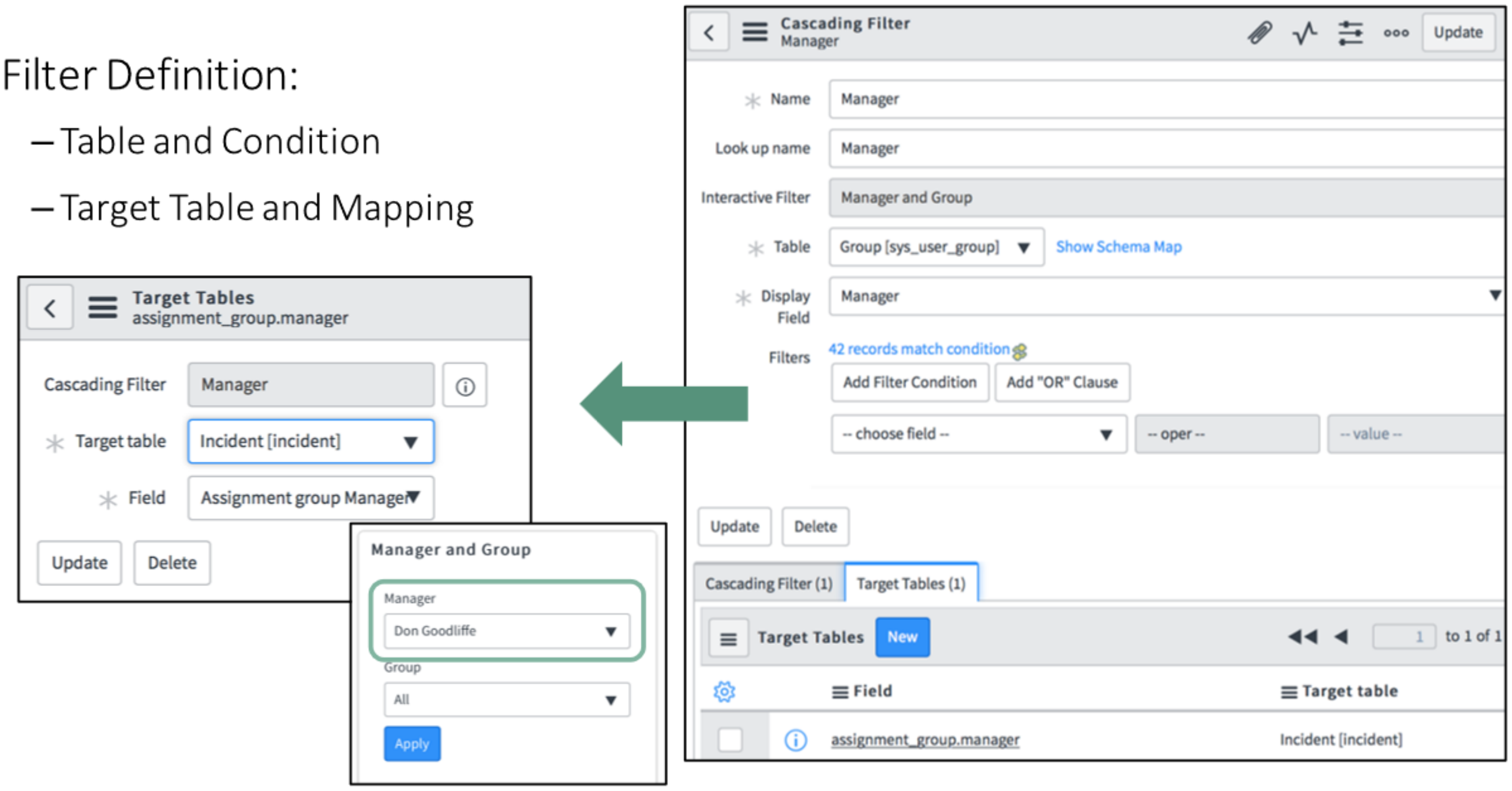
Example:

1. Create a Main Filter **Manager and Group** based on Cascading Filter
2. Add the Level 1 Filter **Manager**
   a. Table: group
   b. Display Field: manager
   c. Parent Reference Field: sys_id
3. The Manager filter **selects all managers from the sys_user_group table**
4. The target table is **incident** and the mapping field is **assignment_group.manager**

1st Level Filter

Filter Definition:

- Table and Condition
- Target Table and Mapping



Cascading Filter
Manager

Name	Manager
Look up name	Manager
Interactive Filter	Manager and Group
Table	Group [sys_user_group]
Display Field	Manager
Filters	42 records match condition

Target Tables

Field	Target table
assignment_group.manager	Incident [incident]

Target Tables
assignment_group.manager

Cascading Filter	Manager
Target table	Incident [incident]
Field	Assignment group Manager

Manager and Group

Manager	Don Goodliffe
Group	All

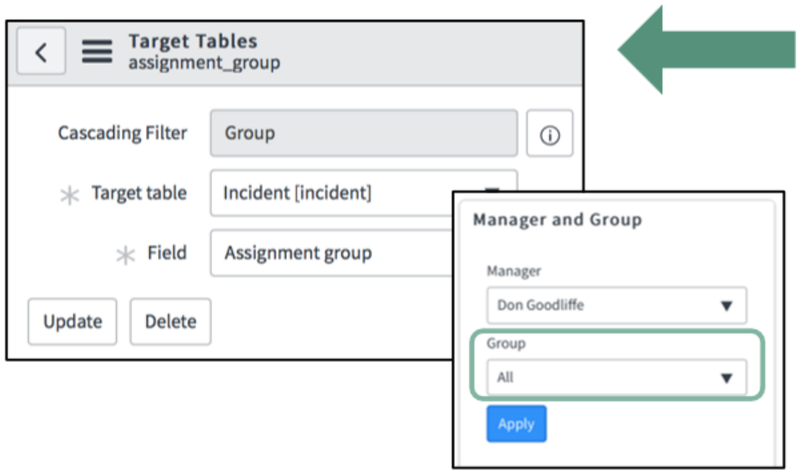
Here are the steps to create a first level Cascading Filter:

1. Create a new Cascading Filter and define the following:
   - Name
   - Table
   - Display Field
   - Filter Condition
2. Create a new Target table and define:
   - Target table
   - Field

The above two steps let the filter know what to display in the filter drop down (Group Managers) and to which Incident field to apply it (Incident's assignment_group.manager).

2nd Level Filter

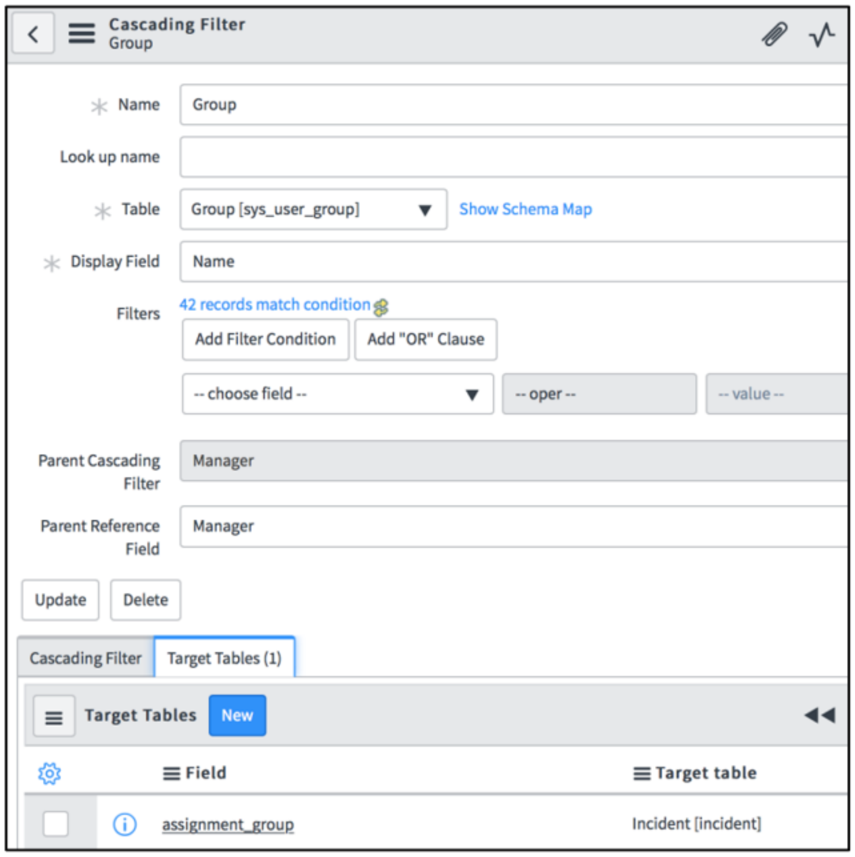
- Relate a new 2nd Level Filter to the 1st Level Filter
- Define the 2nd Level Filter
  - Name / Table / Parent Reference Field
  - Target Table and Mapping



Target Tables
assignment_group

Cascading Filter: Group
Target table: Incident [incident]
Field: Assignment group

Manager and Group
Manager: Don Goodliffe
Group: All



Cascading Filter
Group

Name: Group
Look up name:
Table: Group [sys_user_group]
Display Field: Name
Filters: 42 records match condition
-- choose field -- | -- oper -- | -- value --
Parent Cascading Filter: Manager
Parent Reference Field: Manager

Field	Target table
assignment_group	Incident [incident]

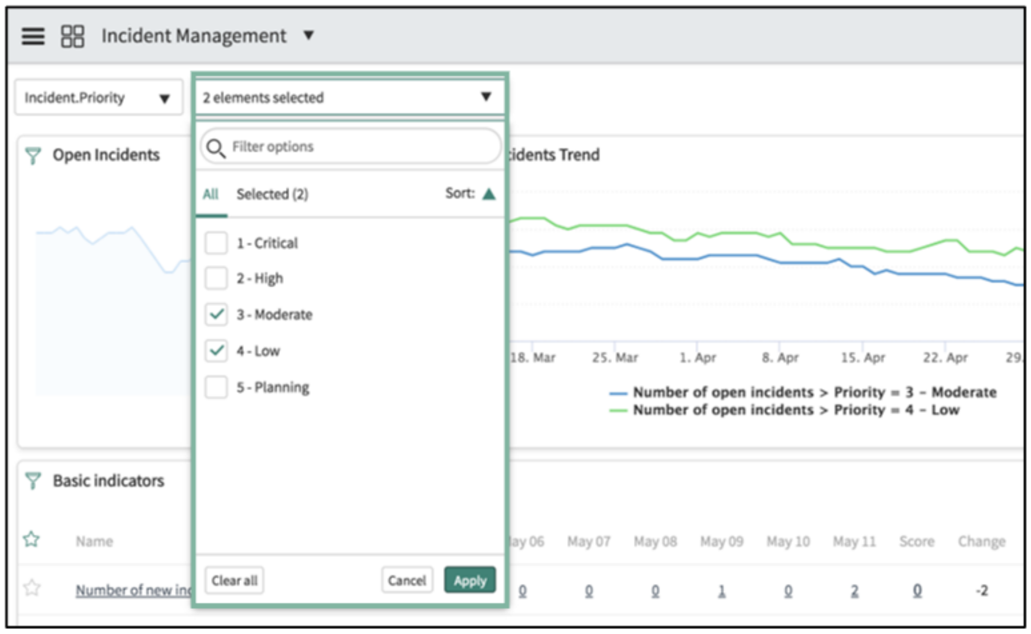
To create the 2nd level of Filter, add a new Cascading Filter from the related list of the first Cascading filter. In the example here, you create the Group as the related Cascading filter to the Manager Cascading filter.

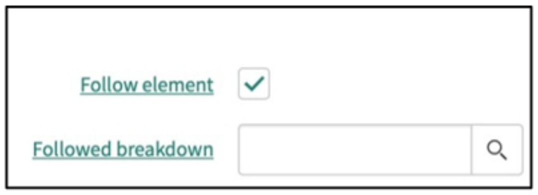
1. Define a filter as follows:
   - Enter a Name (Group)
   - Specify the Table and Field to show in the field dropdown (Group and Name)
   - Optionally, specify a condition to refine the filter table selection
   - Specify the Parent Reference Field – the base table 1st level filter reference field (Manager)
2. Create a new Target Table
   - Define table (incident)
   - Define field to map to (assignment_group)

Note on Targets: A cascading filter hierarchy must specify at least one target. You can define a cascading filter that skips levels in a hierarchy, or a cascading filter that only specifies targets for certain levels in a hierarchy. Defining a Target for the Manager level allows to filter reports as soon as a user selects a manager.

Breakdown Source Dashboards

Dashboard Breakdown Sources let you apply Breakdown elements as Widget data filters




Follow element ☑

Followed breakdown

- Multi-Element select is allowed
- Element selection is persisted
- Widgets need “Follow Element”

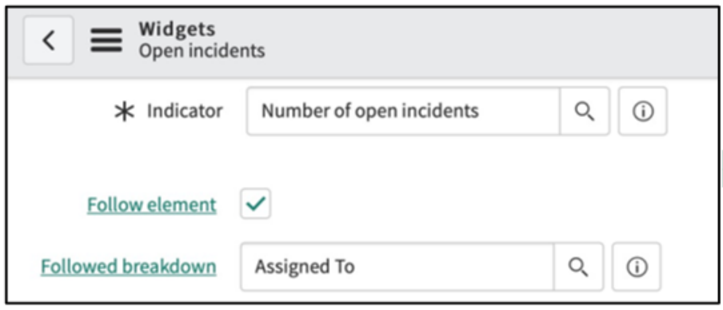
Breakdown dashboards are used to filter the contents of multiple Performance Analytics widgets using one or more Breakdown elements.

In order for widgets to “listen” to the Breakdown source and be filtered, these three properties must be configured:

Follow element: Check to make the widget respond to the selected breakdown on the dashboard.

Followed breakdown: Specify the breakdown to apply if multiple breakdowns using the same Breakdown Source exist on a dashboard

Followed Breakdown Example

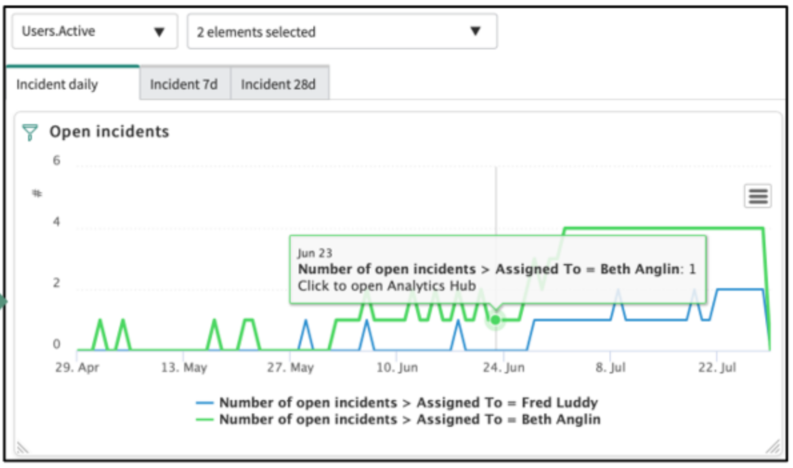


Widgets
Open incidents

* Indicator: Number of open incidents

Follow element ☑

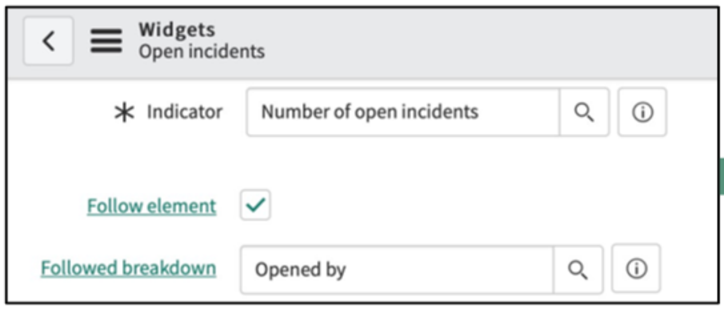
Followed breakdown: Assigned To



Users.Active ▼ | 2 elements selected ▼

Incident daily | Incident 7d | Incident 28d

Open incidents
6
4
2
0
#
Jun 23
Number of open incidents > Assigned To = Beth Anglin: 1
Click to open Analytics Hub
29. Apr
13. May
27. May
10. Jun
24. Jun
8. Jul
22. Jul
Number of open incidents > Assigned To = Fred Luddy
Number of open incidents > Assigned To = Beth Anglin

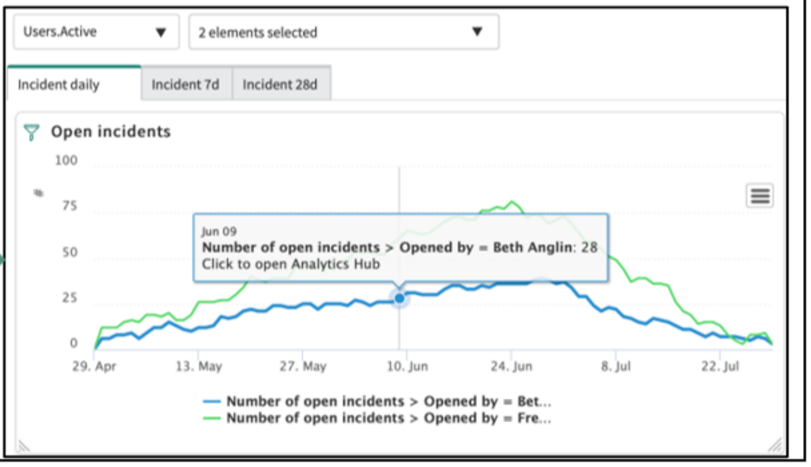


Widgets
Open incidents

* Indicator: Number of open incidents

Follow element ☑

Followed breakdown: Opened by



Users.Active ▼ | 2 elements selected ▼

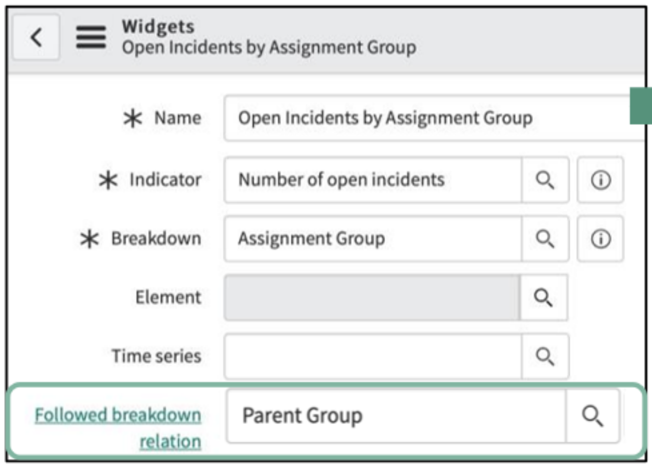
Incident daily | Incident 7d | Incident 28d

Open incidents
100
75
50
25
0
#
Jun 09
Number of open incidents > Opened by = Beth Anglin: 28
Click to open Analytics Hub
29. Apr
13. May
27. May
10. Jun
24. Jun
8. Jul
22. Jul
Number of open incidents > Opened by = Bet...
Number of open incidents > Opened by = Fre...

The above example illustrates the case of specifying a **Followed breakdown**. The **Assigned To** Breakdown and the **Opened by** Breakdown are both based on the **Users.Active** Breakdown Source. To specify which filter to apply, you need to enter the Breakdown in the **Followed breakdown** field. As shown in the example, the widget is filtered using two different Breakdowns.

Followed Breakdown Relation Example

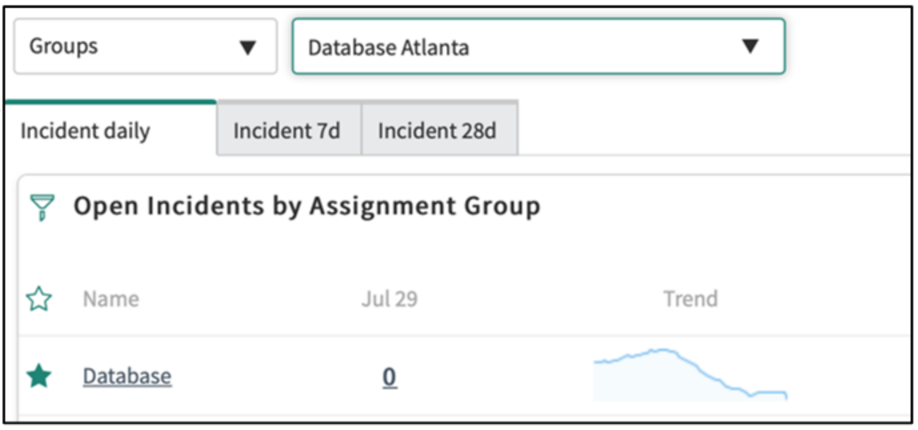
Scorecard visualizations can be configured to follow a Breakdown Relation



Widgets
Open Incidents by Assignment Group

* Name	Open Incidents by Assignment Group
* Indicator	Number of open incidents
* Breakdown	Assignment Group
Element	
Time series	
Followed breakdown relation	Parent Group

Database is the Parent Group for Database Atlanta



Groups ▼ Database Atlanta ▼

Open Incidents by Assignment Group

Name	Jul 29	Trend
Database	0	

The Widget shows the Parent group of Database Atlanta - the Database group

A breakdown widget can display 1st level breakdown elements that are related to the element selected for the dashboard. The widget must be on a breakdown dashboard, and that dashboard must include the breakdown sources of the related breakdowns. In our example, the Open Incidents by Assignment Group widget is configured as follows:

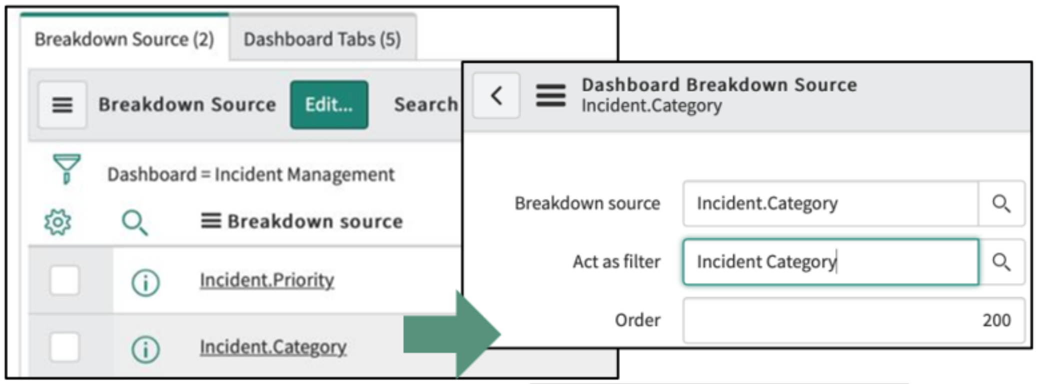
- Type: Breakdown, Visualization: Scorecard
- Breakdown: Assignment Group with Breakdown Source: Groups
- Placed on a Dashboard with the Groups Breakdown source
- Followed breakdown relation: Parent Group

When selecting the Database Atlanta Groups element, the widget only shows the Parent Group of the Database Atlanta group – the Database group

Note that the **Followed breakdown relation** menu works only with Scorecard visualizations. Also, multiple element selection on a breakdown dashboard is not supported on the widget when you set a breakdown relation to follow.

Breakdowns and Charts as Interactive Filters

Breakdown Elements can multitask as Interactive Filters by configuring the **Act as filter** property



Breakdown Source (2) | Dashboard Tabs (5)

Breakdown Source | Edit... | Search

Dashboard = Incident Management

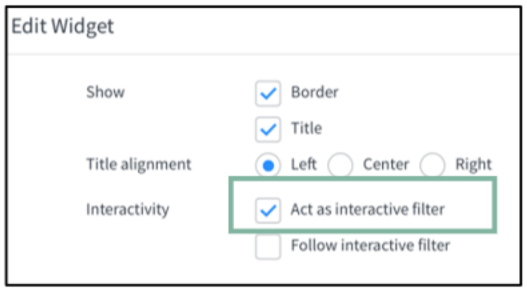
Breakdown source

- Incident.Priority
- Incident.Category

Dashboard Breakdown Source
Incident.Category

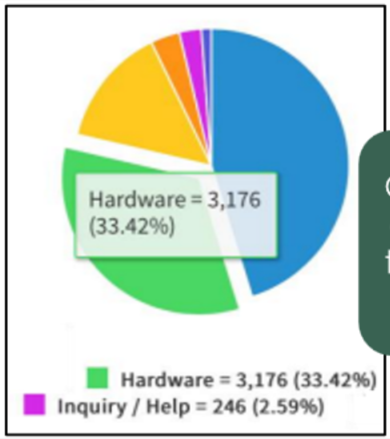
Breakdown source	Incident.Category
Act as filter	Incident Category
Order	200

Summary Charts can act as Interactive Filters by enabling the **Act as interactive filter** property



Edit Widget

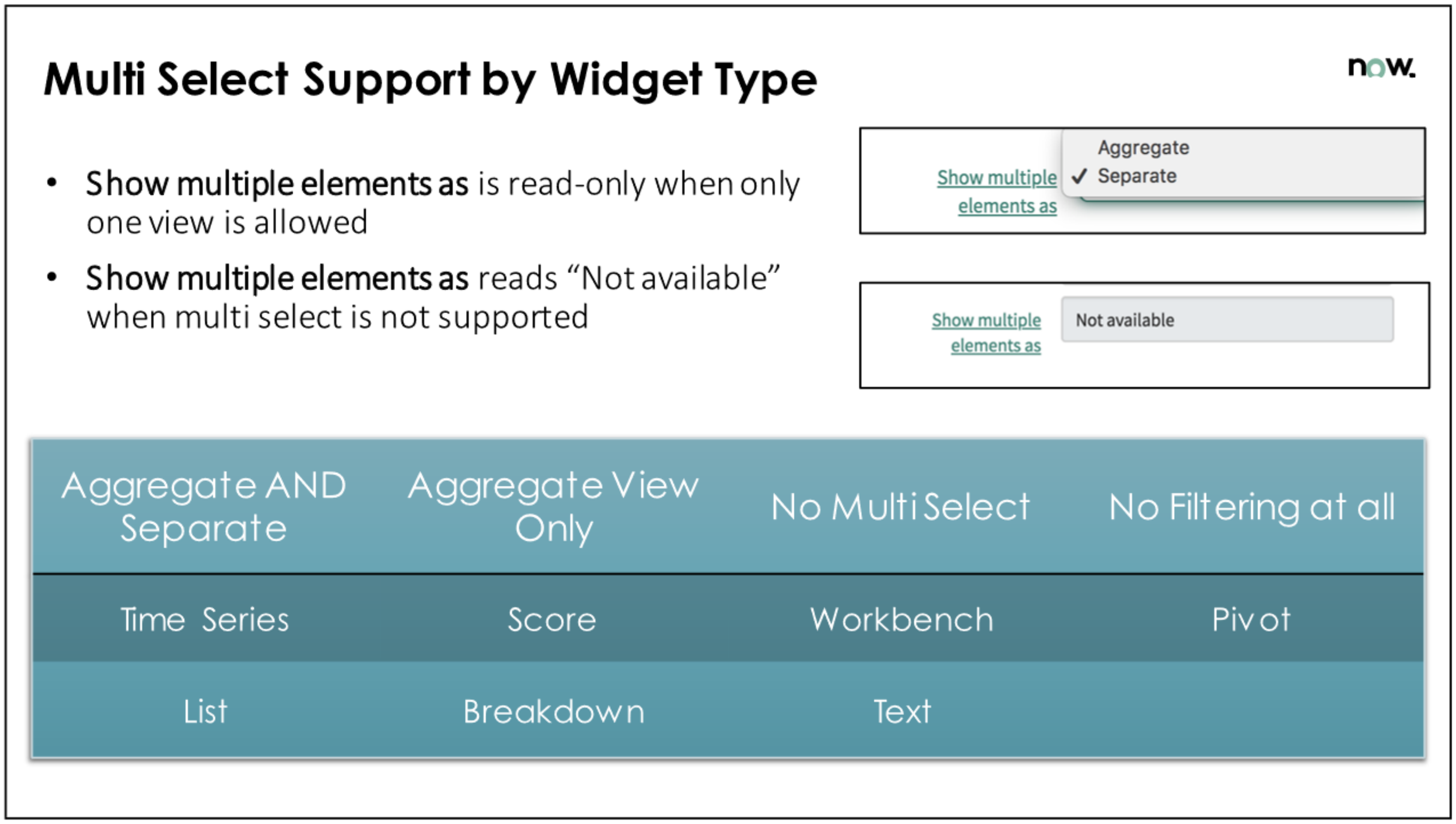
Show	☑ Border ☑ Title
Title alignment	● Left ○ Center ○ Right
Interactivity	☑ Act as interactive filter ☐ Follow interactive filter



Click on a category to apply it as filter

Power Users and PA Admins can configure a Breakdown Source on a dashboard to behave or "act" as an Interactive filter for reports. The breakdown source **Facts table** must match the table that the Interactive filter is based on. In this way, Breakdown elements can filter the contents of both Performance Analytics widgets and Reports.

Report widgets with of type pie, donut, semi donut, funnel, or pyramid can be configured to **Act as Interactive filter** using the Interactivity settings. In this example, the Incident.Category Breakdown source is configured to apply to incident reports and "act like" the corresponding Incident.Category Interactive Filter.



Multi Select Support by Widget Type

- **Show multiple elements as** is read-only when only one view is allowed
- **Show multiple elements as** reads “Not available” when multi select is not supported

Show multiple elements as
Aggregate
✓ Separate

Show multiple elements as
Not available

Aggregate AND Separate	Aggregate View Only	No Multi Select	No Filtering at all
Time Series	Score	Workbench	Pivot
List	Breakdown	Text	

For widgets that allow multiple element selection, the **Show multiple elements as** property can be set to one of these two settings:

- **Aggregate**: The trends are combined. For example, there will be a single line representing High and Critical incidents.
- **Separate**: The trends are separate. For example, there will be two trend lines, one for High and another for the Critical priority incidents

Notes:

- When multiple selection is not allowed, the Show multiple elements as box is greyed out and says “Not available”

In addition to the widget type, the support for multi selection of Breakdown Elements varies according to the type of Indicator and its visualization. The table above lists support for multiple elements by Widget Type. In general, the Aggregate view is not supported for the following Indicator types:

- Formula indicators
- Automated indicators with Aggregation type Average or Count Distinct

Dashboards and Performance Lab

Lab 5.1: Responsive Dashboards

Lab 5.2: Interactive Analysis – Optional Lab

Responsive Dashboards

Lab
5.1
⌛30m

Lab Objectives

As a Customer Success Manager, you require a real-time view of Incident Management processes, Outage statistics, and Customer Satisfaction data

This lab will show you how to do the following:

- Create and populate a dashboard.
- Filter dashboard data using Breakdowns and Interactive Filters
- Manage dashboard configuration

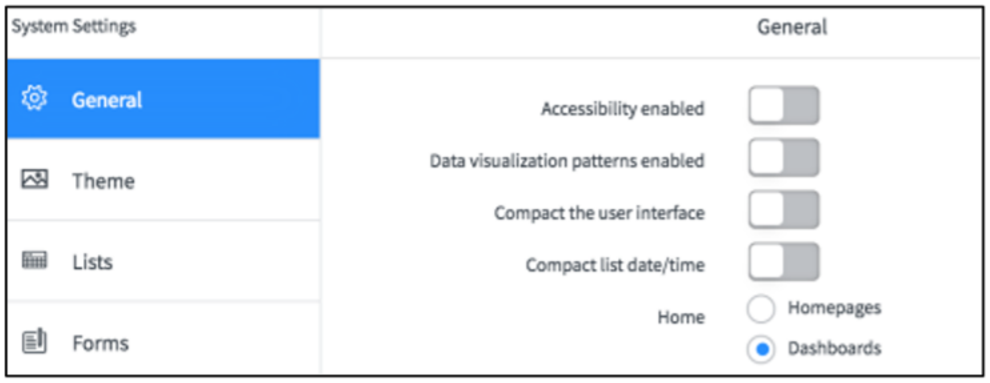
> ***Dependency Note*** *– This lab uses Indicators and Reports created in previous labs.*

A. Properties and Plugins

Personal Settings

1. Navigate to the **System Setting** and select **Dashboards** as your Homepage as shown.

> ***Note:*** *Clicking the ServiceNow logo in the banner takes you to the last visited Dashboard.*



General

Accessibility enabled ○
Data visualization patterns enabled ○
Compact the user interface ○
Compact list date/time ○
Home ○ Homepages ● Dashboards

Plugins Verification

1. Navigate to **System Definition > Plugins**.

2. Confirm that the following plugins are active:

   - Responsive Canvas (com.glideapp.canvas)
   - Responsive Dashboards (com.glideapp.dashboards)

Properties

1. Navigate to **System Properties > Dashboard Properties.**
2. Confirm that the **Enable responsive dashboard** property (glide.cms.enable.responsive_grid_layout) is set to **Yes**.

> ***Tip:*** *In upgraded systems this property may not exist. If that is the case, or if the property is set to true, the Responsive Dashboard functionality is enabled.*

B. New Dashboard

Create Dashboard

1. Navigate to **Performance Analytics > Dashboards**.
2. Click **New** and create a new Dashboard as follows:
   - Name: **Customer Success Advocate**
   - Group: leave empty
   - Order: **10**
   - Active: **checked**
   - Restricted to: **itil** role

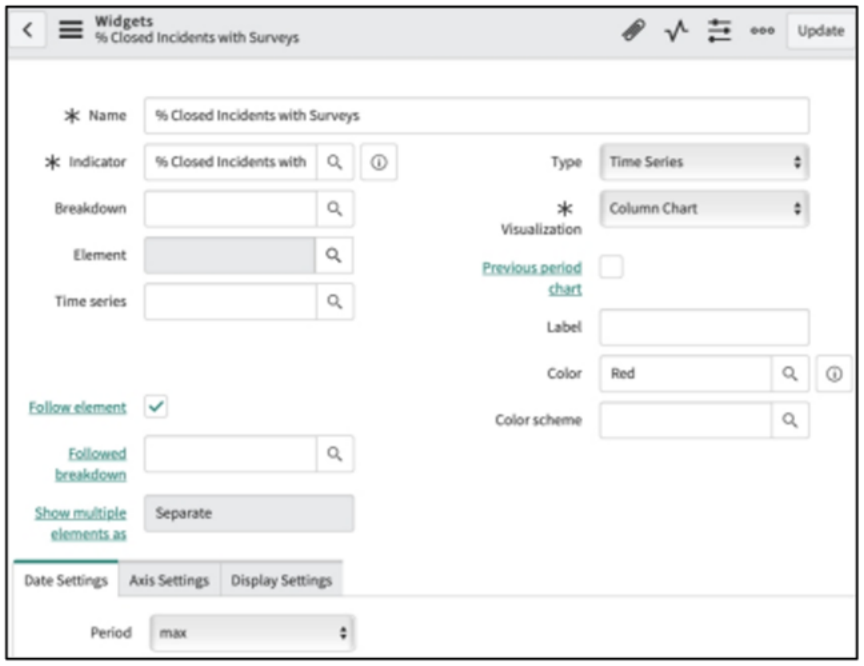
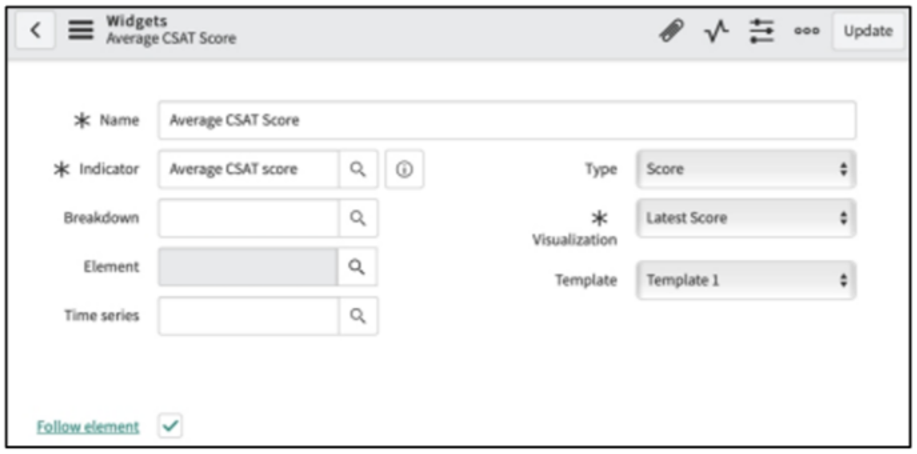
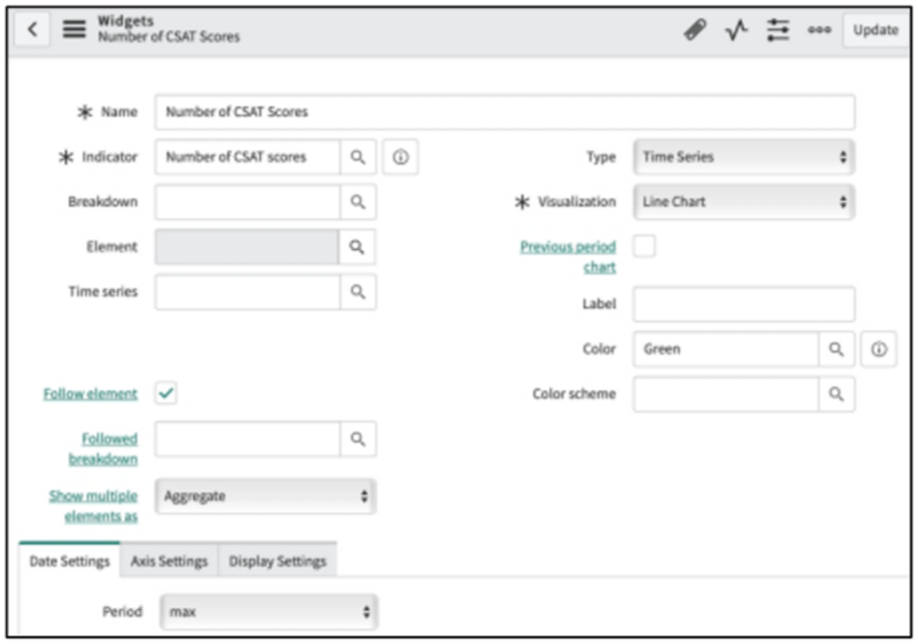
> ***Tips:*** *Leaving the Group empty places the new Dashboard under the* ***Other*** *section.*
>
> *A lower Order setting insures that the dashboard is listed towards the top of a list.*

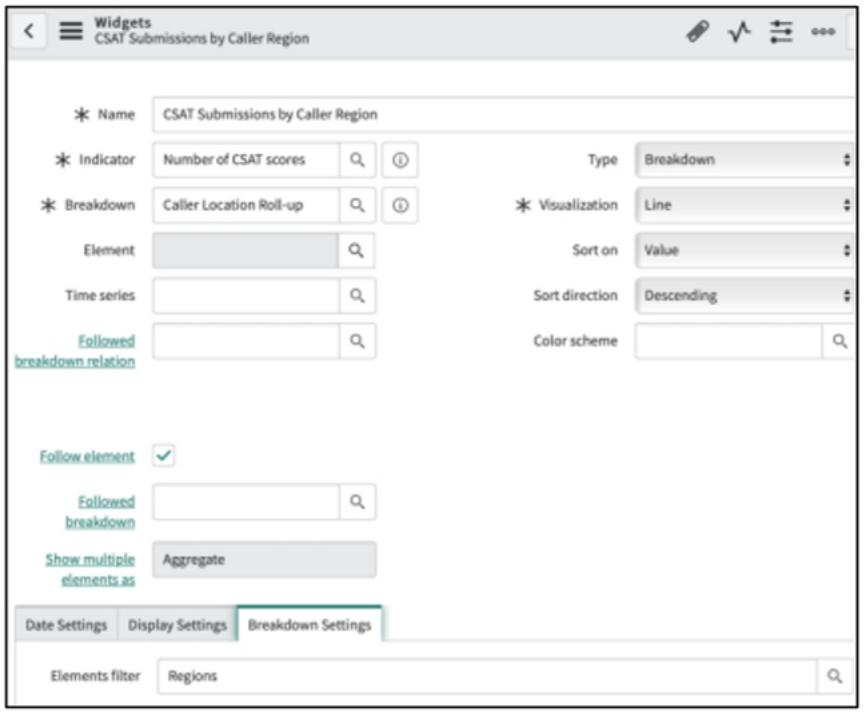
3. Click **Submit**.

Add Dashboard Content

1. Click the **Configuration** button in the new dashboard.
2. Click the **Create Tab** twice and rename the three new tabs as follows:
   - **Incident Management**
   - **Outage Resolution**
   - **Customer Satisfaction**
3. Navigate to the **Customer Satisfaction** dashboard tab.

4. Add these NEW Performance Analytics widgets as specified:

Name: **% Closed Incidents with Surveys** Type: **Time Series** Visualization: **Column Chart** Indicator: **% Closed Incidents with Surveys** Color: **Red** Follow element: **checked** Date Settings: Period: **max** Show date range selector: **checked**	 Widgets % Closed Incidents with Surveys Update * Name: % Closed Incidents with Surveys * Indicator: % Closed Incidents with Type: Time Series Breakdown: * Visualization: Column Chart Element: Previous period chart: □ Time series: Label: Color: Red Follow element: ☑ Color scheme: Followed breakdown: Show multiple elements as: Separate Date Settings / Axis Settings / Display Settings Period: max
Name: **Average CSAT Score** Type: **Score** Visualization: **Latest Score** Indicator: **Average CSAT scores** Template: **Template 1** Follow element: **checked**	 Widgets Average CSAT Score Update * Name: Average CSAT Score * Indicator: Average CSAT score Type: Score Breakdown: * Visualization: Latest Score Element: Template: Template 1 Time series: Follow element: ☑
Name: **Number of CSAT Scores** Type: **Time Series** Visualization: **Line Chart** Indicator: **Number of CSAT scores** Color: **Green** Follow element: **checked** Date Settings: Period: **max** Show date range selector: **checked**	 Widgets Number of CSAT Scores Update * Name: Number of CSAT Scores * Indicator: Number of CSAT scores Type: Time Series Breakdown: * Visualization: Line Chart Element: Previous period chart: □ Time series: Label: Color: Green Follow element: ☑ Color scheme: Followed breakdown: Show multiple elements as: Aggregate Date Settings / Axis Settings / Display Settings Period: max

Name: **CSAT Submissions by Caller Region** Indicator: **Number of CSAT scores** Breakdown: **Caller Location Roll-up** Type: **Breakdown** Visualization: **Line** Follow element: **checked** Breakdown settings > Elements filter: **Regions**	

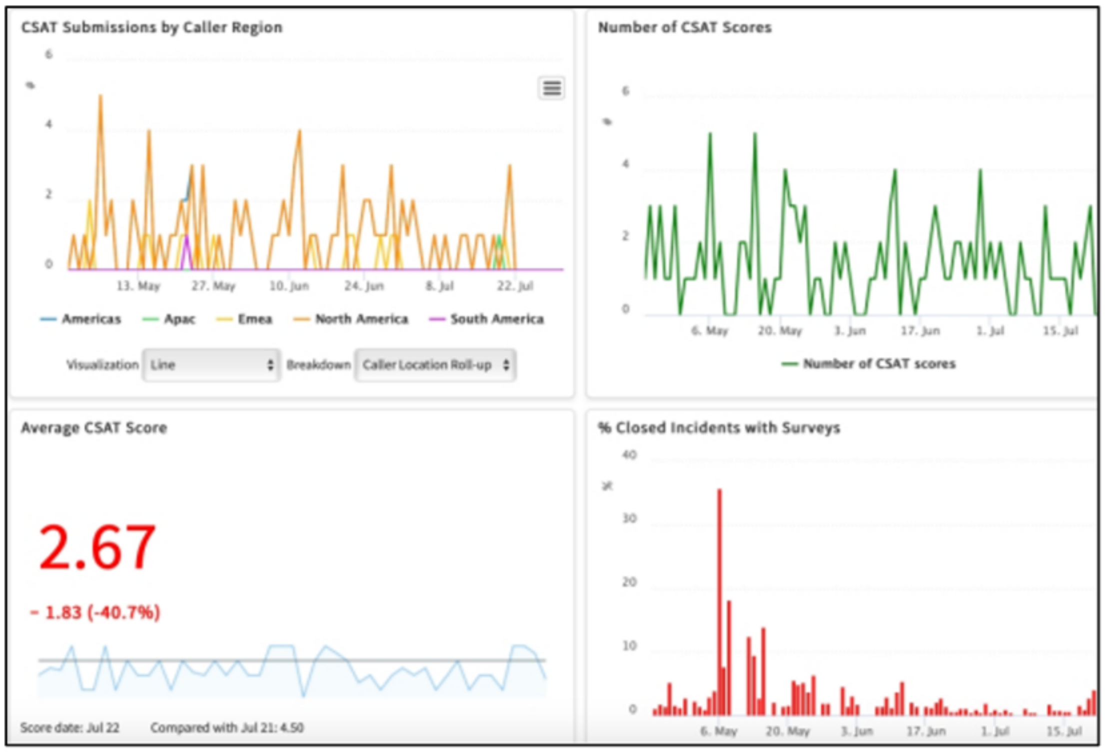
Widgets
CSAT Submissions by Caller Region

- ✱ Name: CSAT Submissions by Caller Region
- ✱ Indicator: Number of CSAT scores
- Type: Breakdown
- ✱ Breakdown: Caller Location Roll-up
- ✱ Visualization: Line
- Element:
- Sort on: Value
- Time series:
- Sort direction: Descending
- Followed breakdown relation:
- Color scheme:
- Follow element: ☑
- Followed breakdown:
- Show multiple elements as: Aggregate

Date Settings | Display Settings | Breakdown Settings

- Elements filter: Regions

5. Arrange the dashboard layout as shown below and verify correct widget display.



Visualization: Line | Breakdown: Caller Location Roll-up

6. Navigate to the **Incident Management** Dashboard tab.

7. Add the following **Reports** to the **Incident Management** tab:

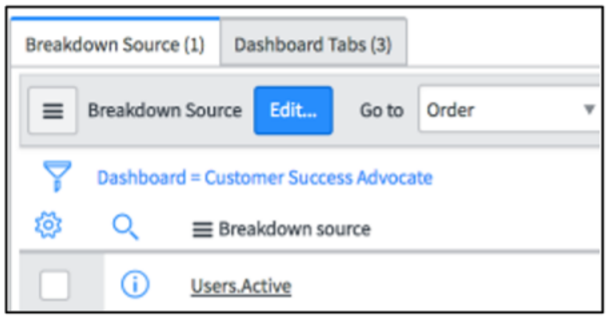
- **Incident Counts by Caller Location**
- **Incidents at Risk**
- **Incident SLAs Completion Breakdown**

- **Incidents by Priority and State with Breached SLAs**

8. Apply the **2x2** layout in the **Configuration** panel.
9. Navigate to the **Outage Resolution** Dashboard tab.
10. Add the **Outages by Root Cause** report to the **Outage Resolution** tab.

C. Dashboard Filtering using Breakdown Source

1. Open the **Dashboard Properties** of the **Customer Success Advocate** dashboard.
2. Click **Edit…** the **Breakdown Sources** Related List.
3. Add the **Users.Active** Breakdown source.



Dashboard = Customer Success Advocate

Breakdown source

Users.Active

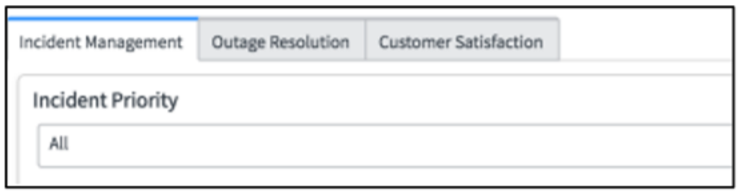
4. Click **Update**.
5. Select **Beth Anglin**, **Fred Luddy**, and **David Loo** in the **Users.Active** breakdown source dropdown.
6. Confirm that all but **Average CSAT Score** widget are filtered.

> ***Note:*** *The Average Score widget is a Score is based on an Indicator with Average aggregation. This type of indicator does not support multi-select.*

D. Dashboard Filtering Using Interactive Filter

Add Interactive Filter

1. Navigate to the **Incident Management** tab.
2. Search for and add the **Incident Priority – Single** Interactive Filter.



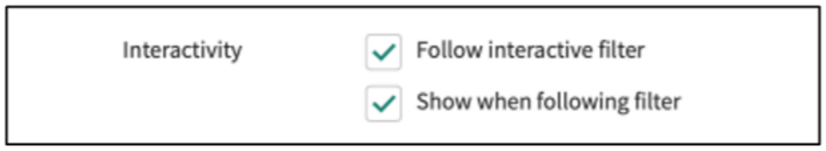
Incident Priority

All

3. Select the **Edit Widget** icon of the **Incidents at Risk** widget.

4. Adjust these **Interactivity** properties:

   - **Follow interactive filter** – checked
   - **Show when following filter** – checked



Interactivity

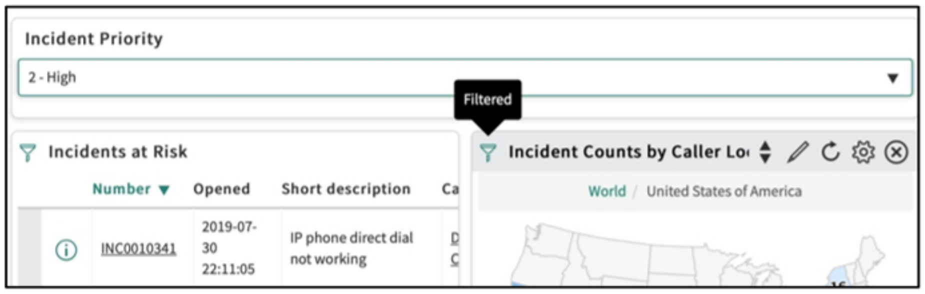
☑ Follow interactive filter

☑ Show when following filter

5. Perform the same configuration on the **Incidents Counts by Caller Location** widget.

6. Set the Incident Priority Interactive Filter to **2 - High.**

7. Confirm that the Incident table reports follow the filter.



Incident Priority

2 - High

Filtered

Incidents at Risk

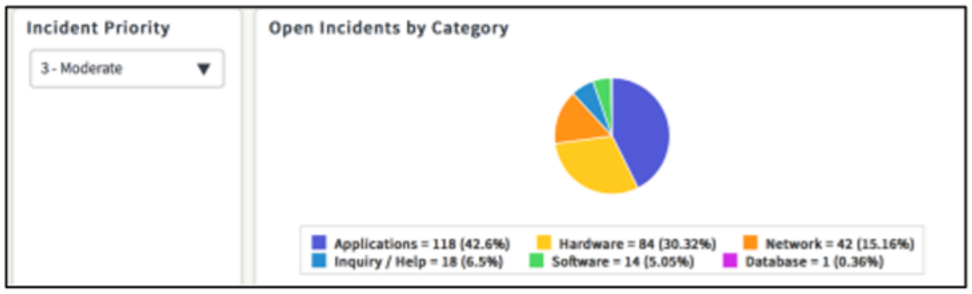
	Number ▼	Opened	Short description	Ca
ⓘ	INC0010341	2019-07-30 22:11:05	IP phone direct dial not working	D C

Incident Counts by Caller Lo

World / United States of America

Report Widget as Interactive Filter

1. Add the **Open Incidents by Category** Report to the **Incident Management** tab:



Incident Priority

3 - Moderate

Open Incidents by Category

2. Select the **Edit Widget** icon of the **Open Incidents by Category** widget.

3. Check the **Act as interactive filter** property.

4. Click the **Software** slice of Incidents in the **Open Incidents by Category** widget.

5. Confirm that the Map and List widget are being filtered accordingly.

6. Remove the **Incident Priority** Interactive Filter and the **Open Incidents by Category** widget from the Incident Management tab.

7. Click **Reset Filter** from the Dashboard menu to reset all Interactive Filters.

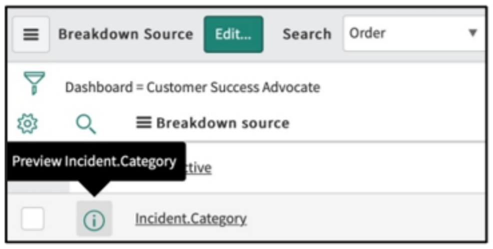
E. Breakdown Source as Interactive Filter

1. Access the **Dashboard Properties** of the Customer Success Advocate Dashboard.

2. Click **Edit…** in the Breakdown Sources Related List.

3. Add **Incident.Category** to the Breakdown Sources List.

4. Open the **Incident.Category** Breakdown Source by clicking the info icon.



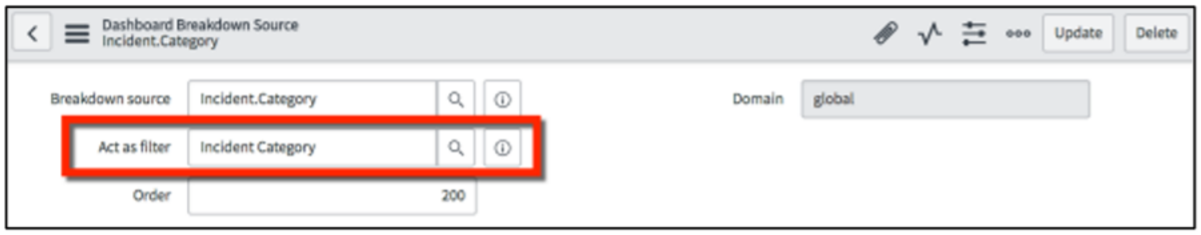
Dashboard = Customer Success Advocate

Breakdown source

…tive

Incident.Category

5. Set **Act as filter** to **Incident Category** (Look up name - **Incident Category – Multiple**).



Dashboard Breakdown Source
Incident.Category

Breakdown source: Incident.Category

Act as filter: Incident Category

Order: 200

Domain: global

6. Click **Update** twice to return to the dashboard.

7. Select the **Incident.Category** Breakdown Source and check the **Software** and **Network** elements.

8. Click **Apply** and validate the following:

   - All reports based on the Incident table in the **Incident Management** are filtered

   - The widgets on the **Customer Satisfaction** tab are filtered

F. Cascading Filter

Review Cascading Filter Configuration

1. Navigate to **Reports > Administration > Interactive Filters**.

2. Open the **Manager – Assignment Group** Cascading filter.

3. Review the configuration of the **Manager – Assignment Group** Cascading filter to become familiar with its filtering capabilities.

Practicing Using Cascading Filter

1. Navigate back to the **Customer Success Advocate** dashboard.

2. Create a new dashboard tab and rename it to **Incident Assignments**.

3. Add the following widgets to the new tab:

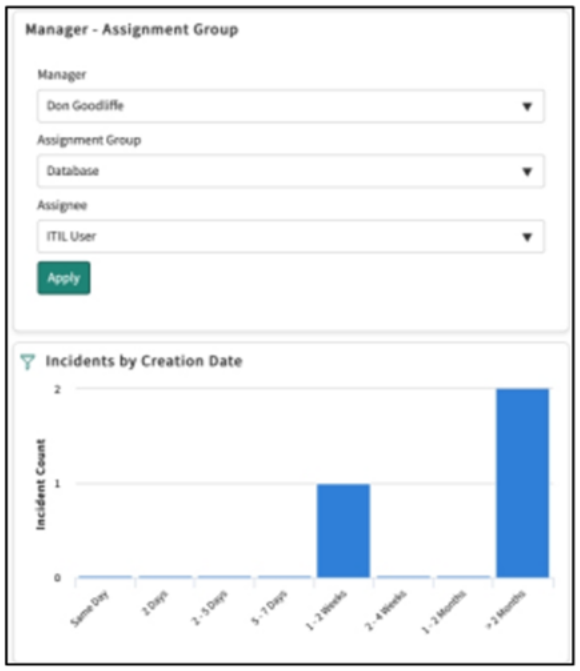
   - Reports Widget – **Incidents by Creation Date**
   - Interactive Filter Widget / Cascading Filter / **Manager – Assignment Group**

4. Configure the **Incidents by Creation Date** report to **Follow the Interactive filter**.

5. **Clear all** Breakdown Source Element filters currently applied to the dashboard.

6. Test the cascading filter by making the following selections:

   - Manager – **Don Goodliffe**
   - Assignment Group – **Database**
   - Assignee – **ITIL User**



Manager - Assignment Group

Manager: Don Goodliffe

Assignment Group: Database

Assignee: ITIL User

7. Confirm that the **Incidents by Creation Date** report displays only Incidents assigned to ITIL User and the Database Assignment Group.

Note: Your report may differ slightly from the screenshot.

You have now completed the Responsive Dashboards lab.

Interactive Analysis

Optional Lab 5.2

⌛10m

Lab Objectives

ITSM Process Managers require the ability to do on-the-fly analysis of current process data. The Interactive Analysis feature provides a deeper understanding of data in an instant dashboard. In addition, it offers easy manipulation of queries, the ability to share and publish, as well as automatic persistence.

This lab will show you how to do the following:

- Launch Interactive Analysis from within a process context.
- Apply Interactive Filters for insightful understanding of analyzed data.

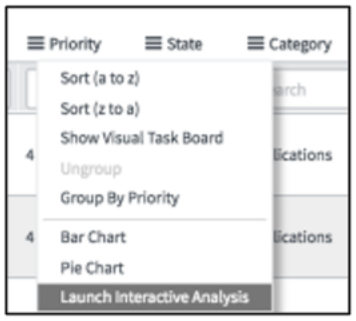
Interactive Analysis

A. Navigation

Viewing Reports and Filters

1. Navigate to **Incident > All**.

2. Right-click the **Priority** field and select **Launch Interactive Analysis**.



3. Adjust the controls at the top of the dashboard as follows:

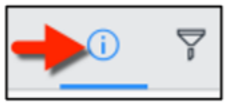
   - Group by: **Priority**
   - Stack by: **State**
   - Aggregation: **Count**

4. If necessary, click the **Filters** button in the banner to open the **Filters** panel.

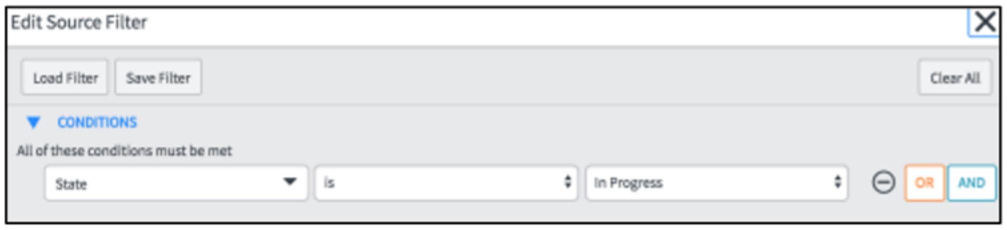
5. Using the **INCIDENT OPENED** filter, view only Incidents opened **Last 30 days**.

6. Using the **INCIDENT PRIORITY** filter, view only **1-Critical** priority incidents.

7. Select **All** in the **INCIDENT OPENED** and **INCIDENT PRIORITY** filter to clear them.

8. Navigate to the **Filters Info** panel. 

9. Click **Edit Source Filter** and make changes to display only Incidents **in Progress**.



Edit Source Filter

Load Filter | Save Filter | Clear All

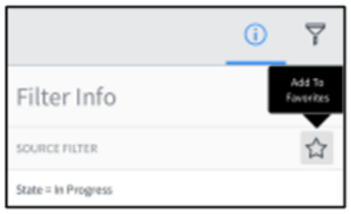
CONDITIONS

All of these conditions must be met

State | is | In Progress | OR | AND

10. Click **Save changes** and note that the Dashboard is refreshed automatically.

11. Edit the **SOURCE FILTER** once again to remove the **State = In Progress** condition and save the changes.

12. Click **Add to Favorites** next to SOURCE FILTER. 

Filter Info

Add To Favorites

SOURCE FILTER

State = In Progress

13. Review the **SHARE** field.

> ***Tip:*** *You can retrieve the Dashboard URL for Sharing purposes from the SHARE* field.

B. Interactive Filters

Change Type Interactive Filter

1. Navigate to the **Change > All**.

2. Click the **Type** field and select **Launch Interactive Analysis** from the context menu.

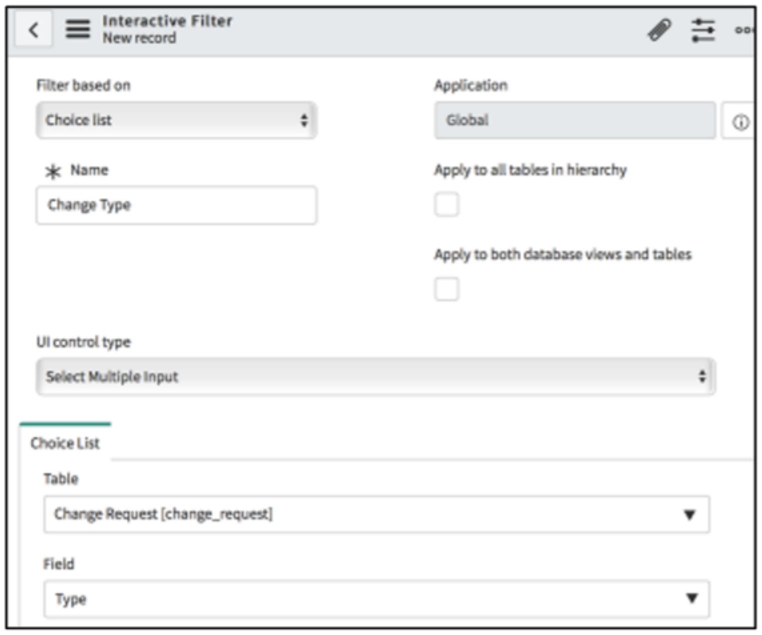
3. Practice applying a non-Count aggregation such as **Average Impact**, **Risk** or **Priority**.

***Note**: There are no Interactive Filters currently defined for the Change Interactive Analysis.*

4. Navigate to **Reports > Administration > Interactive Filters**.

5. Click **New** to create a new Interactive Filter as follows:

   - Filter based on: **Choice List**
   - Name: **Change Type**
   - UI Control Type: **Select Multiple Input**
   - Table: **Change_Request [change_request]**
   - Field: **Type**


Interactive Filter
New record

Filter based on: Choice list

Application: Global

* Name: Change Type

Apply to all tables in hierarchy □

Apply to both database views and tables □

UI control type: Select Multiple Input

Choice List

Table: Change Request [change_request]

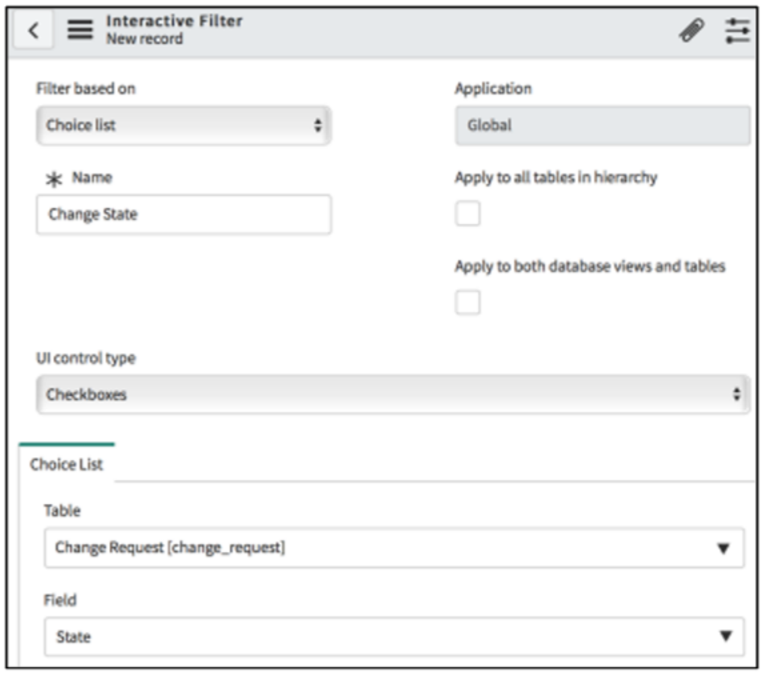
Field: Type

6. Click **Submit**.

Change State Interactive Filters

1. Create a new Interactive Filter as follows:

   - Filter based on: **Choice List**
   - Name: **Change State**
   - UI Control Type: **Checkboxes**
   - Table: **Change_Request [change_request]**
   - Field: **State**


Interactive Filter
New record

Filter based on: Choice list

Application: Global

* Name: Change State

Apply to all tables in hierarchy □

Apply to both database views and tables □

UI control type: Checkboxes

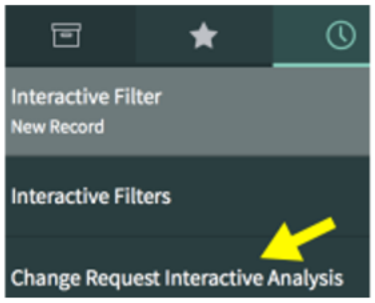
Choice List

Table: Change Request [change_request]

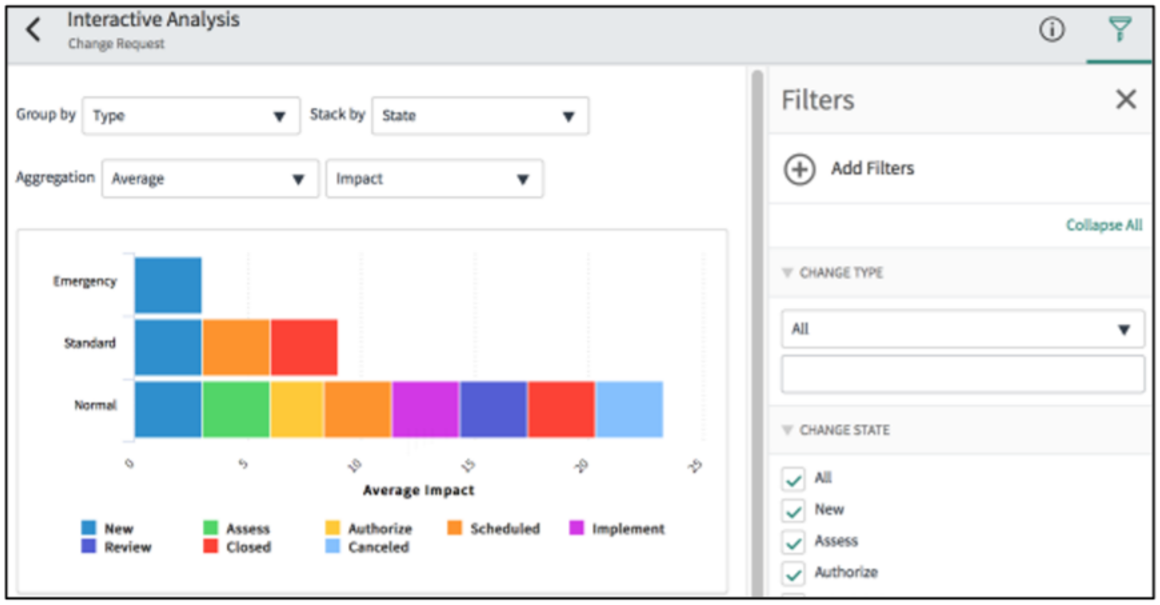
Field: State

2. Click **Submit**.

3. Access the **Change Interactive Dashboard** using the link in your history.

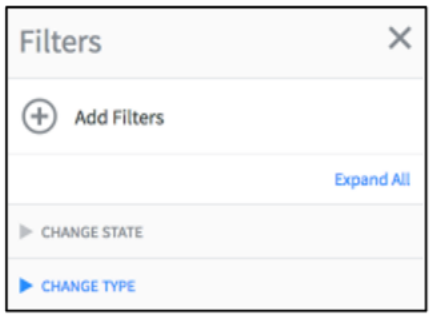


4. Confirm that the two new Interactive Filters are now visible.

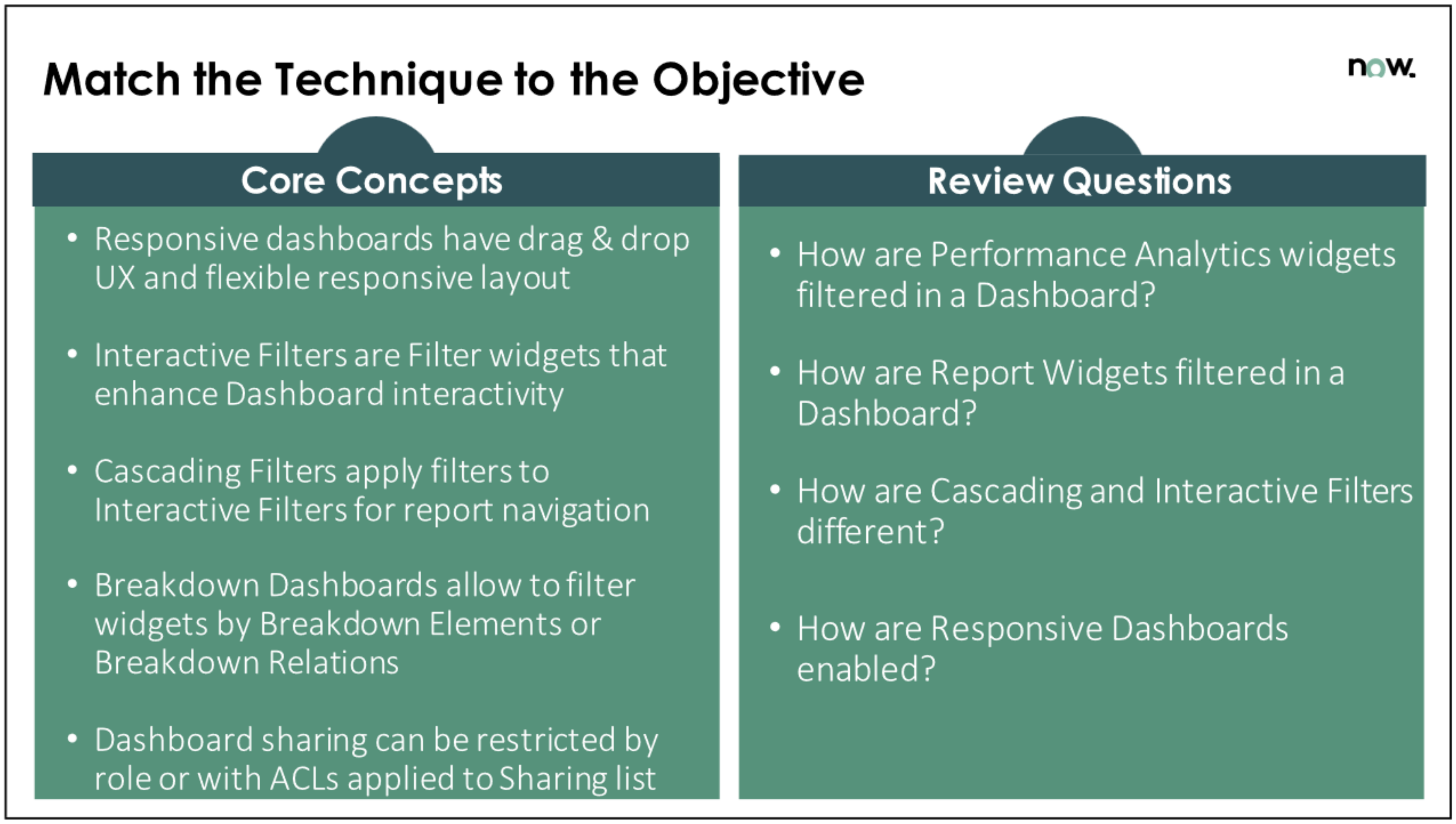

Interactive Analysis
Change Request

Group by: Type
Stack by: State
Aggregation: Average
Impact

***Troubleshooting Tip**: If one or both the **Change State** and **Change Type** Interactive Filters are not on the dashboard, manually add them using the **Add Filters** button.*



You have now completed the Interactive Analysis lab.



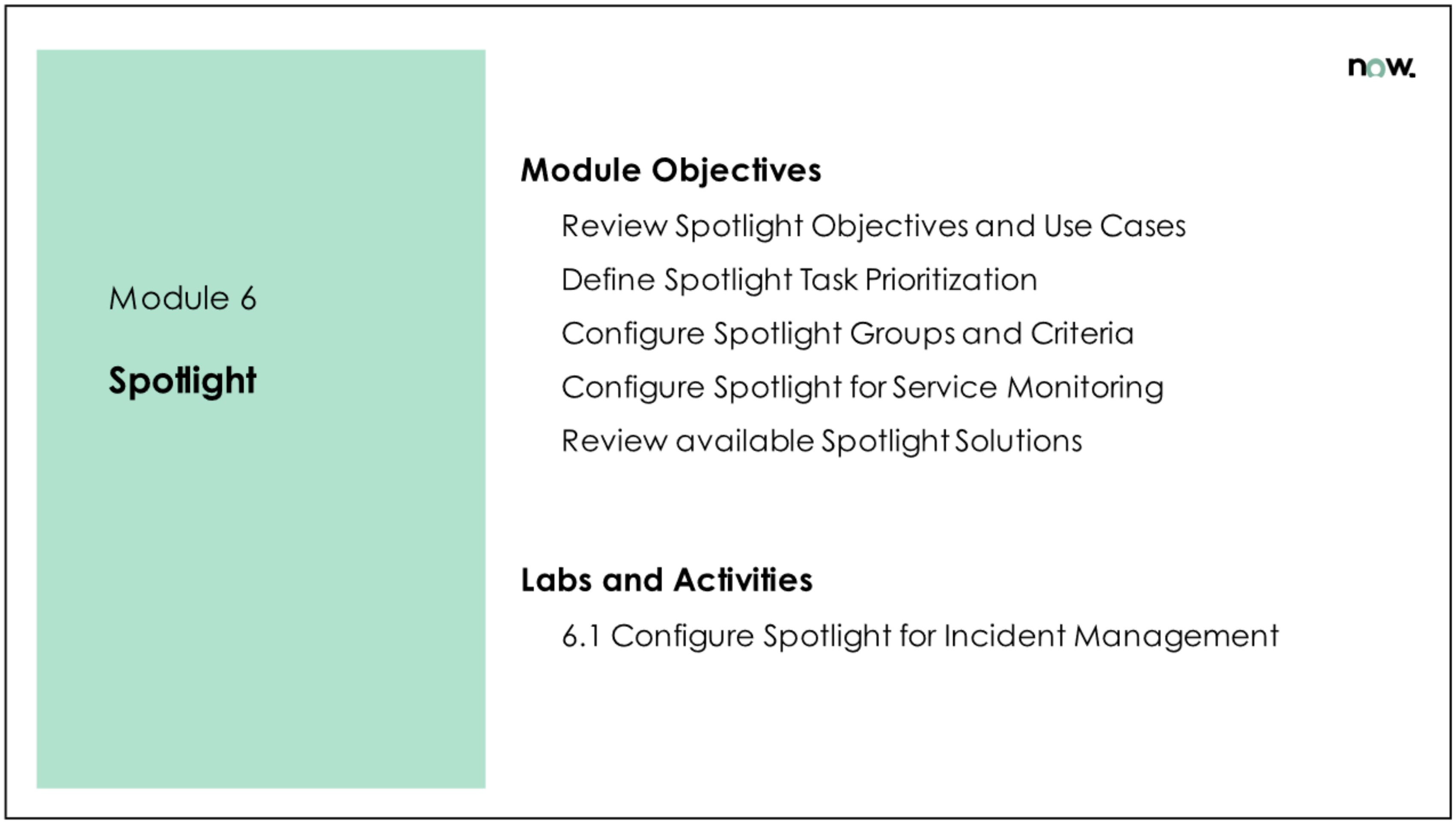
Match the Technique to the Objective

Core Concepts

- Responsive dashboards have drag & drop UX and flexible responsive layout
- Interactive Filters are Filter widgets that enhance Dashboard interactivity
- Cascading Filters apply filters to Interactive Filters for report navigation
- Breakdown Dashboards allow to filter widgets by Breakdown Elements or Breakdown Relations
- Dashboard sharing can be restricted by role or with ACLs applied to Sharing list

Review Questions

- How are Performance Analytics widgets filtered in a Dashboard?
- How are Report Widgets filtered in a Dashboard?
- How are Cascading and Interactive Filters different?
- How are Responsive Dashboards enabled?



now.

Module 6

Spotlight

Module Objectives

Review Spotlight Objectives and Use Cases

Define Spotlight Task Prioritization

Configure Spotlight Groups and Criteria

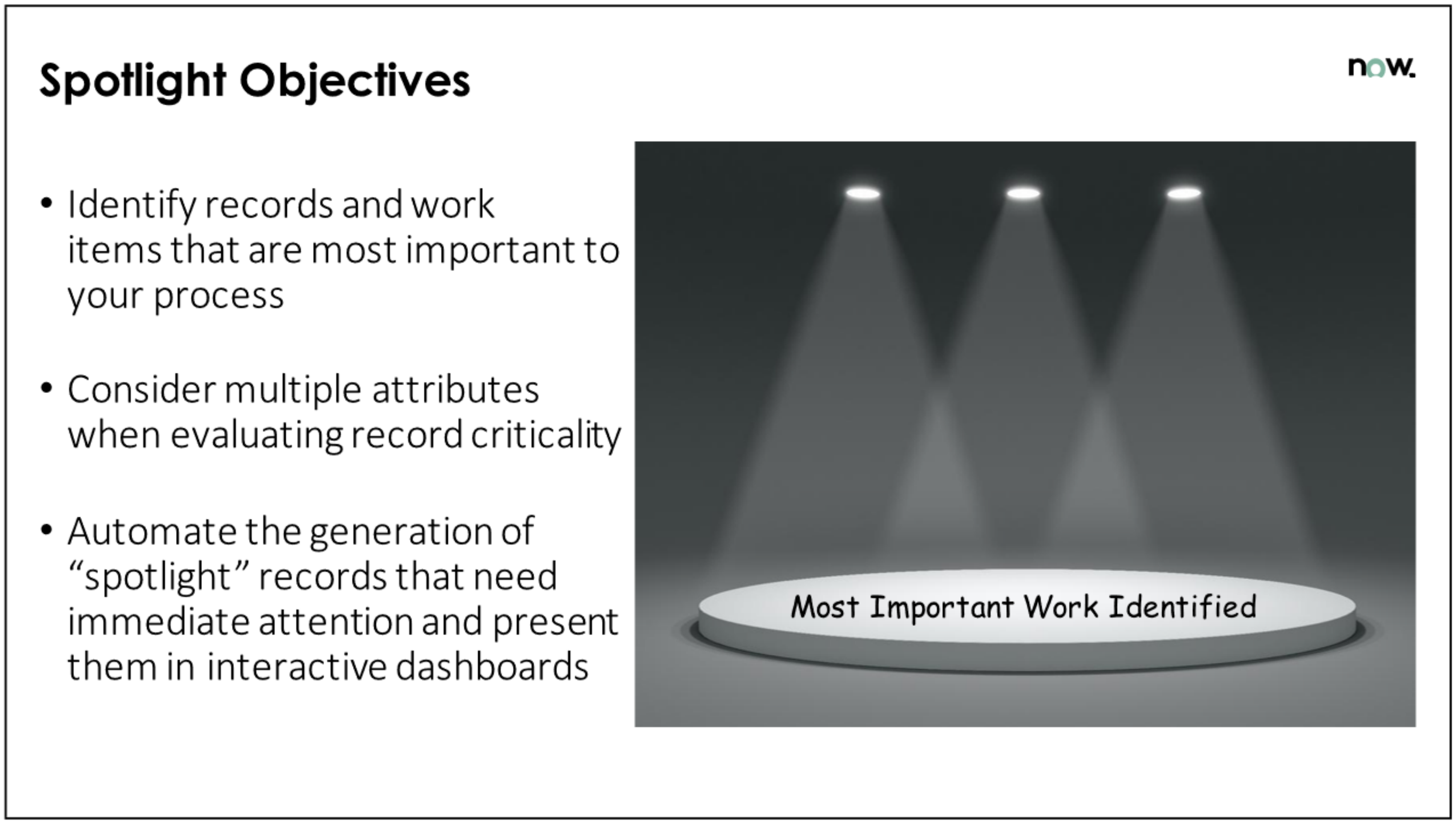
Configure Spotlight for Service Monitoring

Review available Spotlight Solutions

Labs and Activities

6.1 Configure Spotlight for Incident Management

This module introduces the Spotlight application – an advanced analytics feature that combines reporting and performance analytics techniques to uncover the most relevant, pressing, and business critical tasks impacting your processes and services.

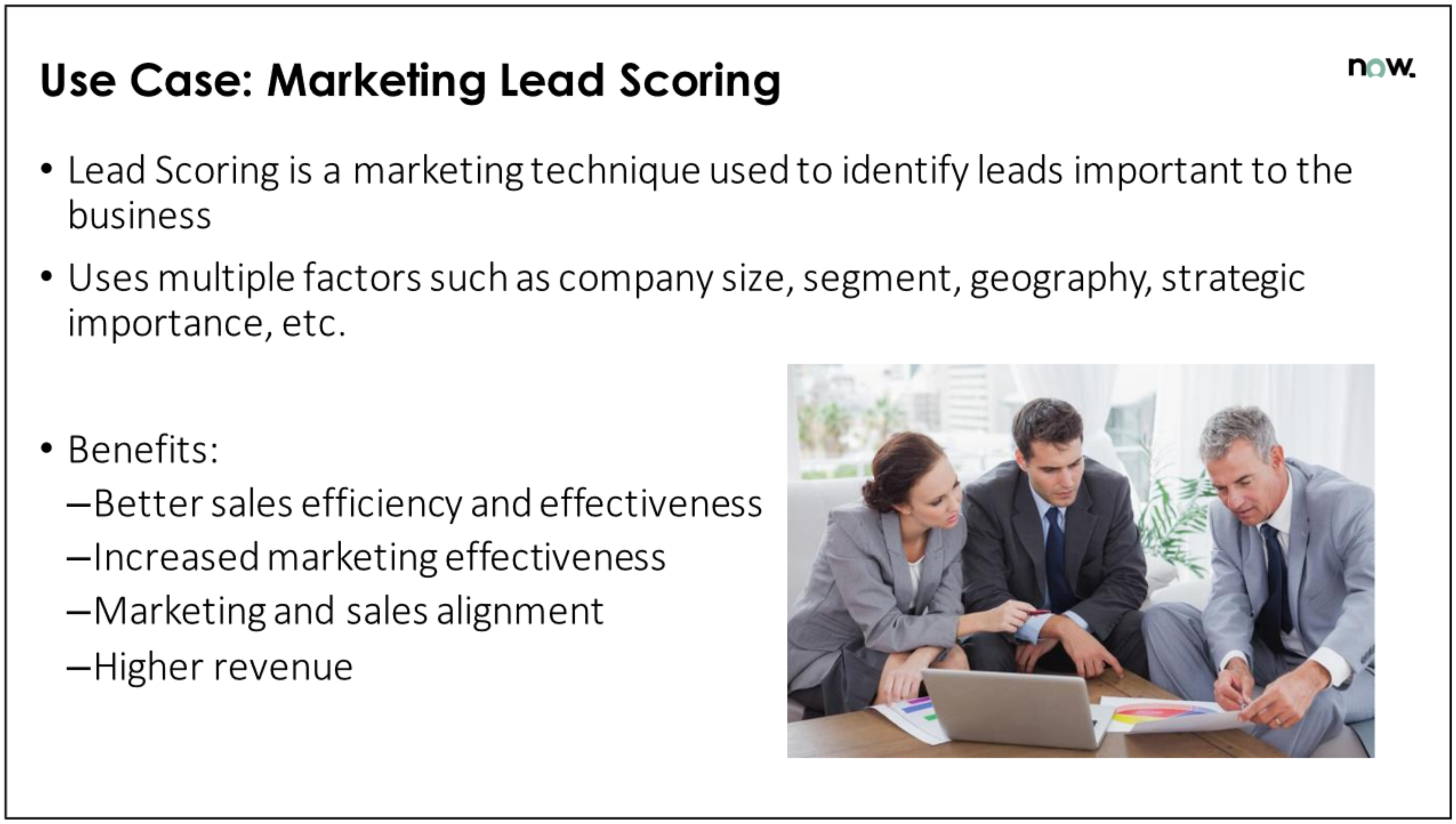


Spotlight Objectives

- Identify records and work items that are most important to your process
- Consider multiple attributes when evaluating record criticality
- Automate the generation of "spotlight" records that need immediate attention and present them in interactive dashboards

The Spotlight can be used in the following situations:

- You need to evaluate and identify a subset of table records based on a dynamic and/or complex selection criteria. And you need to do that on a regular or scheduled basis.
- You need to rank work requests or items based on multiple attributes such as priority, urgency, escalation, impact, etc.
- You need to make decisions on current work based on past performance and trends. With Spotlight, the composite criteria can query both current process data as well as historical information collected by Performance Analytics.



Use Case: Marketing Lead Scoring

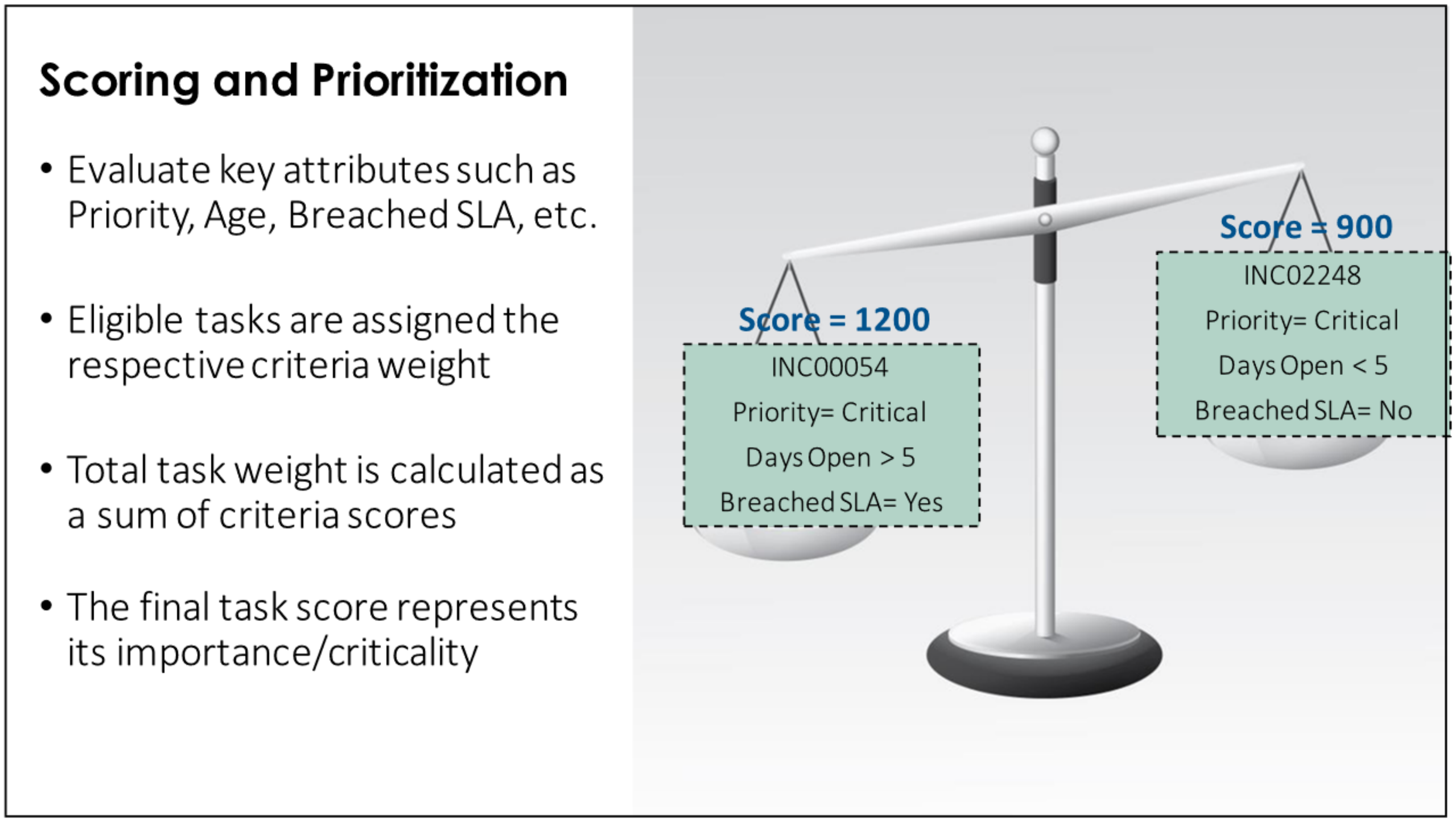
- Lead Scoring is a marketing technique used to identify leads important to the business
- Uses multiple factors such as company size, segment, geography, strategic importance, etc.
- Benefits:
  - Better sales efficiency and effectiveness
  - Increased marketing effectiveness
  - Marketing and sales alignment
  - Higher revenue

Lead scoring is a methodology used to rank prospects against a scale that represents the perceived value each lead represents to the organization. Lead Scoring allows a business to customize a prospect's experience based on his or her buying stage and interest level and greatly improves the quality and "readiness" of leads that are delivered to sales organizations for a follow-up.

The Marketing lead scoring presented here enables marketing team to prioritize leads, send only sales-ready leads to the sales team, nurture the not-yet-sales ready leads until they become qualified and prevents costly lead leakage

In addition to the marketing use case, Spotlight can be used for:

- Triaging Incidents
- Demand and Project Scoring
- Request Ranking
- Sales Lead Scoring



Scoring and Prioritization

- Evaluate key attributes such as Priority, Age, Breached SLA, etc.
- Eligible tasks are assigned the respective criteria weight
- Total task weight is calculated as a sum of criteria scores
- The final task score represents its importance/criticality

Score = 1200

INC00054

Priority= Critical

Days Open > 5

Breached SLA= Yes

Score = 900

INC02248

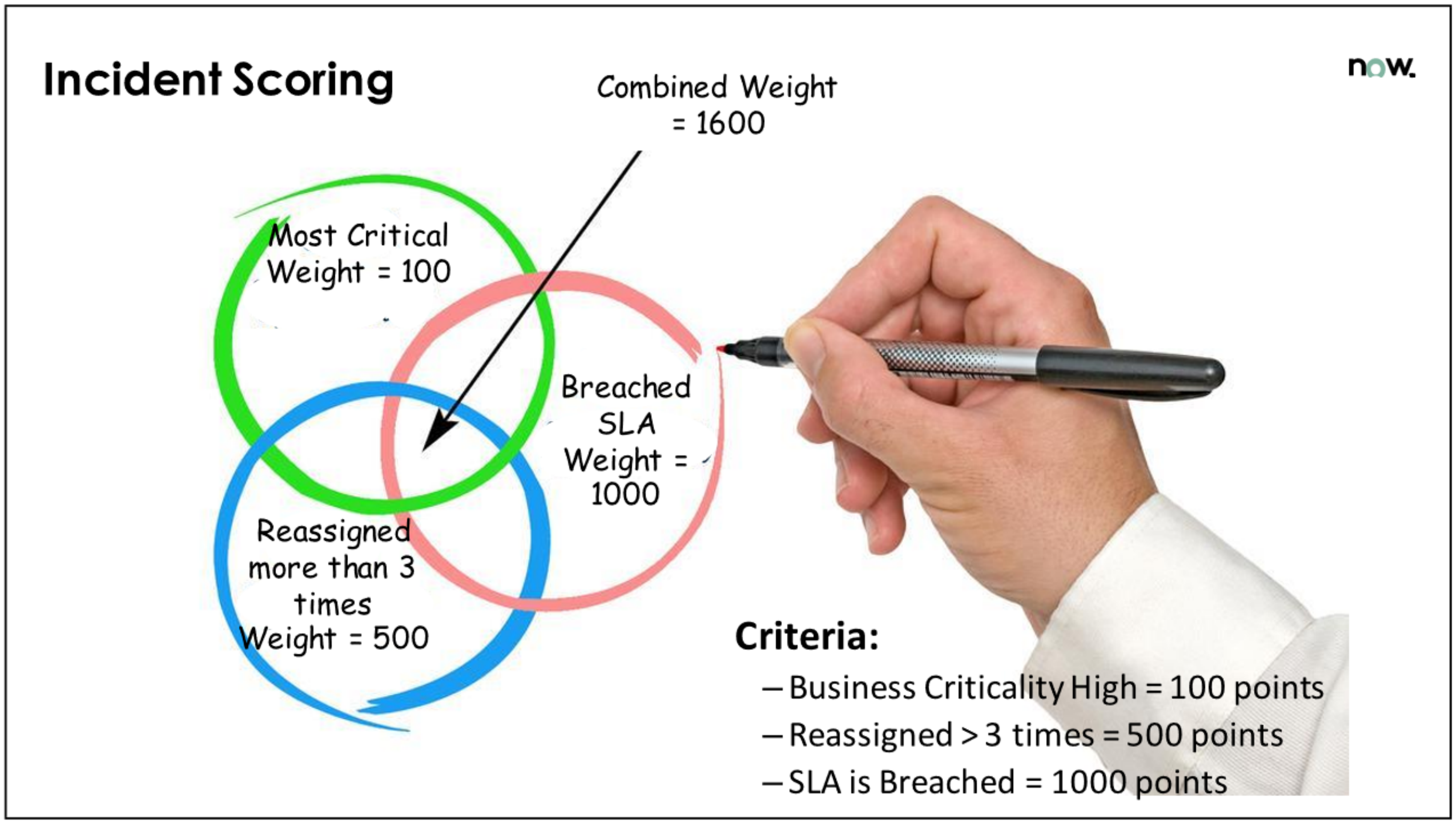
Priority= Critical

Days Open < 5

Breached SLA= No

To identify the most important tasks, you typically prioritize Critical incidents as more important than High Priority incidents. However, if a High priority incident has been open for a long time, breached an SLA, or has been reassigned multiple times, you might consider it more pressing to address than another Critical priority incident.

Spotlight allows you to do this sort of multi-part prioritization by building composite weighted criteria. You may decide to create a composite criteria using the Priority, Open Days, Reassignments, and SLA status properties of an Incident. Each of these criteria parts carries a weight which is used to calculate the total weight or score for the Incident.



Incident Scoring

now.
Combined Weight = 1600
Most Critical Weight = 100
Breached SLA Weight = 1000
Reassigned more than 3 times Weight = 500

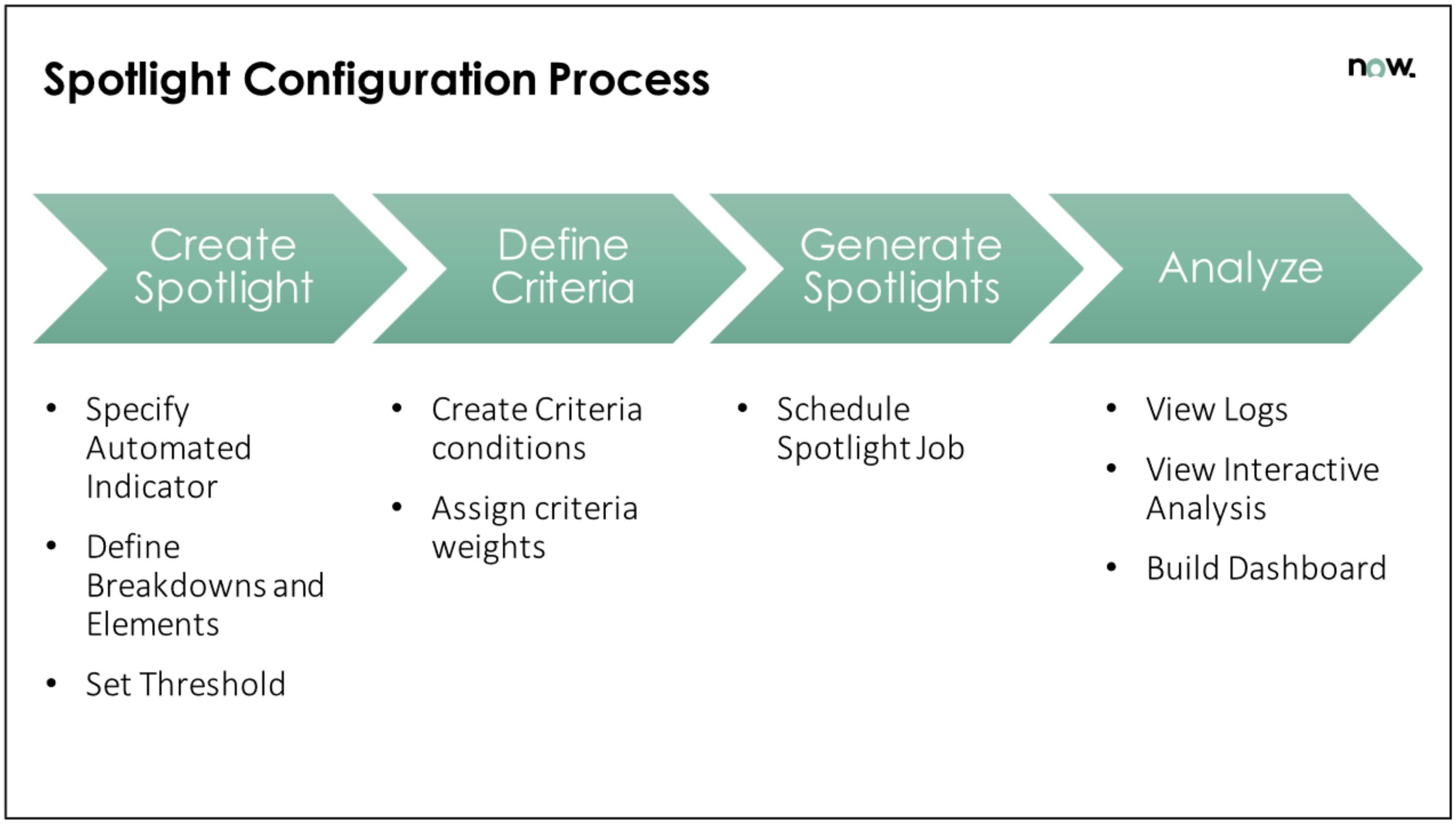
Criteria:

- Business Criticality High = 100 points
- Reassigned > 3 times = 500 points
- SLA is Breached = 1000 points

Spotlight works with the data associated with an Automated Performance Analytics Indicator. In the above example, Spotlight is configured for the Number of Open Incidents indicator.

The scoring process requires a set of criteria to apply to each task in the snapshot. The example criteria applies the following conditions: Business Criticality = Most Critical, SLA is Breached, and Reassignment Count > 3.

If the record meets the criteria, it gets assigned a criteria weight. The total score is the sum of the weight of all criteria that apply to that record.



Spotlight Configuration Process

now.

Create Spotlight

- Specify Automated Indicator
- Define Breakdowns and Elements
- Set Threshold

Define Criteria

- Create Criteria conditions
- Assign criteria weights

Generate Spotlights

- Schedule Spotlight Job

Analyze

- View Logs
- View Interactive Analysis
- Build Dashboard

Before setting up Spotlight, ensure that these tasks have been completed:

- Activate Performance Analytics and configure relevant process Indicators
- Automated data collection for relevant Indicators
- Activate the Spotlight Base Plugin
- Activate any applicable Solution Plugins

Note that the Premium version of Performance Analytics is required if Spotlight is performed for anything other than the Incident table.

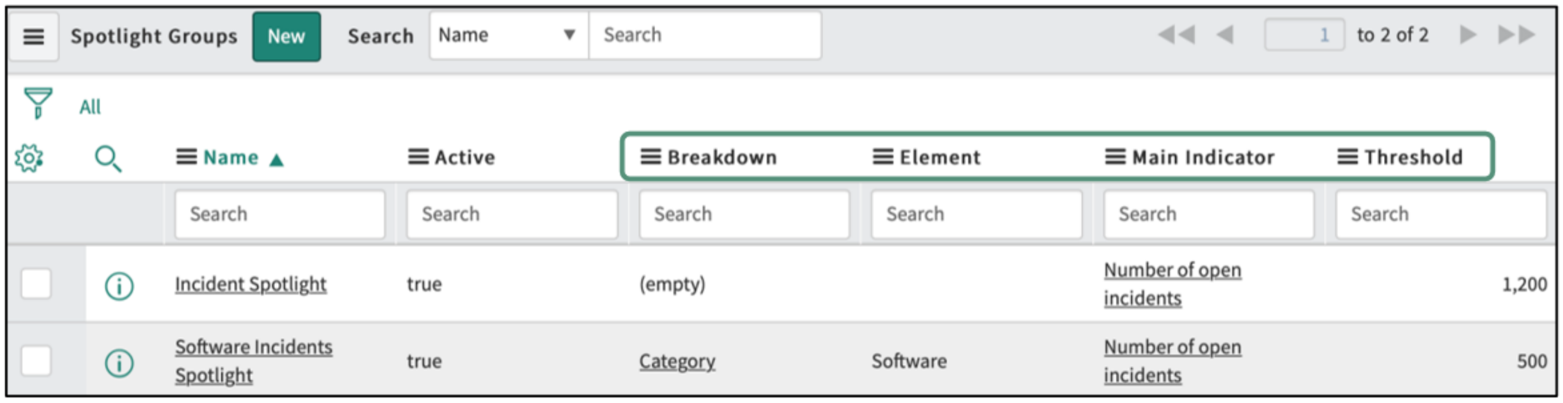
Spotlight Group

Purpose: Define the input data set to be used by Spotlight

MAIN INDICATOR
Defines the data source

BREAKDOWN + ELEMENT
Refines the base data source

THRESHOLD
Triggers creation of Spotlights for records above this score



Name ▲	Active	Breakdown	Element	Main Indicator	Threshold
Incident Spotlight	true	(empty)		Number of open incidents	1,200
Software Incidents Spotlight	true	Category	Software	Number of open incidents	500

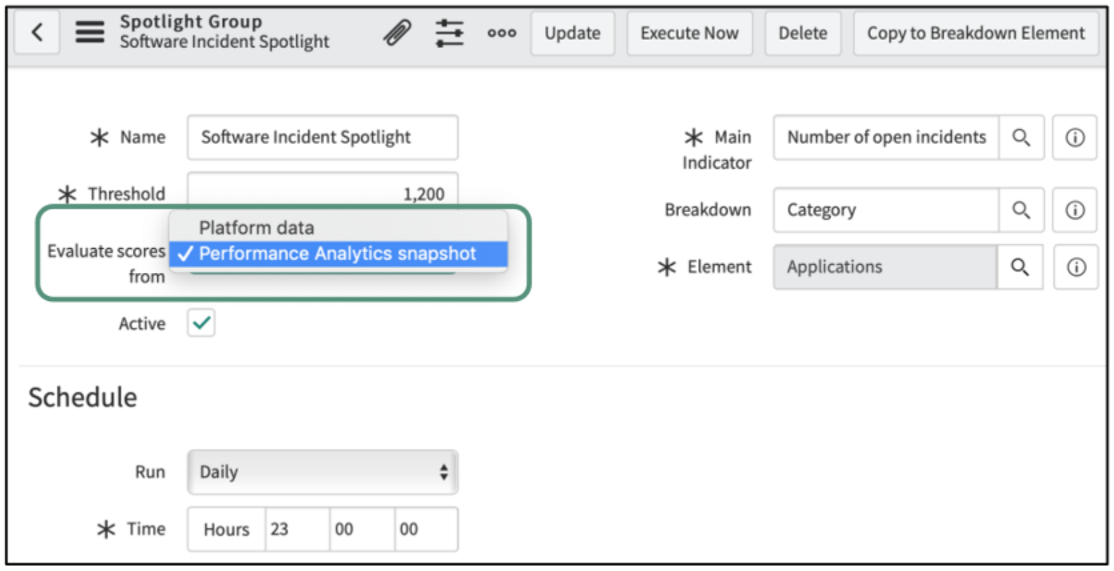
The Spotlight group defines the data to evaluate and the weight threshold needed to create a spotlight. If the score of a record in the data set exceeds the threshold, a spotlight is created for that record. The score of a record is the total weight from all applicable spotlight criteria in the spotlight group.

- **Main Indicator**: the data set to evaluate is defined in a Performance Analytics indicator
- **Breakdown** and **Element** can optionally be defined
- **Threshold**: Any record scoring above at or above this threshold will be included in the Spotlight

Snapshot vs Platform Data

Performance Analytics Snapshot = most recently collected data for main indicator

Platform Data = business table data



* Name	Software Incident Spotlight	* Main Indicator	Number of open incidents
* Threshold	1,200	Breakdown	Category
Evaluate scores from	Platform data ✓ Performance Analytics snapshot	* Element	Applications
Active	✓		

Schedule

Run	Daily
* Time	Hours 23 00 00

The **Evaluate scores from** property determines the source of data to use for generating Snapshots.

By default, a new Spotlight group uses a snapshot of records. The Performance Analytics data collection jobs for the main indicator and the criteria indicators of the Spotlight group create the snapshots. For a Spotlight group to use snapshot records, the main indicator and all the indicators used in the criteria of the Spotlight group have the **Collect Records** option on

Note: Only indicator-based Spotlight criteria evaluate snapshot records.

Query-based criteria always evaluate **platform data**. Spotlight can collect and evaluate records directly from the platform at the time that the Spotlight job runs. The resulting Spotlight scores reflect the state of the platform at the time of the latest Spotlight job.

Note: Evaluating platform data requires more system resources to query the indicator data than evaluating records from a snapshot does.

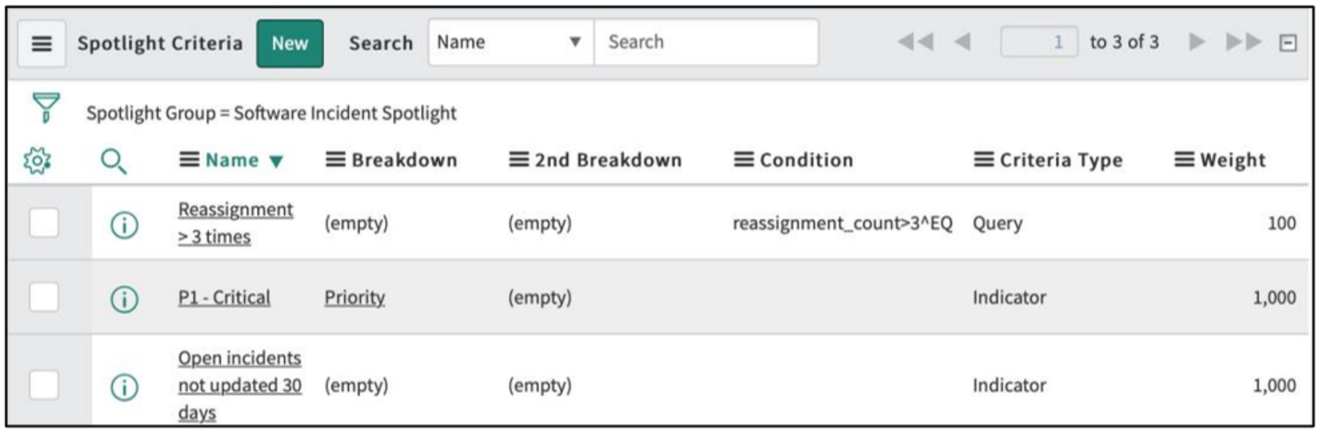
Limitations: Indicator-based criteria *cannot* be used to evaluate platform data when scripted breakdowns are applied.

Spotlight Criteria

- Spotlight scoring logic comprised of multiple conditions
- Every criteria condition has Weight
- Weight is assigned to records that satisfy the criteria

Example:
A Critical Incident not updated in last 30 days and Reassigned 4 times has score =

1000 + 1000 + 100 = 2100



Spotlight Group = Software Incident Spotlight

Name	Breakdown	2nd Breakdown	Condition	Criteria Type	Weight
Reassignment > 3 times	(empty)	(empty)	reassignment_count>3^EQ	Query	100
P1 - Critical	Priority	(empty)		Indicator	1,000
Open incidents not updated 30 days	(empty)	(empty)		Indicator	1,000

The Spotlight criteria defines how to score records based on specific attributes. For example, different criteria weights allow you to assign higher weight to tasks with higher Urgency, and, lower weight to tasks with lower Risk or Impact.

The individual record score is the sum of weights of all applicable criteria conditions in the spotlight group. Using the above definition, a Critical Incident (weight 1000) not updated in 30 days (weight 1000) and reassigned 4 times (weight 100) has a score of 1000 + 1000 + 100 = 2100 as per the defined weights and Criteria.

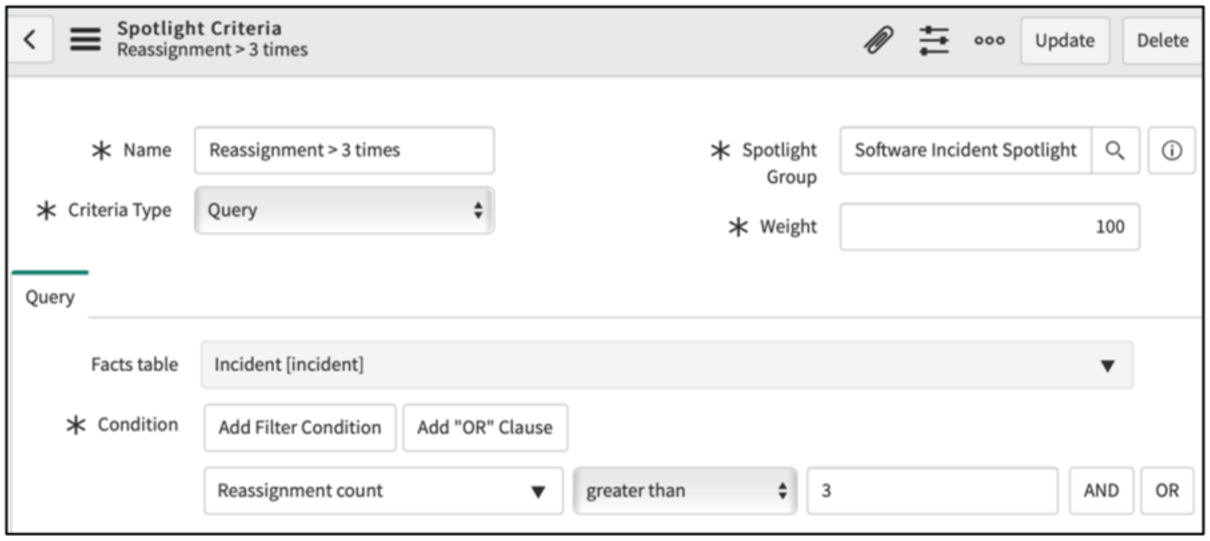
The record score is compared to the Threshold and only records scoring over the Threshold are issued a corresponding Spotlight.

Spotlight Criteria: Query

Criteria Type: QUERY

WEIGHT: Assign this score to all records returned by the Condition

QUERY: Which records will be assigned the weight



* Name: Reassignment > 3 times

* Criteria Type: Query

* Spotlight Group: Software Incident Spotlight

* Weight: 100

Query

Facts table: Incident [incident]

* Condition: Reassignment count | greater than | 3

Example: All Incidents in the data set with Reassignment count > 3 will be assigned score of 100

The Spotlight criteria can either be of type Query or of type Indicator. If you select the **Query** type, you need to specify the **Facts** table and the **Condition** that specifies which records to weight. For consistency and ease of use, the table is already populated with the Facts table of the Main Indicator defined in the Spotlight group.

In the example above, the **Reassignment > 3 times** criteria is going to assign 100 to every incident in the record set which has Reassignment count greater than 3.

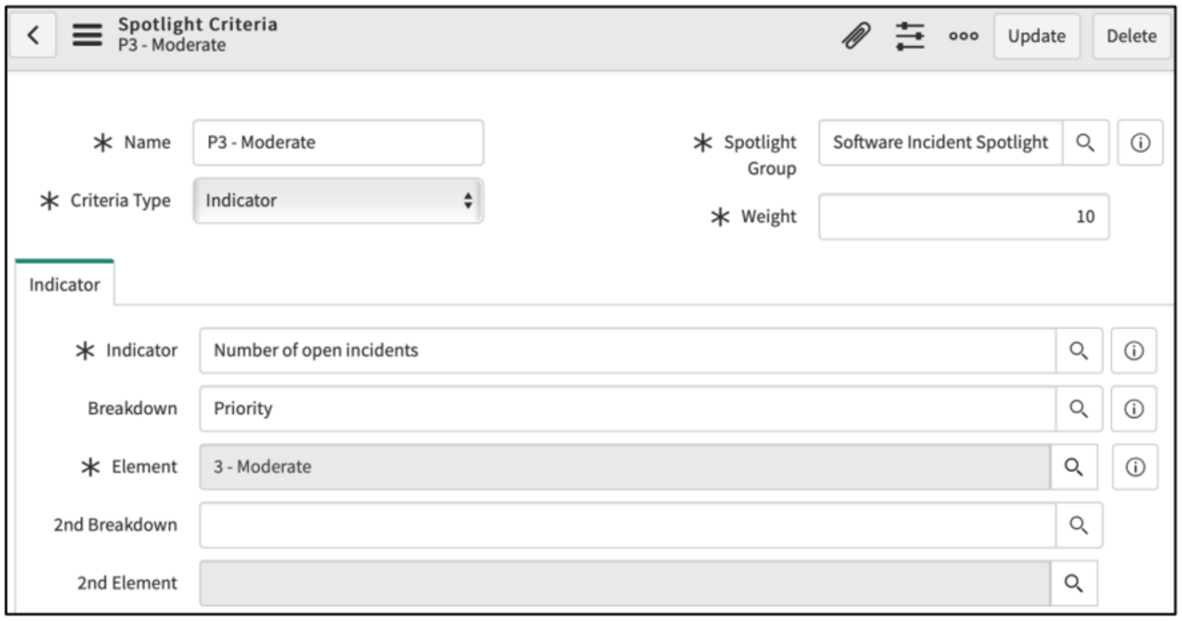
Note: Query-based criteria always evaluate platform data, even if you set the group to evaluate data from a snapshot.

Spotlight Criteria: Indicator

Criteria Type: INDICATOR

WEIGHT: Assign this score to all records returned in the snapshot

1st and 2nd Level Breakdowns & Elements: Refine Data Set



✱ Name	P3 - Moderate	✱ Spotlight Group	Software Incident Spotlight
✱ Criteria Type	Indicator	✱ Weight	10

Indicator

✱ Indicator	Number of open incidents
Breakdown	Priority
✱ Element	3 - Moderate
2nd Breakdown	
2nd Element	

Example: All Moderate Priority Incidents in the Number of Open Incidents snapshot will be assigned a score of 10

Select the **Indicator** Criteria type to assign the weight to all records included in that indicator. Optionally select one or two breakdowns and elements to limit the included records to only those with the specified breakdown element values.

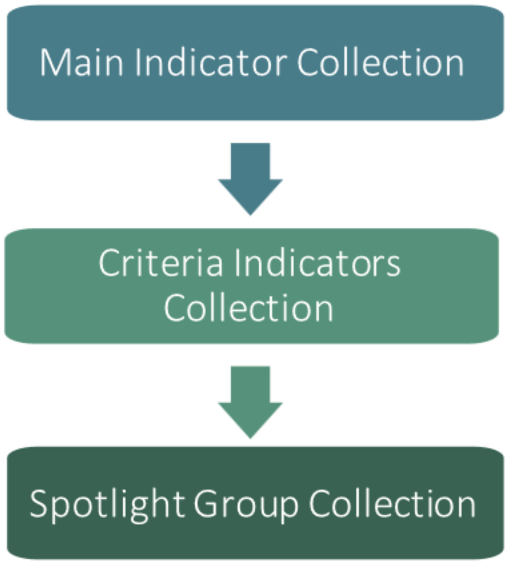
You can only select an Indicator that is sourced from the same Facts table as the Main Indicator of the Spotlight group.

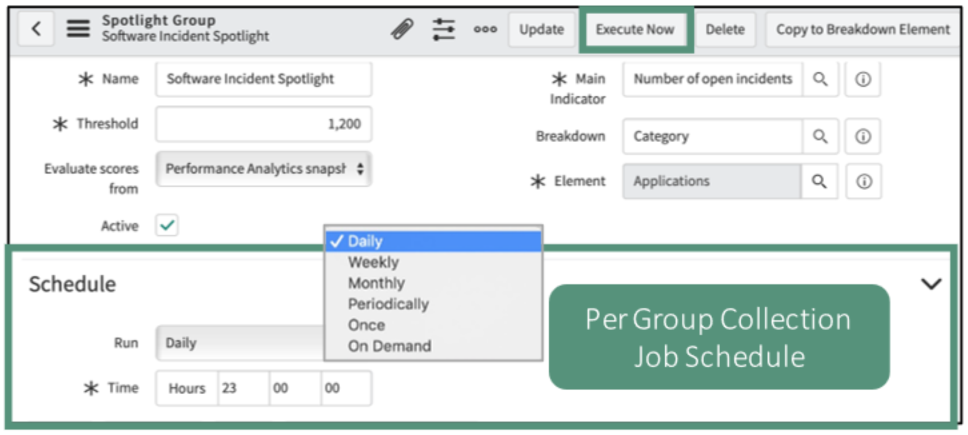
Note: Only indicator-based Spotlight criteria evaluate snapshot records.

Indicator-based criteria cannot be used to evaluate platform data when scripted breakdowns are applied. The specific restrictions are:

- If the indicator of a criterion uses a breakdown that is based on a script, this specific criterion cannot be used. Either collect snapshot instead of platform data or create query-based criteria in place of indicator-based criteria.
- If the main indicator of the Spotlight group uses a breakdown that is based on a script, you cannot evaluate platform data. Configure the Spotlight group to collect data based on snapshots only.

Collecting Spotlight Data


Main Indicator Collection
Criteria Indicators Collection
Spotlight Group Collection



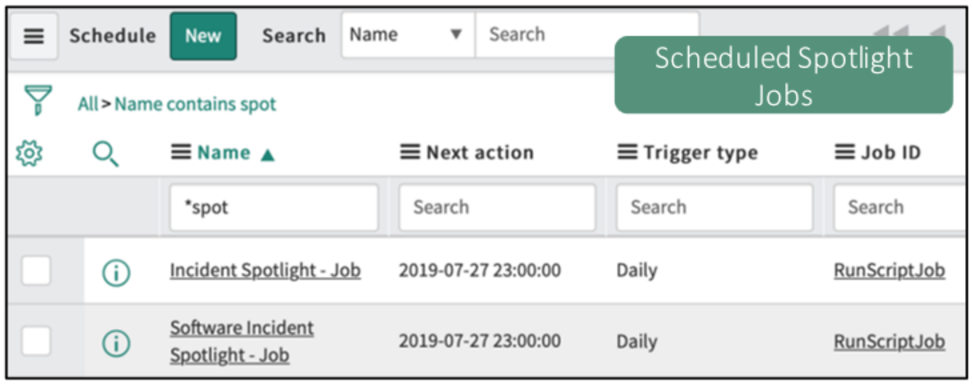
* Name	Software Incident Spotlight	* Main Indicator	Number of open incidents
* Threshold	1,200	Breakdown	Category
Evaluate scores from	Performance Analytics snapsh	* Element	Applications
Active	✓		

Schedule

Run: Daily

* Time: Hours 23 00 00

Per Group Collection Job Schedule



All > Name contains spot

Name ▲	Next action	Trigger type	Job ID
*spot	Search	Search	Search
Incident Spotlight - Job	2019-07-27 23:00:00	Daily	RunScriptJob
Software Incident Spotlight - Job	2019-07-27 23:00:00	Daily	RunScriptJob

Scheduled Spotlight Jobs

Each Spotlight group has a corresponding Collection job. To immediately collect scores for a Spotlight group, go to the Spotlight Group and click **Execute Now**.

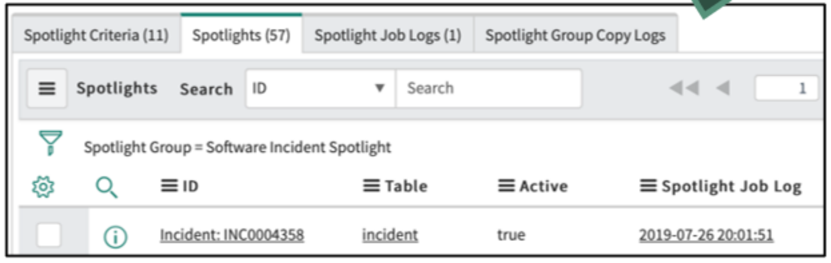
To schedule a Spotlight, check the **Active** property and select a **Run** and **Time** option. Available Run options are: **Daily**, **Weekly**, **Periodically, Once,** and **On Demand**. The Spotlight collection job can be viewed by navigating to **System Scheduler > Scheduled Jobs** and searching for the corresponding Spotlight group name.
Once the Spotlight collection runs, the **Spotlights Job Logs** Related List contains the Spotlight Job Log status record.

Below are some best practices for scheduling the Spotlight collection. Scheduling Spotlight score collection in this way ensures that the results are up-to-date and meaningful.

- The main indicator and all the indicators used in the criteria of the Spotlight group should have the same data collection frequency.
- The data collection jobs for all the Spotlight group indicators run as closely together as possible, to keep the data synchronized. The snapshots of the criteria indicators must have the same date as the last score date of the main indicator. Ideally, the main indicator and all the criteria indicators are in the same data collection job.
- The Spotlight score collection job should runs *at the same frequency* as the data collection jobs, and *as soon as possible after those jobs complete*.

Spotlight Audits

- Spotlight Audits are generated for each Spotlight record
- Only the latest spotlights are retained in the Spotlights table

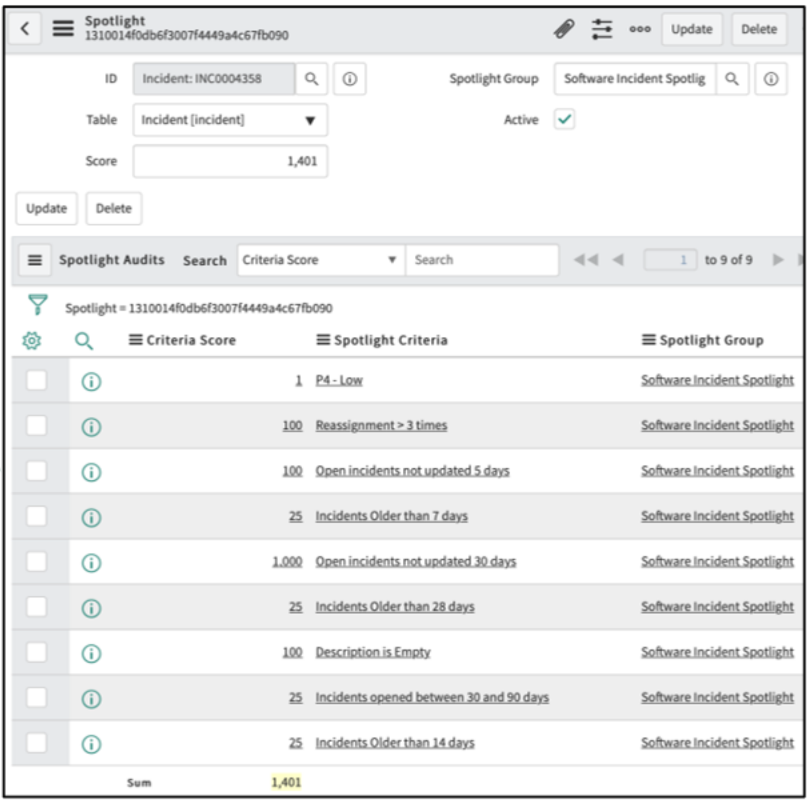


Search: ID

Spotlight Group = Software Incident Spotlight

ID	Table	Active	Spotlight Job Log
Incident: INC0004358	incident	true	2019-07-26 20:01:51

Spotlights per latest Spotlight Group collection



ID: Incident: INC0004358

Spotlight Group: Software Incident Spotlig

Table: Incident [incident]

Active: ✓

Score: 1,401

Search: Criteria Score

Spotlight = 1310014f0db6f3007f4449a4c67fb090

Criteria Score	Spotlight Criteria	Spotlight Group
1	P4 - Low	Software Incident Spotlight
100	Reassignment > 3 times	Software Incident Spotlight
100	Open incidents not updated 5 days	Software Incident Spotlight
25	Incidents Older than 7 days	Software Incident Spotlight
1,000	Open incidents not updated 30 days	Software Incident Spotlight
25	Incidents Older than 28 days	Software Incident Spotlight
100	Description is Empty	Software Incident Spotlight
25	Incidents opened between 30 and 90 days	Software Incident Spotlight
25	Incidents Older than 14 days	Software Incident Spotlight
Sum 1,401		

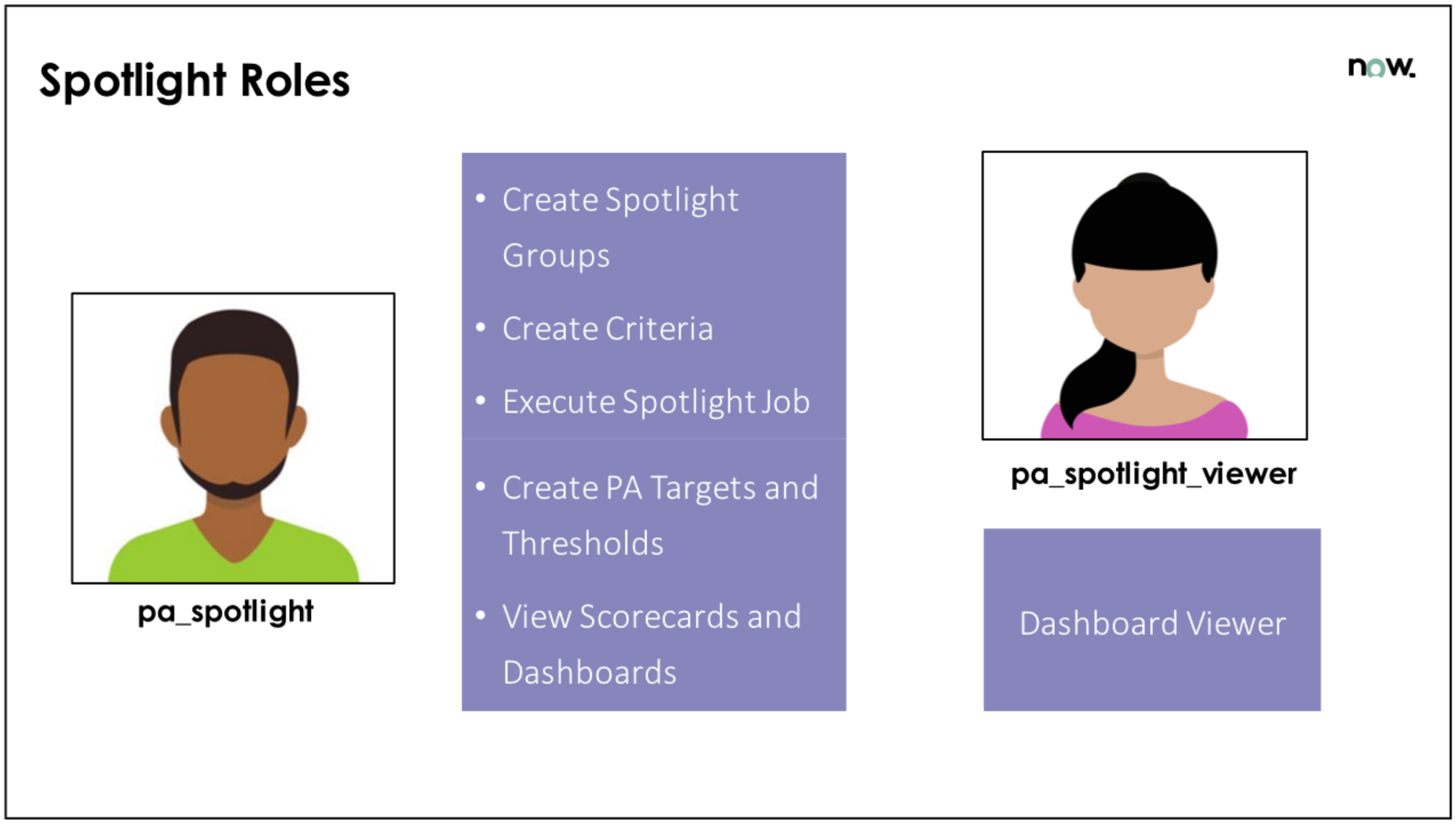
Spotlight Audits = Score Calculation Details

After running the Spotlight Group collection, all records in the data set are assigned a score. Whenever the total weight of a record exceeds the Spotlight group Threshold, a new record is created in the Spotlight [spotlight] table.

The original record's sys_id is stored in the **ID** field and the total weight for that record is in the **Score** field of the Spotlight record. All Spotlights as per the latest Spotlight collection (the records which exceed the threshold) are visible in the **Spotlights** Related List.

Spotlight collection generates Spotlights in the Spotlight table and deletes previous Spotlights. Corresponding Spotlight audit records are created in the [spotlight_audit] table representing per-criteria scores. Spotlight Audits are visible in the Criteria's Spotlight Audits related list.

To summarize, the tables Spotlight **[spotlight]** and Spotlight Audits **[spotlight_audit]** are used to store Spotlight scores. The Spotlight table contains only the most recent Spotlight Group records as per the latest execution of the Spotlight job. The Spotlight Audits table contains a history of all criteria scores.



Spotlight Roles

pa_spotlight

- Create Spotlight Groups
- Create Criteria
- Execute Spotlight Job

- Create PA Targets and Thresholds
- View Scorecards and Dashboards

pa_spotlight_viewer

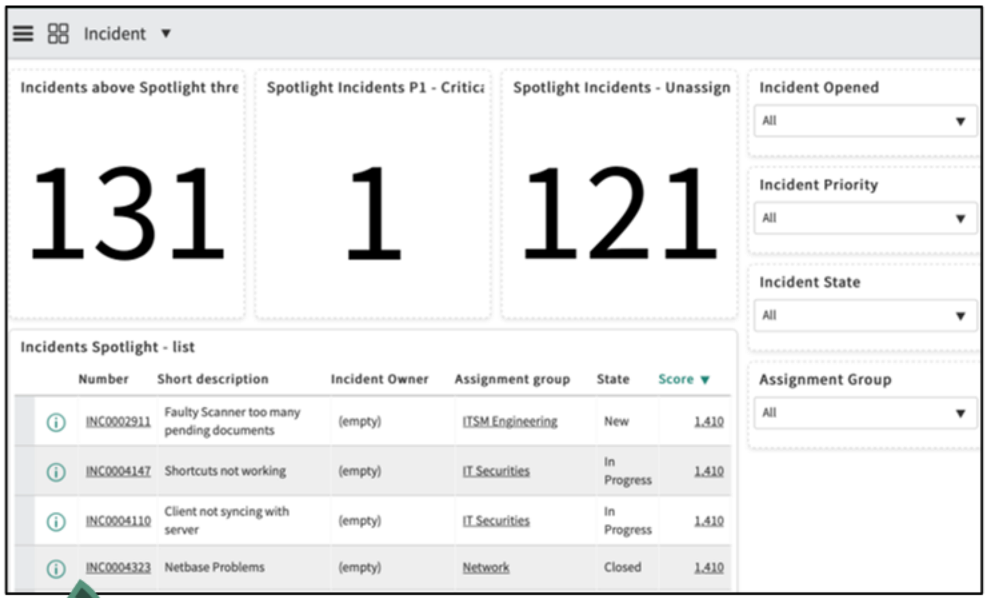
Dashboard Viewer

The **pa_spotlight**, **pa_spotlight_viewer, pa_spotlight_copy_breakdown,** and **pa_spotlight_copy_domain** are four roles included with the Spotlight application. The pa_admin and pa_power_user roles do not have access to the Spotlight application.

- The pa_spotlight contains the pa_viewer role and has Spotlight administrative privileges which allows to create groups, copy groups for different breakdowns, build criteria, execute Spotlight collection, view dashboards shared with pa_viewer, and edit the Spotlight dashboards
- The pa_spotlight_viewer has no privileges out of the box. This role should only be used to allow users to view Spotlight Dashboards
- The pa_spotlight_copy_domain role allows users to copy entire Spotlight Group between domains.
- The pa_spotlight_copy_breakdown role allows users to copy entire Spotlight Group with different breakdown and elements.

Spotlight Dashboard

- Score Reports, Interactive Filters, Spotlight DB View
- Shared with itil, pa_spotlight, and pa_spotlight_viewer



Incidents above Spotlight thre	Spotlight Incidents P1 - Critic:	Spotlight Incidents - Unassign
131	1	121

Incident Opened: All

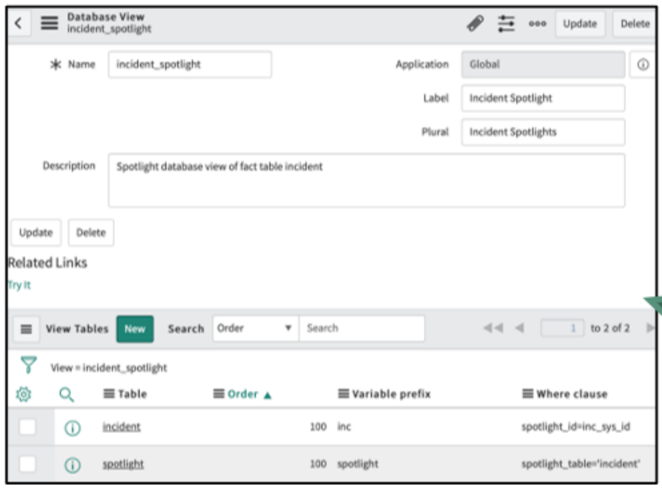
Incident Priority: All

Incident State: All

Assignment Group: All

Incidents Spotlight - list

Number	Short description	Incident Owner	Assignment group	State	Score ▼
INC0002911	Faulty Scanner too many pending documents	(empty)	ITSM Engineering	New	1,410
INC0004147	Shortcuts not working	(empty)	IT Securities	In Progress	1,410
INC0004110	Client not syncing with server	(empty)	IT Securities	In Progress	1,410
INC0004323	Netbase Problems	(empty)	Network	Closed	1,410



✱ Name	incident_spotlight	Application	Global
		Label	Incident Spotlight
		Plural	Incident Spotlights
Description	Spotlight database view of fact table incident		

View = incident_spotlight

Table	Order ▲	Variable prefix	Where clause
incident	100	inc	spotlight_id=inc_sys_id
spotlight	100	spotlight	spotlight_table='incident'

Database view joins Incident and Spotlight tables using the Incident's sys_id

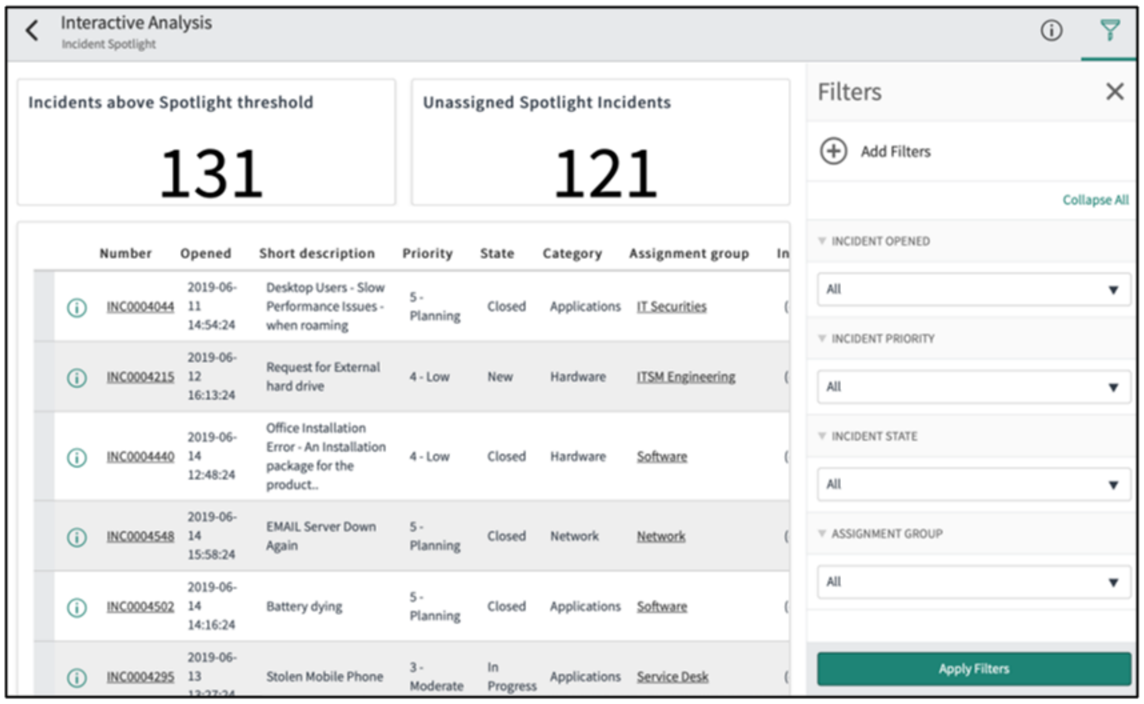
The Incident Spotlight solution provides an out-of-the-box dashboard with the following widgets:

- Score report of Incidents Above the Spotlight threshold
- Score report of Incidents Above the Spotlight threshold that have Critical Priority
- Score report of Incidents Above the Spotlight threshold that are Unassigned
- Several interactive Incident filters are automatically added to the dashboard – Assignment Group, State, Priority, Date Opened
- A list view of the records in the Spotlight based on a database view

Shown here is the Incident Spotlight database view that joins the Spotlight and Incident tables. It lets you easily view incident record details along with their Spotlight weight,. You can use it as an example to create your own database views to other tables. Join the tables so that the Spotlight ID matches the source record's sys_id value.

Spotlight Interactive Analysis

- Shows the Spotlight output
- Visible to **pa_spotlight_viewer**



Interactive Analysis
Incident Spotlight

Incidents above Spotlight threshold
131

Unassigned Spotlight Incidents
121

Number	Opened	Short description	Priority	State	Category	Assignment group	In
INC0004044	2019-06-11 14:54:24	Desktop Users - Slow Performance Issues - when roaming	5 - Planning	Closed	Applications	IT Securities	(
INC0004215	2019-06-12 16:13:24	Request for External hard drive	4 - Low	New	Hardware	ITSM Engineering	(
INC0004440	2019-06-14 12:48:24	Office Installation Error - An Installation package for the product..	4 - Low	Closed	Hardware	Software	(
INC0004548	2019-06-14 15:58:24	EMAIL Server Down Again	5 - Planning	Closed	Network	Network	(
INC0004502	2019-06-14 14:16:24	Battery dying	5 - Planning	Closed	Applications	Software	(
INC0004295	2019-06-13	Stolen Mobile Phone	3 - Moderate	In Progress	Applications	Service Desk	(

Filters

Add Filters

Collapse All

INCIDENT OPENED
All

INCIDENT PRIORITY
All

INCIDENT STATE
All

ASSIGNMENT GROUP
All

Apply Filters

The spotlights Interactive Analysis dashboard presents similar information to that on the Spotlight Dashboard. Additional Interactive filters can be configured and added as needed to visualize the Spotlight data.

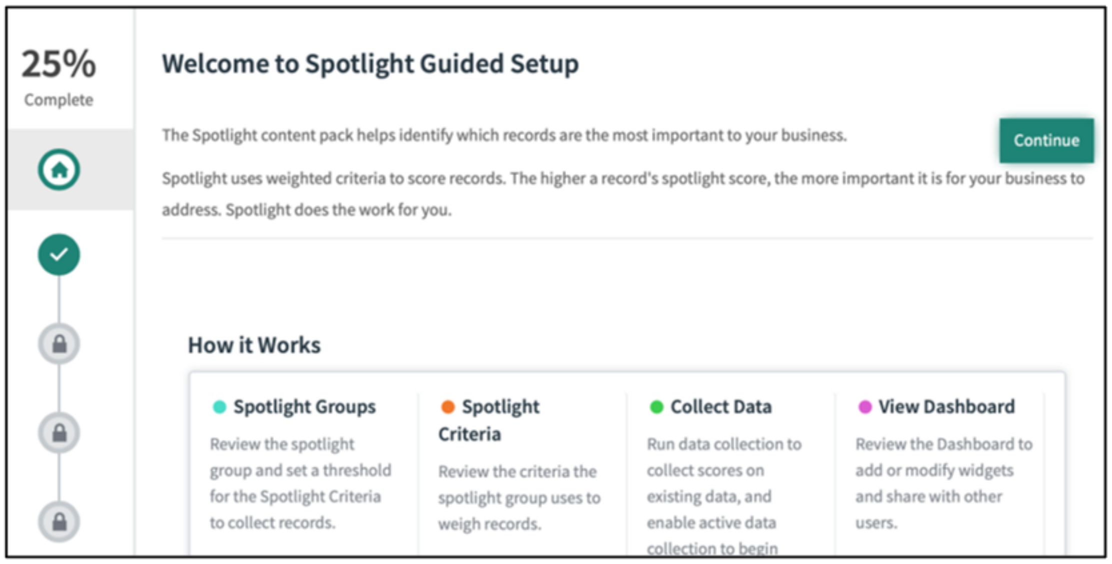
Interactive Analysis contains the following information:

- The number of Spotlight records. These are all fact table records that exceeded the Spotlight Group threshold.
- The number of unassigned Spotlights.
- The database view that joins the Spotlight table with the facts table of the main indicator.

Share this dashboard with anyone who needs to see the final output of the Spotlight collection. The user with whom you share the interactive analysis must have the **pa_spotlight_viewer** role.

Activating Spotlight Solutions

- Guided Setup for:
  - Incident Spotlight
  - Problem Spotlight
  - Change Spotlight
  - Request Spotlight
- Spotlight solutions provide:
  - Spotlight Group
  - Spotlight Criteria
  - Dashboard



Welcome to Spotlight Guided Setup

The Spotlight content pack helps identify which records are the most important to your business.

Spotlight uses weighted criteria to score records. The higher a record's spotlight score, the more important it is for your business to address. Spotlight does the work for you.

How it Works

Spotlight Groups
Review the spotlight group and set a threshold for the Spotlight Criteria to collect records.

Spotlight Criteria
Review the criteria the spotlight group uses to weigh records.

Collect Data
Run data collection to collect scores on existing data, and enable active data collection to begin

View Dashboard
Review the Dashboard to add or modify widgets and share with other users.

To activate Spotlight, enable the base plugin com.snc.pa.spotlight.

The following preconfigured Spotlight solutions are available:

- Performance Analytics - Spotlight - Change Spotlight content pack (com.snc.pa.spotlight.change)
- Performance Analytics - Spotlight - Incident Spotlight content pack (com.snc.pa.spotlight.incident)
- Performance Analytics - Spotlight - Problem spotlight content pack (com.snc.pa.spotlight.problem)
- Performance Analytics - Spotlight - Request spotlight content pack (com.snc.pa.spotlight.request)

In addition, a **Spotlight > Guided Setup** module is provided to assist with the Spotlight deployment.

Spotlight Lab

now.

Lab 6.1 Configure Spotlight for Incident Management

Spotlight for Incident Management

Lab 6.1

⌛25m

Lab Objectives

The Cloud Dimensions team would like to leverage reports and indicators to prioritize their work and focus on the most pressing incidents. In this lab, you perform the following activities:

- Activate and configure the Spotlight application for Incident Management
- Analyze service performance using Spotlight visualizations

Activate and Configure Spotlight

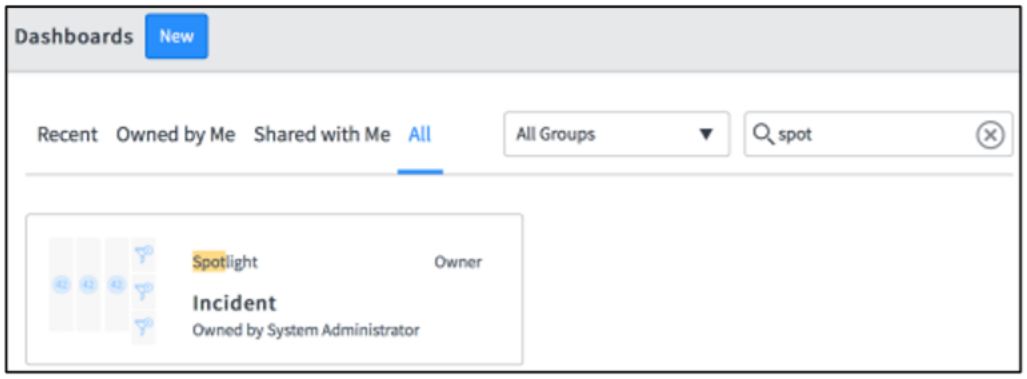
A. Solution Activation

Activation and Verification of required plugins

In this section, you configure the base Spotlight plugin and the Spotlight for Incidents solution.

1. Log in to the lab environment as a **System Administrator**.
2. Navigate to **System Definition > Plugins**.
3. Activate the following plugins:
   - Performance Analytics – Spotlight (com.snc.pa.spotlight)
   - Performance Analytics - Spotlight - Incident Spotlight content pack (com.snc.pa.spotlight.incident)

   > ***Note:*** *Activating the Performance Analytics - Spotlight - Incident Spotlight content pack plugin will automatically activate the Performance Analytics – Spotlight plugin.*
4. Navigate to **Performance Analytics > Dashboards.**
5. Open the **Spotlight** dashboard to validate successful plugin installation.



6. Confirm that widgets on the Spotlight Incident dashboard show no data.

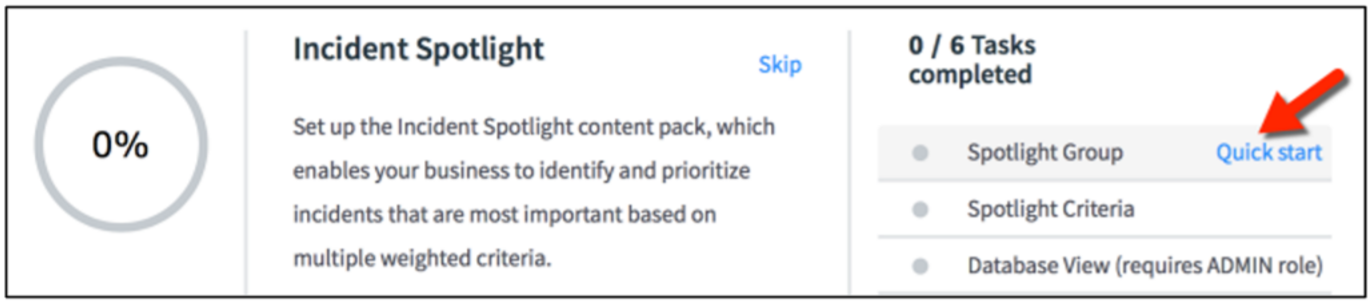
> ***Note:*** *Data becomes available after configuration is completed and collection is performed.*

B. Review of Spotlight for Incidents using Guided Setup

In this section, you will review the existing setup of the Spotlight for Incidents solution using the Guided Setup application.

Spotlight Group

1. Navigate to **Spotlight > Guided Setup**.
2. Select the **Get Started** button.
3. Focus on the **Incident Spotlight** configuration task (top of the list).
4. Hover over the **Spotlight Group** task and click the **Quick start** link.



Set up the Incident Spotlight content pack, which enables your business to identify and prioritize incidents that are most important based on multiple weighted criteria.

5. Open the **Incident Spotlight Group**.

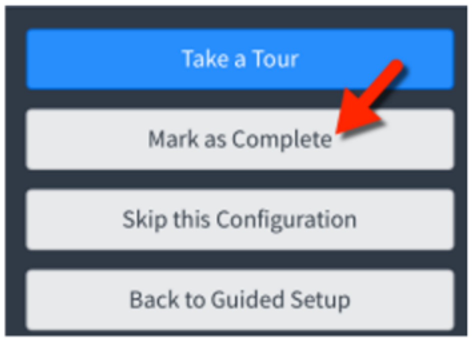
> ***Questions:***
>
> *Which Indicator is this spotlight based on?*
>
> *What is the cut-off score for generating a spotlight record?*

6. Enable the **Active** checkbox and **Save**.

> ***Note**: The Execute Now button is now visible on the form.*

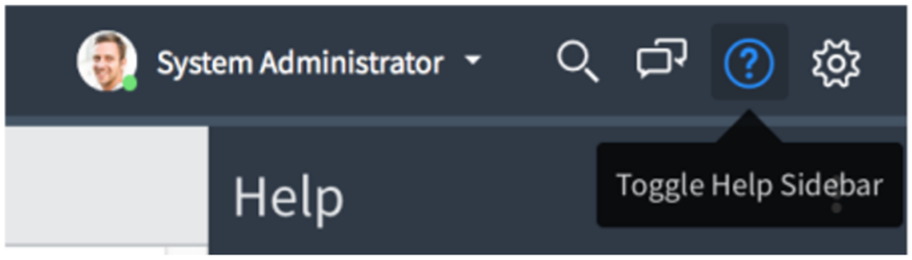
7. Click the **Back** button on the form to return to the list of Spotlight groups.

8. Click the **Mark as Complete** button in the **Help** panel.



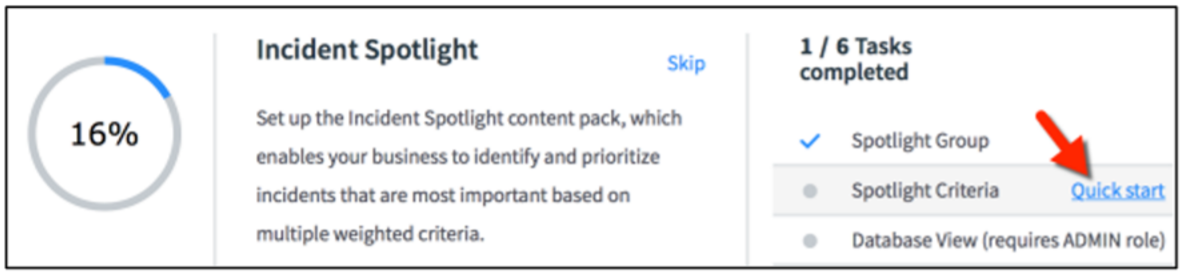
9. Verify that your current completion status is **16%**.

> ***Note:** if at any point you close the **Help** panel/sidebar, press the Help icon on the banner to restore it.*



Spotlight Criteria

1. Hover over the **Spotlight Criteria** task and click the **Quick start** link.



Set up the Incident Spotlight content pack, which enables your business to identify and prioritize incidents that are most important based on multiple weighted criteria.

2. Review the components of the Spotlight Criteria.

> ***Questions:***
>
> *How many **Query** Criteria have been defined?*
>
> *How many **Indicator** Criteria have been defined?*
>
> *What is the base Indicator source for all Indictor queries?*

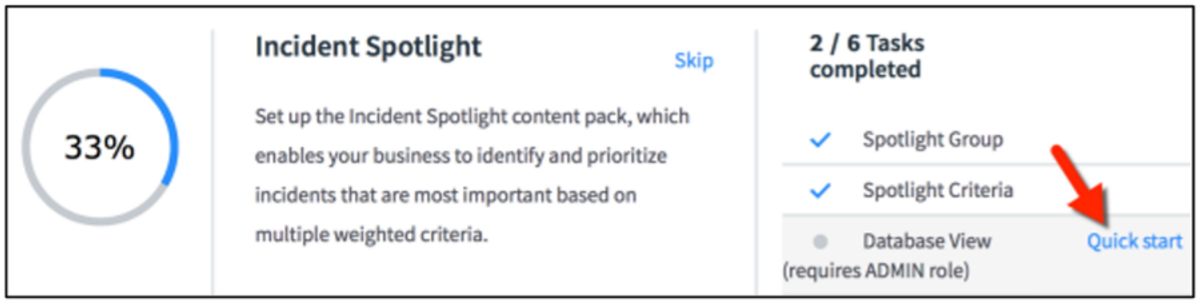
> *Which are the Criteria with highest **Weight**?*
>
> *Which criteria will use snapshot data which will use platform data?*

3. Click the **Mark as Complete** button in the **Help** panel.

4. Verify that your current completion status is **33%**.

Database View

1. Hover over the **Database View** task and click the **Quick start** link.

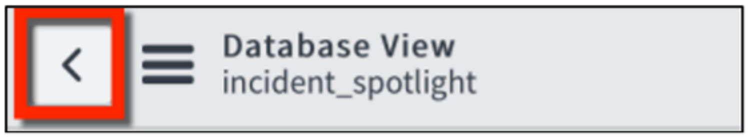


Set up the Incident Spotlight content pack, which enables your business to identify and prioritize incidents that are most important based on multiple weighted criteria.

2. Open the **incident_spotlight** view and review its configuration.

> ***Question:*** *Which field is used to join the incident and the spotlight tables?*

3. Press the Back button of the database view.

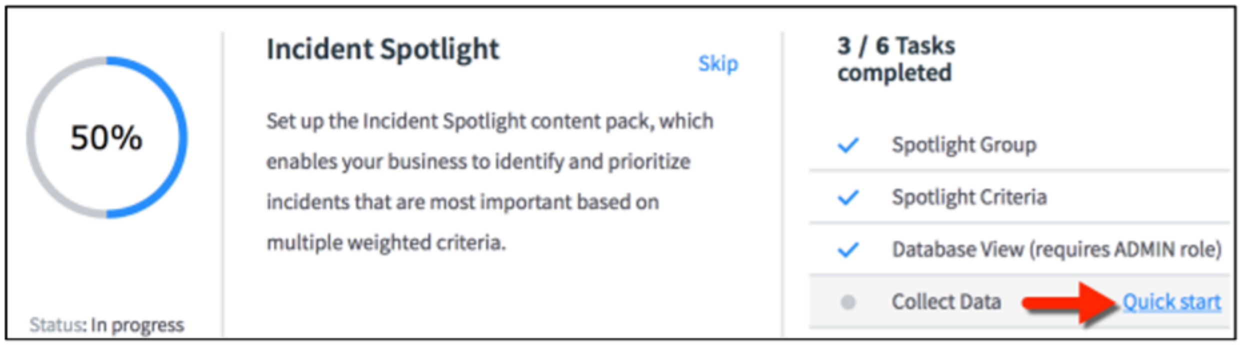


4. Click the **Mark as Complete** button in the **Help** panel.

5. Verify that your current completion status is **50%**.

Collect Data

1. Hover over the **Collect Data** task and click the **Quick start** link.



Set up the Incident Spotlight content pack, which enables your business to identify and prioritize incidents that are most important based on multiple weighted criteria.

Status: In progress

2. Review the **Time** attribute under **Schedule**.

> ***Note***: *Note: Make sure that the Spotlight collection happens after the Indicator collection for most accurate results.*

3. Press the **Execute Now** button.

> ***Tip***: *Allow some time for the job to complete calculating and generating a list of Incidents that meet the spotlight criteria.*

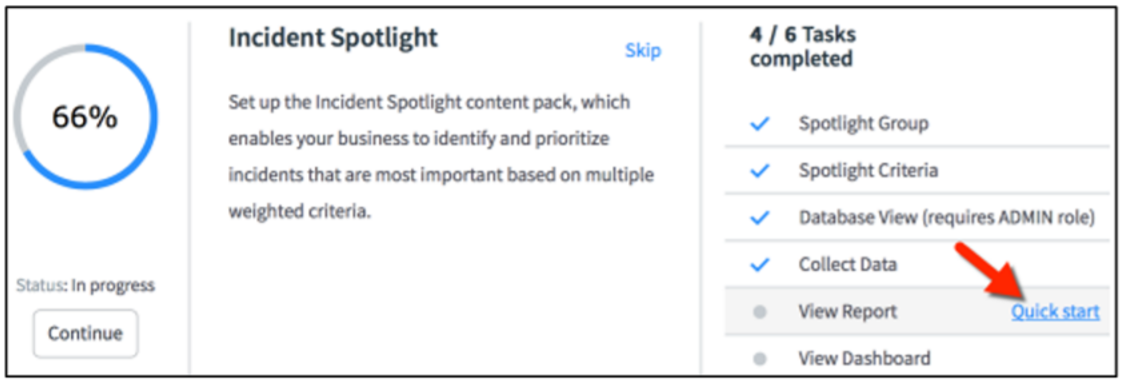
4. Refresh the form and note that you have new records in the **Spotlights** Related List.

5. Click the **Mark as Complete** button in the **Help** panel.

6. Verify that your current completion status is **66%**.

View Report

1. Hover over the **View Report** task and click the **Quick start** link.



Incident Spotlight

Set up the Incident Spotlight content pack, which enables your business to identify and prioritize incidents that are most important based on multiple weighted criteria.

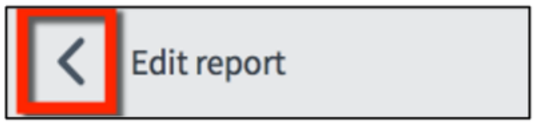
4 / 6 Tasks completed

- ✓ Spotlight Group
- ✓ Spotlight Criteria
- ✓ Database View (requires ADMIN role)
- ✓ Collect Data
- ● View Report
- ● View Dashboard

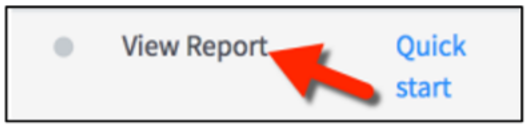
2. Confirm that 80+ incident spotlights have been generated and shown in the list.

> ***Note:*** *This report retrieves records from the incident_spotlight view with Spotlight Group set to Incident Spotlight.*

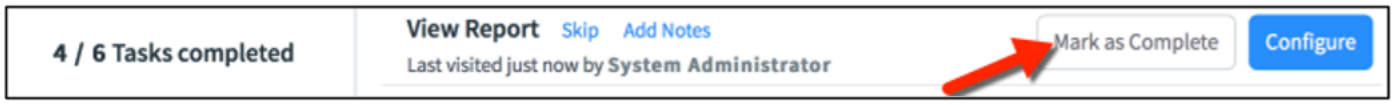
3. Click the **Back** button to return to the Guided Setup.



4. Click **View Report** to open the task and update its status.



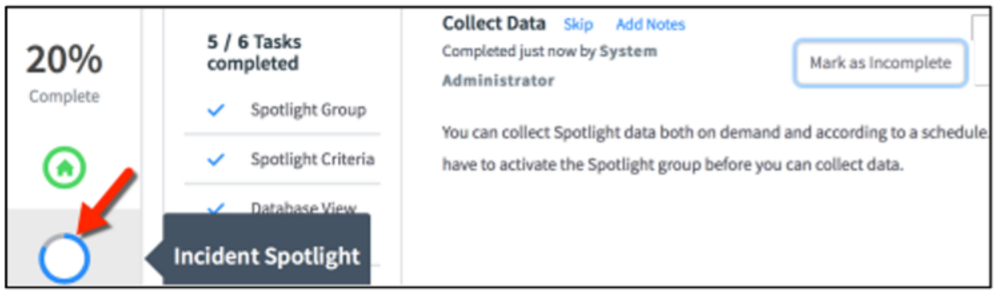
5. Mark the **View Report** task as Complete.



Last visited just now by System Administrator

View Dashboard

1. Click the **Incident Spotlight** task (circle) in the Guided Setup menu.



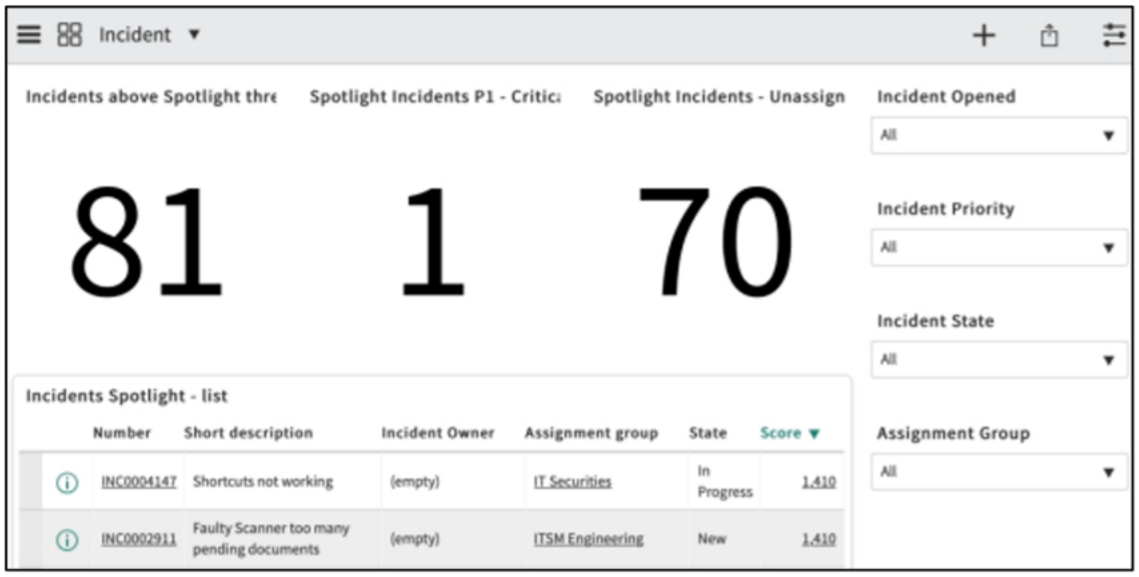
Completed just now by System Administrator

You can collect Spotlight data both on demand and according to a schedule
have to activate the Spotlight group before you can collect data.

2. Navigate to the **View Dashboard** step in the Guided Setup.

3. Click the corresponding **Quick start** link.

4. Confirm that the Incident Dashboard displays data in all Widgets.


Incident
Incidents above Spotlight thre
81
Spotlight Incidents P1 - Critic
1
Spotlight Incidents - Unassign
70
Incident Opened
All
Incident Priority
All
Incident State
All
Assignment Group
All

Incidents Spotlight - list

	Number	Short description	Incident Owner	Assignment group	State	Score ▼
ⓘ	INC0004147	Shortcuts not working	(empty)	IT Securities	In Progress	1,410
ⓘ	INC0002911	Faulty Scanner too many pending documents	(empty)	ITSM Engineering	New	1,410

***Note:** Your results may differ slightly from the example.*

5. Access the dashboard menu and select **Create Favorite**.

6. Confirm that the **Incident Dashboard** shortcut has been created.

7. Click the **Mark as Complete** button in the **Help** panel.

8. Confirm that you have completed all Incident Spotlight configuration tasks.

9. Navigate to **Spotlight > Spotlights**. Confirm that the list of Spotlights contains the same incidents displayed in the **Incident Spotlights** – list widget.

10. Navigate to **Spotlight > Spotlight Job Logs**.

> ***Note***: *The list contains a record of all Spotlight job executions.*

Modify Spotlight

C. Adjust Spotlight Threshold

1. Navigate to the **Spotlight > Spotlight Groups**.

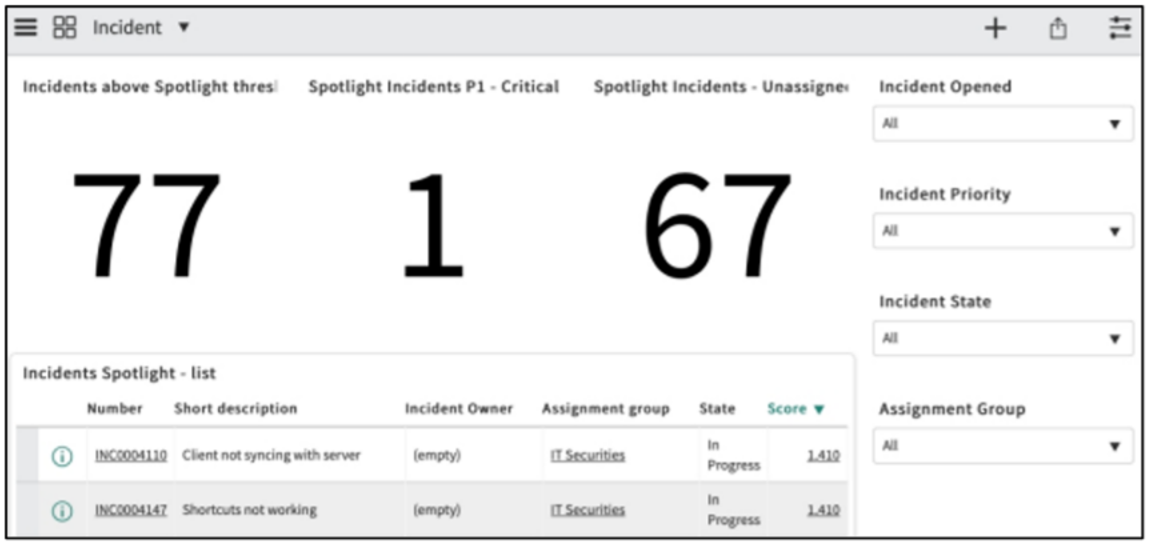
2. Open the **Incident Spotlight** group record.

3. Set the **Threshold** to **1,200**.

4. Click **Execute Now** to run the Spotlight job again to re-calculate the incident spotlight records.

5. Select the **Incident Dashboard** shortcut created earlier.

6. Confirm the dashboard displays new Spotlight information.



Incidents above Spotlight thres| 77

Spotlight Incidents P1 - Critical 1

Spotlight Incidents - Unassigne| 67

Incident Opened: All ▾

Incident Priority: All ▾

Incident State: All ▾

Assignment Group: All ▾

Incidents Spotlight - list

Number	Short description	Incident Owner	Assignment group	State	Score ▾
INC0004110	Client not syncing with server	(empty)	IT Securities	In Progress	1,410
INC0004147	Shortcuts not working	(empty)	IT Securities	In Progress	1,410

> ***Note:*** *There should now be fewer Spotlights as your threshold is higher.*

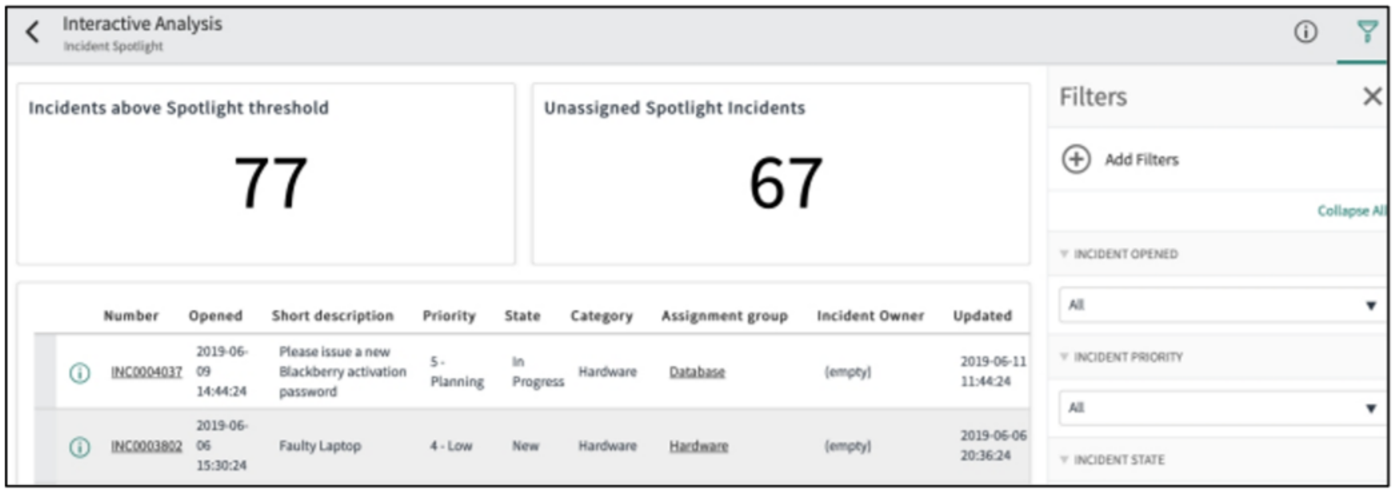
7. Navigate to **Spotlight > Spotlight Logs** and open the latest **Spotlight Job Log**. How many inserts and deletes have been recorded for the latest job execution?

8. Navigate to **Spotlight > Spotlight**. Confirm that the list of Spotlights contains the same incidents displayed in the **Incident Spotlights** – list widget.

Service Performance Analysis using Interactive Analysis

D. Spotlight Interactive Analysis

1. Navigate to **Spotlight > Spotlight Groups**.
2. Open the **Incident Spotlight** group.
3. Click the **Launch Spotlight Interactive Analysis** Related Link.
4. Review all available interactive filters.



Interactive Analysis
Incident Spotlight

Incidents above Spotlight threshold
77

Unassigned Spotlight Incidents
67

Number	Opened	Short description	Priority	State	Category	Assignment group	Incident Owner	Updated
INC0004037	2019-06-09 14:44:24	Please issue a new Blackberry activation password	5 - Planning	In Progress	Hardware	Database	(empty)	2019-06-11 11:44:24
INC0003802	2019-06-06 15:30:24	Faulty Laptop	4 - Low	New	Hardware	Hardware	(empty)	2019-06-06 20:36:24

Filters

Add Filters

Collapse All

INCIDENT OPENED: All

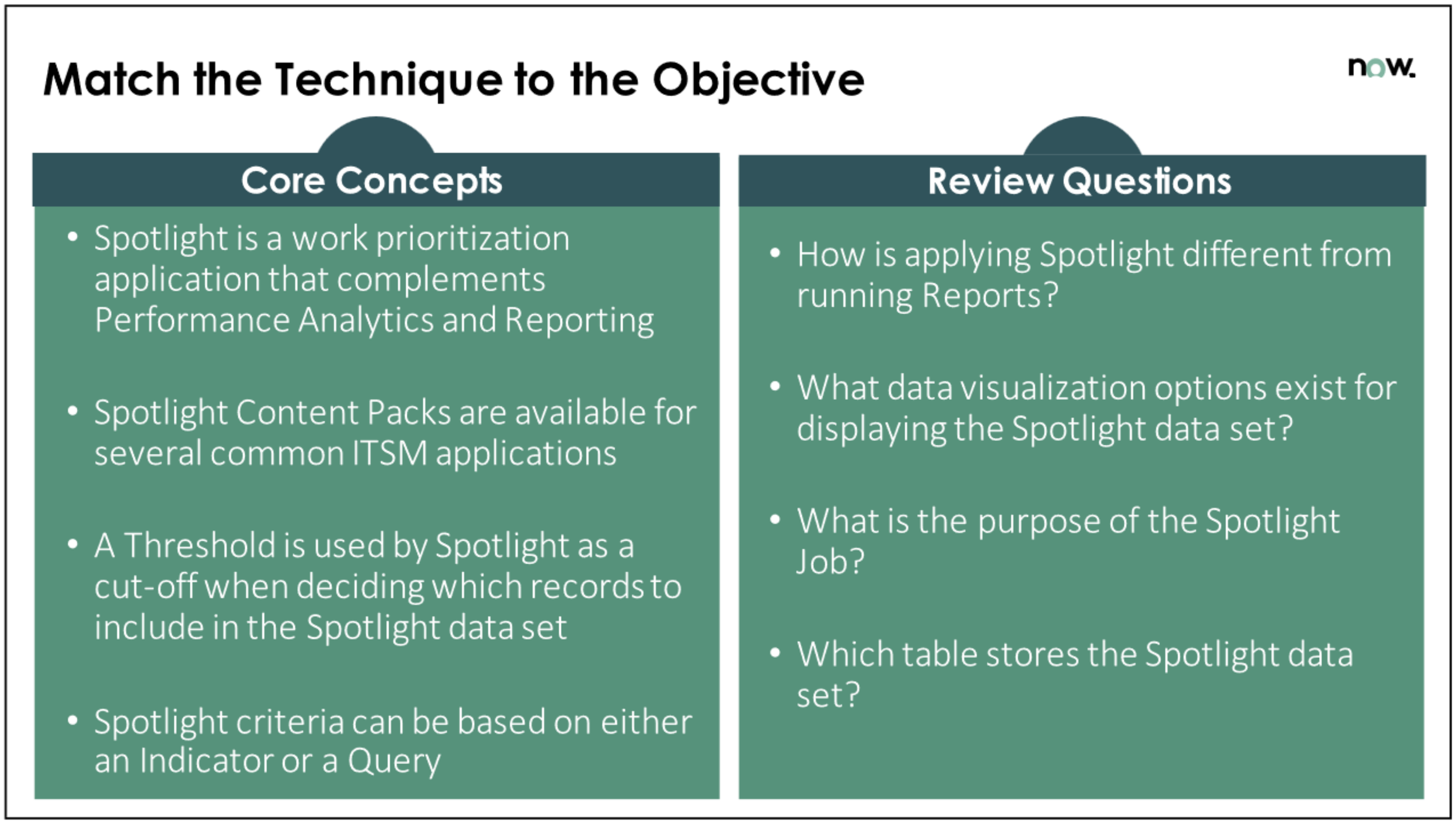
INCIDENT PRIORITY: All

INCIDENT STATE

Discussions

1. Your goal is to trend the daily number of generated Spotlights. Describe a solution that would achieve this goal.
2. Your goal is to implement a Spotlight solution for Incidents with low survey scores. Describes the necessary Spotlight group, criteria, and other setup required.

You have now completed the Spotlight lab.



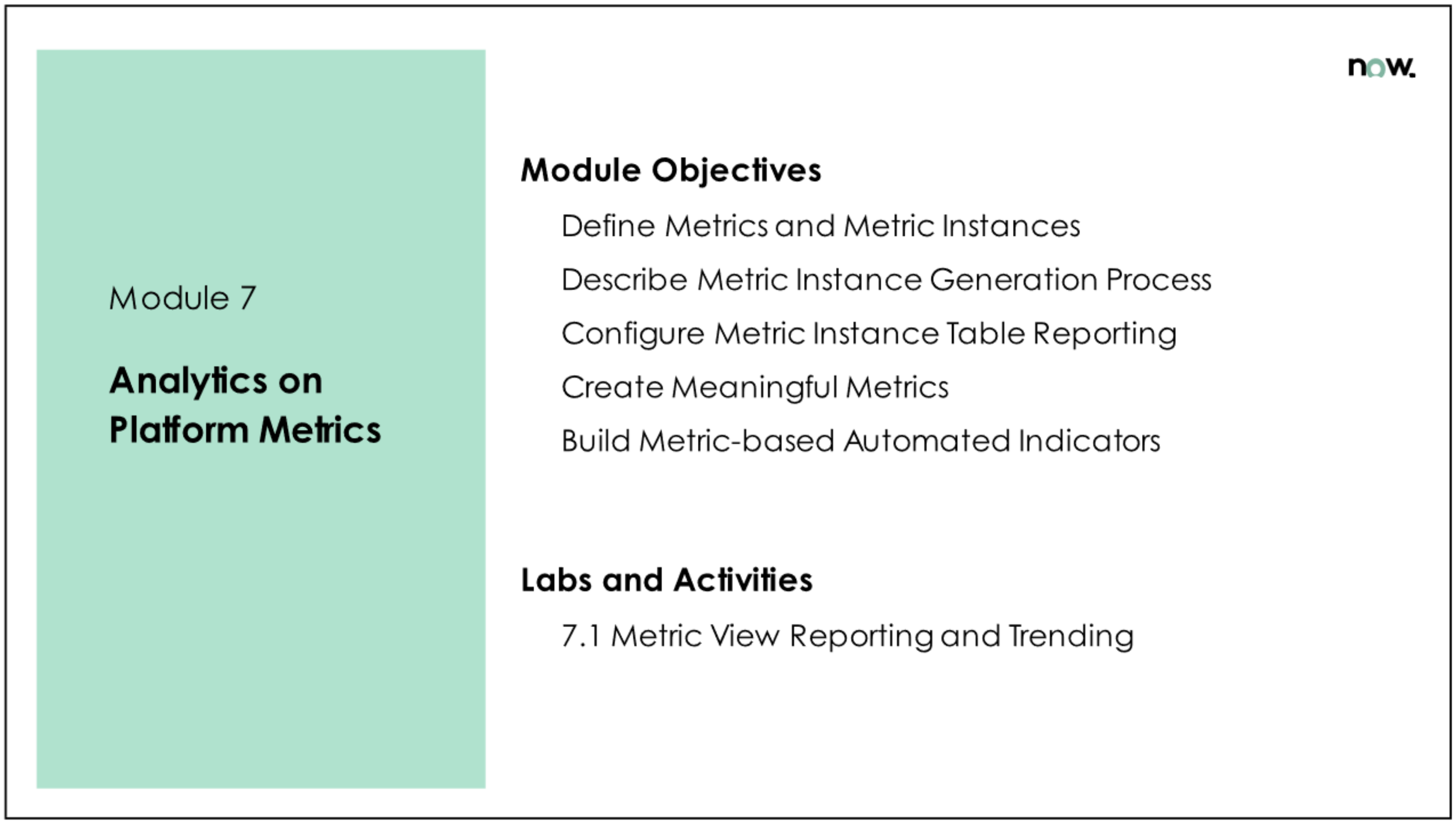
Match the Technique to the Objective

Core Concepts

- Spotlight is a work prioritization application that complements Performance Analytics and Reporting
- Spotlight Content Packs are available for several common ITSM applications
- A Threshold is used by Spotlight as a cut-off when deciding which records to include in the Spotlight data set
- Spotlight criteria can be based on either an Indicator or a Query

Review Questions

- How is applying Spotlight different from running Reports?
- What data visualization options exist for displaying the Spotlight data set?
- What is the purpose of the Spotlight Job?
- Which table stores the Spotlight data set?



Module 7

Analytics on Platform Metrics

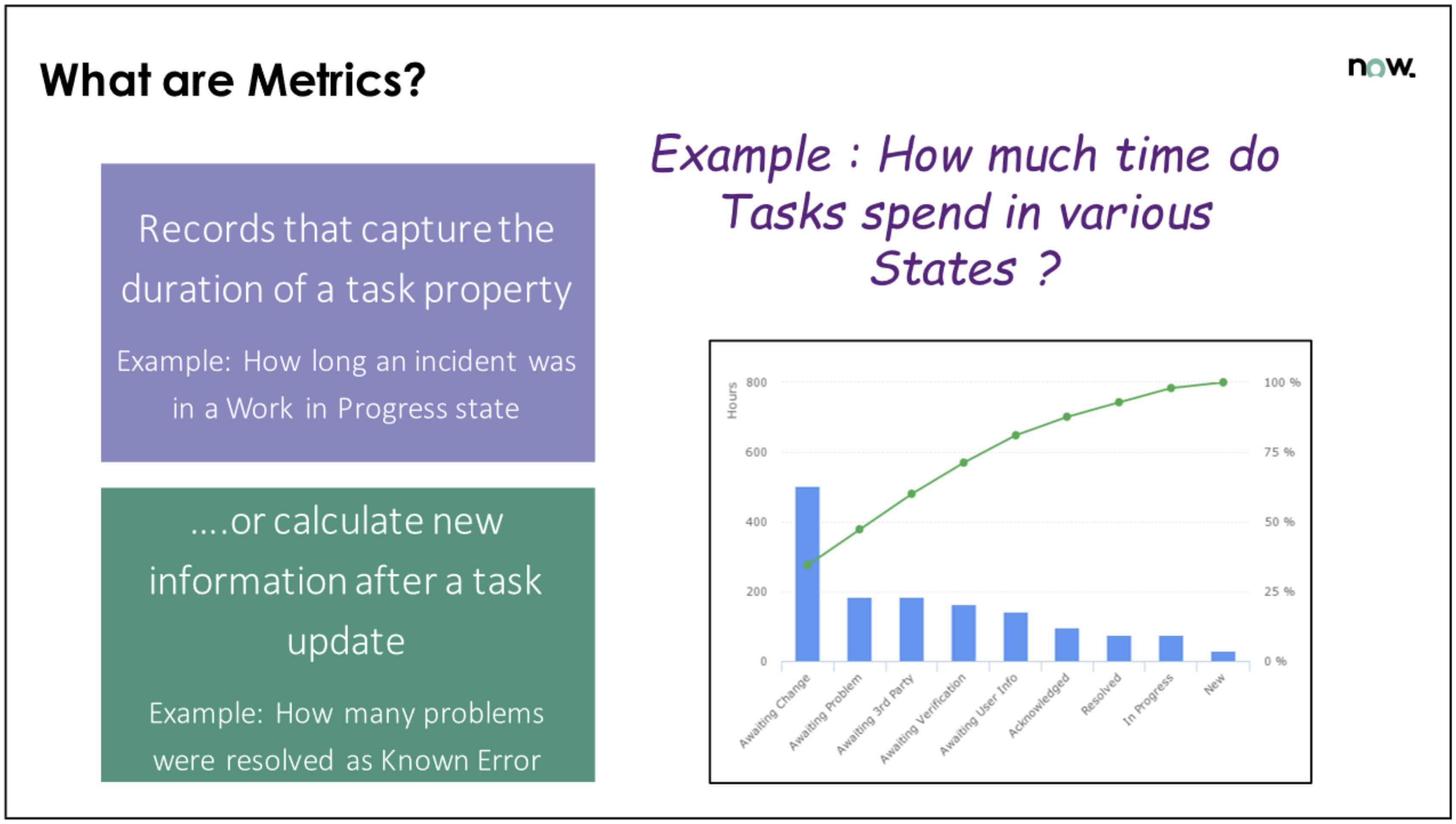
Module Objectives

- Define Metrics and Metric Instances
- Describe Metric Instance Generation Process
- Configure Metric Instance Table Reporting
- Create Meaningful Metrics
- Build Metric-based Automated Indicators

Labs and Activities

- 7.1 Metric View Reporting and Trending

This module puts the focus on automating process information using Metric tables as a data source. Metric definitions and Metric instances are a platform feature that allows to capture instantaneously any property change of a task and record the amount of time a property has had a certain value. Reporting and additional Analytics can be built to summarize and track metric behavior over time.



What are Metrics?

Records that capture the duration of a task property

Example: How long an incident was in a Work in Progress state

....or calculate new information after a task update

Example: How many problems were resolved as Known Error

Example : How much time do Tasks spend in various States ?

Hours
800
600
400
200
0
100 %
75 %
50 %
25 %
0 %
Awaiting Change
Awaiting Problem
Awaiting 3rd Party
Awaiting Verification
Awaiting User Info
Acknowledged
Resolved
In Progress
New

A metric captures a task update event as defined by the metric logic. Most commonly, the task event is a property change such as State or an Assignment change. For instance, a metric can be used to capture the duration of time a task has its State set to Open or the duration of time a task is assigned to the Help Desk Assignment group.

More complex metric logic allows for the calculation of a new virtual property using current parameters of the business task being monitored.

Metrics are a ServiceNow platform capability. Enable Metrics by activating the com.glide.metrics plugin.

Metrics provide a Historical Trend of Task Updates

now.

Why do Assigned Incidents take so long ?

We need a Metric to store duration of each assignment

How can I track Assignment duration?

Then we can report against the Metric table and trend assignment durations

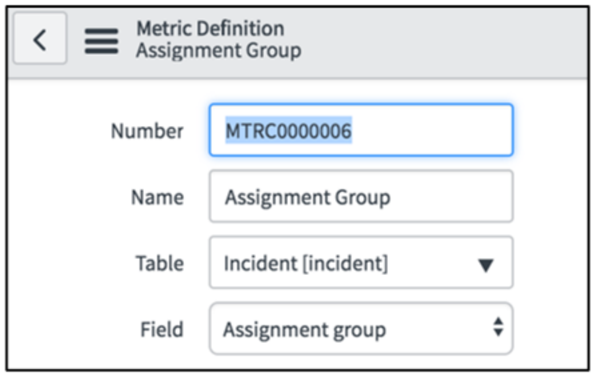
Metrics measure and evaluate the effectiveness of a business process. You implement metrics when you need more insight into a task or a process but the base task or process table does not contain this information. Consider this example:

- An incident gets reassigned multiple times before it gets resolved. You need to know how much time is spent in each Assignment Group.
- Reporting can provide average duration of all incidents in the incident table and break that down by Assignment Group. However, it does not track historical data about Incident reassignments. So any assignment changes prior to today are not tracked and timed.
- A metric can be implemented to track incident reassignments and record duration of each assignment. Any assignment change is timed and tracked with a metric instance.
- Historical information on incident assignment durations can be obtained by from the metric and metric instance tables.

Metric Definition and Instances

A **Metric Definition** describes the property or field to watch and measure

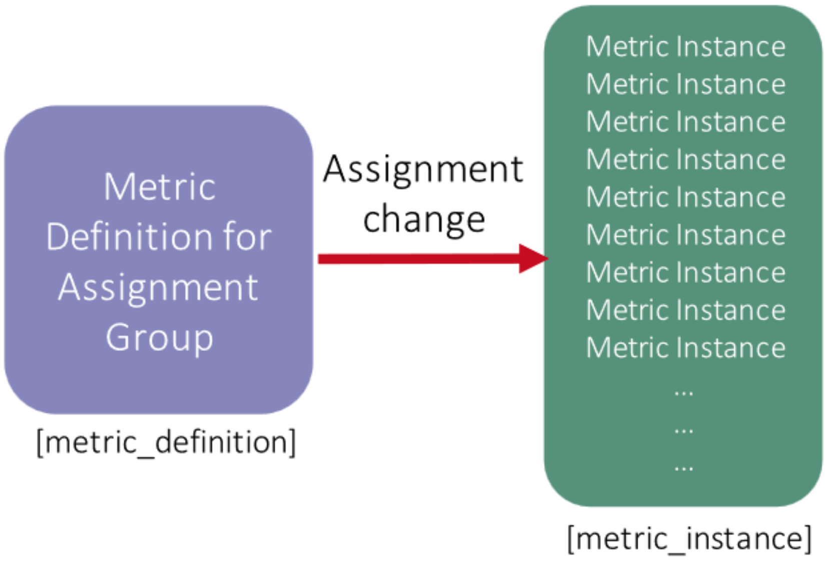
- Assignment Group / Assignee
- State
- Category



Metric Definition
Assignment Group

Number	MTRC0000006
Name	Assignment Group
Table	Incident [incident]
Field	Assignment group

A property update triggers a **Metric Instance** which records the field duration or performs a scripted calculation



Metric Definition for Assignment Group

[metric_definition]

Assignment change

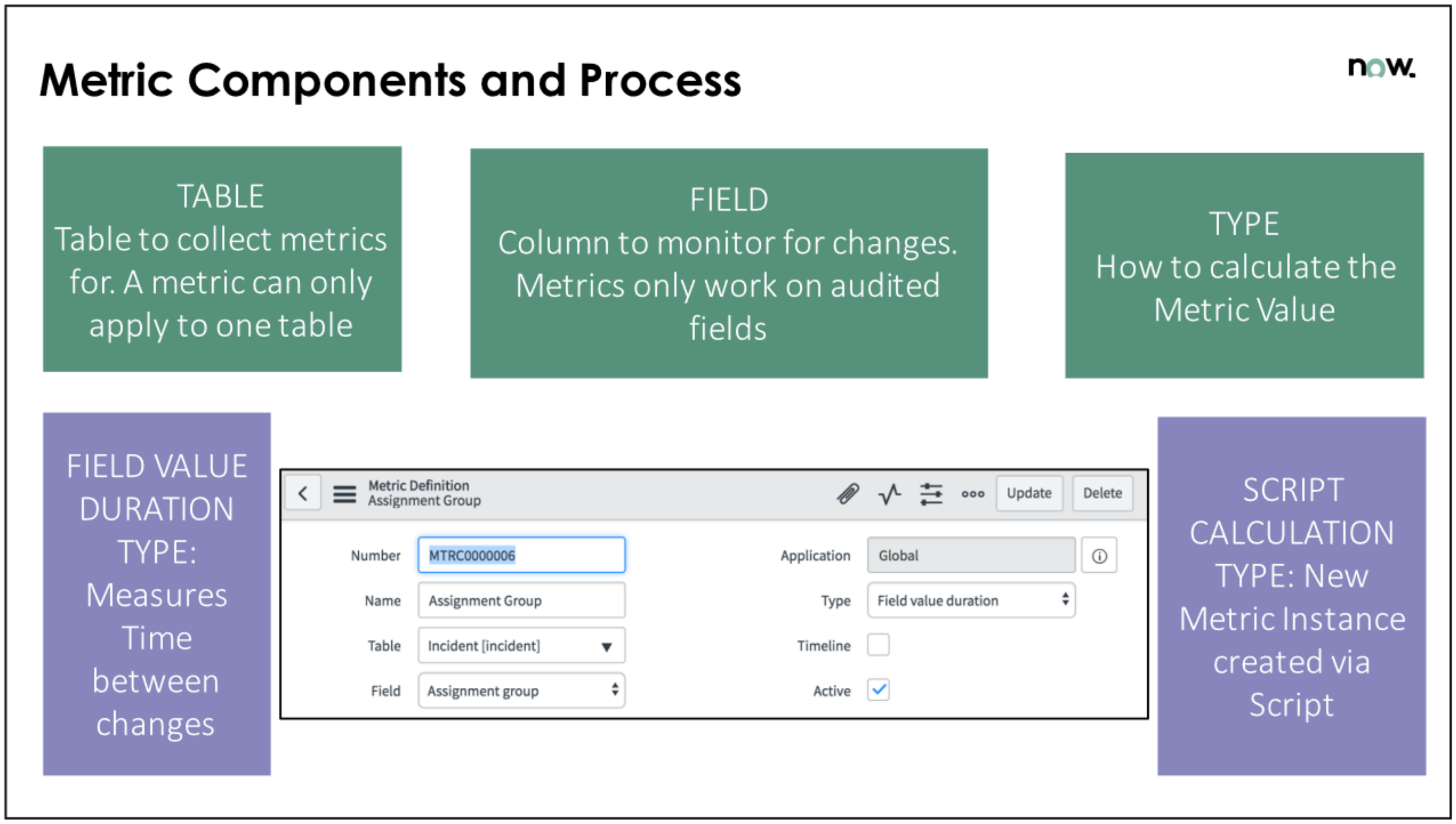
Metric Instance
Metric Instance
Metric Instance
Metric Instance
Metric Instance
Metric Instance
Metric Instance
Metric Instance
Metric Instance
…
…
…

[metric_instance]

Here is how you can use the Metrics application to measure the duration of time an incident is with a specific Assignment Group:

1. Create a new *Metric Definition*
2. Set the type to *Field value duration*
3. Set the *Table* to incident
4. Set the *Field* to *Assignment Group*

Once the Metric Definition is in place, a new metric instance will be created whenever the incident's Assignment Group attribute changes. The Metric Instance captures the amount of time the incident has spent assigned to a specific Assignment Group.



Metric Components and Process

TABLE
Table to collect metrics for. A metric can only apply to one table

FIELD
Column to monitor for changes. Metrics only work on audited fields

TYPE
How to calculate the Metric Value

FIELD VALUE DURATION TYPE: Measures Time between changes

Metric Definition
Assignment Group

Number	MTRC0000006	Application	Global
Name	Assignment Group	Type	Field value duration
Table	Incident [incident]	Timeline	□
Field	Assignment group	Active	☑

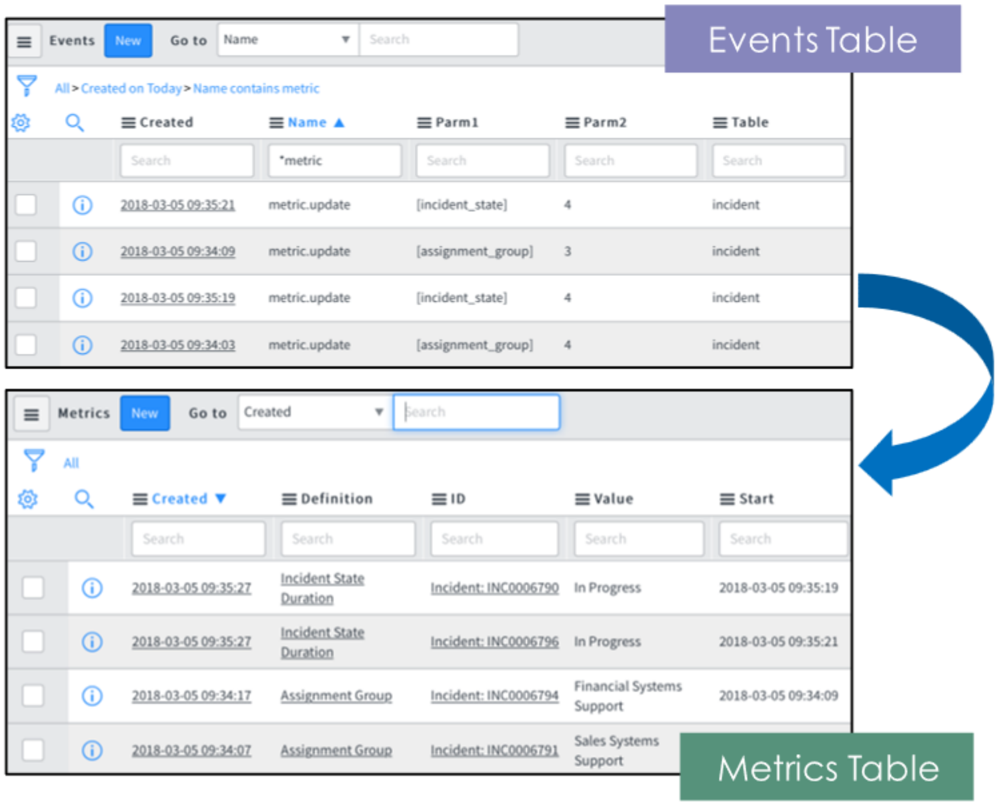
SCRIPT CALCULATION TYPE: New Metric Instance created via Script

Here are the most important configuration aspects of a Metric:

- **Table**: Specifies the table to collect metrics for. A metric can only apply to one table.
- **Field**: The column to monitor for changes. Remember that metrics only work on audited fields.
- **Type**:
  - **Field value duration**: Measure the duration of time from when the value of the specified field is set until it is changed.
  - **Script calculation**: Creates a metric instance using a script. The script has access to the current row in the table (such as the incident task) and the metric definition. The script can then perform any calculation and insert data into the metric_instance table. The script can calculate any type of value, not only a duration, and store it in the metric_instance table.

Metric Process Flow Deep Dive

1. Process Record is Updated
2. Business Rule queues up Event
3. Scheduled Job Processes Event
4. Event run Script Action
5. Metric Instance is created



Created	Name ▲	Parm1	Parm2	Table
Search	*metric	Search	Search	Search
2018-03-05 09:35:21	metric.update	[incident_state]	4	incident
2018-03-05 09:34:09	metric.update	[assignment_group]	3	incident
2018-03-05 09:35:19	metric.update	[incident_state]	4	incident
2018-03-05 09:34:03	metric.update	[assignment_group]	4	incident

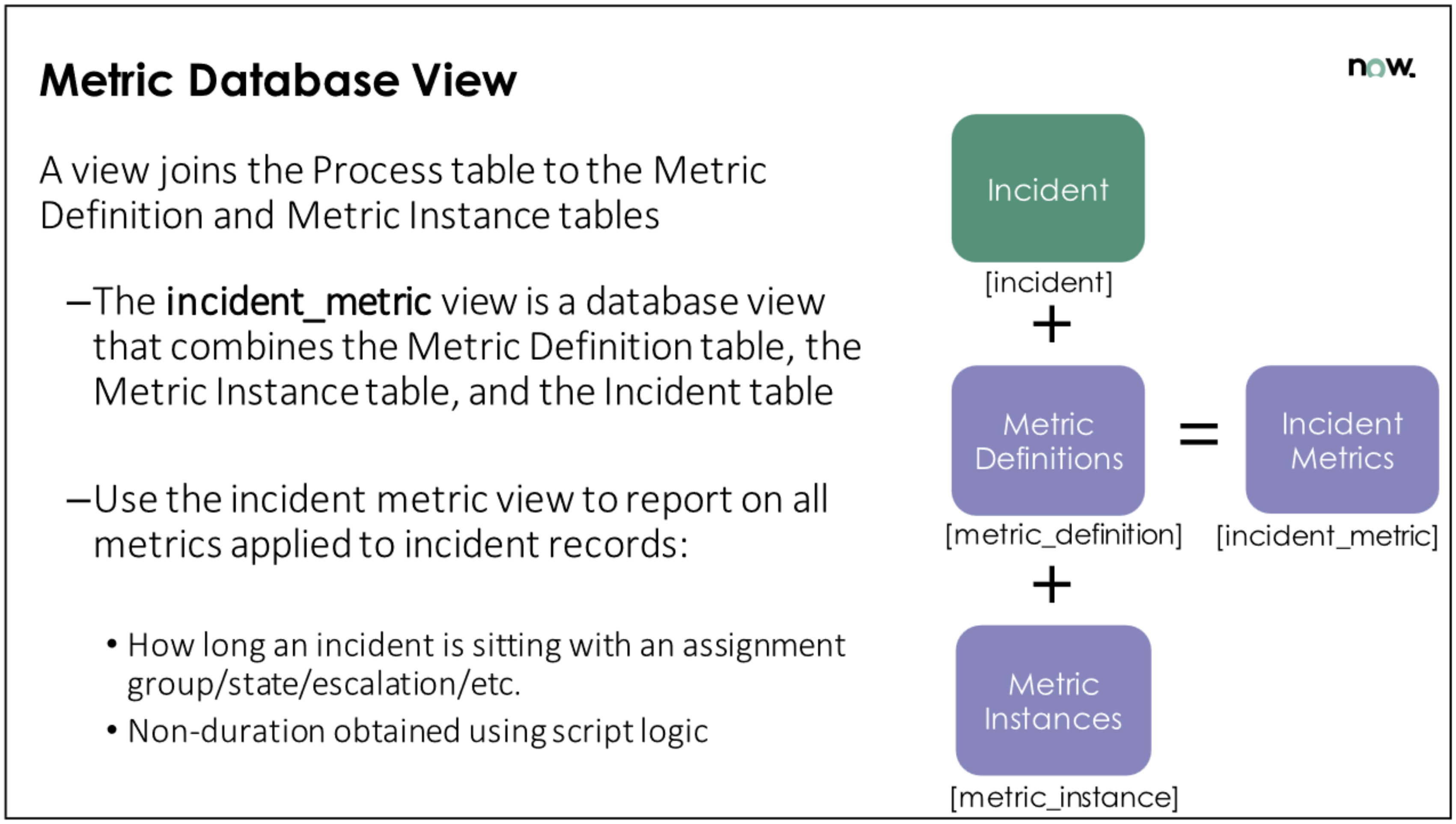
Created ▼	Definition	ID	Value	Start
Search	Search	Search	Search	Search
2018-03-05 09:35:27	Incident State Duration	Incident: INC0006790	In Progress	2018-03-05 09:35:19
2018-03-05 09:35:27	Incident State Duration	Incident: INC0006796	In Progress	2018-03-05 09:35:21
2018-03-05 09:34:17	Assignment Group	Incident: INC0006794	Financial Systems Support	2018-03-05 09:34:09
2018-03-05 09:34:07	Assignment Group	Incident: INC0006791	Sales Systems Support	

Here are the deep-dive details of metric processing:

- The flow is triggered by a record update such as an Assignment Group or State change.
- Business Rule **metric events** runs and creates and queues up a metric_update event. This is the business logic which creates a new metric.update event:

  gs.eventQueue('metric.update', current, fields, current.sys_mod_count, 'metric_update');

- A Scheduled Job runs periodically and processes all queued-up metric.update events. The **metric update events process** system job runs every 5 seconds and processes Script Actions that result in the creation of Metric Instances.
- As part of event processes, the Event's Script Action **Metric Update** is run.
- As a result, a new metric Instance is created.



Metric Database View

A view joins the Process table to the Metric Definition and Metric Instance tables

- The **incident_metric** view is a database view that combines the Metric Definition table, the Metric Instance table, and the Incident table

- Use the incident metric view to report on all metrics applied to incident records:
  - How long an incident is sitting with an assignment group/state/escalation/etc.
  - Non-duration obtained using script logic

Incident
[incident]
+
Metric Definitions
[metric_definition]
+
Metric Instances
[metric_instance]
=
Incident Metrics
[incident_metric]

As you recall from the earlier discussions, there are situations in which the data you need does not live in a single table. In these situations, you need to create a database view to join one or more tables for reporting purposes.

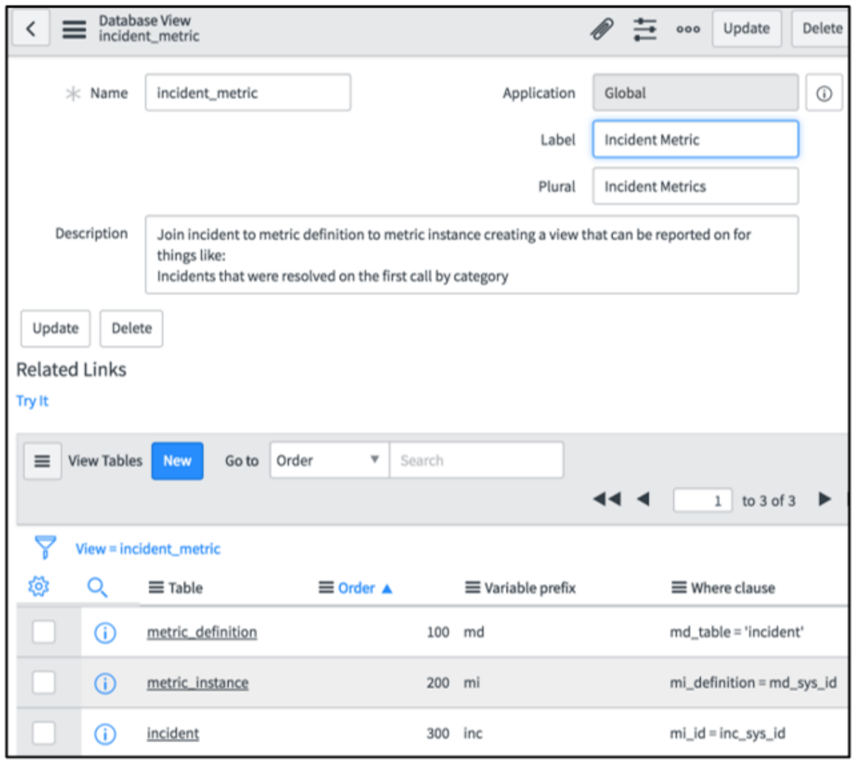
The incident_metric view joins incident, metric_definition, and metric_instance tables.

Metric View Creation

10 Process Metric Views: incident change, problem, release, request, etc.

Example:

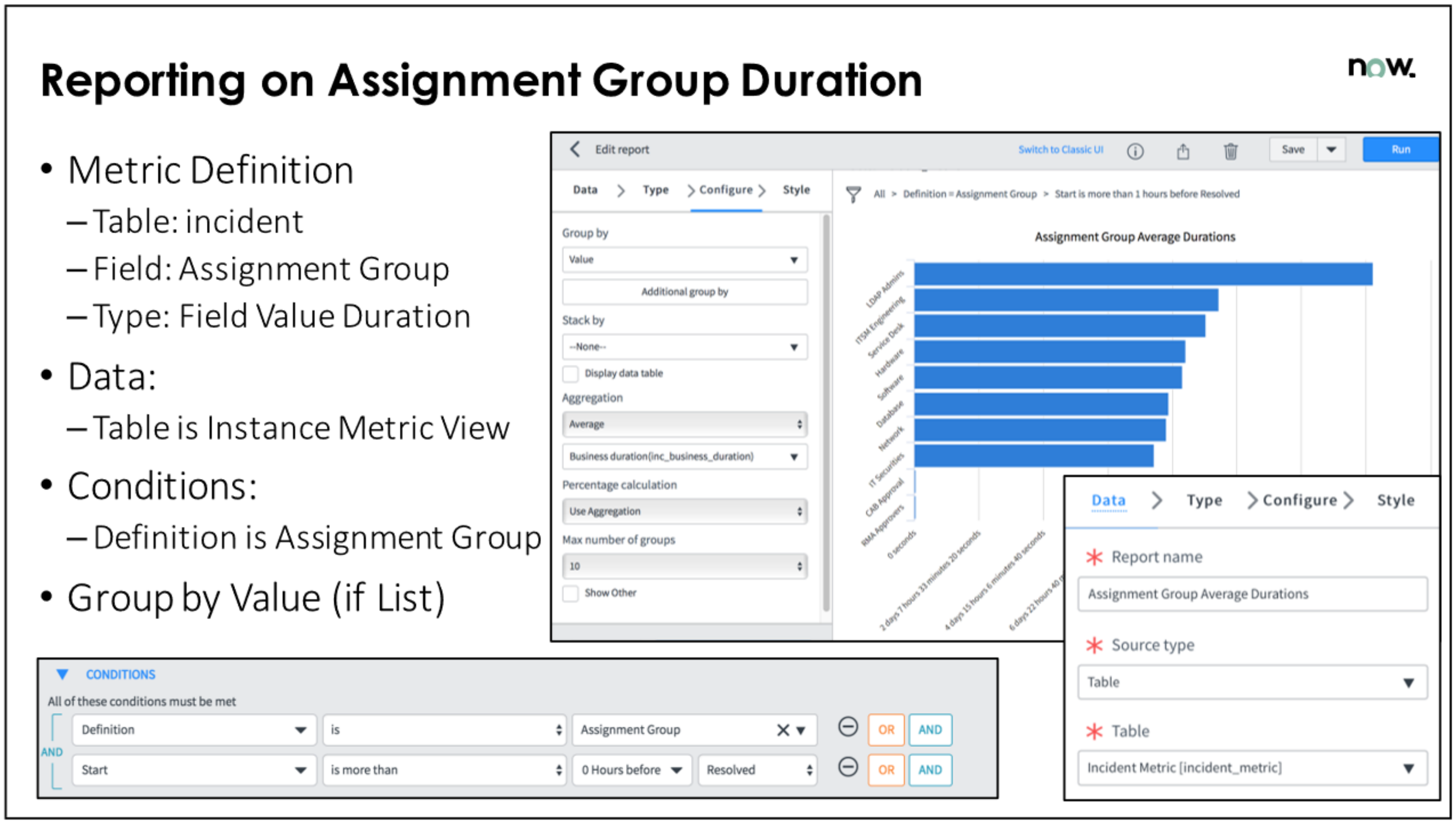
- Incident_metric view tables:
  - md_table = incident
    - Select only incident records from the metric table:
  - mi_definition = md_sys_id
    - Join metric_definition and metric_instance tables
  - mi_id = inc_sys_id
    - Join incident and metric_instance tables



Name	incident_metric
Application	Global
Label	Incident Metric
Plural	Incident Metrics
Description	Join incident to metric definition to metric instance creating a view that can be reported on for things like: Incidents that were resolved on the first call by category

Table	Order ▲	Variable prefix	Where clause
metric_definition	100	md	md_table = 'incident'
metric_instance	200	mi	mi_definition = md_sys_id
incident	300	inc	mi_id = inc_sys_id

Many metric views already exist in the baseline ServiceNow instance. System and Reporting administrators can create additional views and alter existing views to build the required metric reporting objects. System Administrators and users with the **metric_admin** role can manage metrics.



Reporting on Assignment Group Duration

- Metric Definition
  - Table: incident
  - Field: Assignment Group
  - Type: Field Value Duration
- Data:
  - Table is Instance Metric View
- Conditions:
  - Definition is Assignment Group
- Group by Value (if List)

Edit report

All > Definition = Assignment Group > Start is more than 1 hours before Resolved

Group by: Value

Stack by: --None--

□ Display data table

Aggregation: Average

Business duration(inc_business_duration)

Percentage calculation: Use Aggregation

Max number of groups: 10

□ Show Other

CONDITIONS

All of these conditions must be met

Definition | is | Assignment Group

AND

Start | is more than | 0 Hours before | Resolved

Report name: Assignment Group Average Durations

Source type: Table

Table: Incident Metric [incident_metric]

Once metrics have been defined and metric instances have been generated, you can report on Assignment Group durations by building a report against the metric_instance table. The following are important fields for reporting building:

- **Definition:** the metric definition for which this metric instance was gathered. This is Assignment Group in this example.
- **Value**: this is the value of the table field for which duration is calculated. In the case of the Assignment Group metric, the Value is the Assignment Group to which the incident is assigned. For other metrics, the value can be any value calculated by the metric.

Other useful [metric_instance] reporting fields are:

- **ID**: Identifies the specific record for which the metric is gathered, such as the specific incident.
- **Duration**: Time duration for a *Field value duration* metric.
- **Created**: Metric updates are time stamped so you can use the Created field for a Trend report.

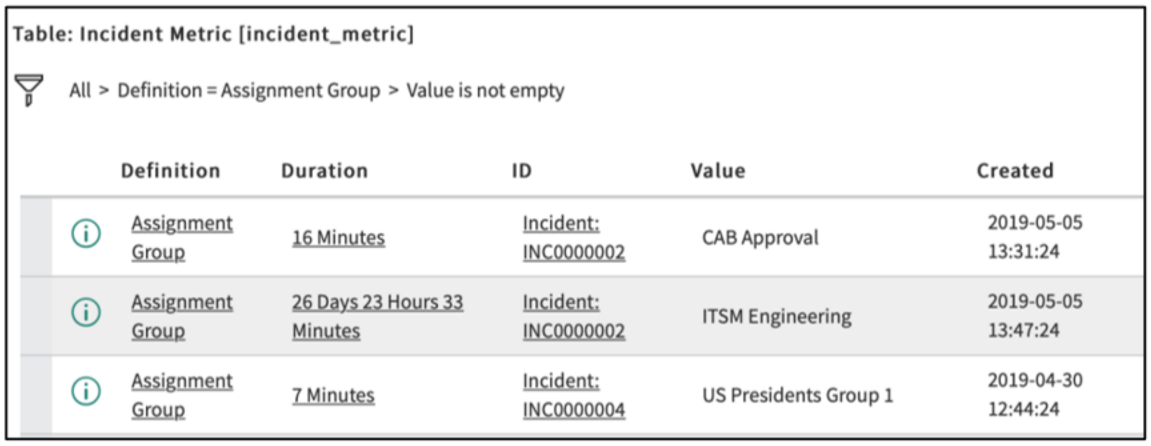
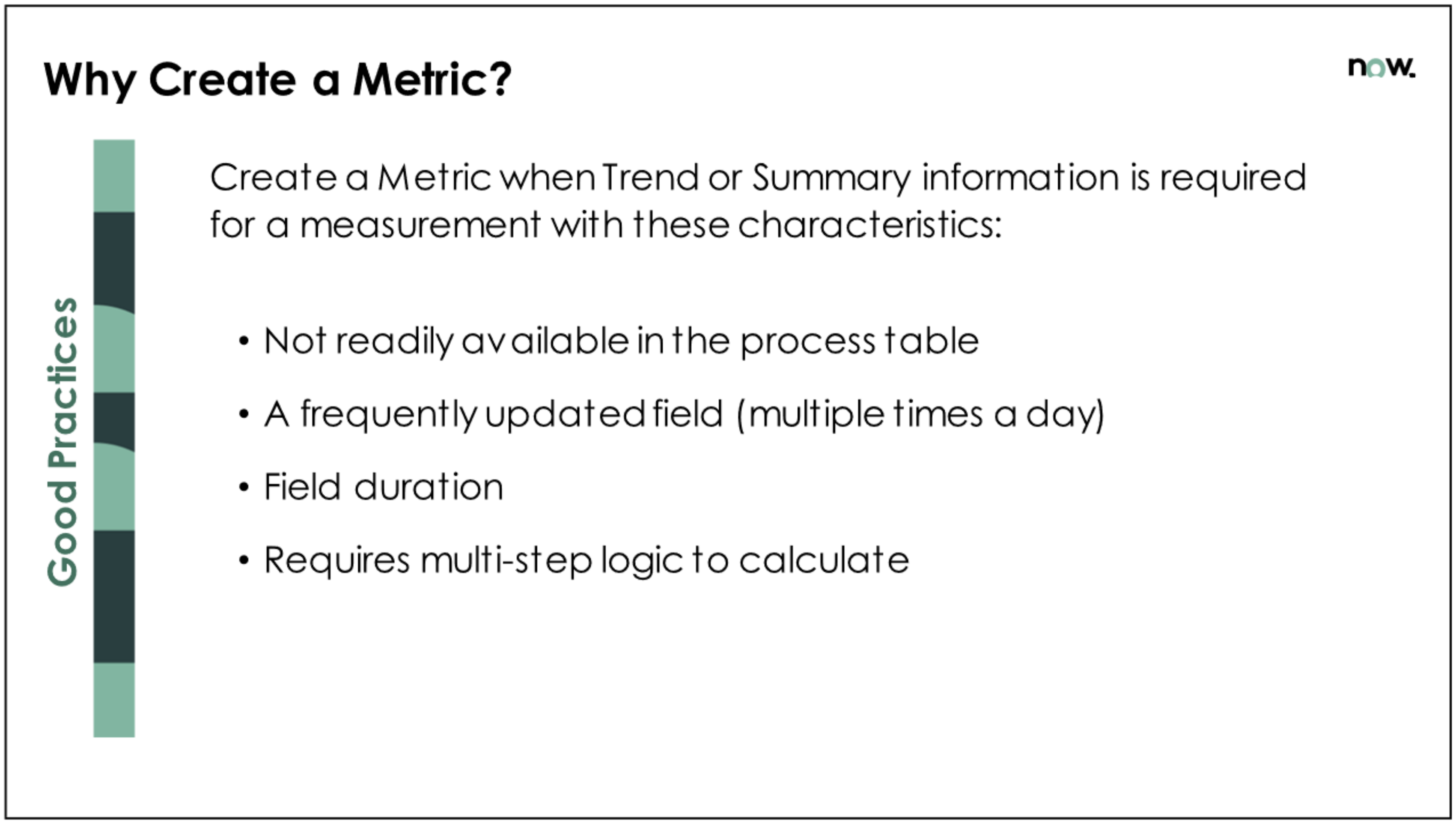


Table: Incident Metric [incident_metric]

All > Definition = Assignment Group > Value is not empty

Definition	Duration	ID	Value	Created
Assignment Group	16 Minutes	Incident: INC0000002	CAB Approval	2019-05-05 13:31:24
Assignment Group	26 Days 23 Hours 33 Minutes	Incident: INC0000002	ITSM Engineering	2019-05-05 13:47:24
Assignment Group	7 Minutes	Incident: INC0000004	US Presidents Group 1	2019-04-30 12:44:24



Why Create a Metric?

now.

Good Practices

Create a Metric when Trend or Summary information is required for a measurement with these characteristics:

- Not readily available in the process table
- A frequently updated field (multiple times a day)
- Field duration
- Requires multi-step logic to calculate

Consider the above best practices and examples when implementing new Metrics.

And here are some example Use Cases:

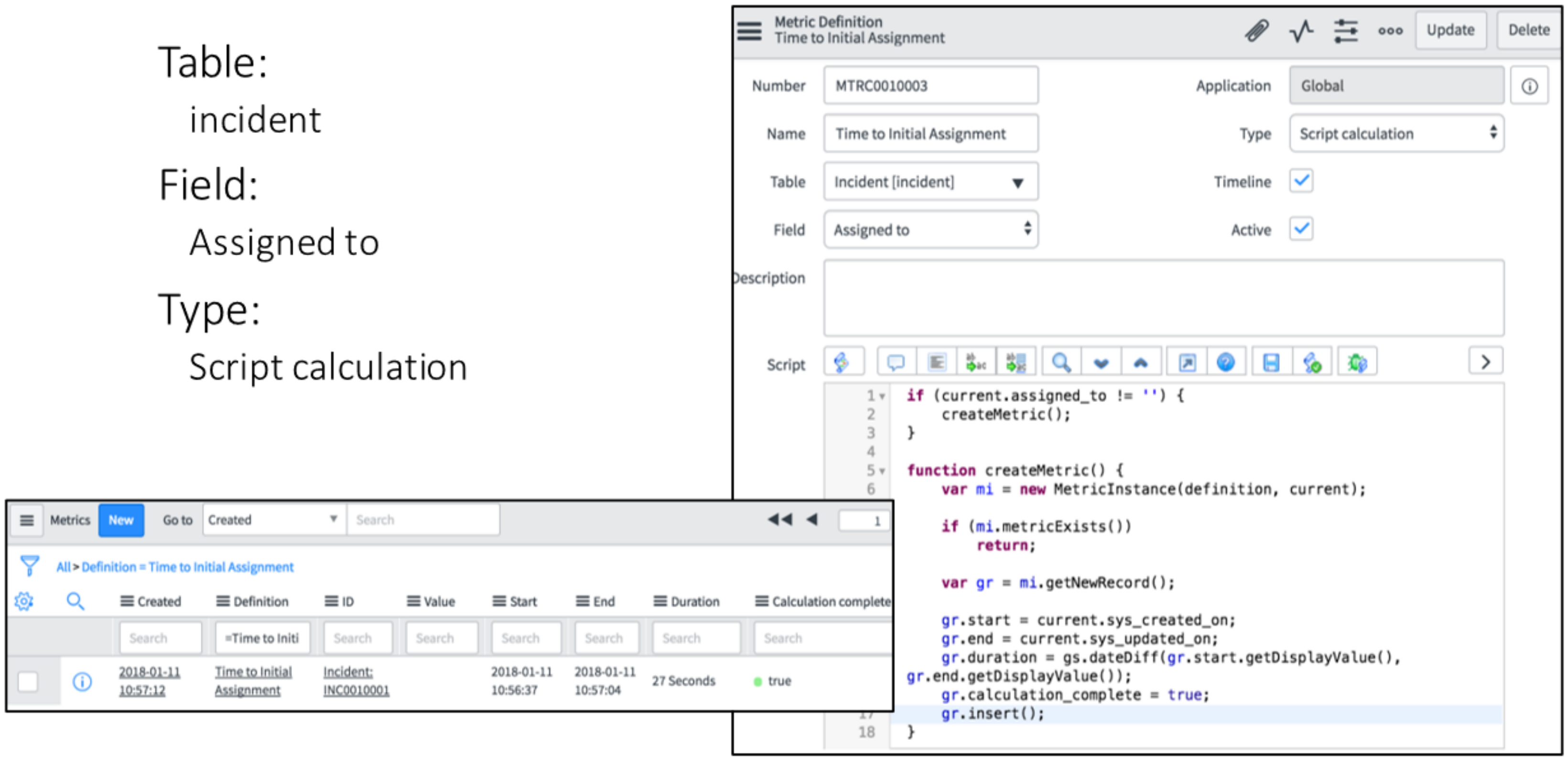
- What is the average time to obtain the first individual assignment?
- How long is the average incident in Escalation
- What is the average time incidents spend in “In Progress” state
- How long does it take, on average, to resolve an incident once assigned to Tier 3

Example Task: Measure Initial time to assign new incident

Table:
incident

Field:
Assigned to

Type:
Script calculation



Metric Definition
Time to Initial Assignment

Number	MTRC0010003	Application	Global
Name	Time to Initial Assignment	Type	Script calculation
Table	Incident [incident]	Timeline	☑
Field	Assigned to	Active	☑
Description			

Script

```
if (current.assigned_to != '') {
    createMetric();
}

function createMetric() {
    var mi = new MetricInstance(definition, current);

    if (mi.metricExists())
        return;

    var gr = mi.getNewRecord();

    gr.start = current.sys_created_on;
    gr.end = current.sys_updated_on;
    gr.duration = gs.dateDiff(gr.start.getDisplayValue(),
gr.end.getDisplayValue());
    gr.calculation_complete = true;
    gr.insert();
}
```

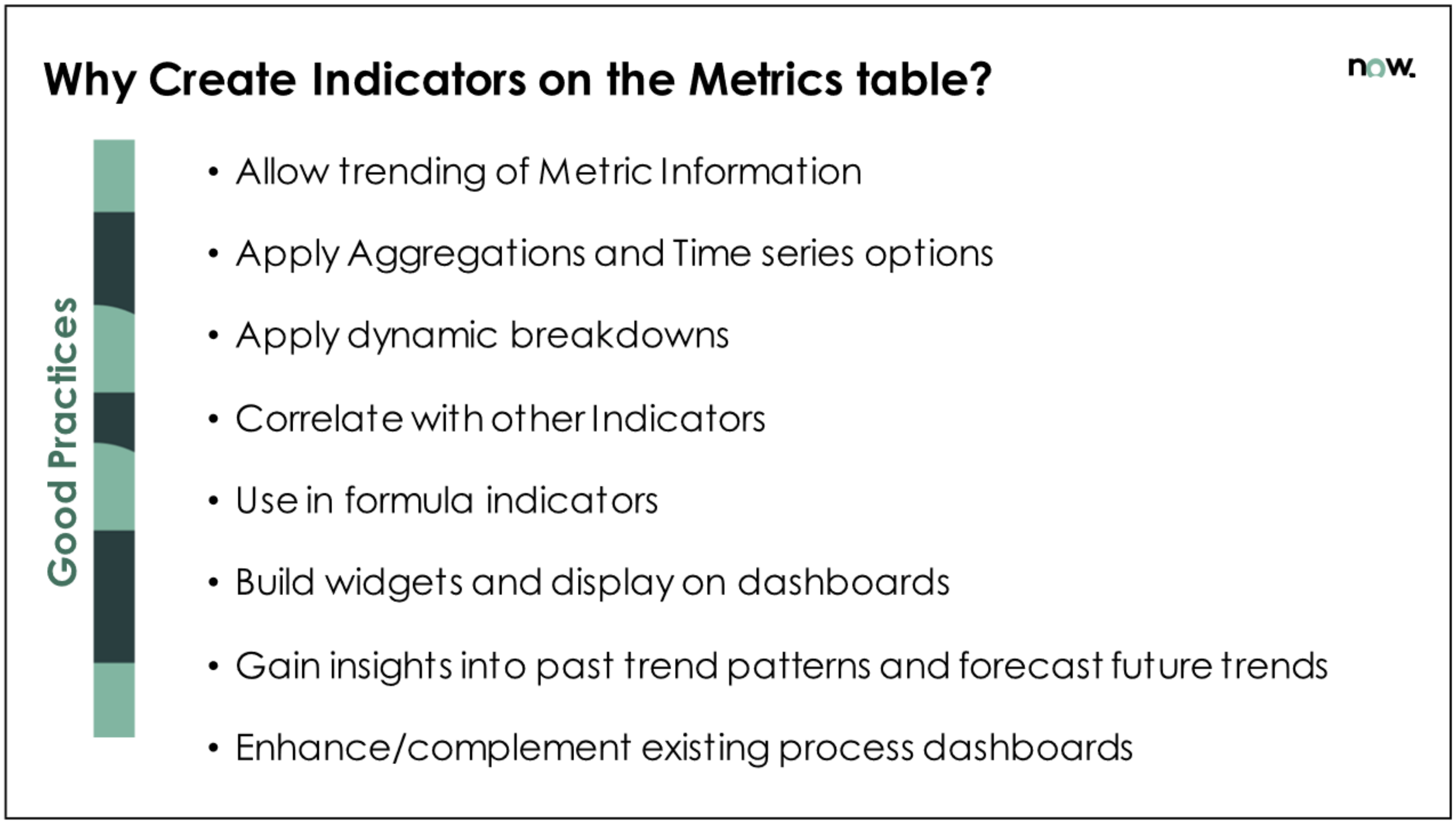
All > Definition = Time to Initial Assignment

Created	Definition	ID	Value	Start	End	Duration	Calculation complete
Search	=Time to Initi	Search	Search	Search	Search	Search	Search
2018-01-11 10:57:12	Time to Initial Assignment	Incident: INC0010001		2018-01-11 10:56:37	2018-01-11 10:57:04	27 Seconds	● true

In this example, a script is used to measure how long it takes to assign an incident. The script logic is as follows:

- When the *Assigned To* is populated, a new metric is created.
- The script calculates the new metric's duration as the time difference between the current time and the creation time of the incident record
- The scripts sets the *calculation_complete* attribute to true

For reporting purposes, you can average out the Value attribute of all metric instances with Definition = Time to Initial Assignment

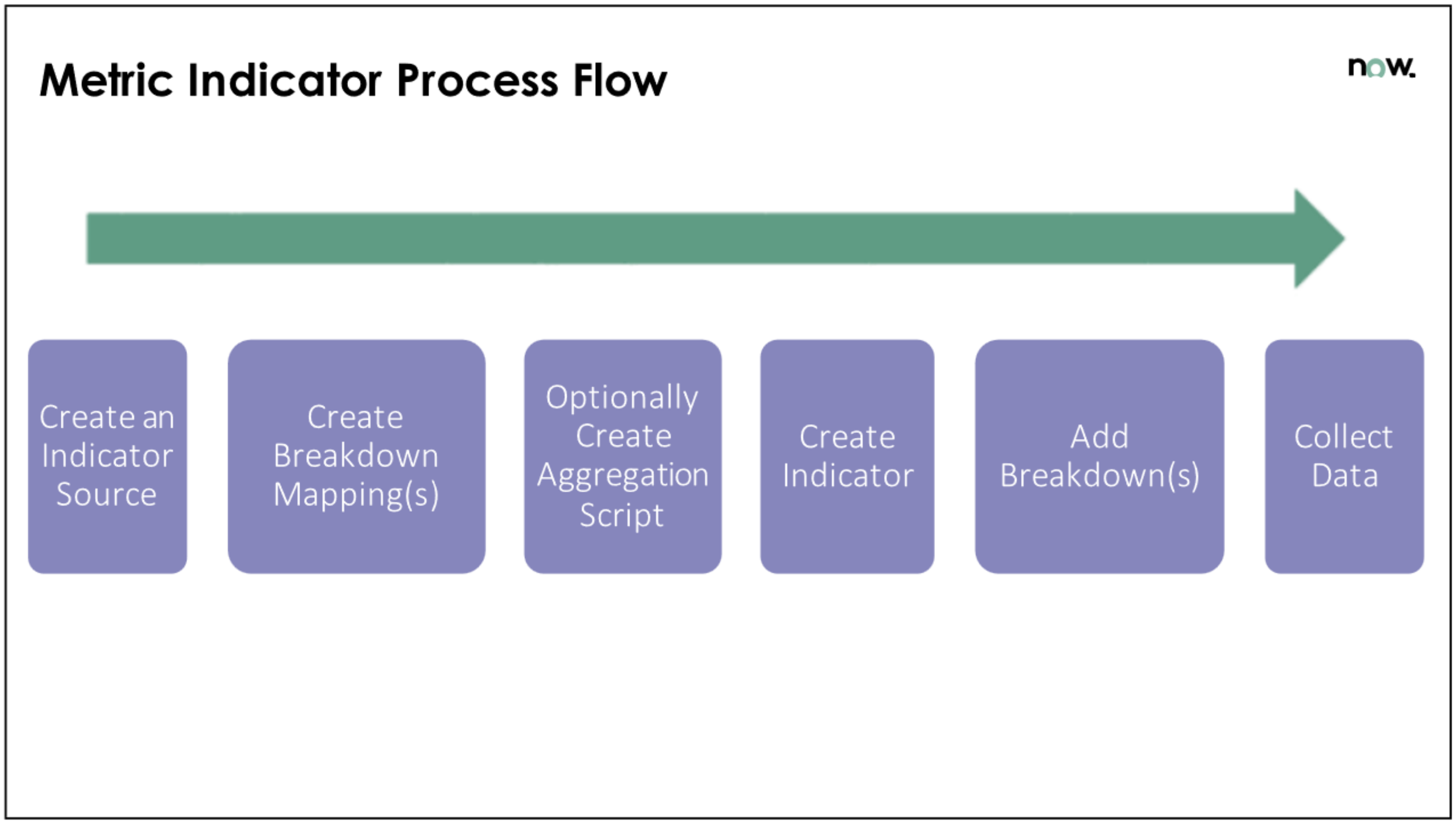


Why Create Indicators on the Metrics table?

Good Practices

- Allow trending of Metric Information
- Apply Aggregations and Time series options
- Apply dynamic breakdowns
- Correlate with other Indicators
- Use in formula indicators
- Build widgets and display on dashboards
- Gain insights into past trend patterns and forecast future trends
- Enhance/complement existing process dashboards

In addition to reporting against the metric tables, further insight can be gained by setting up Automated Indicators against the metric table source.

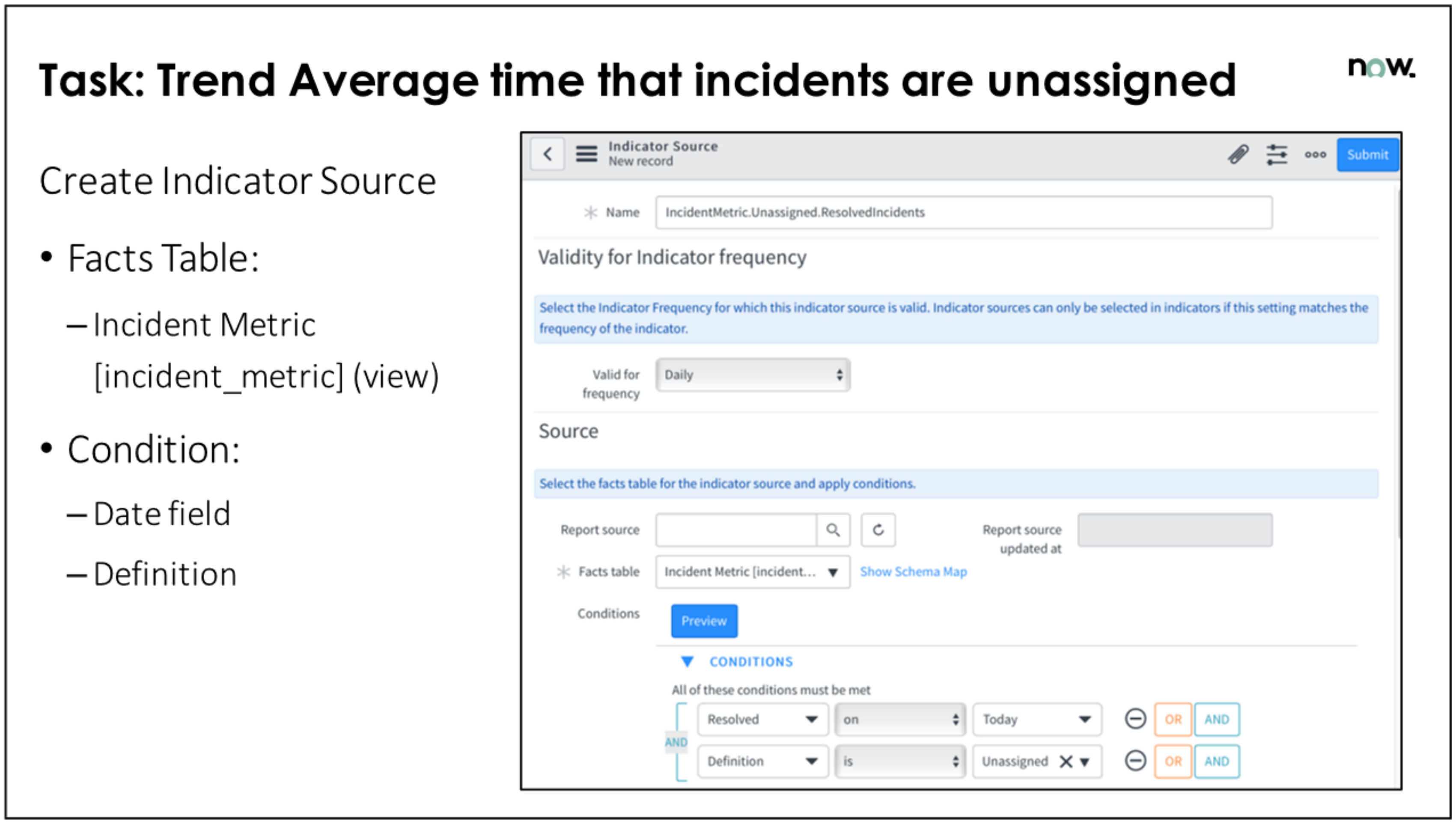


Metric Indicator Process Flow

now.
Create an Indicator Source
Create Breakdown Mapping(s)
Optionally Create Aggregation Script
Create Indicator
Add Breakdown(s)
Collect Data

These are the steps required fore creating an Automated Indicator using a Metric source:

- Use the metric source table or view to create an Indicator Source
- Create new Breakdowns or just Breakdown Mappings if the Breakdowns already exist
- To optionally aggregate the metric data in a customized fashion, create a Script
- Create the Automated Indicator and add the Breakdowns
- Collect and validate correct Widget and Analytics Hub display



Task: Trend Average time that incidents are unassigned

Create Indicator Source

- Facts Table:
  - Incident Metric [incident_metric] (view)
- Condition:
  - Date field
  - Definition

Name: IncidentMetric.Unassigned.ResolvedIncidents

Validity for Indicator frequency

Select the Indicator Frequency for which this indicator source is valid. Indicator sources can only be selected in indicators if this setting matches the frequency of the indicator.

Valid for frequency: Daily

Source

Select the facts table for the indicator source and apply conditions.

Report source:

Report source updated at:

Facts table: Incident Metric [incident... Show Schema Map

Conditions: Preview

CONDITIONS

All of these conditions must be met

Resolved on Today

AND

Definition is Unassigned

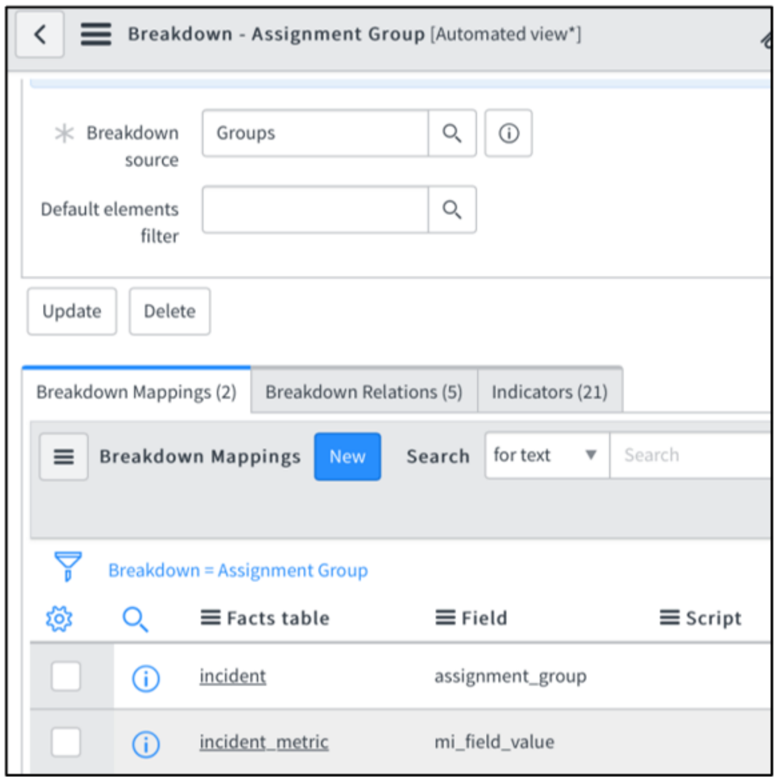
The above example implements an Indicator to track the average time that a resolved Incident is not assigned. The *Unassigned* metric definition is used calculate the duration of time until an incident is assigned to an Assignment Group and an individual Assignee.

As a fist step, you need to create an Indicator source as follows:

- The Indicator Source uses the incident_metric view as its Facts table
- The Conditions are:
  - Resolved on Today AND
  - Definition is Unassigned

Breakdown and Breakdown Mapping

- Create a Breakdown to analyze metric data by the **Field value**
- Create a Breakdown Mapping for the desired Process Metric Table



Breakdown - Assignment Group [Automated view*]

Breakdown source	Groups
Default elements filter	

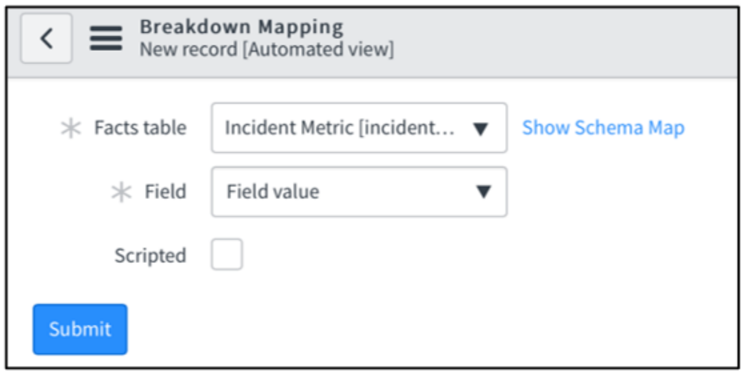
Update Delete

Breakdown Mappings (2) | Breakdown Relations (5) | Indicators (21)

Breakdown Mappings New **Search** for text Search

Breakdown = Assignment Group

Facts table	Field	Script
incident	assignment_group	
incident_metric	mi_field_value	



Breakdown Mapping
New record [Automated view]

Facts table	Incident Metric [incident…
Field	Field value
Scripted	□

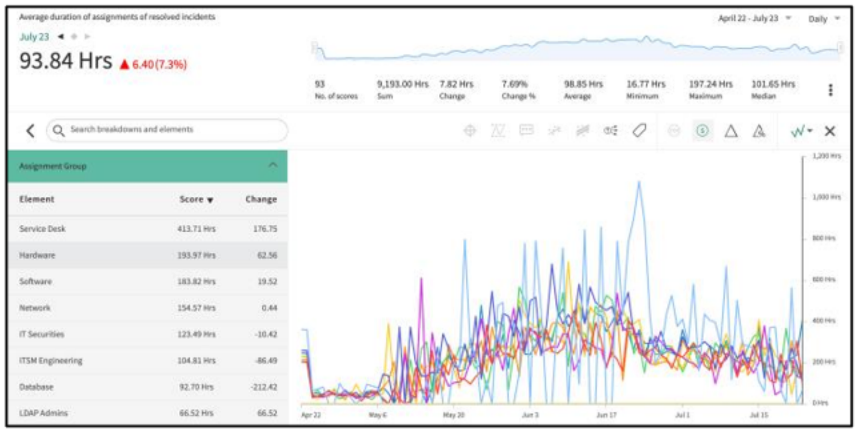
Show Schema Map

Submit

The Unassigned metric sets the **Value** attribute of the Metric Instance to the newly assigned Assignment Group. Therefore, a breakdown is needed to be able to navigate a Scorecard or a Widget by Assignment group. Since the Assignment group breakdown already exists out of baseline, only a new breakdown mapping is needed to describe which incident_metric field contains the Assignment group. This is the mi_field_value field.

New Automated Indicator

- Use metric-based Indicator Source
- Add Breakdowns
- Add to a Collection Job



Average duration of assignments of resolved incidents

April 22 - July 23 · Daily

July 23

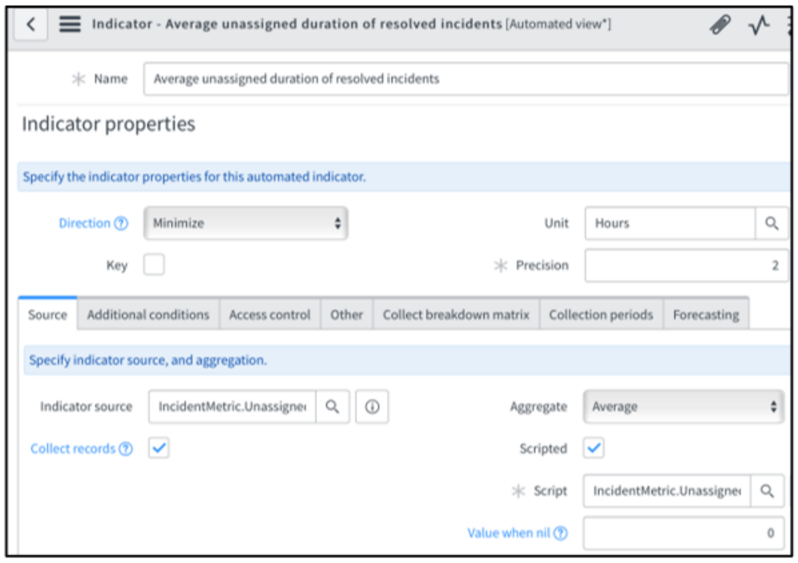
93.84 Hrs ▲ 6.40 (7.3%)

93	9,193.00 Hrs	7.82 Hrs	7.69%	98.85 Hrs	16.77 Hrs	197.24 Hrs	101.65 Hrs
No. of scores	Sum	Change	Change %	Average	Minimum	Maximum	Median

Search breakdowns and elements

Assignment Group

Element	Score ▼	Change
Service Desk	413.71 Hrs	176.75
Hardware	193.97 Hrs	62.56
Software	183.82 Hrs	19.52
Network	154.57 Hrs	0.44
IT Securities	123.49 Hrs	-10.42
ITSM Engineering	104.81 Hrs	-86.49
Database	92.70 Hrs	-212.42
LDAP Admins	66.52 Hrs	66.52



Indicator - Average unassigned duration of resolved incidents [Automated view*]

* Name: Average unassigned duration of resolved incidents

Indicator properties

Specify the indicator properties for this automated indicator.

Direction: Minimize

Unit: Hours

Key: □

* Precision: 2

Specify indicator source, and aggregation.

Indicator source: IncidentMetric.Unassigne

Aggregate: Average

Collect records: ☑

Scripted: ☑

* Script: IncidentMetric.Unassigne

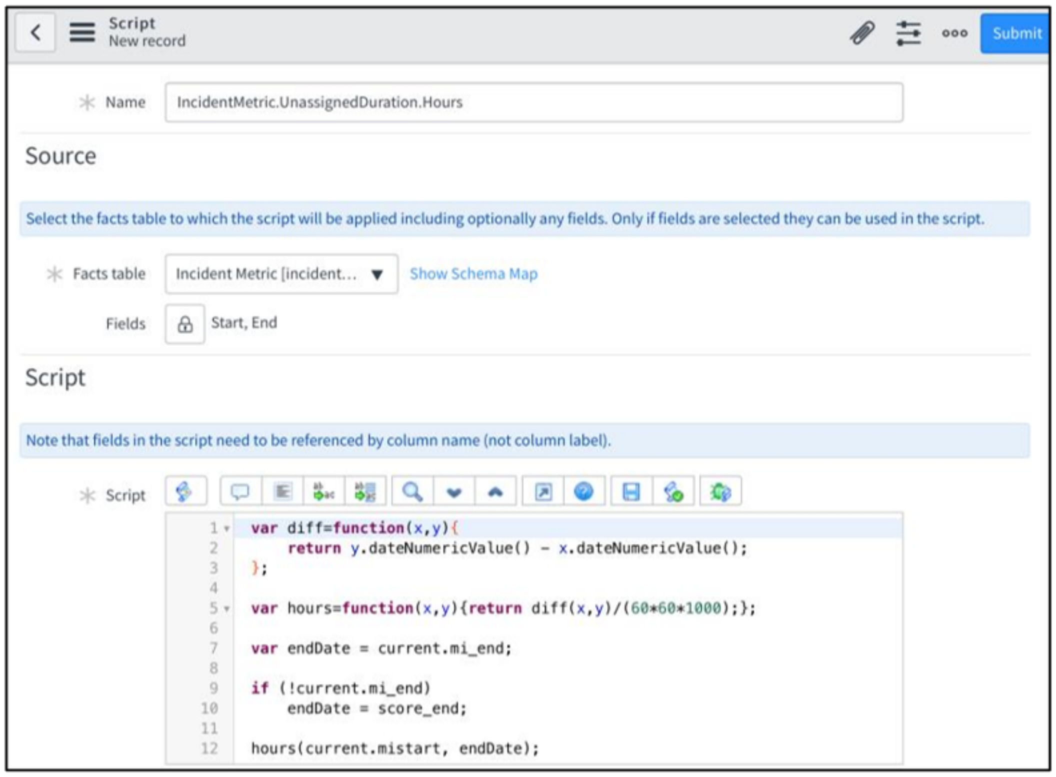
Value when nil: 0

- Set the Aggregate to Average Duration OR
- Use Scripted Aggregation to obtain the Duration in the desired format

To create an automated Indicator, use the metric-based Indicator source you created and select the desired Aggregate. In this example, we are using an Aggregation Script to present the durations in the desired format. But you can also simply set the Aggregate to **Average** and the Field to **Duration (mi_duration).**

Indicator Aggregation Script

- Facts table
  - Incident Metric
- Fields
  - Start, End
- Script
  - Return the difference between Metric Start and Metric End in Hours



Name: IncidentMetric.UnassignedDuration.Hours

Source

Select the facts table to which the script will be applied including optionally any fields. Only if fields are selected they can be used in the script.

Facts table: Incident Metric [incident... ▼ Show Schema Map

Fields: Start, End

Script

Note that fields in the script need to be referenced by column name (not column label).

Script:

```
var diff=function(x,y){
    return y.dateNumericValue() - x.dateNumericValue();
};

var hours=function(x,y){return diff(x,y)/(60*60*1000);};

var endDate = current.mi_end;

if (!current.mi_end)
    endDate = score_end;

hours(current.mistart, endDate);
```

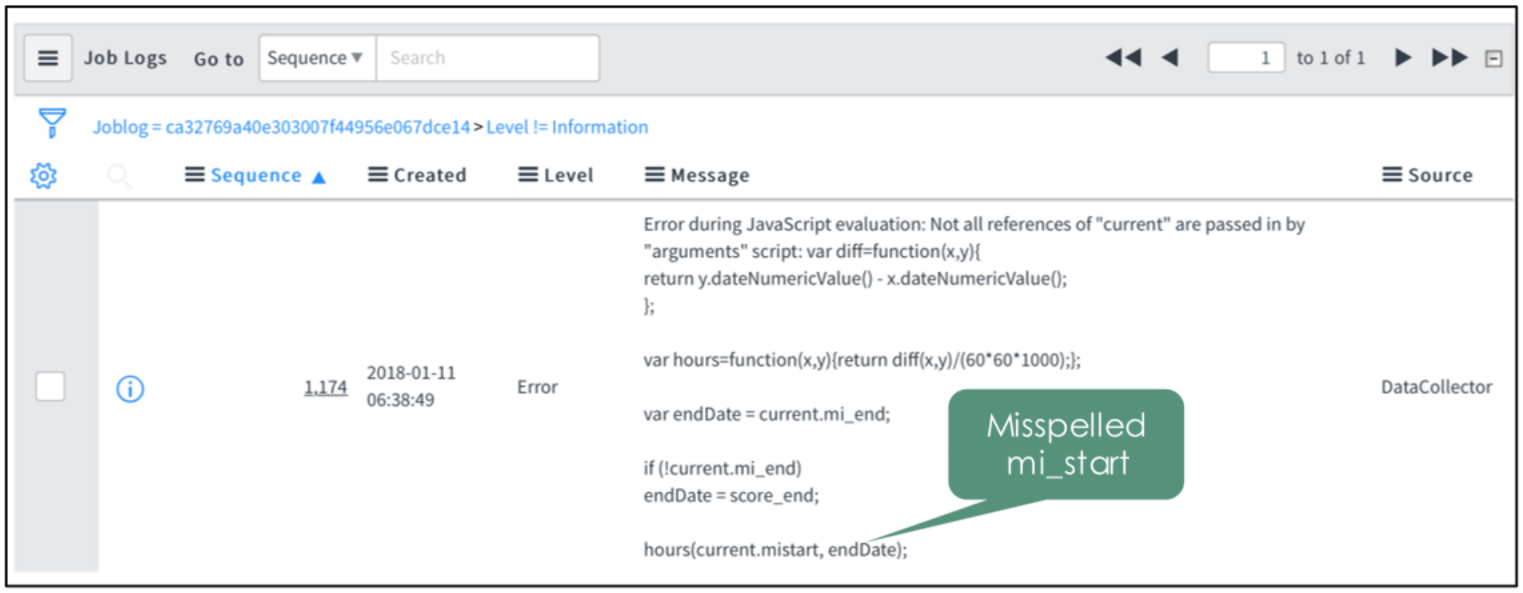
The Aggregation Script returns the difference between the Start and End attributes of an incident metric instance. In addition to calculating the difference between metric end and metric start, the script also formats the result from milliseconds to Hours.

If the End date for a metric instance does not exist because the incident is still open, the **Collection** date (score_end) is used. This makes the script fit to be used for open process records as well.

Note that the mi_start and mi_end variables are automatically available to all Performance Analytics scripts.

Job Errors

Script Errors: Error during JavaScript evaluation: Not all references of "current" are passed in by "arguments"



Joblog = ca32769a40e303007f44956e067dce14 > Level != Information

	Sequence ▲	Created	Level	Message	Source

□	1,174	2018-01-11 06:38:49	Error	Error during JavaScript evaluation: Not all references of "current" are passed in by "arguments" script: var diff=function(x,y){
				return y.dateNumericValue() - x.dateNumericValue();
				};
				
				var hours=function(x,y){return diff(x,y)/(60*60*1000);};
				
				var endDate = current.mi_end;
				
				if (!current.mi_end)
				endDate = score_end;
				
				hours(current.mistart, endDate);
					DataCollector

Misspelled mi_start

Use caution when implementing scripted aggregations and in general using script logic. Always check the Collection log for any error messages resulting from the processing of JavaScript.

Metric Analytics Lab

Lab 7.1: Metric Reporting and Trending

Metric Reporting and Trending

Lab
7.1
⌛25m

Lab Objectives

The Incident Process Manager need to measure the effectiveness of the incident handling process by reporting on assignment durations and trending the *Average duration of assignments broken down by Assignment Groups*

This lab will show you how to do the following:

- Report on the average time an incident spends within certain assignment groups
- Create an indicator based on a Metric View and a Metric Definition

> ***Dependency Note** – These two configurations capturing assignment statistics already exist:*
> **incident_metric view** – database view that stores metric instances
> **Assignment Group Metric Definition** – business logic to generate metric instances

Incident Assignment Durations Reporting

A. Current Incident Duration Tracking

Assignment Group Metric Definition

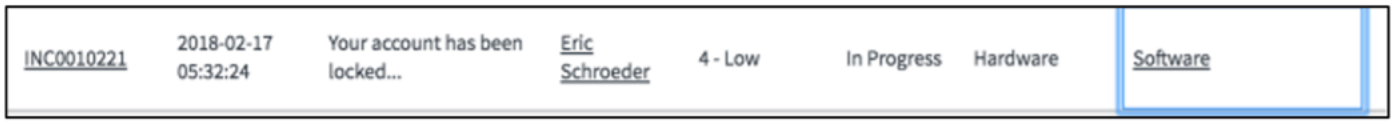
1. Navigate to the **Metrics > Definitions** module.

2. Review the **Assignment Group (MTR0000006)** metric definition.

> ***Note:** This metric tracks how long an Incident is assigned to a specific Assignment group.*

3. To test the metric logic, navigate to **Incident > All**.

4. Select Incident **INC0010221** and change its **Assignment Group** to **Software**.



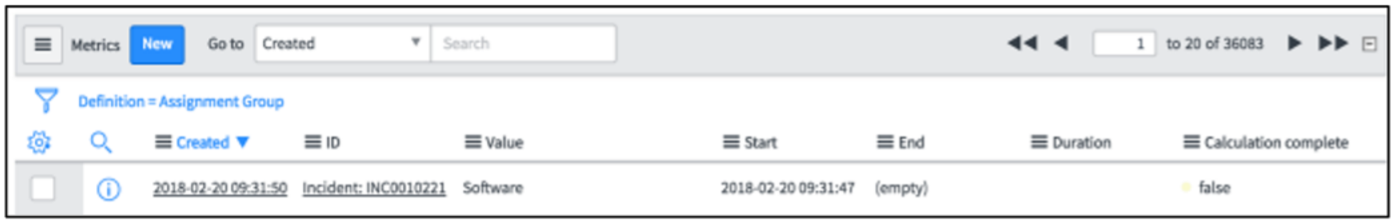
INC0010221	2018-02-17 05:32:24	Your account has been locked...	Eric Schroeder	4 - Low	In Progress	Hardware	Software

> ***Tip:** Any other incident with a non-empty Assignment Group can also be used.*

5. Navigate back to **Metrics > Definitions** and open the **Assignment Group** definition.

6. Sort the **Metrics** Related List by **Created**.

7. Confirm that a new metric instance has been created for the incident you modified.



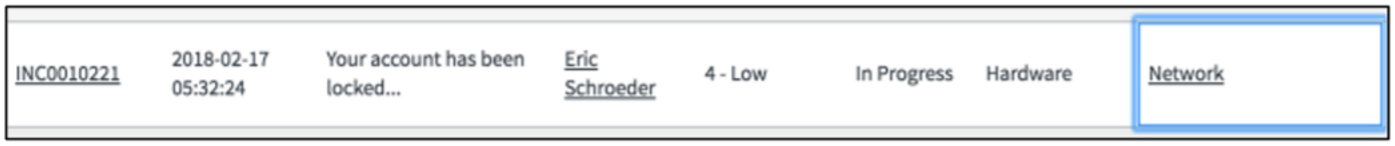
Definition = Assignment Group

Created ▼	ID	Value	Start	End	Duration	Calculation complete
2018-02-20 09:31:50	Incident: INC0010221	Software	2018-02-20 09:31:47	(empty)		false

Question*: Why is the* ***End*** *attribute of the new metric instance empty?*

8. Navigate to **Incident > All** and search for **INC0010221** (or the one modified earlier).

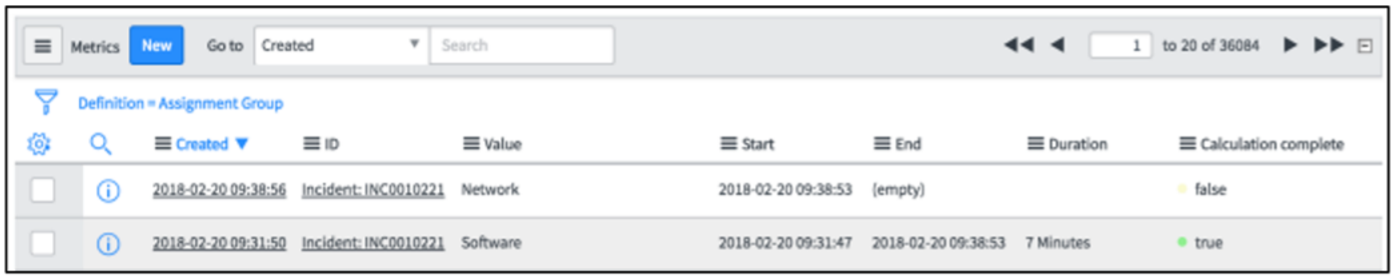
9. Change the **Assignment group** to **Network**.



INC0010221	2018-02-17 05:32:24	Your account has been locked...	Eric Schroeder	4 - Low	In Progress	Hardware	Network

Question: *What metric events do you think will be triggered? And what metric instances do you expect to be generated in the metric_instance table?*

10. Navigate to the **Metrics** related list of the **Assignment Group** metric.



Definition = Assignment Group

Created ▼	ID	Value	Start	End	Duration	Calculation complete
2018-02-20 09:38:56	Incident: INC0010221	Network	2018-02-20 09:38:53	(empty)		false
2018-02-20 09:31:50	Incident: INC0010221	Software	2018-02-20 09:31:47	2018-02-20 09:38:53	7 Minutes	true

Questions: *How long was the duration of the Software assignment?*

Which group assignment duration for the INC0010221 incident is currently being tracked?

B. Metric Reporting

Report Designer

1. Navigate to **Reports > Create New** and create a new report as follows:

   Name: **Assignment Group Average Durations**
   Table: **incident_metric**

Tip: The incident_metric table is a database view that combines the Metric Definition table, Metric Instance table, and Incident table. It reports on all metrics applied to incident records.

Type: **Horizontal Bar**
Group by: **Value**

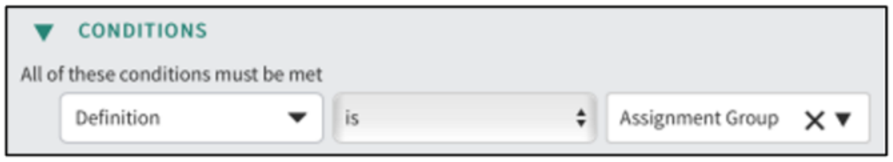
***Tip:** Value holds the Assignment Group for the Assignment Group Metric Definition.*

Stack by: **None**
Aggregation: **Average**
Aggregation Field: **Duration (mi_duration)**
Show Other: **uncheck**

***Tip:** Duration represents the length of time the incident was assigned to a specific group.*

2. Add this **Condition** to return only metrics for the Assignment Group definition:

   Field: **Definition**
   Operator: **Is**
   Value: **Assignment Group**

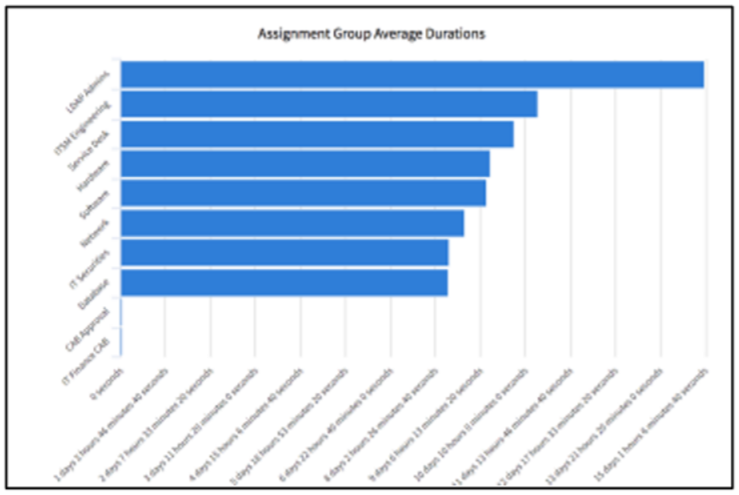


CONDITIONS

All of these conditions must be met

Definition	is	Assignment Group

3. Save and view the report. Confirm that you see durations by assignment group.



Incident Metric Duration Trending

C. Create Indicator Source

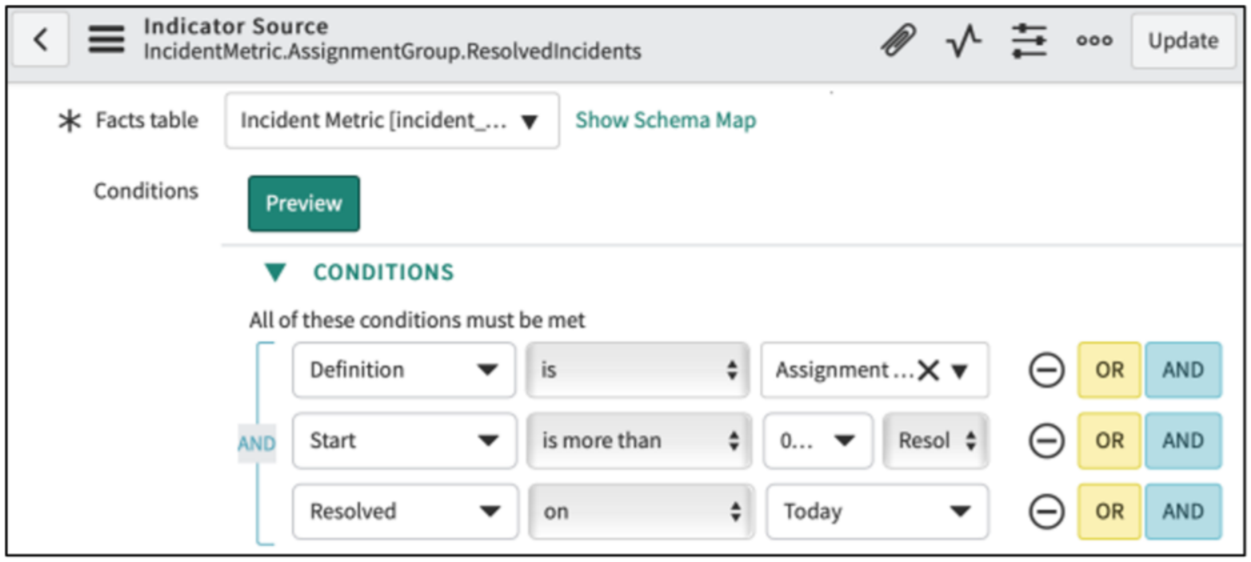
Using Database View as Indicator Source

1. Navigate to **Performance Analytics > Sources > Indicator Sources**.

2. Create a new indicator source to capture Assignment Group metrics for Resolved incidents as follows:

   Name: **IncidentMetric.AssignmentGroup.ResolvedIncidents**
   Valid for frequency: **Daily**
   Facts table**: Incident Metric [incident_metric]**
   Conditions:

Resolved	**on**	**Today** AND	
Definition	**is**	**Assignment Group** AND	
Start	**is more than**	**0 Hours before**	**Resolved**



Indicator Source
IncidentMetric.AssignmentGroup.ResolvedIncidents

Facts table: Incident Metric [incident_...

Conditions

CONDITIONS

All of these conditions must be met

	Definition	is	Assignment ...	
AND	Start	is more than	0...	Resol
	Resolved	on	Today	

__Note:__ The Conditions exclude assignment changes that occur after the incident is resolved.

3. Click **Submit**.

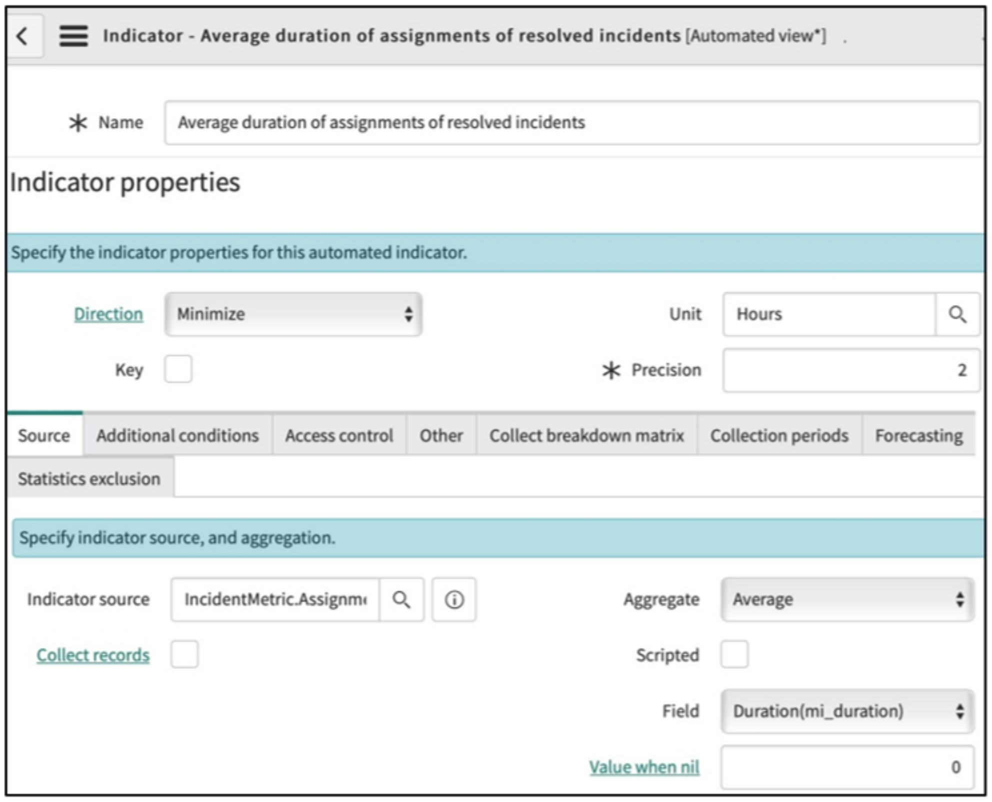
D. Create New Indicator

1. Navigate to **Performance Analytics > Indicators > Automated Indicators**.

2. Click **New** to create an indicator for incident assignments duration:

   Name: **Average duration of assignments of resolved incidents**
   Frequency: **Daily**
   Direction: **Minimize**
   Unit: **Hours**
   Precision: **2**
   Indicator source: **IncidentMetric.AssignmentGroup.ResolvedIncidents**
   Collect records: **UNCHECKED**

Aggregate: **Average**
Field: **Duration (mi_duration)**
Value when nil: **0**



Indicator - Average duration of assignments of resolved incidents [Automated view*]

* Name: Average duration of assignments of resolved incidents

Indicator properties

Specify the indicator properties for this automated indicator.

Direction: Minimize
Unit: Hours
Key: □
* Precision: 2

Specify indicator source, and aggregation.

Indicator source: IncidentMetric.Assignm
Aggregate: Average
Collect records: □
Scripted: □
Field: Duration(mi_duration)
Value when nil: 0

3. Check **Publish on Analytics Hub** under **Access control**.

4. **Save** the Indicator to stay on the Indicator form.

Assignment Group Breakdown Mapping

1. Click the **Manage Breakdowns** button.

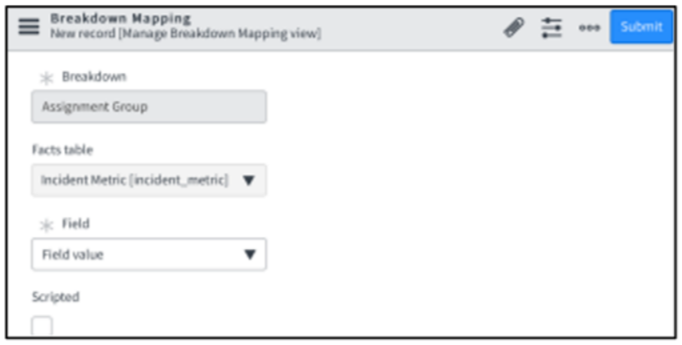
2. Drag the **Assignment Group** breakdown into the **Selected Breakdowns** column.

3. Fill out the **Breakdown Mapping** form as follows:

   Breakdown: **Assignment Group** (already populated)

   Facts table: **Incident Metric [incident_metric]** (already populated)

   Field: **Field value**



Breakdown Mapping
New record [Manage Breakdown Mapping view]

* Breakdown: Assignment Group
Facts table: Incident Metric [incident_metric]
* Field: Field value
Scripted: □

4. Return to the Indicator and verify that you have added a new Breakdown.

Data Collection

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Create a historic scores collection for the **Average duration of assignments of resolved incidents indicator** for the last 3 months.

3. Execute the job.

4. When the collection finishes, verify that the collection log looks similar to this:

   Inserts: ~**2,500**   Warnings: **0**   Errors: **0**

> ***Note***: *The inserts may differ slightly, but warnings and errors should be 0.*

Dashboard Visualization

1. Navigate to the **Customer Success Advocate** dashboard.

2. View the **Incident Assignments** tab.

3. Add a new **Breakdown** Widget as follows:

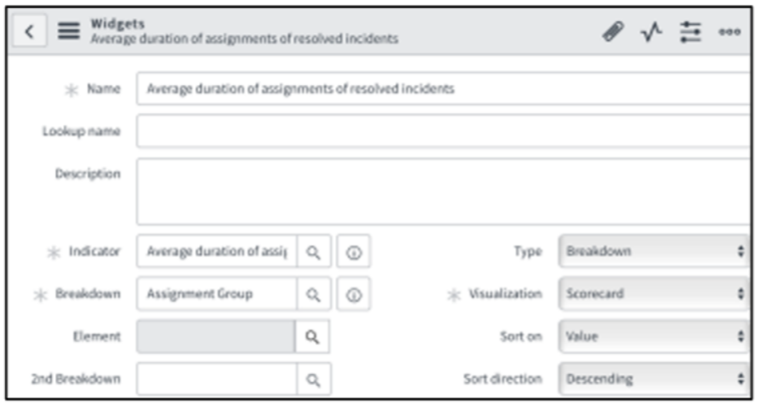
   Type: **Breakdown**

   Name: **Average duration of assignments of resolved incidents**

   Indicator: **Average duration of assignments of resolved incidents**

   Breakdown: **Assignment Group**

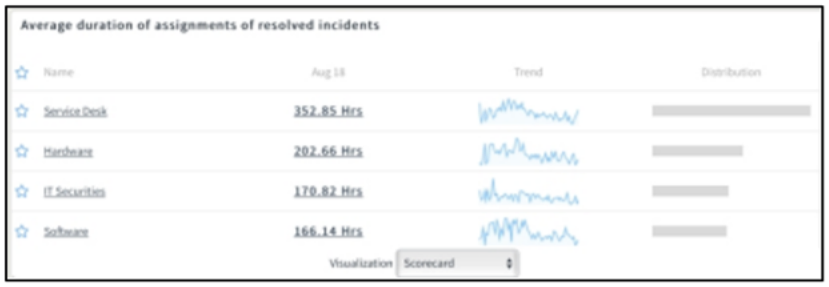
   Visualization: **Scorecard**



Widgets
Average duration of assignments of resolved incidents

✳ Name	Average duration of assignments of resolved incidents		
Lookup name			
Description			
✳ Indicator	Average duration of assi	Type	Breakdown
✳ Breakdown	Assignment Group	✳ Visualization	Scorecard
Element		Sort on	Value
2nd Breakdown		Sort direction	Descending

4. Verify that your new Breakdown widget look similar to this:

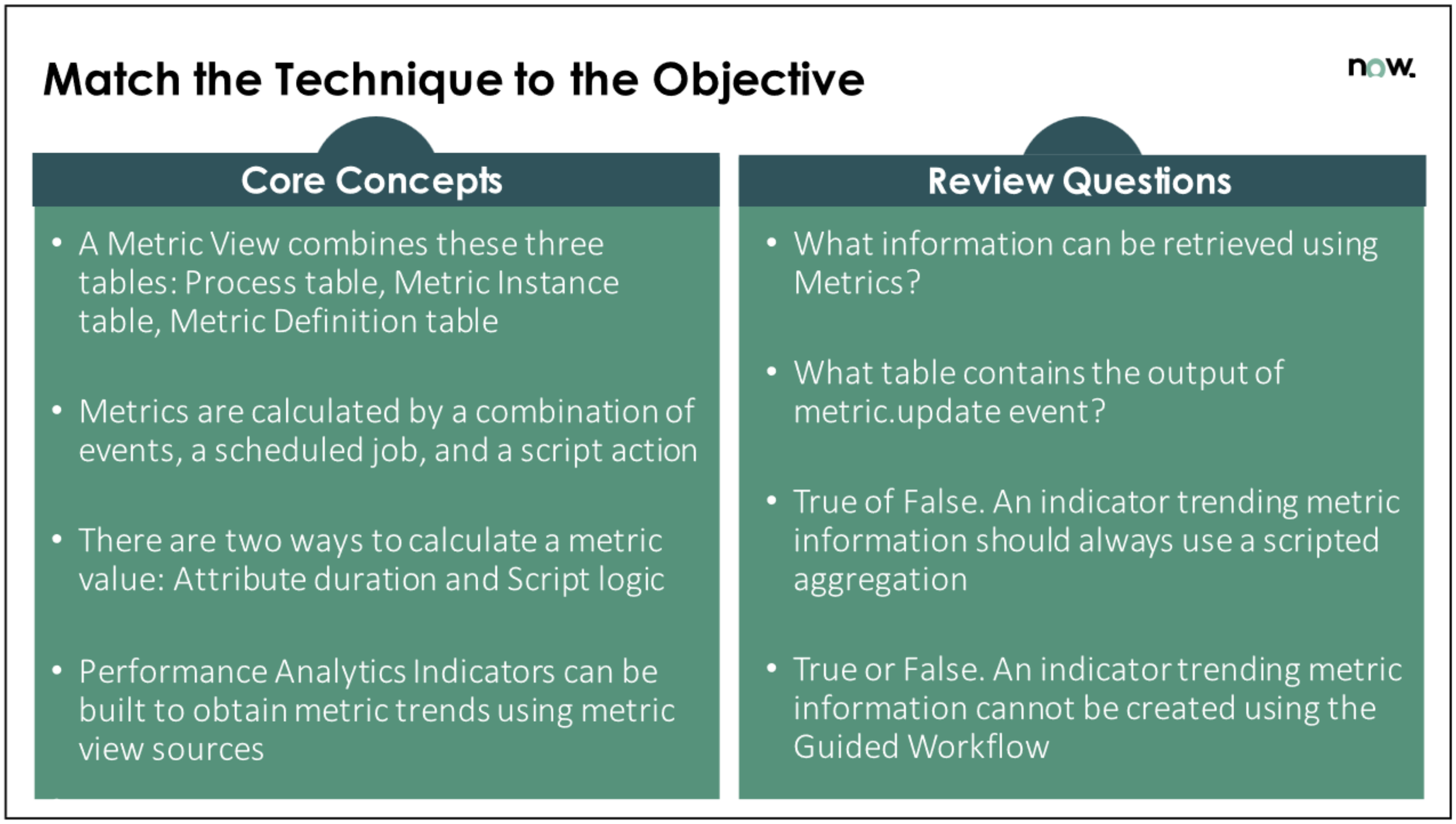


Average duration of assignments of resolved incidents

☆ Name	Aug 18	Trend	Distribution
☆ Service Desk	352.85 Hrs		
☆ Hardware	202.66 Hrs		
☆ IT Securities	170.82 Hrs		
☆ Software	166.14 Hrs		

Visualization: Scorecard

You have now completed the Metric Reporting and Trending lab.



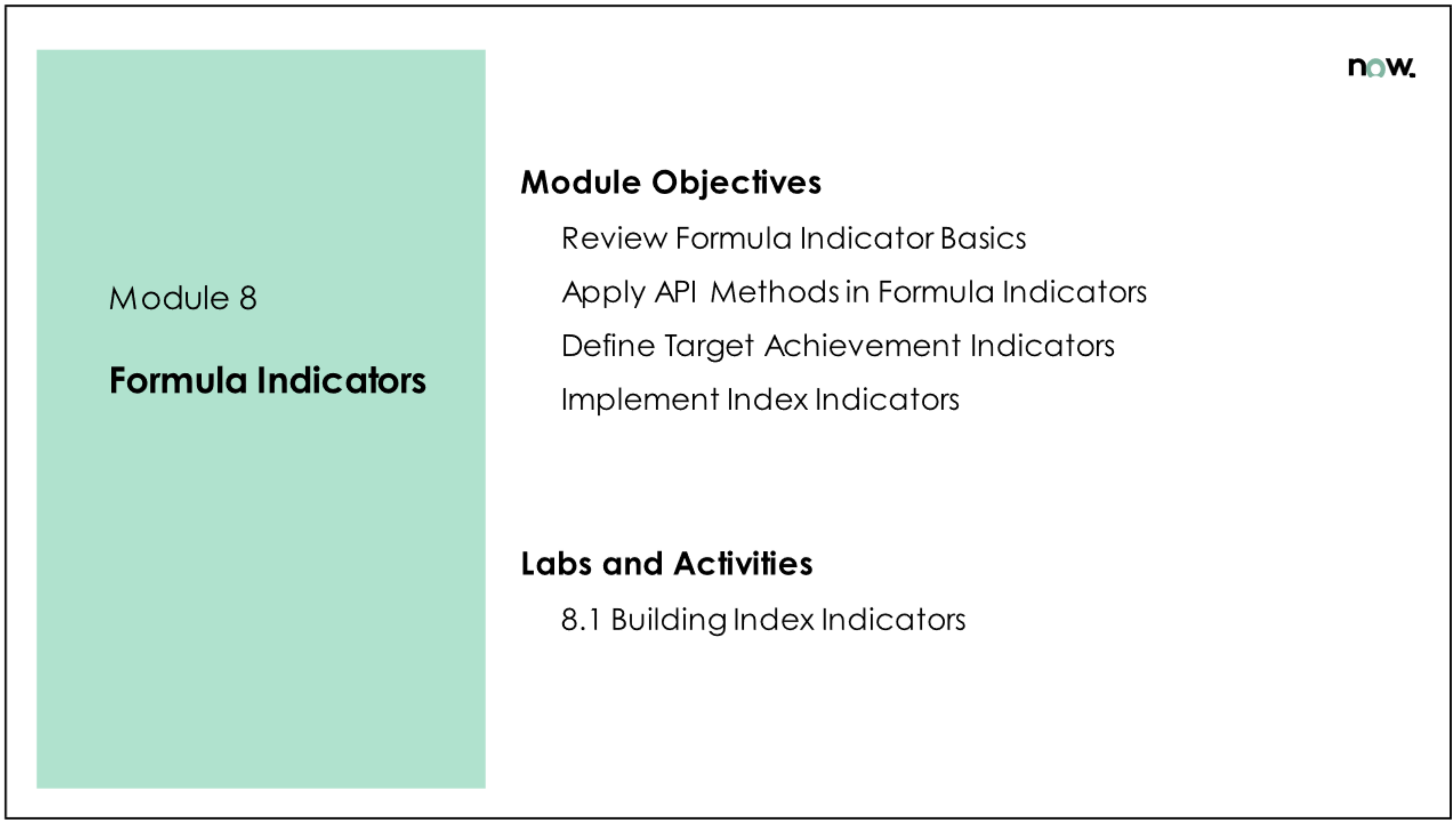
Match the Technique to the Objective

Core Concepts

- A Metric View combines these three tables: Process table, Metric Instance table, Metric Definition table
- Metrics are calculated by a combination of events, a scheduled job, and a script action
- There are two ways to calculate a metric value: Attribute duration and Script logic
- Performance Analytics Indicators can be built to obtain metric trends using metric view sources

Review Questions

- What information can be retrieved using Metrics?
- What table contains the output of metric.update event?
- True of False. An indicator trending metric information should always use a scripted aggregation
- True or False. An indicator trending metric information cannot be created using the Guided Workflow



Module 8

Formula Indicators

Module Objectives

Review Formula Indicator Basics

Apply API Methods in Formula Indicators

Define Target Achievement Indicators

Implement Index Indicators

Labs and Activities

8.1 Building Index Indicators

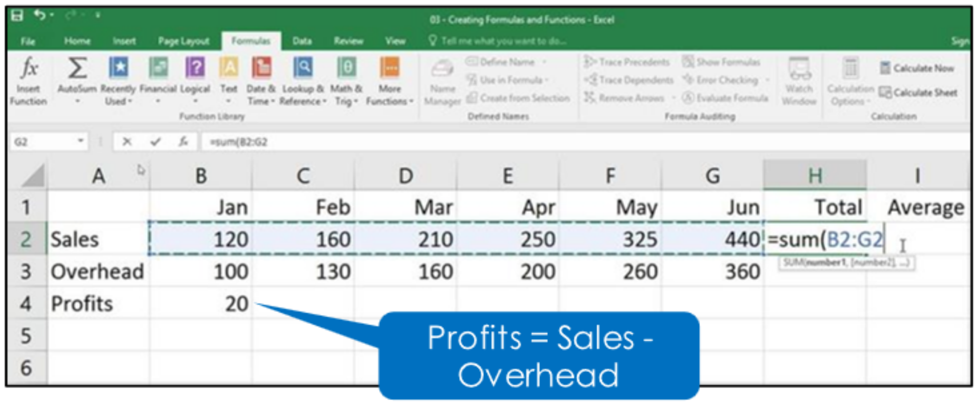
In this module, you explore techniques and capabilities to go beyond the raw score trends and gain deeper insight into your data. The following capabilities let you present scores differently and improve your ability to analyze and understand the underlying business process data:

- Formula Indicators – Calculated Indicator trends that combine multiple indicators and business logic to derive new metrics.
- Index Indicators – A special case of formula indicators that visualize the performance of several Indicators.

Formula Indicator Basics

Why

Obtain new metrics from existing Indicators



G2 =sum(B2:G2

	A	B	C	D	E	F	G	H	I
1		Jan	Feb	Mar	Apr	May	Jun	Total	Average
2	Sales	120	160	210	250	325	440	=sum(B2:G2	
3	Overhead	100	130	160	200	260	360		
4	Profits	20							
5									
6									

Profits = Sales - Overhead

What

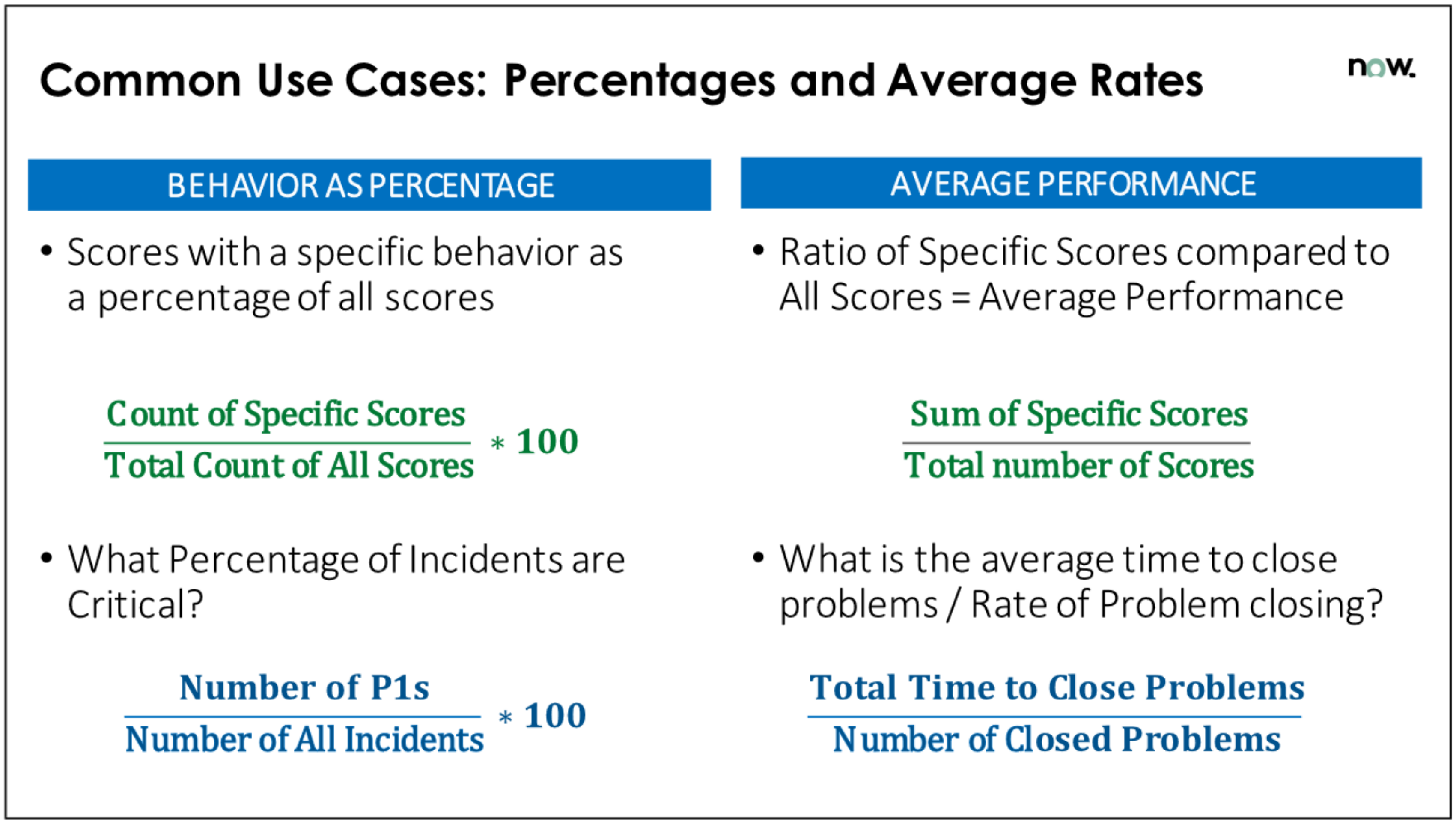
JavaScript statement referencing Indicators and using APIs

How

- The Formula is processed at Indicator view time
- The Formula output is not persisted
- Formula calculations may impact performance

Formulas calculate new metrics from existing Indicators. The calculation involves mathematical operations using JavaScript syntax. Formula indicators are calculated at display time (in Scorecards and Widgets) and their value is not persisted. Because of the fact that formulas are executed for every data point in a trend, complex calculations and long running scripts can potentially slow down Scorecard and Dashboard loading.

Note: There is some overlap between what a Performance Analytics Script can do and a Performance Analytics Formula Indicators. The main difference is that a Performance Analytics Script is run at collection time and the results are stored in the database. PA Formula Indicators are calculated at display time and the results are not materialized.



Common Use Cases: Percentages and Average Rates

now.

BEHAVIOR AS PERCENTAGE

- Scores with a specific behavior as a percentage of all scores

$$\frac{\text{Count of Specific Scores}}{\text{Total Count of All Scores}} * 100$$

- What Percentage of Incidents are Critical?

$$\frac{\text{Number of P1s}}{\text{Number of All Incidents}} * 100$$

AVERAGE PERFORMANCE

- Ratio of Specific Scores compared to All Scores = Average Performance

$$\frac{\text{Sum of Specific Scores}}{\text{Total number of Scores}}$$

- What is the average time to close problems / Rate of Problem closing?

$$\frac{\text{Total Time to Close Problems}}{\text{Number of Closed Problems}}$$

The most common use case for a Formula Indicator is to visualize the performance of a subcategory of scores, such as:

- Percentage Incidents with Surveys
- Percentage of Incidents Resolved by the First Assignment Group
- Percentage Incidents with Breached SLAs

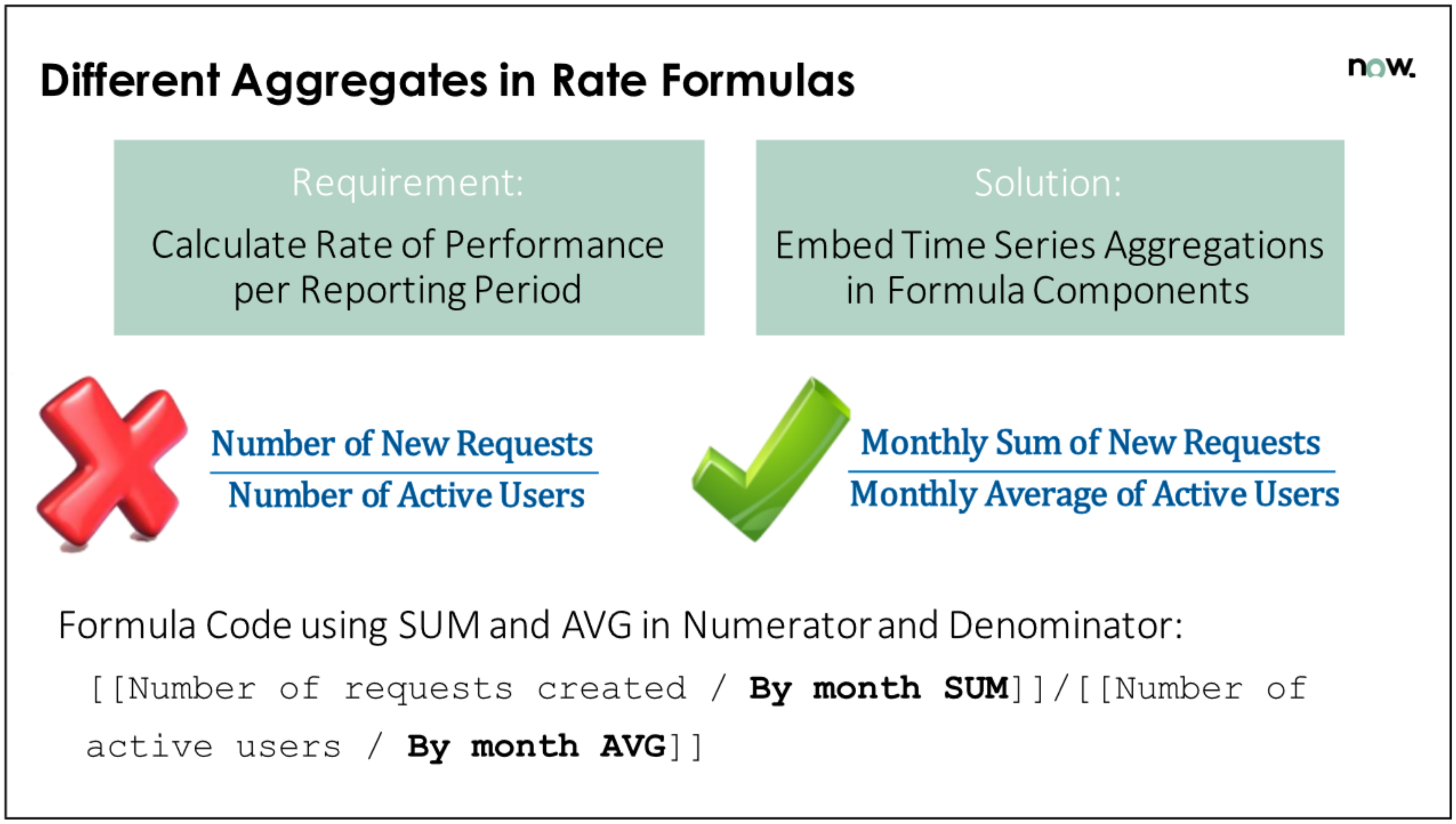
The basic syntax is the Ration of Specific and All Scores multiplied by 100. Formula indictors that calculate fractions as percentage should have their Unit set to Percent.

Calculating the Average Performance is a variation of the Percentage calculation where the result is interpreted as a fraction or a proportion. A common objective is to derive an average or a unit rate such as:

- Average Resolution Time per Task
- Average Cost per Outage
- Planned Budget Cost per Project

Another application of the rate calculation is the comparison or the correlation use case. Calculate the ratio of any two indicators to visualize the difference as a rate:

- Planned Costs / Actual Costs
- Incident Resolution Time Overall / Incident Resolution Specific Team



Different Aggregates in Rate Formulas

now.

Requirement:

Calculate Rate of Performance per Reporting Period

Solution:

Embed Time Series Aggregations in Formula Components

✗ Number of New Requests / Number of Active Users

✓ Monthly Sum of New Requests / Monthly Average of Active Users

Formula Code using SUM and AVG in Numerator and Denominator:

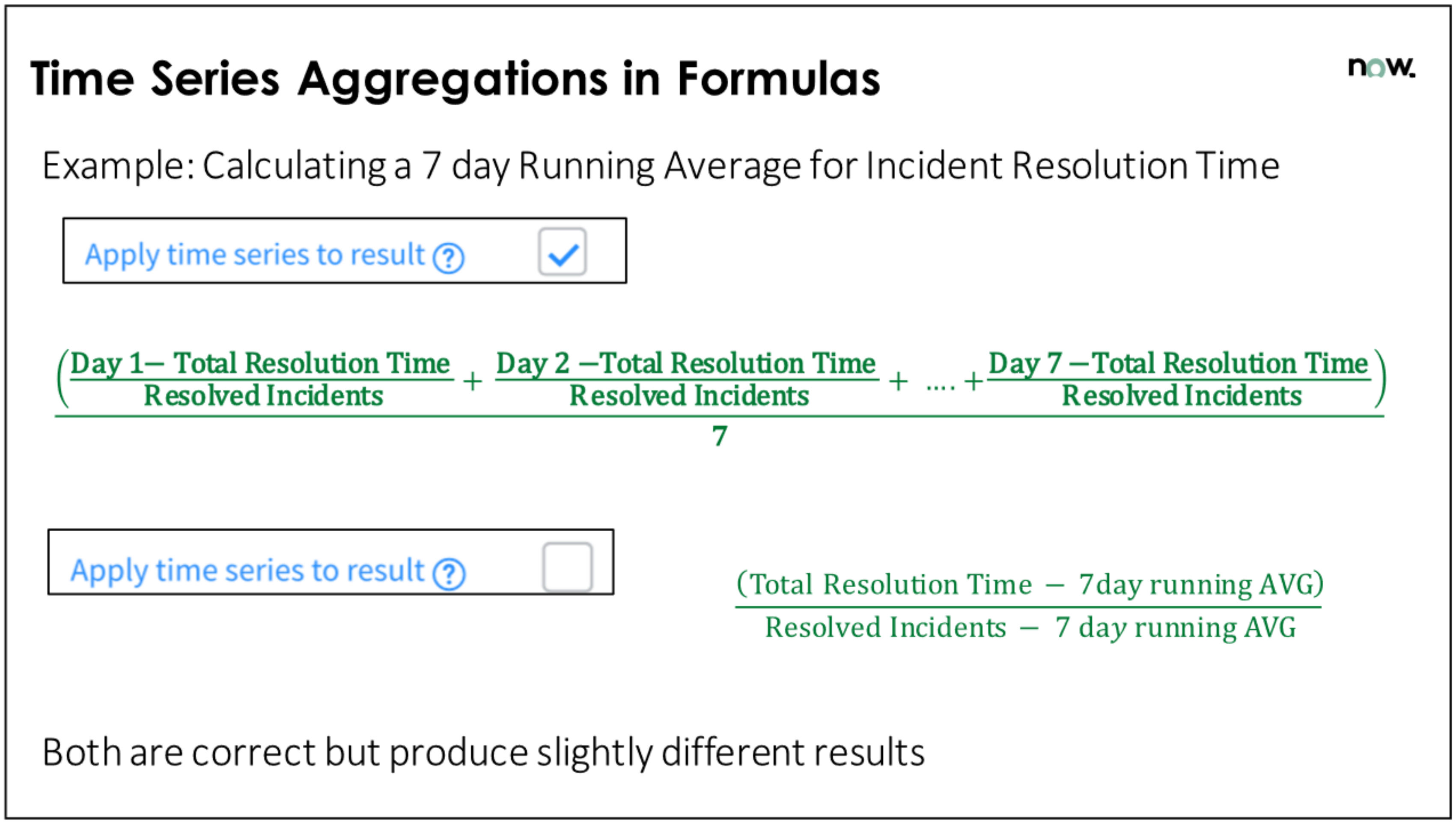
```
[[Number of requests created / By month SUM]]/[[Number of
active users / By month AVG]]
```

When presenting averages and rates in aggregations other than Daily, a Time Series aggregate needs to be applied to all indicators in the formula. If interested in viewing Monthly statistics, apply the By Month SUM aggregate to the numerator and the By Month AVG aggregate to the denominator.

Do not forget to add the appropriate Time Series Exclusions to the formula definition.

Here are some additional rate examples that are implemented by embedding different aggregations in the formula:

- [[Number of knowledge article views / By month SUM]] / [[Number of active users / By month AVG]]
- [[Number of incidents created / By month SUM]] / [[Number of active users / By month AVG]]



Time Series Aggregations in Formulas

now.

Example: Calculating a 7 day Running Average for Incident Resolution Time

Apply time series to result ? ☑

$$\frac{\left(\frac{\text{Day 1– Total Resolution Time}}{\text{Resolved Incidents}} + \frac{\text{Day 2 –Total Resolution Time}}{\text{Resolved Incidents}} + \text{....} + \frac{\text{Day 7 –Total Resolution Time}}{\text{Resolved Incidents}}\right)}{7}$$

Apply time series to result ? ☐

$$\frac{(\text{Total Resolution Time} - \text{7day running AVG})}{\text{Resolved Incidents} - \text{7 day running AVG}}$$

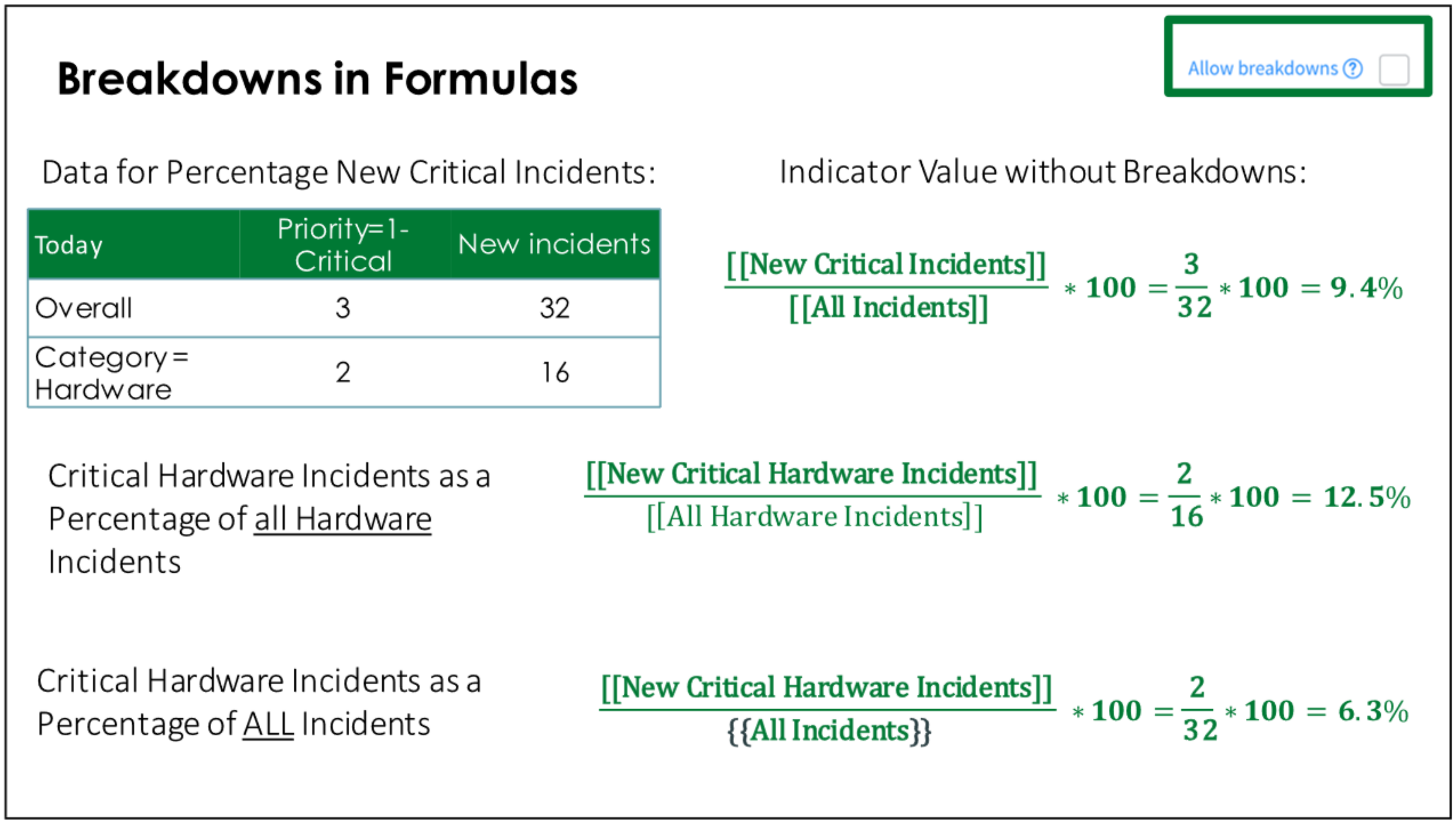
Both are correct but produce slightly different results

To control how Time series are applied to a formula, use the **Apply time series to result** property. This property has these two settings:

- Checked: Apply the Time Series after calculating the formula result
- Unchecked: Apply the Time Series to each formula component individually before calculating the result

The first example shows how the system calculates the 7 day running average for a multi indicator formula.

The second example demonstrates how to more accurately calculate the 7d running average.



Breakdowns in Formulas

Allow breakdowns ⓘ □

Data for Percentage New Critical Incidents:

Today	Priority=1-Critical	New incidents
Overall	3	32
Category=Hardware	2	16

Indicator Value without Breakdowns:

$$\frac{[[\text{New Critical Incidents}]]}{[[\text{All Incidents}]]} * 100 = \frac{3}{32} * 100 = 9.4\%$$

Critical Hardware Incidents as a Percentage of all Hardware Incidents

$$\frac{[[\text{New Critical Hardware Incidents}]]}{[[\text{All Hardware Incidents}]]} * 100 = \frac{2}{16} * 100 = 12.5\%$$

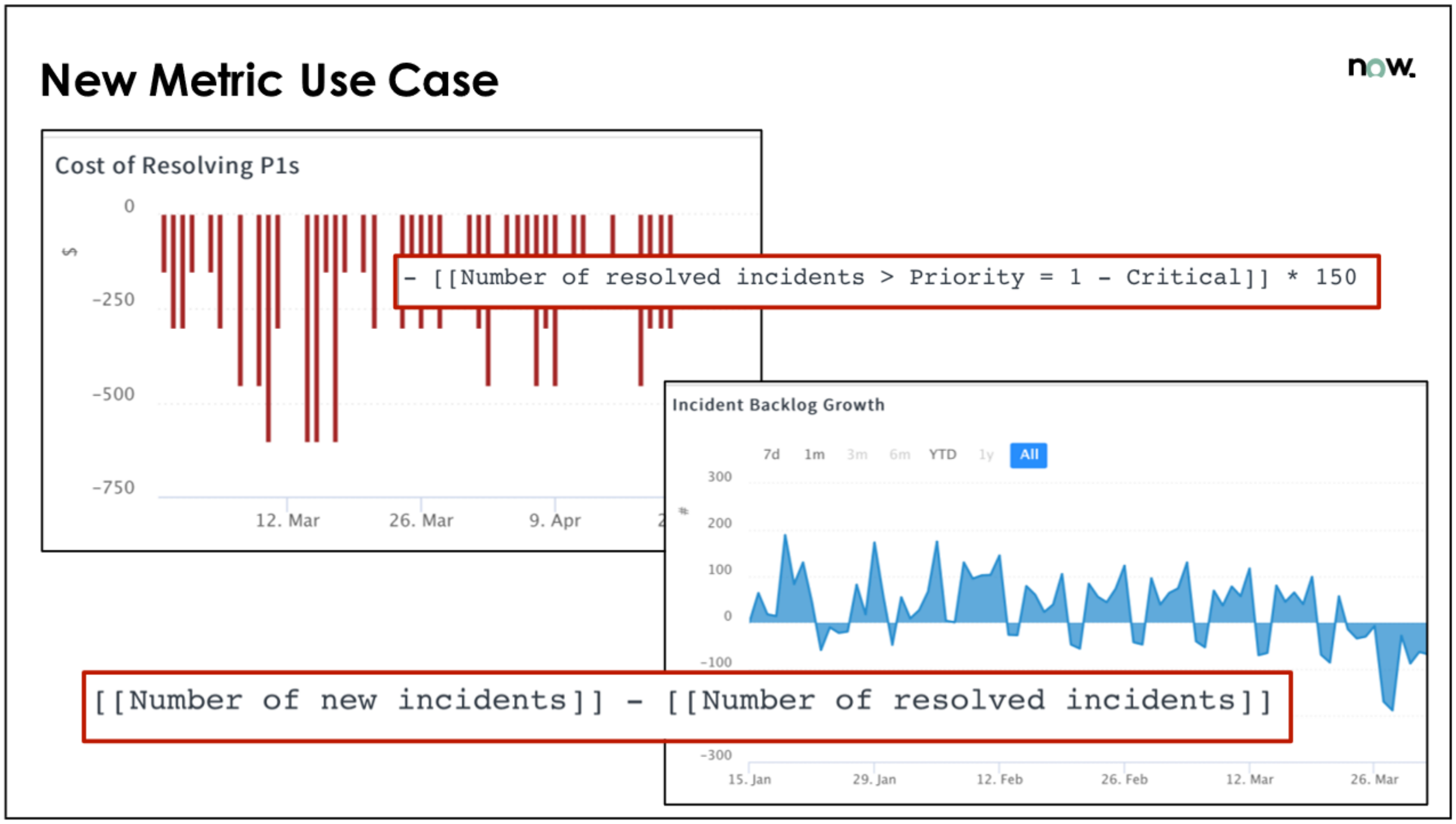
Critical Hardware Incidents as a Percentage of ALL Incidents

$$\frac{[[\text{New Critical Hardware Incidents}]]}{\{\{\text{All Incidents}\}\}} * 100 = \frac{2}{32} * 100 = 6.3\%$$

By default, dashboard and scorecard breakdowns are applied to all formula components. This behavior can be controlled using the **Allow Breakdown** checkbox. The formula result and meaning changes depending on how a breakdown is applied to the formula components:

- In the first example, no breakdowns are applied, so the value of the Percentage New critical Incidents indicator is 9.38%.
- In the second example, the Category Breakdown is applied to both components of the Formula. So the result represents the critical incidents of category Hardware as a percentage of *only the incidents of category Hardware*. This percentage is 12.5%.
- In the third example, the Category Breakdown is applied only to the numerator. So the result represents the critical incidents of category Hardware as a percentage of *all incidents*. This percentage is 6.25%. In this case the Denominator is configured to NOT allow breakdowns at Formula design time.

Do not allow a breakdown to be applied by checking the Allow breakdown checkbox.



New Metric Use Case

Cost of Resolving P1s

`- [[Number of resolved incidents > Priority = 1 - Critical]] * 150`

Incident Backlog Growth

`[[Number of new incidents]] - [[Number of resolved incidents]]`

Formula Indicators can be used to derive any metric that is not yet tracked and stored in ServiceNow tables. Here is an example:

- Cost of resolving Priority 1 Incidents – Even if the rate for working priority 1 incidents is not obtainable from Service Now, you can create a formula like this – Number of P1 Incidents * $150
- Net Inserts = Daily Inserts – Daily Deletes

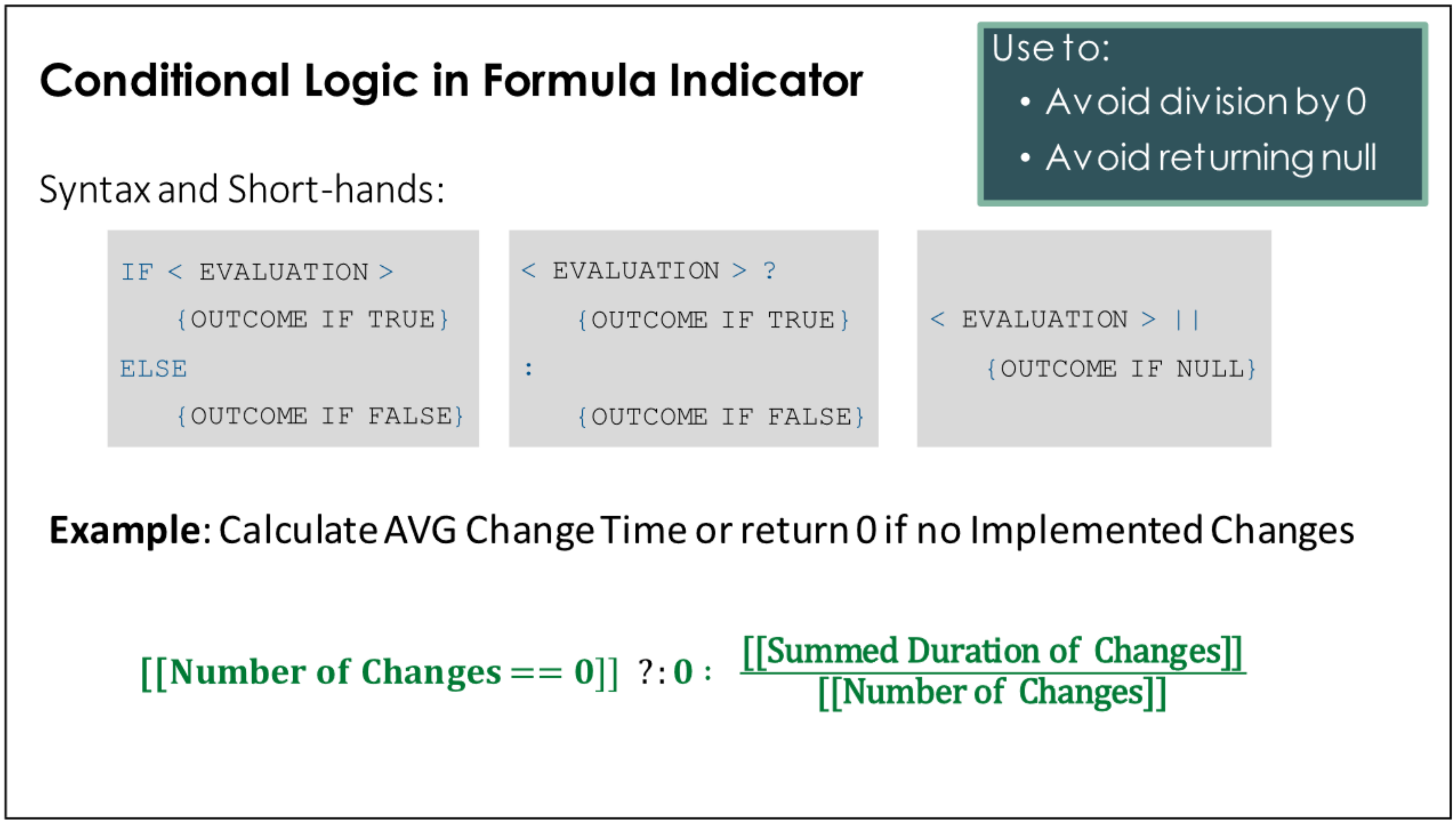
Formulas can also be used to perform conversions such as:

- Express the result in weeks/days/hours/minutes
  - Conversion of Milliseconds to Minutes:

```
[[PA: Daily Total Run Time]] / (1000 * 60)
```

  - Expressing something on a custom scale or as a negative:

```
-1 * [[PA: Daily Deletes]]
```



Conditional Logic in Formula Indicator

Use to:

- Avoid division by 0
- Avoid returning null

Syntax and Short-hands:

```
IF < EVALUATION >
    {OUTCOME IF TRUE}
ELSE
    {OUTCOME IF FALSE}
```

```
< EVALUATION > ?
    {OUTCOME IF TRUE}
:
    {OUTCOME IF FALSE}
```

```
< EVALUATION > ||
    {OUTCOME IF NULL}
```

Example: Calculate AVG Change Time or return 0 if no Implemented Changes

$$\text{[[Number of Changes == 0]]} \ ?: 0 : \frac{\text{[[Summed Duration of Changes]]}}{\text{[[Number of Changes]]}}$$

The Formula box supports JavaScript syntax. Because of that, simple JavaScript statements can be used to handle conditional logic. There is no built in parser for your JavaScript code at development time, so always test the logic before you include it in the Formula.

Use conditional syntax in the following situations:

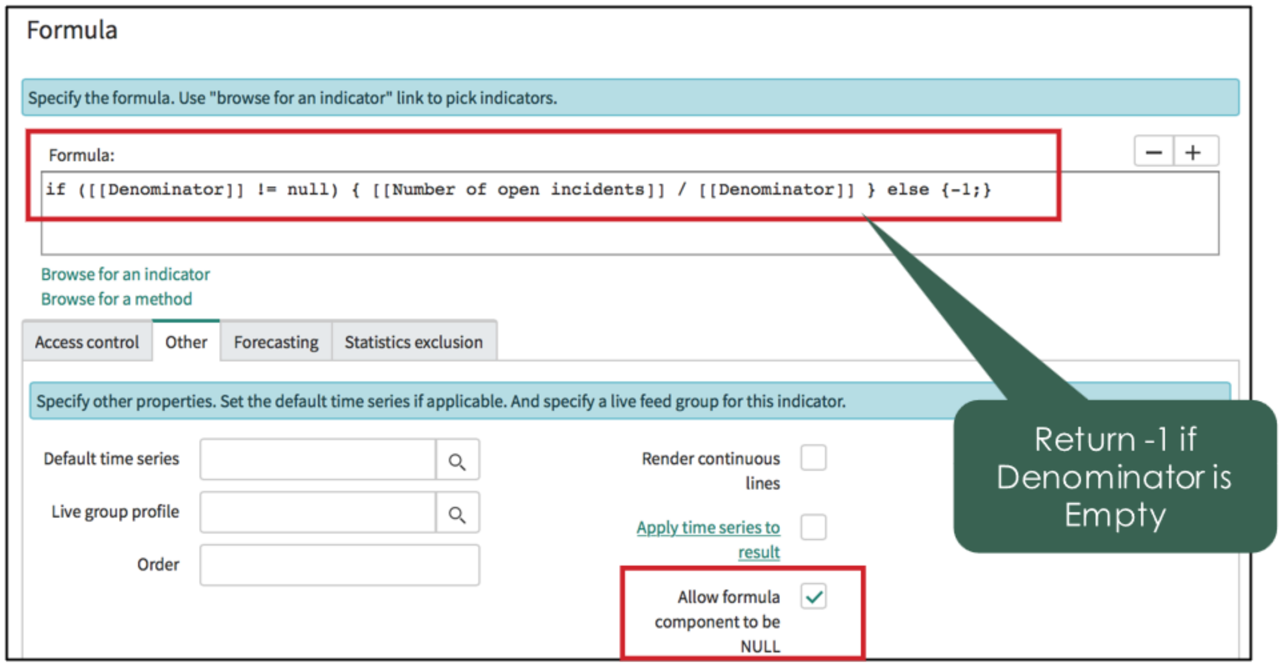
- To choose between two actions based on the evaluation of a score
- To return a different result based on score or current date and time

In the Example above, you need to calculate the Average time it takes to review implemented Changes. The problem is that on some days we have no implemented changes. On such days, the denominator is 0 and a null value is returned because division by 0 cannot be evaluated. To get around this problem, apply conditional logic that returns a 0 any time the denominator value is 0.

Handling NULL values

Allow Formula Component to be Null to allow Null check

```
If Indicator
!=null

   {Indicator}

else

   {Constant}
```



Formula

Specify the formula. Use "browse for an indicator" link to pick indicators.

Formula:

```
if ([[Denominator]] != null) { [[Number of open incidents]] / [[Denominator]] } else {-1;}
```

Browse for an indicator
Browse for a method

Access control | Other | Forecasting | Statistics exclusion

Specify other properties. Set the default time series if applicable. And specify a live feed group for this indicator.

Default time series:
Live group profile:
Order:
Render continuous lines ☐
Apply time series to result ☐
Allow formula component to be NULL ☑

Return -1 if Denominator is Empty

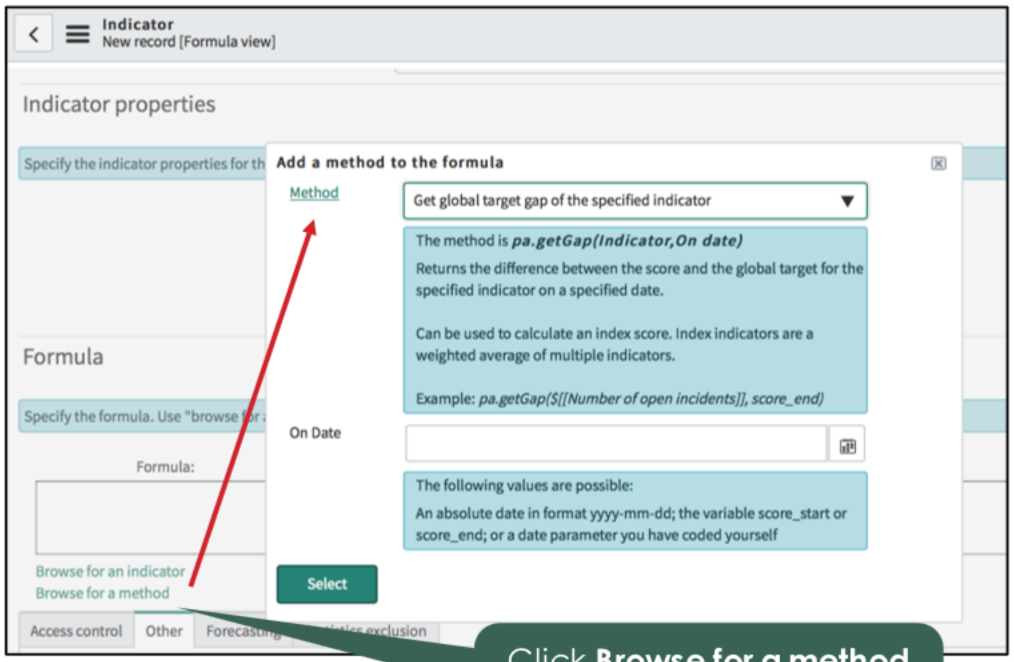
If any of the formula components are Null, the formula will simply abort and return a No Score message. To perform a Null check and return an alternative value, select the **Allow formula component to be NULL**.

Note that an attempt to divide by 0 will always result in a No Score formula return value.

Include API Methods in Formula

Insert the following and use as input to the formula:

- Indicator Score
- Target
- Target Gap
- Change
- Time series aggregation
- Breakdown
- Breakdown element



Get global target gap of the specified indicator

The method is ***pa.getGap(Indicator,On date)***

Returns the difference between the score and the global target for the specified indicator on a specified date.

Can be used to calculate an index score. Index indicators are a weighted average of multiple indicators.

Example: *pa.getGap($[[Number of open incidents]], score_end)*

The following values are possible:

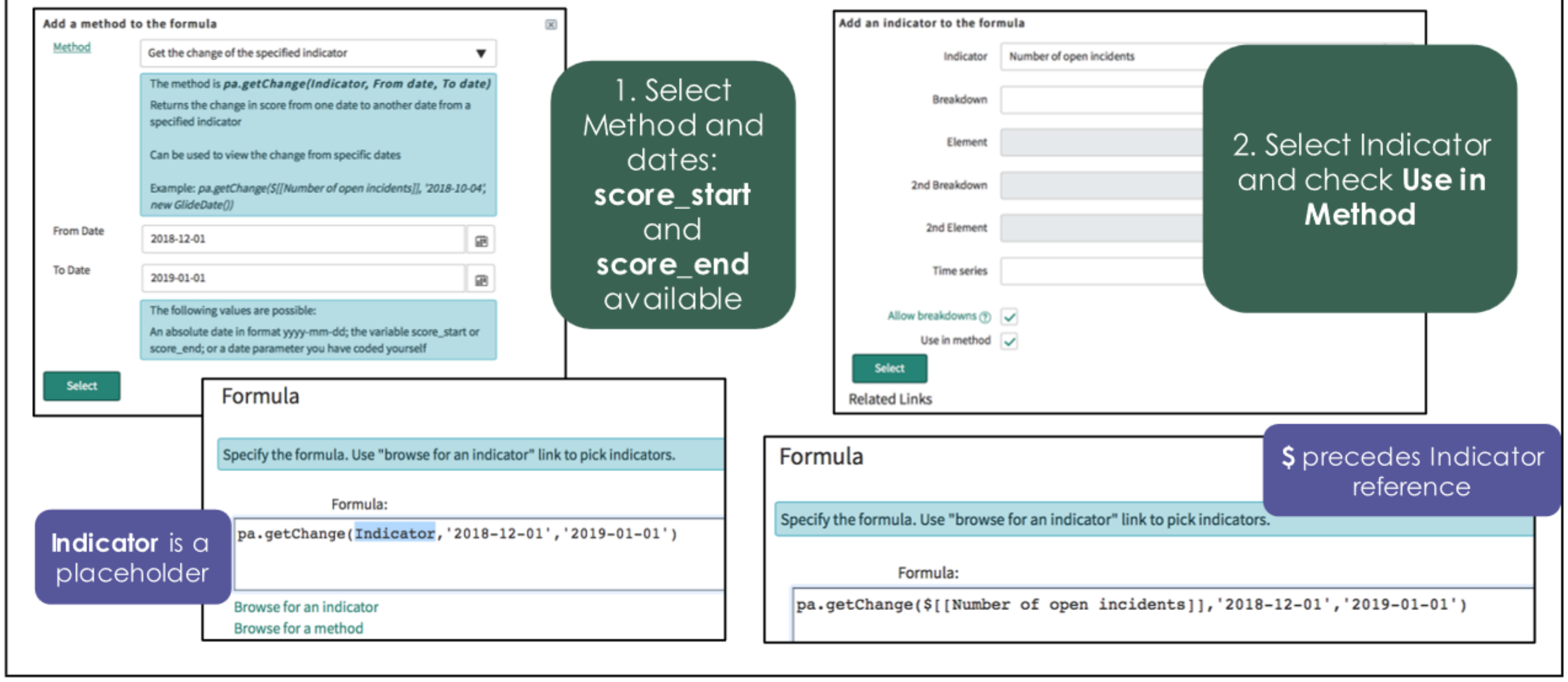
An absolute date in format yyyy-mm-dd; the variable score_start or score_end; or a date parameter you have coded yourself

Click **Browse for a method** to insert a calculated value from the Analytics Hub

To insert a calculated value from the Analytics Hub into a formula, use a method in the formula. You can use a value that was calculated in the Analytics Hub as input for a formula. That value can be from any indicator, including from the current formula indicator itself.

Method Parameters

Most APIs require the following parameters: **Method**, **From Date**, **To Date**, **Indicator**



Add a method to the formula

Method: Get the change of the specified indicator ▼

The method is ***pa.getChange(Indicator, From date, To date)***

Returns the change in score from one date to another date from a specified indicator

Can be used to view the change from specific dates

Example: *pa.getChange($[[Number of open incidents]], '2018-10-04', new GlideDate())*

From Date: 2018-12-01

To Date: 2019-01-01

The following values are possible:

An absolute date in format yyyy-mm-dd; the variable score_start or score_end; or a date parameter you have coded yourself

1. Select Method and dates: **score_start** and **score_end** available

Add an indicator to the formula

Indicator: Number of open incidents

Breakdown:

Element:

2nd Breakdown:

2nd Element:

Time series:

Allow breakdowns ☑

Use in method ☑

2. Select Indicator and check **Use in Method**

Formula

Specify the formula. Use "browse for an indicator" link to pick indicators.

Formula:

```
pa.getChange(Indicator,'2018-12-01','2019-01-01')
```

Indicator is a placeholder

Formula

Specify the formula. Use "browse for an indicator" link to pick indicators.

Formula:

```
pa.getChange($[[Number of open incidents]],'2018-12-01','2019-01-01')
```

$ precedes Indicator reference

Two date parameters are available to any formula: **score_start** and **score_end**.

The variable **score_start** is the first date on which scores for the indicator were collected, depending on the frequency.

The variable **score_end** is the last date on which scores for the indicator were collected, depending on the frequency.

If you have an indicator with a daily frequency and you are calculating for today:

score_start is yesterday and score_end is today

If you are calculating that indicator for yesterday:

score_start is two days ago and score_end is yesterday

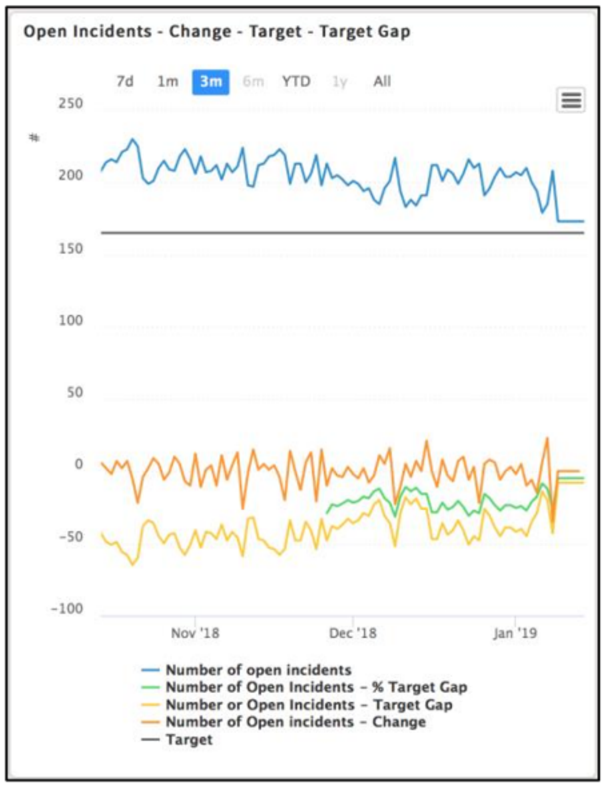
In the example above, after inserting the method, the formula code is pa.getChange(**Indicator**, '2018-12-01','2019-01-01').

After specifying an Indicator to use in method, the code becomes: pa.getChange**($[[Number of open incidents]]**,'2018-12-01','2019-01-01').

Note that instead of hard coded dates, the method can use start_score and end_score.

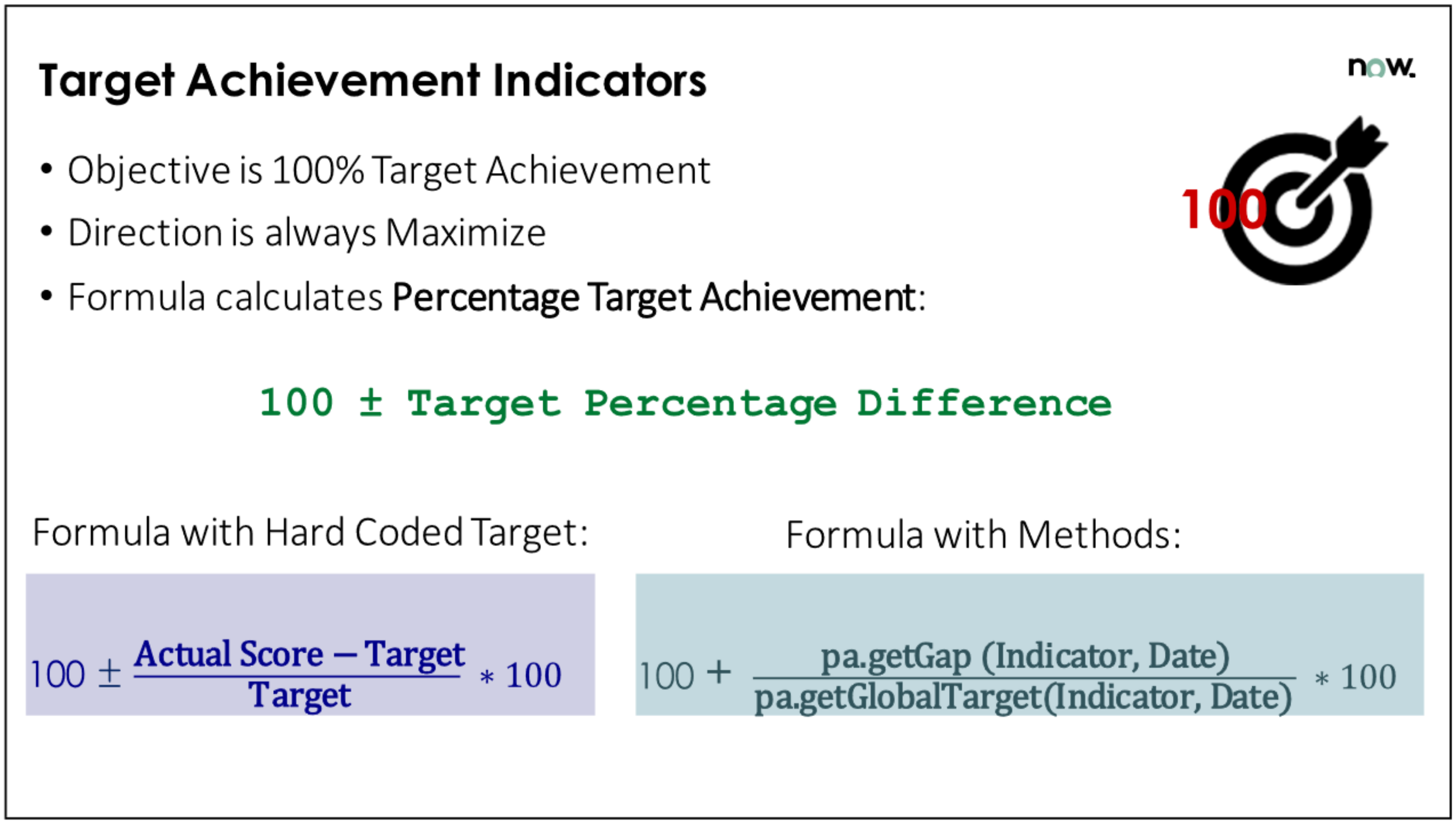
Method Examples

```
//Trend Change or Change Percentage
pa.getChange($[[Number of open incidents]], score_start,
score_end);
pa.getChangePercentage($[[Number of open incidents]],
score_start, score_end);

//Trend Global or Personal Target
pa.getGlobalTarget($[[Number of open incidents]], score_start);
pa.getPersonalTarget($[[Number of open incidents]], score_start);

//Trend the sum of two scores:
pa.getScore($[[Number of open incidents]], score_start) +
pa.getScore($[[Number of new incidents]], score_start)
```



To get a value from the Analytics Hub representing Target, Target Gap, or even calculate a new value, insert a method from **PAFormulaUtils()** into the formula.

For information on all available APIs, visit https://docs.servicenow.com/bundle/newyork-performance-analytics-and-reporting/page/use/performance-analytics/PAFormulaUtil/concept/PAFormulaUtils.html.



Target Achievement Indicators

- Objective is 100% Target Achievement
- Direction is always Maximize
- Formula calculates **Percentage Target Achievement**:

`100 ± Target Percentage Difference`

Formula with Hard Coded Target:

$$100 \pm \frac{\textbf{Actual Score} - \textbf{Target}}{\textbf{Target}} * 100$$

Formula with Methods:

$$100 + \frac{\text{pa.getGap (Indicator, Date)}}{\text{pa.getGlobalTarget(Indicator, Date)}} * 100$$

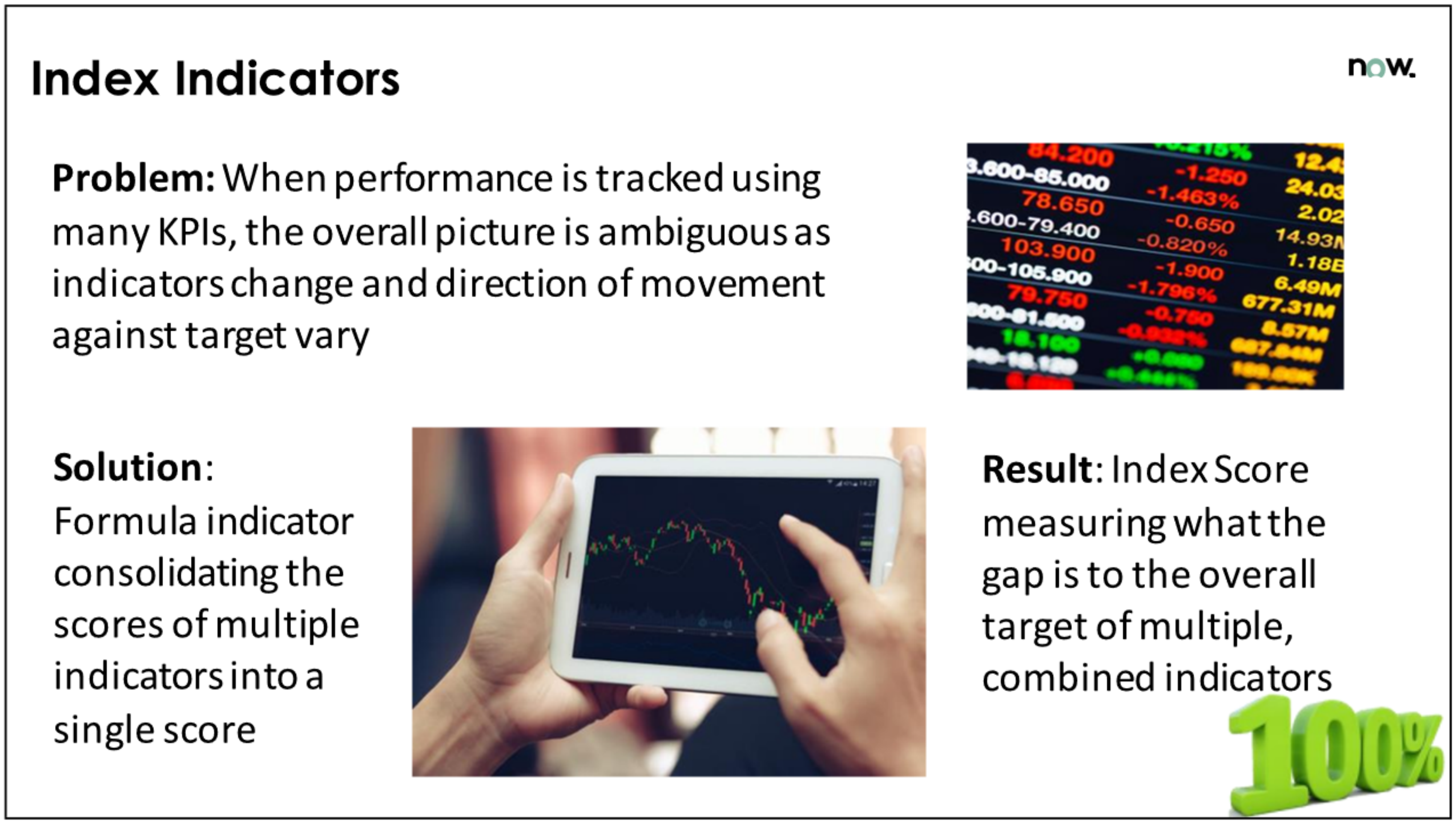
The Target Achievement Indicators are a special case of Formula indicators. The idea of a Target Achievement Index is to calculate the percentage Target Achievement.

Consider these examples for the Number of Open Incidents Indicator with a target of 20 and a Minimize direction:

- The daily score is 20. You have achieved the Target at 100%. (You are right on Target)
- The daily score is 10. You have achieved the Target at 200%. (You have over achieved the Target)
- The daily score is 40. You have achieved the Target at 50%. (You are at 50% of the desired Target)

Implementation notes:

- The Target should always be set at 100% target achievement
- An improving indicator should always result in an increasing index regardless of direction
- The Target value needs to be “hardcoded” in the formula and cannot be obtained from the indicator
- Indicators that have a target of 0 cannot be used in the index indicator as this would result in a division by 0 in the formula which is not allowed.



Index Indicators

Problem: When performance is tracked using many KPIs, the overall picture is ambiguous as indicators change and direction of movement against target vary

Solution: Formula indicator consolidating the scores of multiple indicators into a single score

Result: Index Score measuring what the gap is to the overall target of multiple, combined indicators

The performances of processes, services, and groups are usually tracked and monitored using more than one indicator. When viewing and analyzing performance of these processes, business services, or workgroups, the overall picture can be confusing and ambiguous.

Consider this hypothetical situation regarding a service you are responsible for and monitor with 5 indicators: Scores for 3 process indicators improved somewhat, the scores for 2 of them are still below target and 1 is above target.

So, what is the overall picture? Is the overall performance of the process/service/group still at or above the desired level? Did the overall performance improve?

An index indicator can answer these questions. Index indicator is just a formula indicator that combines the output of other indicators to produce a single score that is easier to interpret and analyze.

Think of this stock market analogy: To get a sense of how the market did on any given day, you don't just look at a handful of stock symbols, rather, you use a composite index like the NASDAQ or the S&P 500 as a daily indicator of stock market performance.

Customer Experience Index – Example Use Case

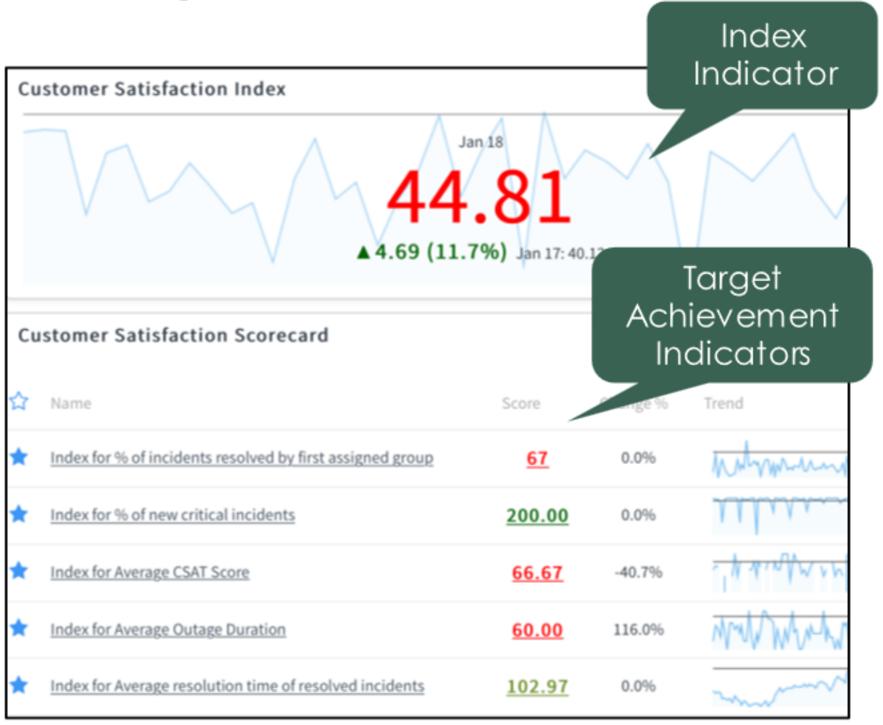
(Target Achievement of Resolution by First Assign. Group +

Target Achievement of New Critical Incidents +

Target Achievement of Average Resolution Time +

Target Achievement of CSAT Score +

Target Achievement of Outage Duration) / 5



Customer Satisfaction Index

44.81

Customer Satisfaction Scorecard

☆	Name	Score	Change %	Trend
★	Index for % of incidents resolved by first assigned group	67	0.0%	
★	Index for % of new critical incidents	200.00	0.0%	
★	Index for Average CSAT Score	66.67	-40.7%	
★	Index for Average Outage Duration	60.00	116.0%	
★	Index for Average resolution time of resolved incidents	102.97	0.0%	

The Index is a Weighted Average of Multiple Target Achievement KPIs

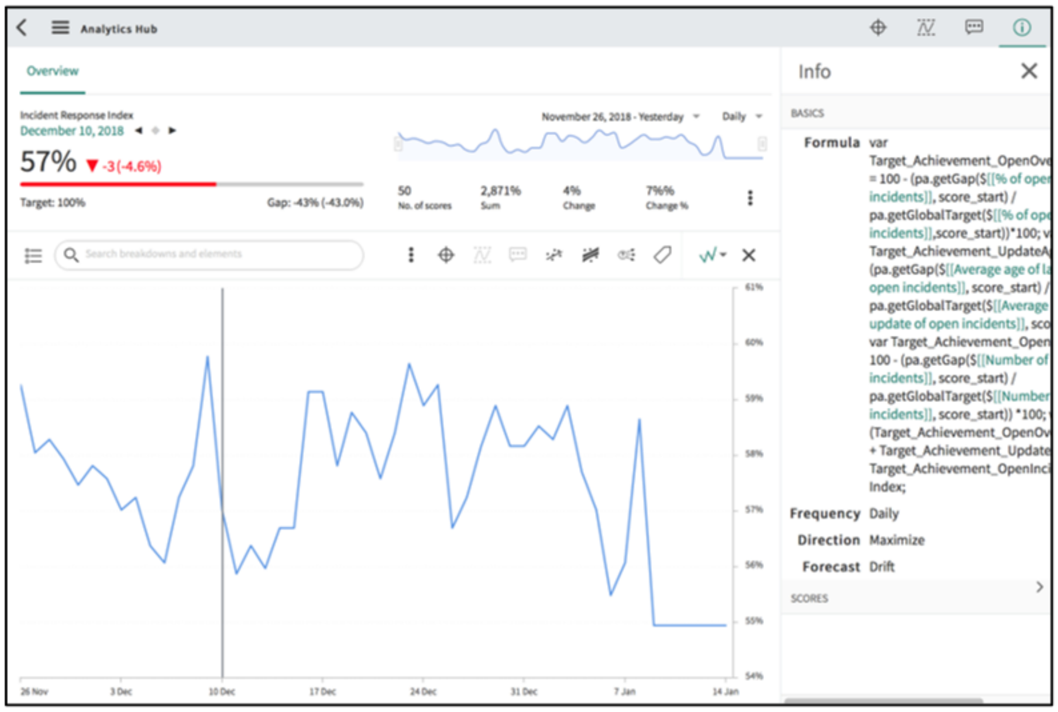
With an Index Indicator, the scores of multiple indicators are aggregated into a single score. The Index is a weighted average of ***x*** indicators. If the weighted sum of these ***x*** indicators is improving, the calculated score of the index formula goes up. Like any other indicator, the index indicator shows if the score is good or not and if the score has improved or not.

Index indicators are just formula indicators that combine the output of other indicators to produce a single score that is easier to interpret and analyze. In the example above, the Customer Experience Index is a weighted average of several Target Achievement Index Indicators.

Note that the terms *Target Achievement Indicator* and *Index Indicator* describe the formula implementation approach and are not actual productized Indicator Types.

Index Indicator Implementation

- The Index calculates a weighted average of several Target Achievement indicators in a Formula
- To calculate Target Achievement, an Indicator must have Direction and Target
- Use the pa.getGap API to calculate the Target Gap



Formula
```
var
Target_Achievement_OpenOve
= 100 - (pa.getGap($[[% of oper
incidents]], score_start) /
pa.getGlobalTarget($[[% of ope
incidents]],score_start))*100; v
Target_Achievement_UpdateA
(pa.getGap($[[Average age of la
open incidents]], score_start) /
pa.getGlobalTarget($[[Average
update of open incidents]], sco
var Target_Achievement_Open
100 - (pa.getGap($[[Number of
incidents]], score_start) /
pa.getGlobalTarget($[[Number
incidents]], score_start)) *100;
(Target_Achievement_OpenOv
+ Target_Achievement_Update
Target_Achievement_OpenInci
Index;
```

Frequency Daily

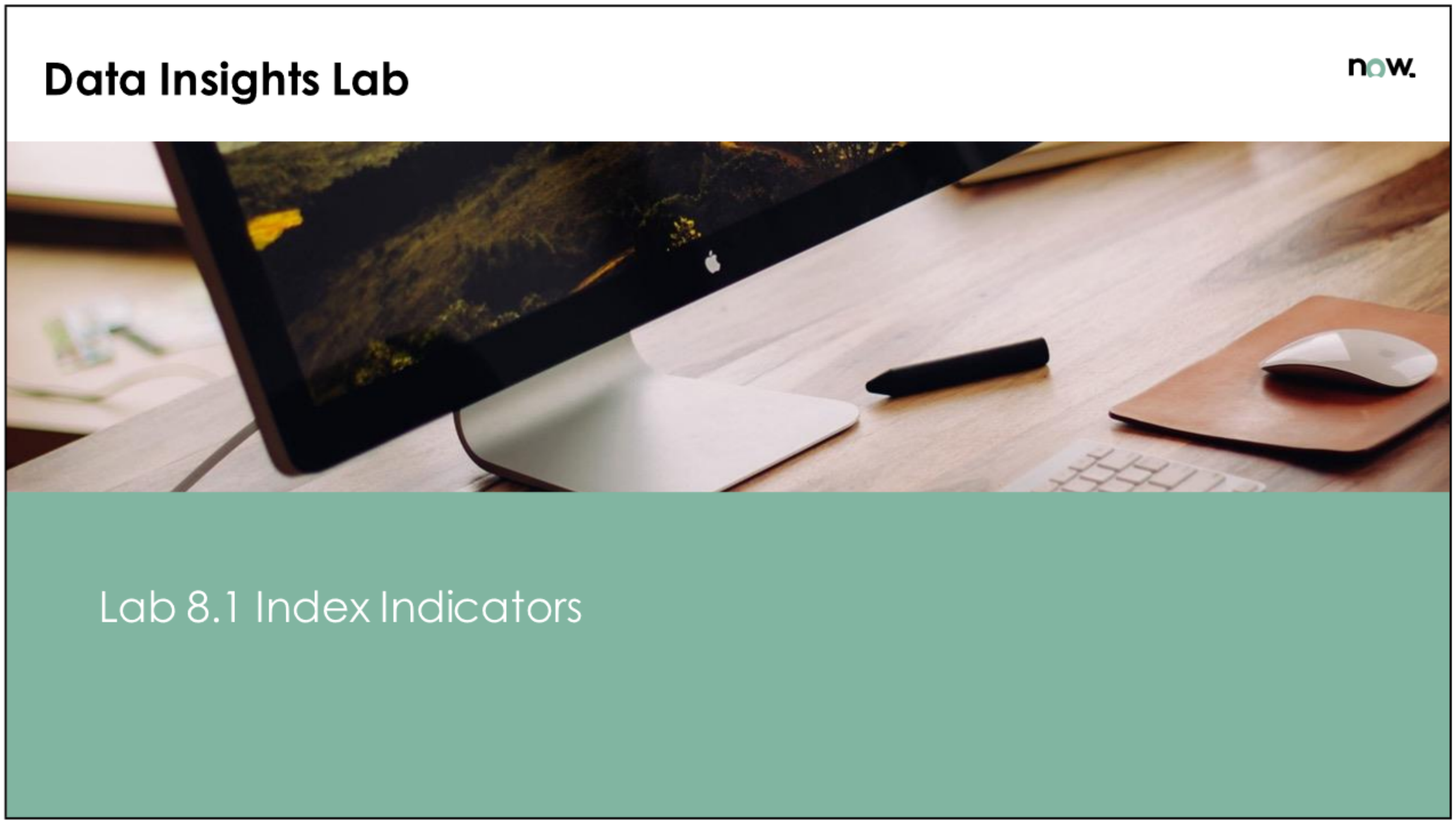
Direction Maximize

Forecast Drift

The Incident Resolution Index uses the following formula:

```
var Target_Achievement_OpenOverdueIncidents =
        100 + (pa.getGap($[[% of open overdue incidents ]], score_start) /
        pa.getGlobalTarget($[[% of open overdue incidents]],score_start)) * 100;

var Target_Achievement_UpdateAge =
        100 + (pa.getGap($[[Average age of last update of open incid ents]], score_start) /
        pa.getGlobalTarget($[[Average age of last update of open incidents]], score_start)) * 100;

var Target_Achievement_OpenIncidents =
        100 + (pa.getGap($[[Number of open incidents]], score_st art) /
        pa.getGlobalTarget($[[Number of open incidents]], score_start)) * 100;

var Index = (Target_Achievement_OpenOverdueIncidents +
            Target_Achievement_UpdateAge +
            Target_Achievement_OpenIncidents) / 3;
Index;
```

Note that instead of calculating the Target Achievement for each Indicator in the Index formula, you can also create Target Achievement Formula indicators and reference them in the Index formula.



Data Insights Lab

Lab 8.1 Index Indicators

Index Indicators

Lab 8.1

⌛30m

Lab Objectives

The Customer Success Manager at Cloud Dimensions requires a single indicator that shows how satisfied customers are with the overall incident handling and customer interaction processes. This process index indicator will be based on these underlying indicators:

- Average CSAT Score
- Average Duration of Outages

This lab will show you how to do the following:

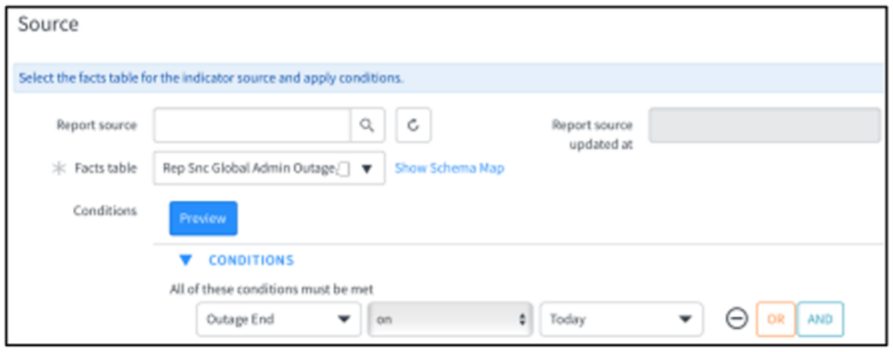
- Create Formula Indicators to represent individual Indicator vs Target performance
- Create a combined Index indicator to track multiple Indicator performance

Outage Indicator

Indicator Source and Indicator

1. Navigate to **Performance Analytics > Sources > Indicator Sources**
2. Create the following new Indicator Source:

   Name: **Outages**
   Facts Table: **Rep Snc Global Admin System Outages [u_rep_snc_global_admin_system_outages]**
   Conditions: **Outage End on Today**



Source

Select the facts table for the indicator source and apply conditions.

Report source:

Report source updated at:

Facts table: Rep Snc Global Admin Outage

Conditions

CONDITIONS

All of these conditions must be met

Outage End | on | Today

> ***Note**: The Report Designer automatically prefixes **Rep SNC Global Admin** to whatever external data source import name you provide when creating an external source report.*

3. Click **Submit**.

4. Navigate to **Performance Analytics > Indicators > Automated Indicators.**

5. Create the following new Indicator:

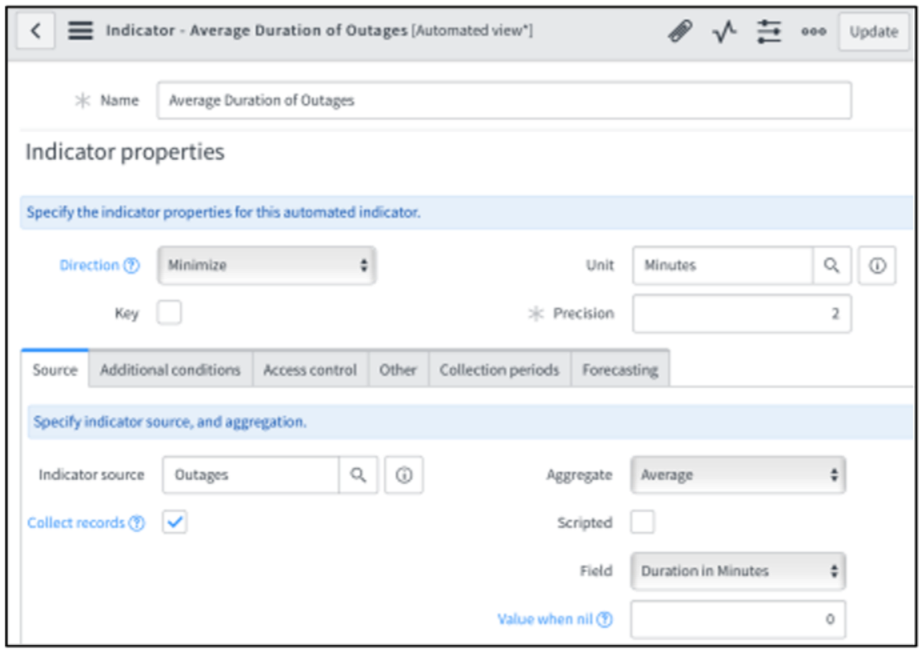
   Name: **Average Duration of Outages**

   Frequency: **Daily**

   Direction: **Minimize**

   Unit: **Minutes**
   Precision: **2**

   Indicator Source: **Outages**



Indicator - Average Duration of Outages [Automated view*]

* Name: Average Duration of Outages

Indicator properties

Specify the indicator properties for this automated indicator.

Direction: Minimize

Unit: Minutes

Key: □

* Precision: 2

Specify indicator source, and aggregation.

Indicator source: Outages

Aggregate: Average

Collect records: ☑

Scripted: □

Field: Duration in Minutes

Value when nil: 0

   Aggregate: **Average**   Field: **Duration in Minutes**

   Value when nil: **0**   Publish on Analytics Hub: **checked**

6. Click **Submit**.

Data Collection

1. Navigate to **Performance Analytics > Data Collector > Jobs**.

2. Create a new job to collect data for the **Average Duration of Outages** indicator for the last 3 months.

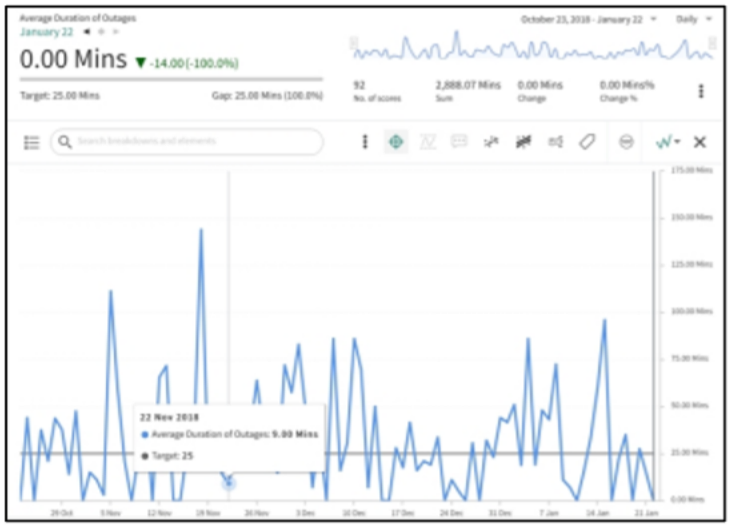
3. Execute the job.

Data Verification

1. Navigate to the **Analytics Hub**.

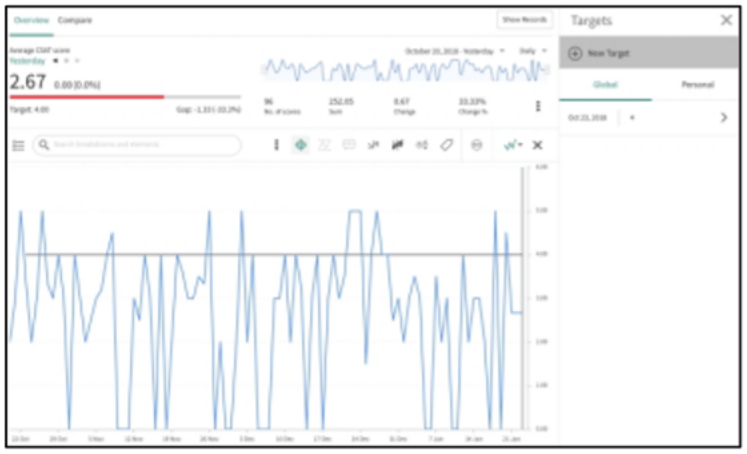
2. Confirm the **Average Duration of Outages** Analytics Hub displays valid outage data.

3. Add a global **Target** starting about 3 months ago at **25 minutes**.


Average Duration of Outages
January 22
0.00 Mins ▼ -14.00 (-100.0%)
Target: 25.00 Mins
Gap: 25.00 Mins (100.0%)
October 23, 2018 - January 22
Daily
92
No. of scores
2,888.07 Mins
Sum
0.00 Mins
Change
0.00 Mins%
Change %
Search breakdowns and elements
22 Nov 2018
Average Duration of Outages: 0.00 Mins
Target: 25

CSAT Score Target Creation

1. Navigate to **Performance Analytics > Analytics Hub**
2. View the **Average CSAT Score** Indicator
3. Create a new **Global Target** set at **4** and starting about **3 months ago**:



Index Indicator Creation

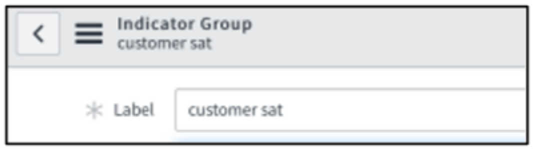
A. Target Performance Tracking

In this section, you create two new formula indicators to track the Target performance of each of the Indicators reviewed in the previous section.

Indicator Group

1. Navigate to **Performance Analytics > Indicators > Indicator Groups**.
2. Click **New** and create the following new group:

Label: **customer sat**



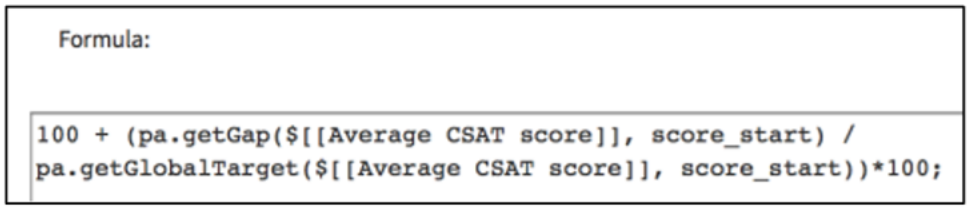
* Label: customer sat

Index for Average CSAT Score

1. Navigate to **Performance Analytics > Indicators > Formula Indicators**.

2. Click **New** and create a new Formula Indicator as follows:

   Indicator group: **customer sat**
   Name: **Index for Average CSAT scores**
   Direction: **Maximize**
   Unit: **%**
   Precision: **2**
   Formula:



Formula:

```
100 + (pa.getGap($[[Average CSAT score]], score_start) /
pa.getGlobalTarget($[[Average CSAT score]], score_start))*100;
```

3. **Save** the indicator.

4. Navigate to the **Targets** Related List and create a new **Target** as follows:

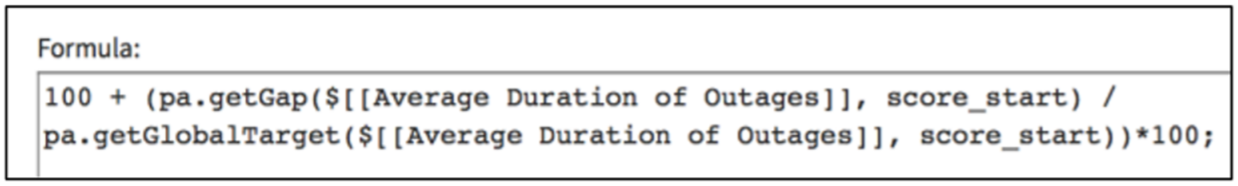
   - Target at: **about 3 months ago**
   - Value: **100**

5. Return to the list of Formula indicators.

Index for Average Outage Duration

1. Click **New** and create a new Formula Indicator as follows:

   Indicator group: **customer sat**
   Name: **Index for Average Outage Duration**
   Direction: **Maximize**
   Unit: **%**
   Precision: **2**
   Formula:



Formula:

```
100 + (pa.getGap($[[Average Duration of Outages]], score_start) /
pa.getGlobalTarget($[[Average Duration of Outages]], score_start))*100;
```

2. **Save** the indicator.

3. Navigate to the **Targets** Related List and create a new **Target** as follows:

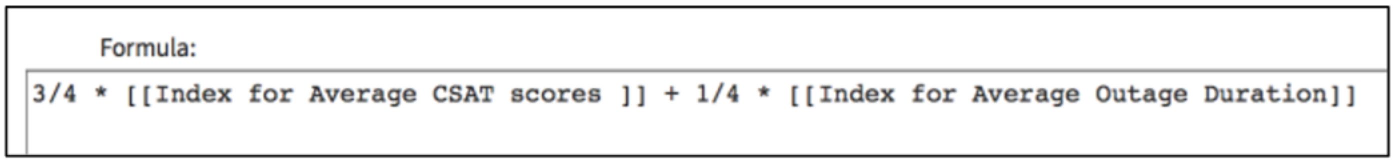
   - Target at: **about 3 months ago**
   - Value: **100**

B. Customer Satisfaction Index

The two formula index indicators created above are now going to be used to create the overall Index indicator for the Customer Satisfaction sentiment.

1. Navigate to **Performance Analytics > Indicators > Formula Indicators**.
2. Create a new Formula Indicator as follows:

   Name: **Customer Satisfaction Index**
   Direction: **Maximize**
   Unit: **%**
   Formula:



Formula:

```
3/4 * [[Index for Average CSAT scores ]] + 1/4 * [[Index for Average Outage Duration]]
```

> ***Note**: The above is a weighted average of two indicators. You can over- or under-weight them as needed to represent stronger/weaker influence on the Customer satisfaction index.*

3. **Save** the Indicator.

4. Navigate to the **Targets** Related List and create a new **Target** as follows:

   - Target at: **about 3 months ago**
   - Value: **100**

Dashboard Visualization

With the hard work out of the way, it is time to create a widget on the Customer Success Advocate dashboard.

1. Navigate to the **Customer Success Advocate** dashboard.

2. Create a new tab and rename it to **Customer Satisfaction Index**.

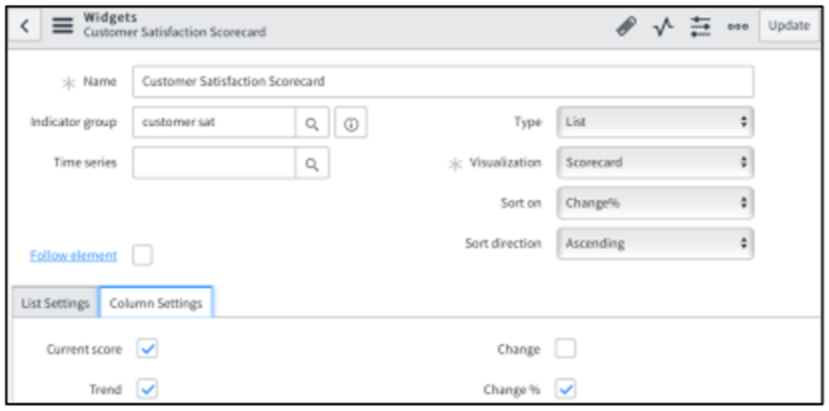
3. Add a new **List** Widget and configure it as follows:

Name: **Customer Satisfaction Scorecard**

Indicator group: **customer sat**

Type: **List**
Visualization: **Scorecard**

Column Settings: **Current score**, **Trend**, **Change %**



Widgets
Customer Satisfaction Scorecard

✳ Name	Customer Satisfaction Scorecard		
Indicator group	customer sat	Type	List
Time series		✳ Visualization	Scorecard
		Sort on	Change%
Follow element	□	Sort direction	Ascending

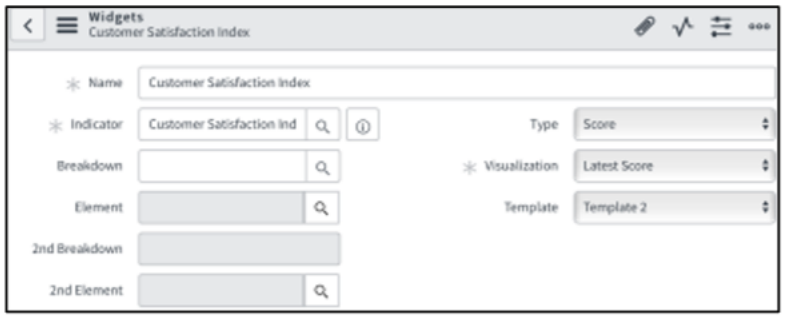
List Settings | Column Settings

Current score	☑	Change	□
Trend	☑	Change %	☑

4. Add a new Performance Analytics **Score** Widget and configure it as follows:

   Name: **Customer Satisfaction Index**

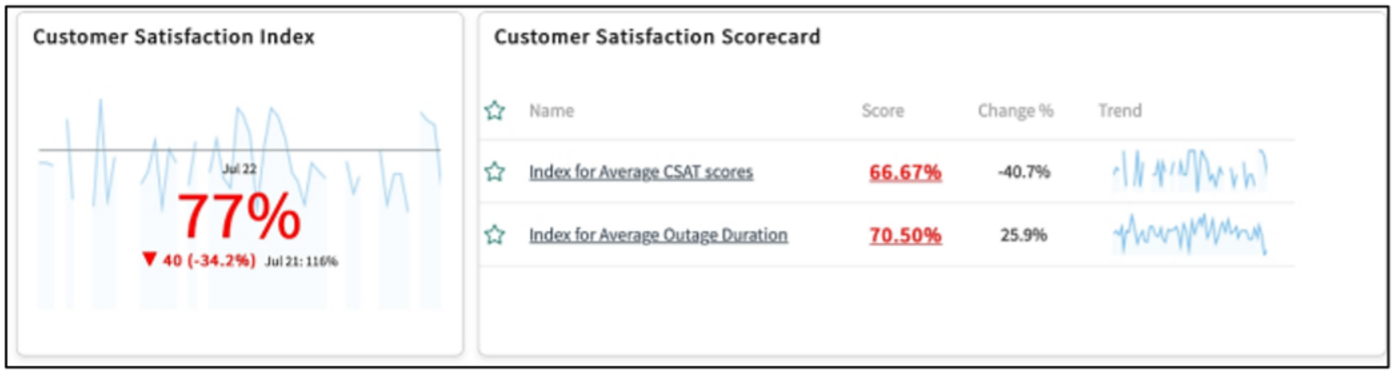
   Type: **Score**
   Visualization: **Latest Score**
   Template: **Template 2**



Widgets
Customer Satisfaction Index

✳ Name	Customer Satisfaction Index		
✳ Indicator	Customer Satisfaction Ind	Type	Score
Breakdown		✳ Visualization	Latest Score
Element		Template	Template 2
2nd Breakdown			
2nd Element			

5. Reposition the **Customer Satisfaction Index** tab as the **first** tab on the dashboard.



Customer Satisfaction Index

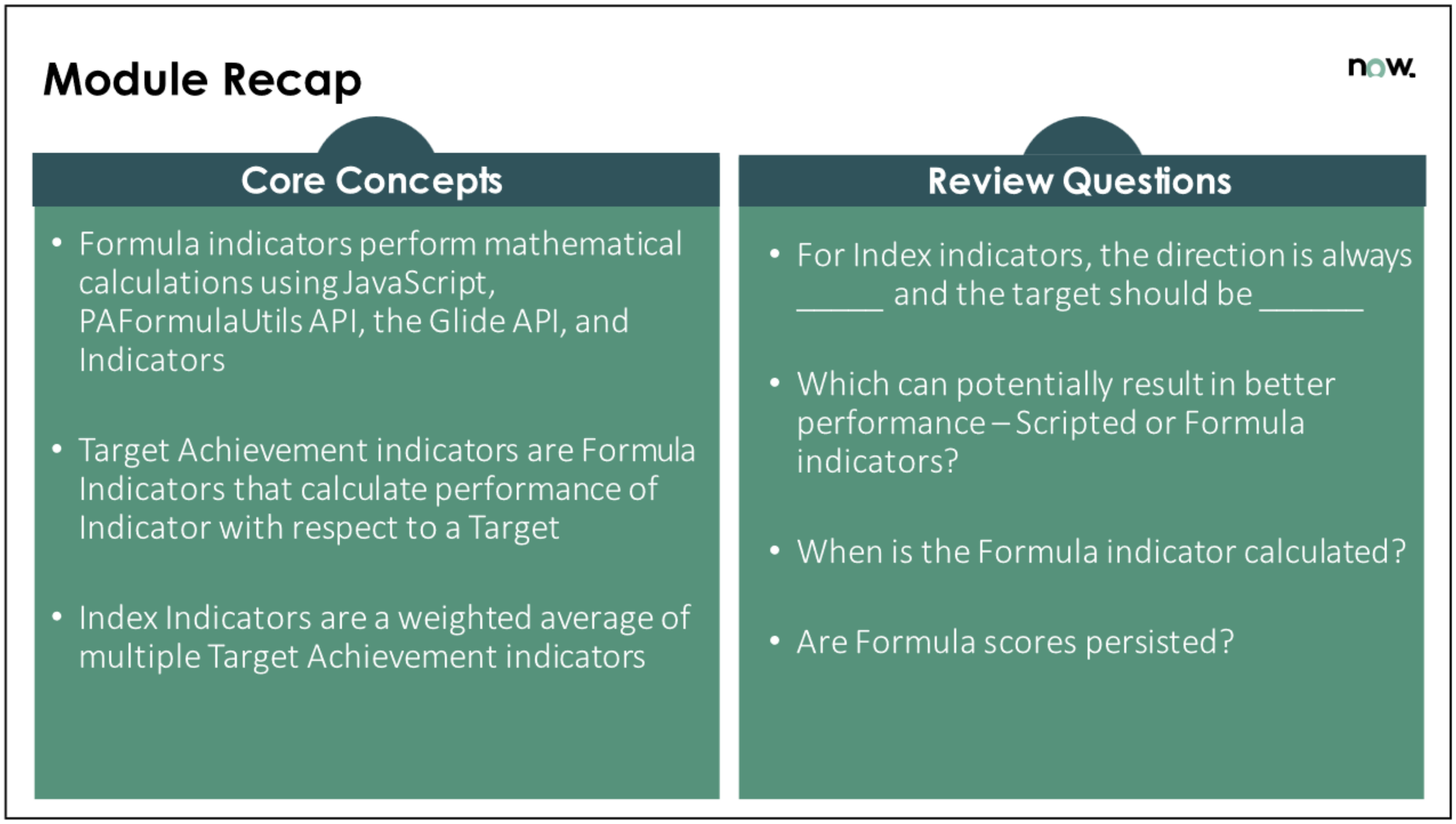
Customer Satisfaction Scorecard

	Name	Score	Change %	Trend
☆	Index for Average CSAT scores	66.67%	-40.7%	
☆	Index for Average Outage Duration	70.50%	25.9%	

- Which index is under-performing?

- Which index is over-performing?

You have now completed the Index Indicators lab.



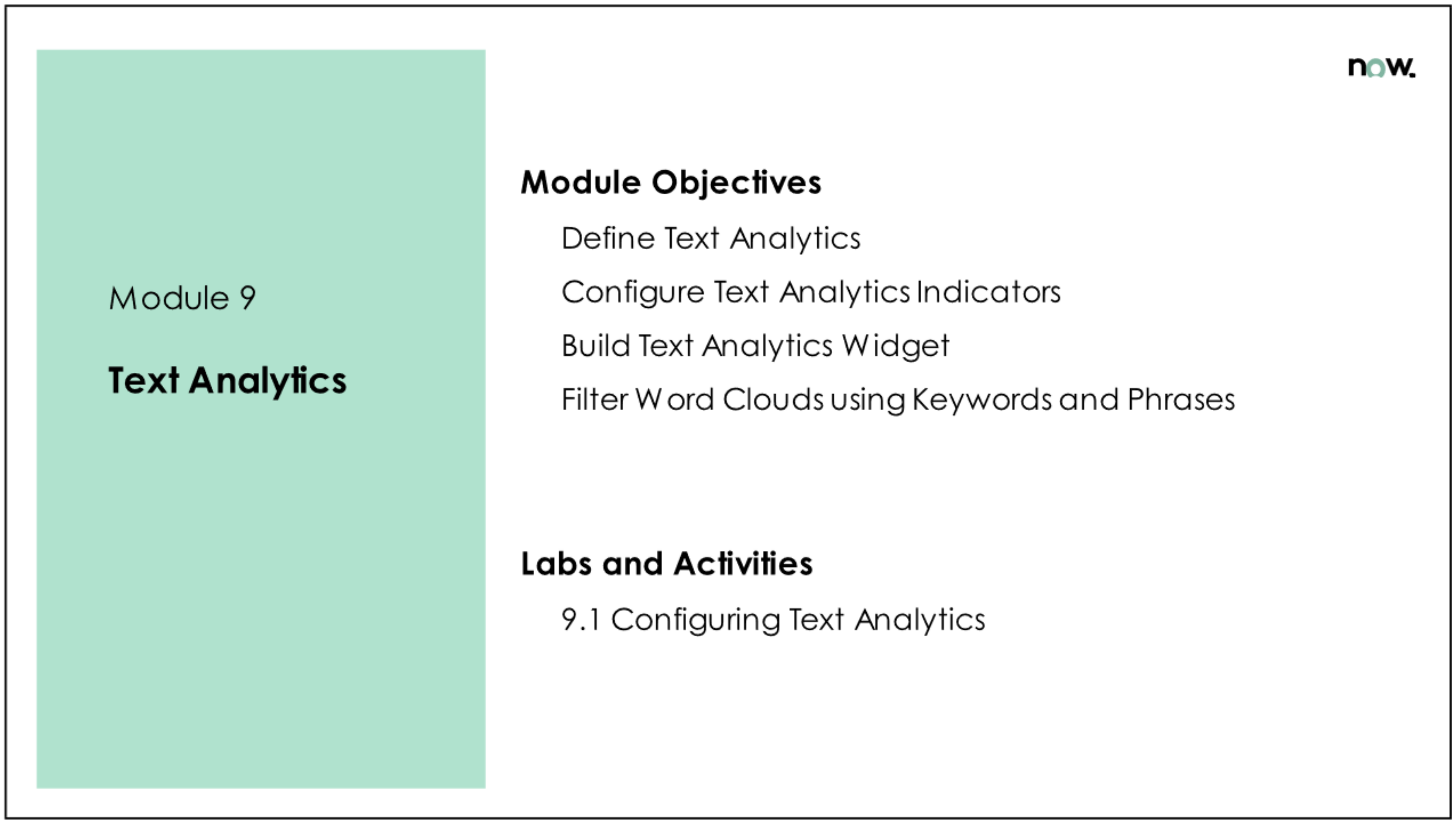
Module Recap

Core Concepts

- Formula indicators perform mathematical calculations using JavaScript, PAFormulaUtils API, the Glide API, and Indicators
- Target Achievement indicators are Formula Indicators that calculate performance of Indicator with respect to a Target
- Index Indicators are a weighted average of multiple Target Achievement indicators

Review Questions

- For Index indicators, the direction is always ______ and the target should be _______
- Which can potentially result in better performance – Scripted or Formula indicators?
- When is the Formula indicator calculated?
- Are Formula scores persisted?



Module 9

Text Analytics

Module Objectives

- Define Text Analytics
- Configure Text Analytics Indicators
- Build Text Analytics Widget
- Filter Word Clouds using Keywords and Phrases

Labs and Activities

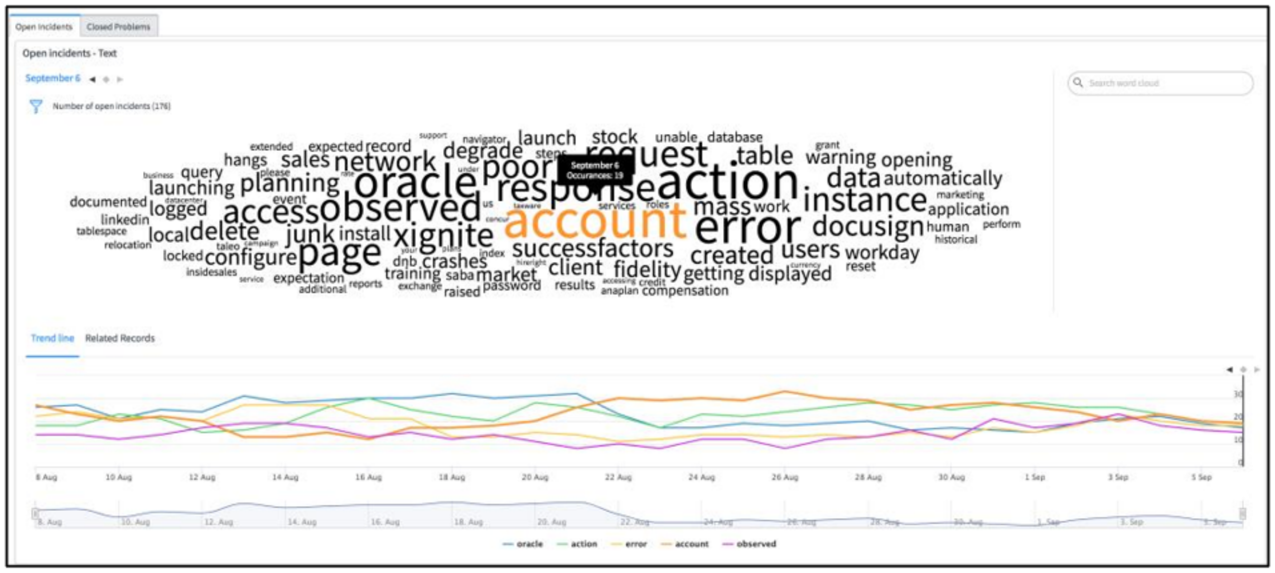
- 9.1 Configuring Text Analytics

In this module, you explore Text Analytics. Text Analytics uses Text Indicators that trend and analyze words in String values in order to make sense out of non-numeric data.

Text Analytics Defined

Text Analytics is the analysis of non-numeric and unstructured data

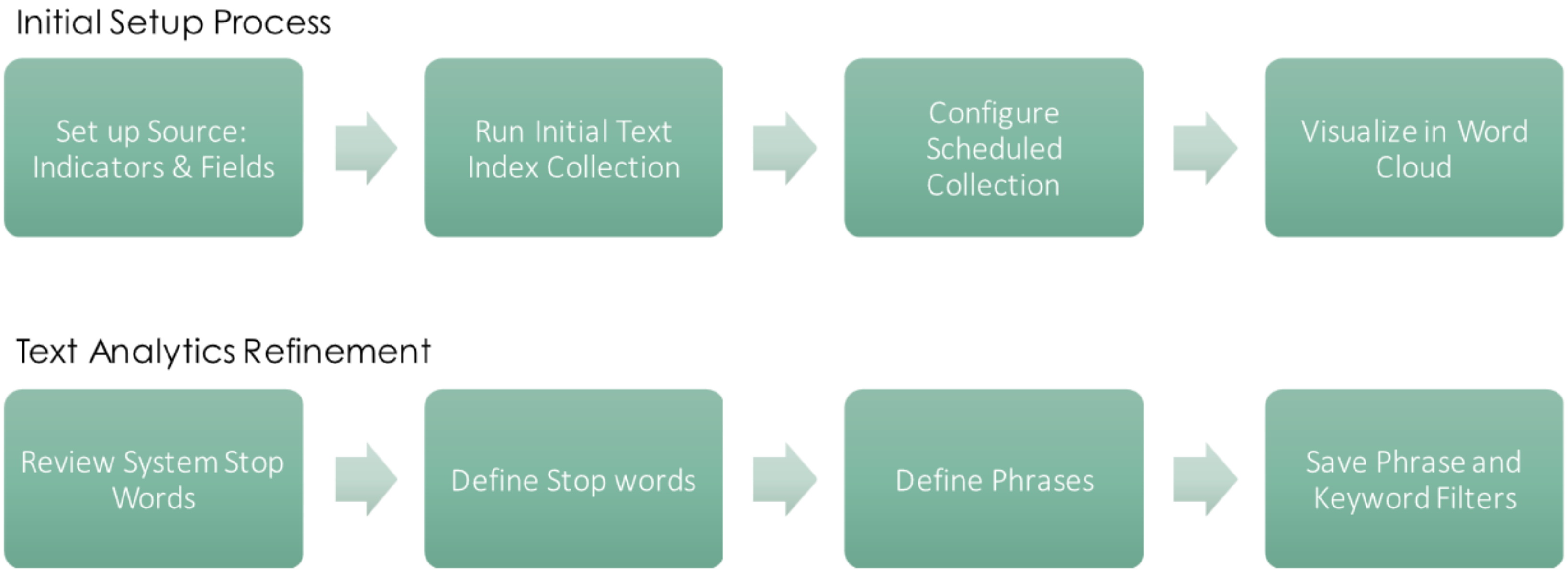
Text Analytics allows to gain insight from Text Fields by analyzing word and phrase patterns and trends



Text Analytics key features:

- Text Analytics is the analysis of unstructured data such as Descriptions, Comments, Notes, or other non-numeric data that is of type String.
- Standard Indicator types such as Automated, Manual, and Formula are geared to trend and analyze numerical data recorded as scores in the pa_scores and pa_snapshots tables.
- The Text Analytics Indicator is a new type of visualization that relies on the creation of Text Indexes during collection time.
- The “Performance Analytics – Premium” plugin must be installed to make this feature available.

Text Analytics Setup



Initial Setup Process

Set up Source: Indicators & Fields → Run Initial Text Index Collection → Configure Scheduled Collection → Visualize in Word Cloud

Text Analytics Refinement

Review System Stop Words → Define Stop words → Define Phrases → Save Phrase and Keyword Filters

The Initial Text Analytics setup involves these steps:

- Define the Sources, Indicators, and the String Fields to analyze
- Run one time historical Text Index collection
- Configure scheduled collection to retrieve the text index
- Create a Widget of Type Text with the Word Cloud visualization and add to a dashboard

The subsequent refinement involves these steps:

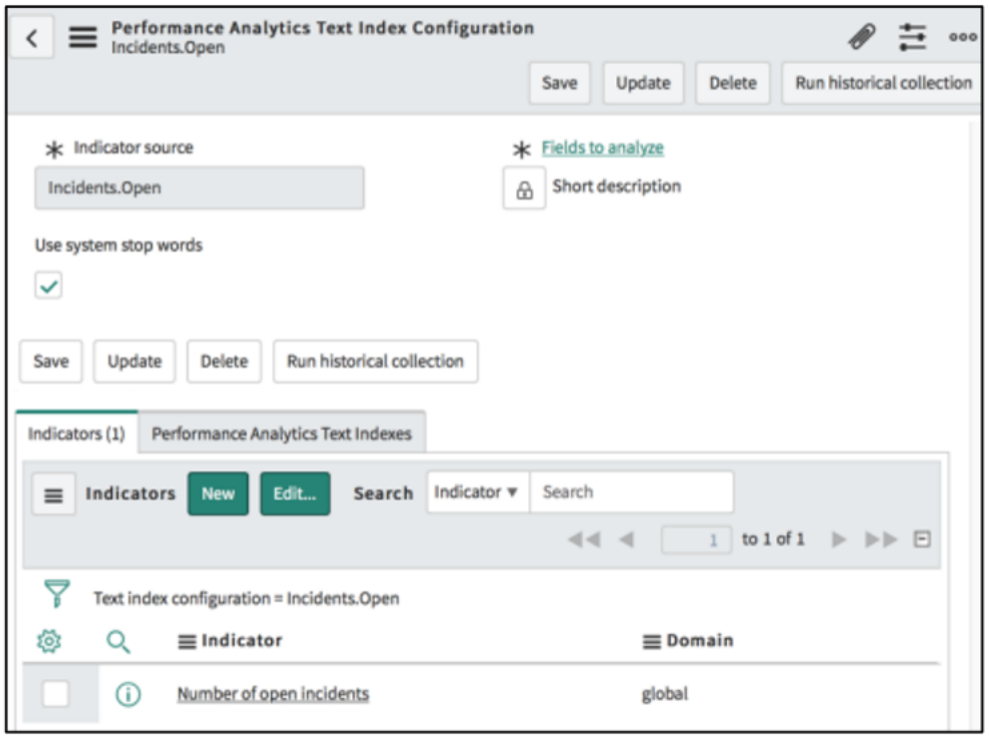
- Review existing system stop words
- Define new stop words and phrases to exclude from the Word Cloud
- Define filters or drilldowns to automatically filter the words in the Word Cloud

Technical Notes:

- Invalid keywords added in the Keyword form will not be filtered out automatically if not applicable to the word cloud
- Only String fields are allowed for indexing.

Define Indicators and Fields for Text Analytics

- Specify the **Indicator source** to perform indexing on
- Check if planning to **Use system stop words**
  - – **Stop Words are trivial words to ignore**
- Select one or more String fields in **Fields to analyze**
- Add Indicators to analyze in the **Indicators** related list


Performance Analytics Text Index Configuration
Incidents.Open
Save
Update
Delete
Run historical collection
Save
Update
Delete
Run historical collection
Indicators (1)
Performance Analytics Text Indexes
Indicators
New
Edit...
Search
Indicator ▾
Search
1 to 1 of 1

Indicator source: Incidents.Open

Fields to analyze: Short description

Use system stop words: ☑

Text index configuration = Incidents.Open

Indicator	Domain
Number of open incidents	global

The first step in the process is to define the Setup Fields. Navigate to **Performance Analytics -> Text Analytics** > **Setup** to begin configuring the following:

- Indicator Source
- Fields to analyze with Text Analytics
  - Common fields are Description, Short Description, or any other String field
- Whether to use system stop words
  - Stop words are common trivial words such as: the, a, etc. that should be ignored

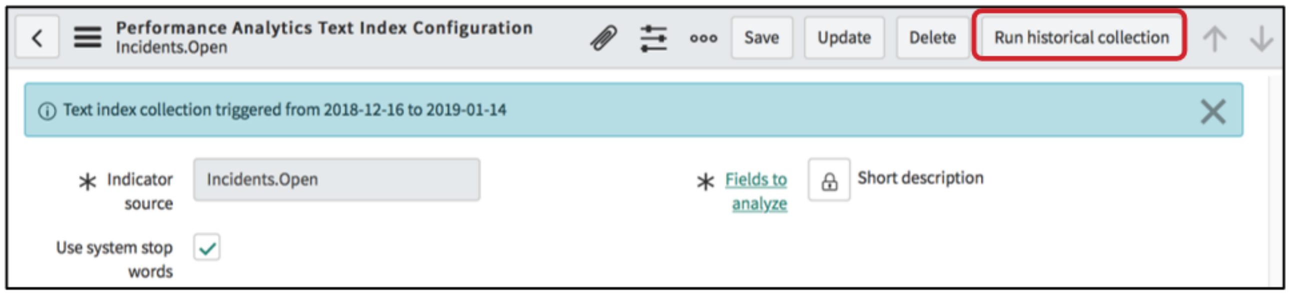
  Note that user-defined stop words can be used with the System stop words. To view system stop words, navigate to **System Definition > Text Index Stop Words**.

- Related list of Indicators

The above setup requires the pa_power_user or pa_admin role.

Run Historical Ad-Hoc Collection + Schedule Daily Collection

Run Historical Collection



Performance Analytics Text Index Configuration
Incidents.Open
Save
Update
Delete
Run historical collection

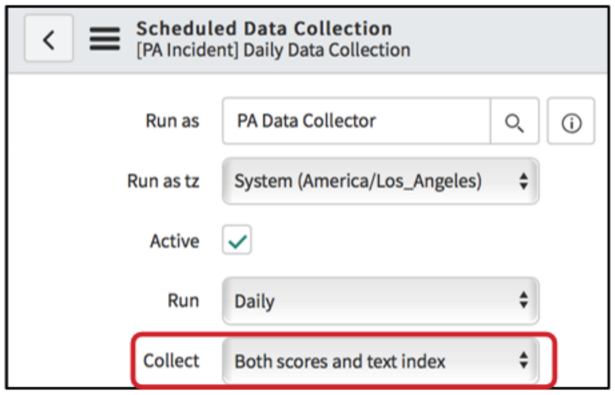
ⓘ Text index collection triggered from 2018-12-16 to 2019-01-14

* Indicator source: Incidents.Open

* Fields to analyze: Short description

Use system stop words: ✓

Configure Regular scheduled Collection (daily):



Scheduled Data Collection
[PA Incident] Daily Data Collection

Run as: PA Data Collector

Run as tz: System (America/Los_Angeles)

Active: ✓

Run: Daily

Collect: Both scores and text index

Scores Only

Disable the collection of all text indexes

Text indexes only

Do not collect scores, only text indexes

Both scores and text index

Collect all scores and all text indexes (Default)

As soon as the Performance Analytics Text Index Configuration is complete, the **Run historical collection** button becomes available. Click **Run historical collection** to populate metrics for the new Text Analytics Indicator(s).

The Collection Job form has been enhanced with these two additional options:

- Text Indexes Only
- Both Scores and Text Index

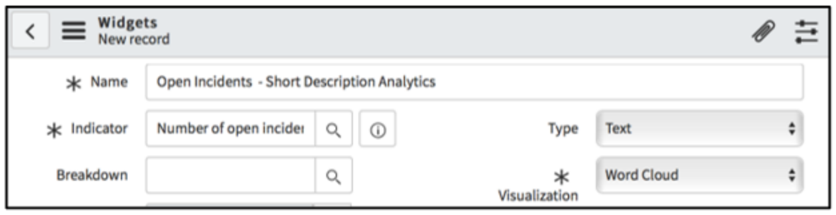
The **Collect** field is set to Both Scores and Text Index by default on z-booted and upgraded instances.

Visualize Text Analytics in Word Cloud

Select Widget of type **Text** and visualization **Word Cloud**

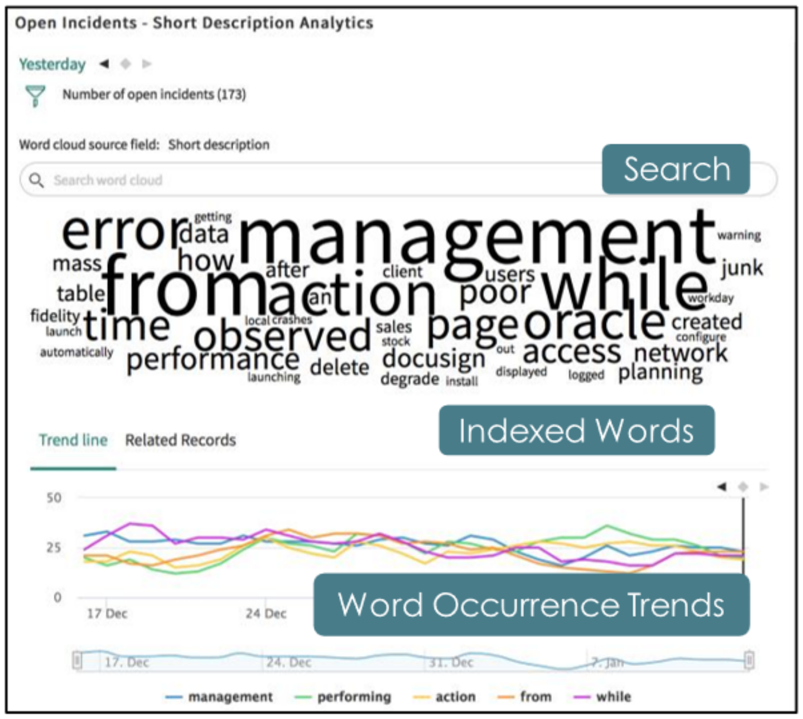
Main Configuration:

- Type/Visualization
- Indicator / Breakdown / Element



Widgets
New record

* Name	Open Incidents - Short Description Analytics
* Indicator	Number of open incidei
Type	Text
Breakdown	
* Visualization	Word Cloud



Open Incidents - Short Description Analytics

Yesterday

Number of open incidents (173)

Word cloud source field: Short description

Search word cloud

Search

Indexed Words

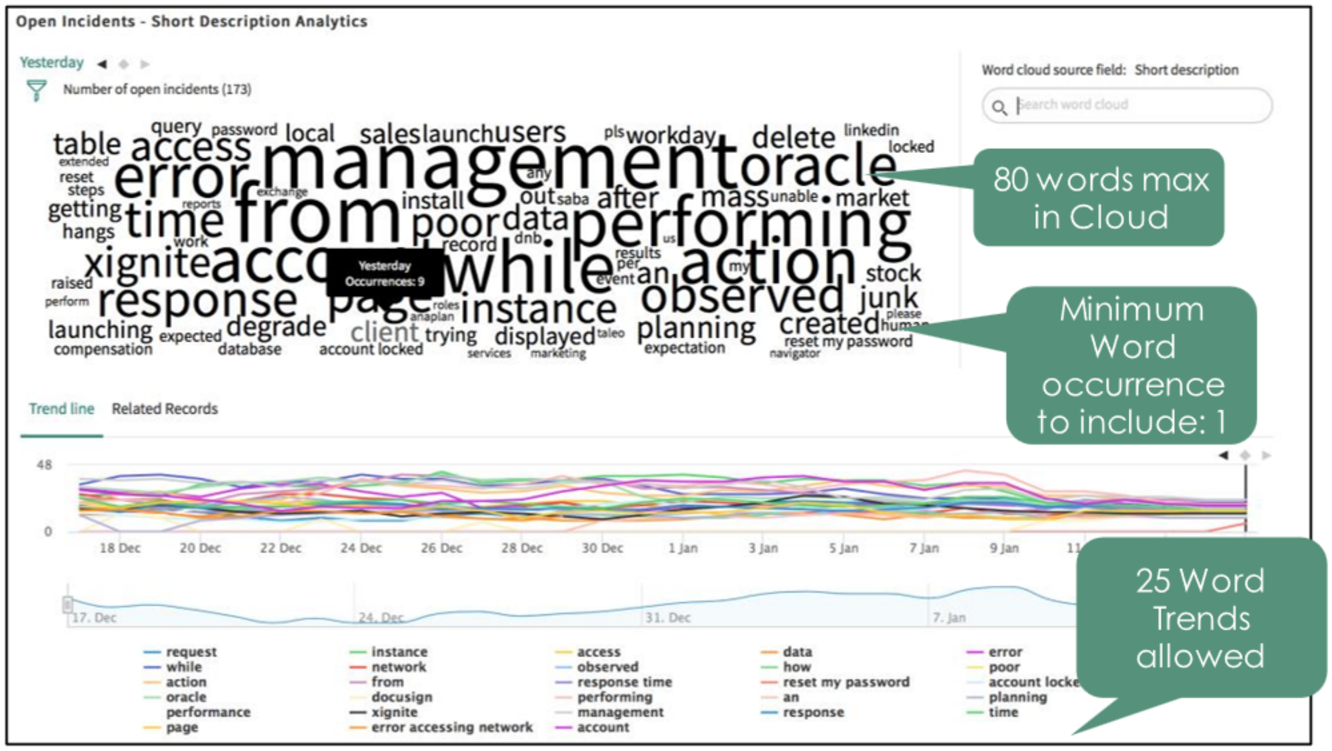
Word Occurrence Trends

The **Text** Widget is a new type of widget introduced to support the Text analytics capability. The available Text visualization is **Word Cloud**.

To create a Text Widget, a Text Analytics Index Configuration must already exist

Word Cloud Display Configuration

- Text Field (when multiple)
- Max Words in Word Cloud
- % or Count for occurrence calculation
- Min word occurrence to include
- # word trends to display



Open Incidents - Short Description Analytics

Yesterday

Number of open incidents (173)

Word cloud source field: Short description

80 words max in Cloud

Minimum Word occurrence to include: 1

25 Word Trends allowed

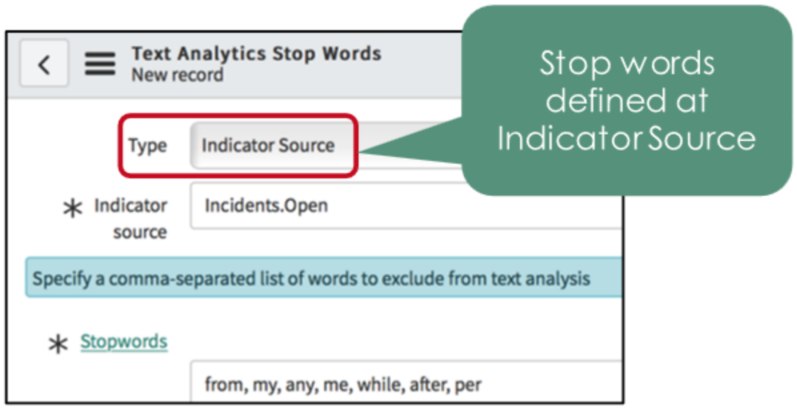
The Word Cloud visualization has an expanded ability to specify the cutoff criteria for showing a word or phrase in a text widget. Previously, you could set only a minimum word occurrence. Now you can define:

- Whether the cutoff value is a numerical count or a relative percentage. For example, you can set the cutoff at 5 occurrences of a word or phrase, or at 10% of all the words in the field being that word or phrase.
- Whether the widget shows words or phrases when there are at least as many occurrences as the cutoff value. In other words, the widget shows only the most common words.
- Whether the widget shows words or phrases when there are no more occurrences than the cutoff value. In other words, the widget shows only the least common words.

Define New Stop Words

Stop Word is a common word which is excluded from Text Analysis

Examples: an, a, the, it's, are, won't, etc.



Text Analytics Stop Words
New record

Type: Indicator Source

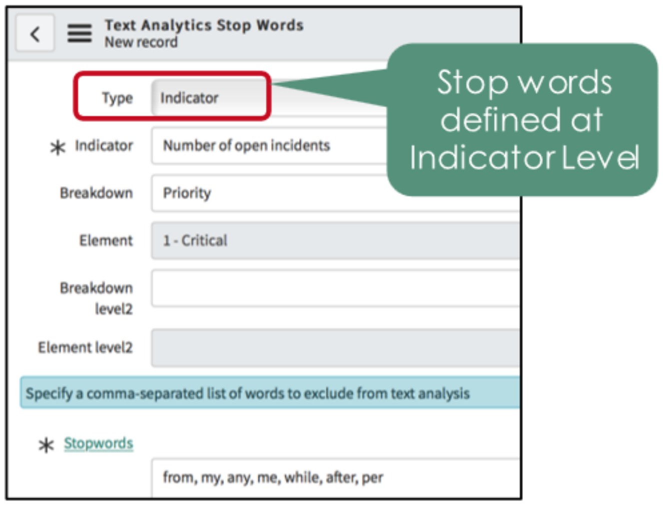
* Indicator source: Incidents.Open

Specify a comma-separated list of words to exclude from text analysis

* Stopwords: from, my, any, me, while, after, per

Stop words defined at Indicator Source

Stop words are removed from data collection to keep the index lean



Text Analytics Stop Words
New record

Type: Indicator

* Indicator: Number of open incidents

Breakdown: Priority

Element: 1 - Critical

Breakdown level2:

Element level2:

Specify a comma-separated list of words to exclude from text analysis

* Stopwords: from, my, any, me, while, after, per

Stop words defined at Indicator Level

Stop words remain in the index but are removed from the Widget

Stop Words are commonly used words that a search engine should ignore, both when indexing entries for searching, and when retrieving a search query. System Stop words exist by default in every instances. Additional Stop words can be configured by the pa_admin or the pa_analyst as follows:

Stop Words on Indicator Source:

- When specified on the indicator source, stop words are removed from data collection to keep the index lean. However, you cannot immediately bring these stop words back into the widget by removing them from the **Stopwords** field. They are not removed until the next data collection.

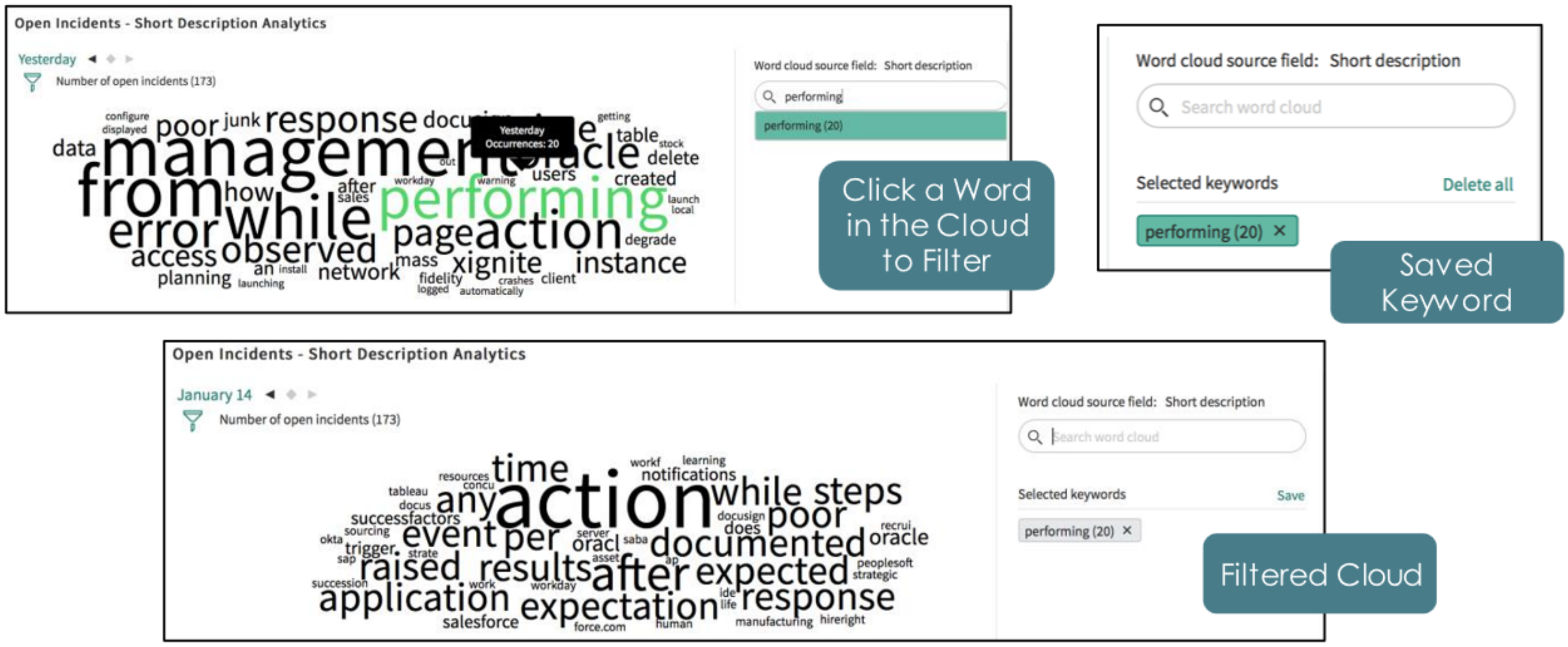
Stop Words on Indicator:

- Stop words that are specified on the indicator remain in the index. These stop words are removed from the widget immediately.

To configure additional stop words for use with PA Text Analytics, navigate to **Text Analytics** > **Stop Words** and click **New**. In the **Stopwords** field, enter a comma-separated list of words to exclude from the text analysis.

Keywords

Keywords are saved word cloud Searches



Open Incidents - Short Description Analytics

Yesterday

Number of open incidents (173)

Word cloud source field: Short description

performing

performing (20)

Click a Word in the Cloud to Filter

Word cloud source field: Short description

Search word cloud

Selected keywords — Delete all

performing (20) ×

Saved Keyword

Open Incidents - Short Description Analytics

January 14

Number of open incidents (173)

Word cloud source field: Short description

Search word cloud

Selected keywords — Save

performing (20) ×

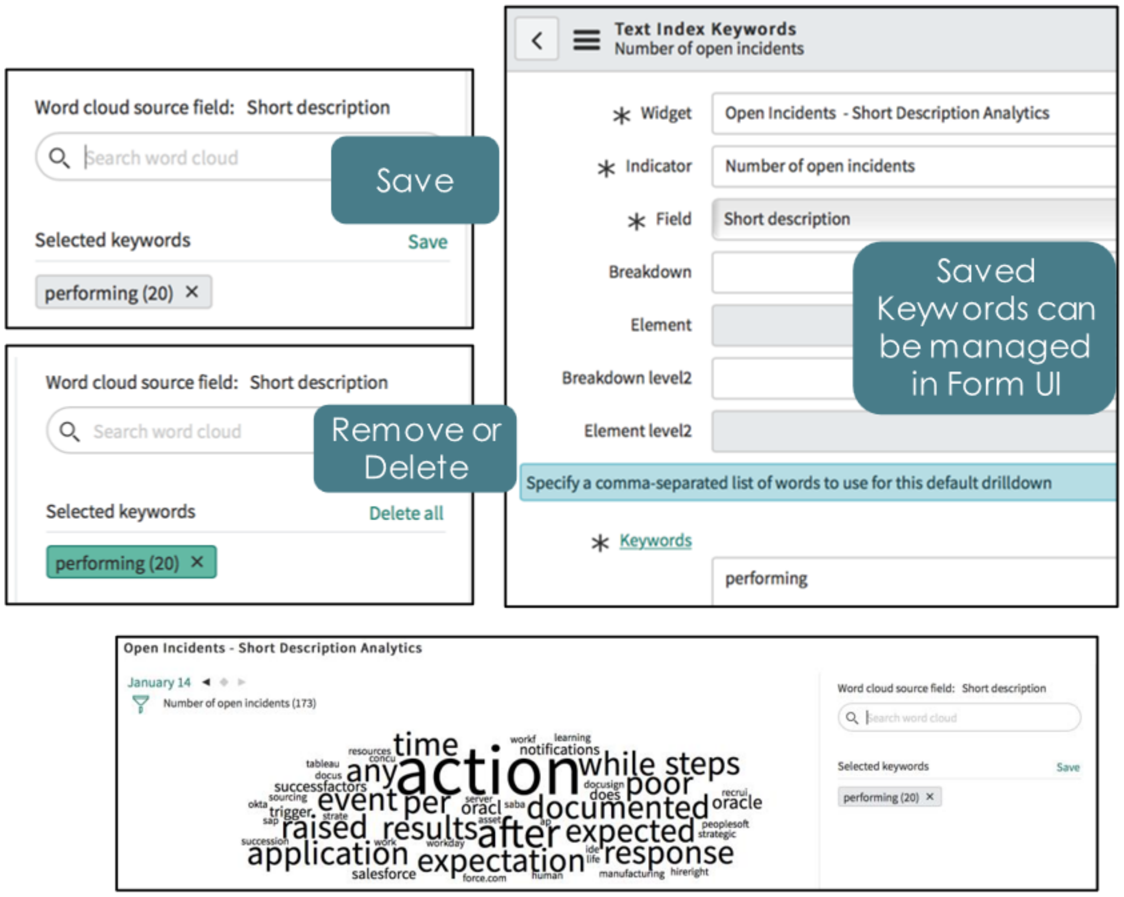
Filtered Cloud

To filter a word cloud by keywords, click the words in the cloud. You can save a list of the keywords, which will be used whenever someone views the widget. Saved keywords always filter a text analytics widget. You can save them directly on the widget in a dashboard, choosing from the words in the word cloud.

Users with the roles pa_analyst role can define keywords from a form by navigating to **Text Analytics** > **Keywords** > **New**. Saved keywords can be removed while viewing the Word Cloud and deleted by users with the pa_analyst role.

Managing Keywords

- Save a Search word to create a keyword
- Remove a Keyword from the current filter
- Delete all Keywords
- Upon reload, the widget reverts to the saved keywords
- User selection is not persisted



Word cloud source field: Short description

Search word cloud

Selected keywords Save

performing (20) ×

Save

Word cloud source field: Short description

Search word cloud

Selected keywords Delete all

performing (20) ×

Remove or Delete

Text Index Keywords
Number of open incidents

* Widget	Open Incidents - Short Description Analytics
* Indicator	Number of open incidents
* Field	Short description
Breakdown	
Element	
Breakdown level2	
Element level2	

Specify a comma-separated list of words to use for this default drilldown

* Keywords: performing

Saved Keywords can be managed in Form UI

Open Incidents - Short Description Analytics

January 14

Number of open incidents (173)

Word cloud source field: Short description

Search word cloud

Selected keywords Save

performing (20) ×

Selecting a keyword acts as a filter for indexed text fields that only contain the selected keywords. Perform the following to define the default keyword from the widget UI:

- Select a keyword either by clicking on a word, or by searching the word cloud in the search box
- Click **Save**

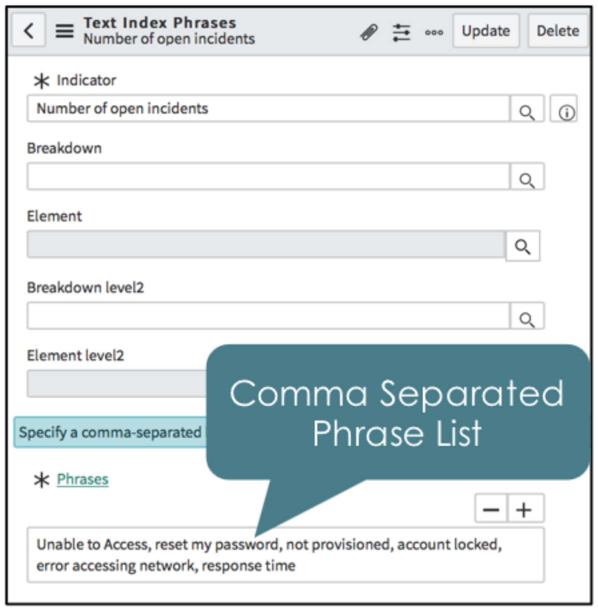
Note that when applying breakdown and element combination, the selected keywords will be stored for that specific combination of breakdowns and elements.

Users can modify or delete the saved keywords from the UI using the **Delete all** action according to their role:

- pa_viewers can change selected keywords as desired to perform analysis or drilldown by other keywords
- pa_viewers cannot share keywords they select with other users

Phrases to Include in the Word Cloud

Phrases are saved multi-word Keywords used as Word Cloud filters



Text Index Phrases
Number of open incidents

Update | Delete

* Indicator: Number of open incidents

Breakdown:

Element:

Breakdown level2:

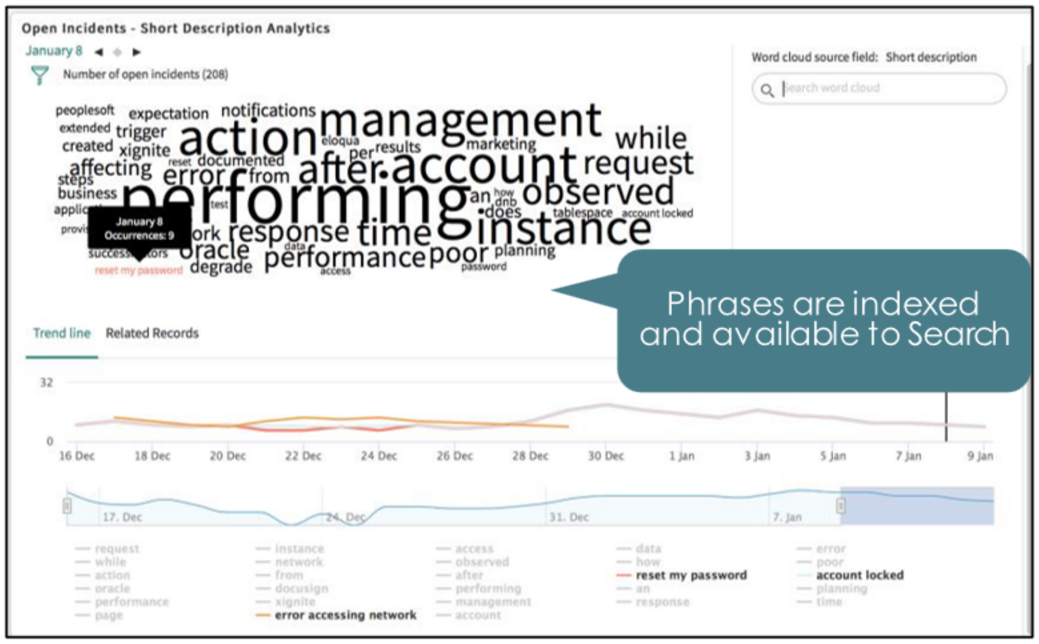
Element level2:

Specify a comma-separated

* Phrases

Unable to Access, reset my password, not provisioned, account locked, error accessing network, response time

Comma Separated Phrase List



Open Incidents - Short Description Analytics

January 8

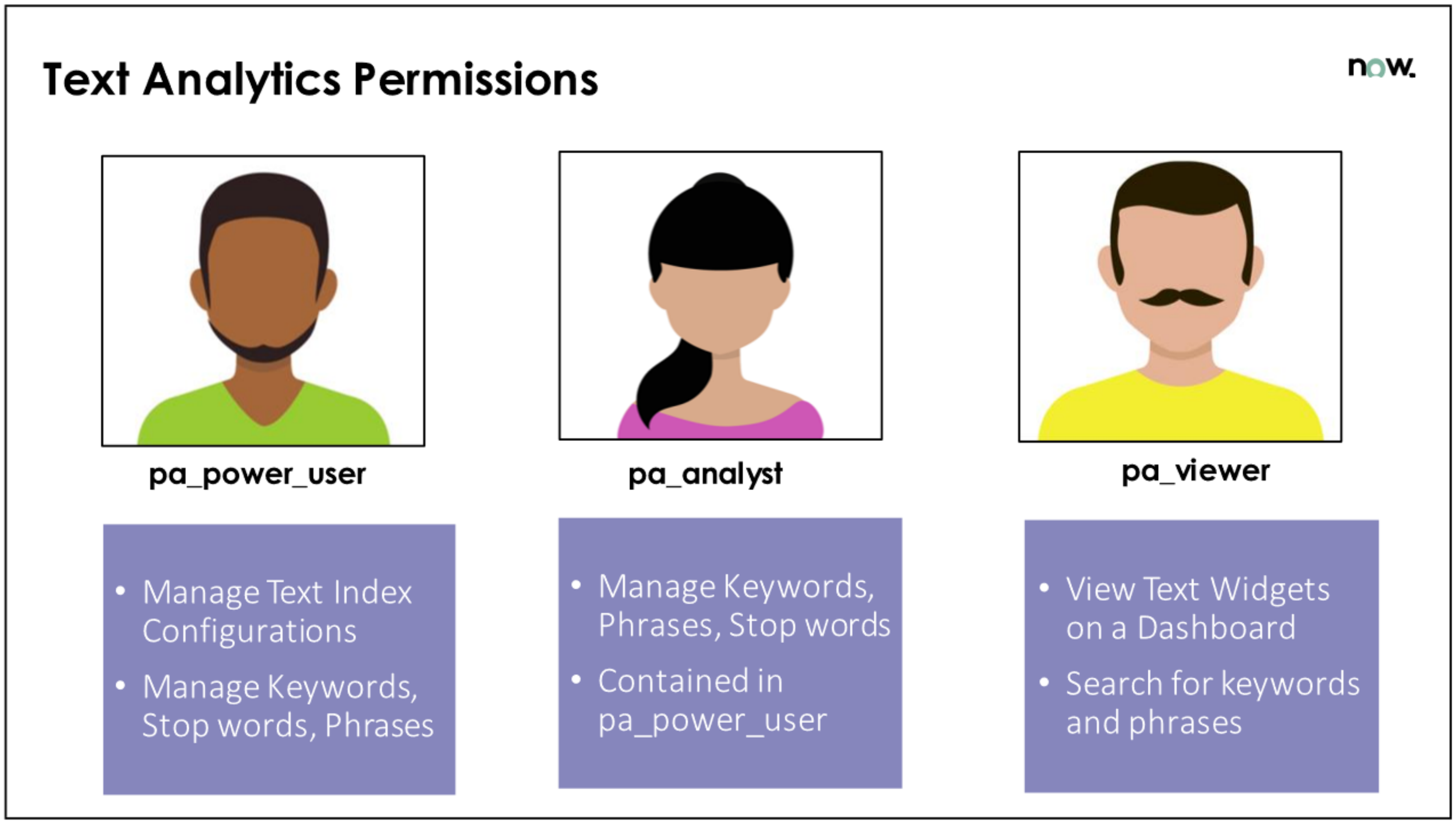
Number of open incidents (208)

Word cloud source field: Short description

Phrases are indexed and available to Search

Users with the pa_analyst role can specify phrases that text analytics searches for, instead of searching for only the most frequent individual words. Phrases are indexed and searched just like the Keywords.

Note that phrases can be created only from the form view by navigating to **Performance Analytics > Text Analytics > Phrases.** The phrases are defined as a comma separated list.



Text Analytics Permissions

pa_power_user

- Manage Text Index Configurations
- Manage Keywords, Stop words, Phrases

pa_analyst

- Manage Keywords, Phrases, Stop words
- Contained in pa_power_user

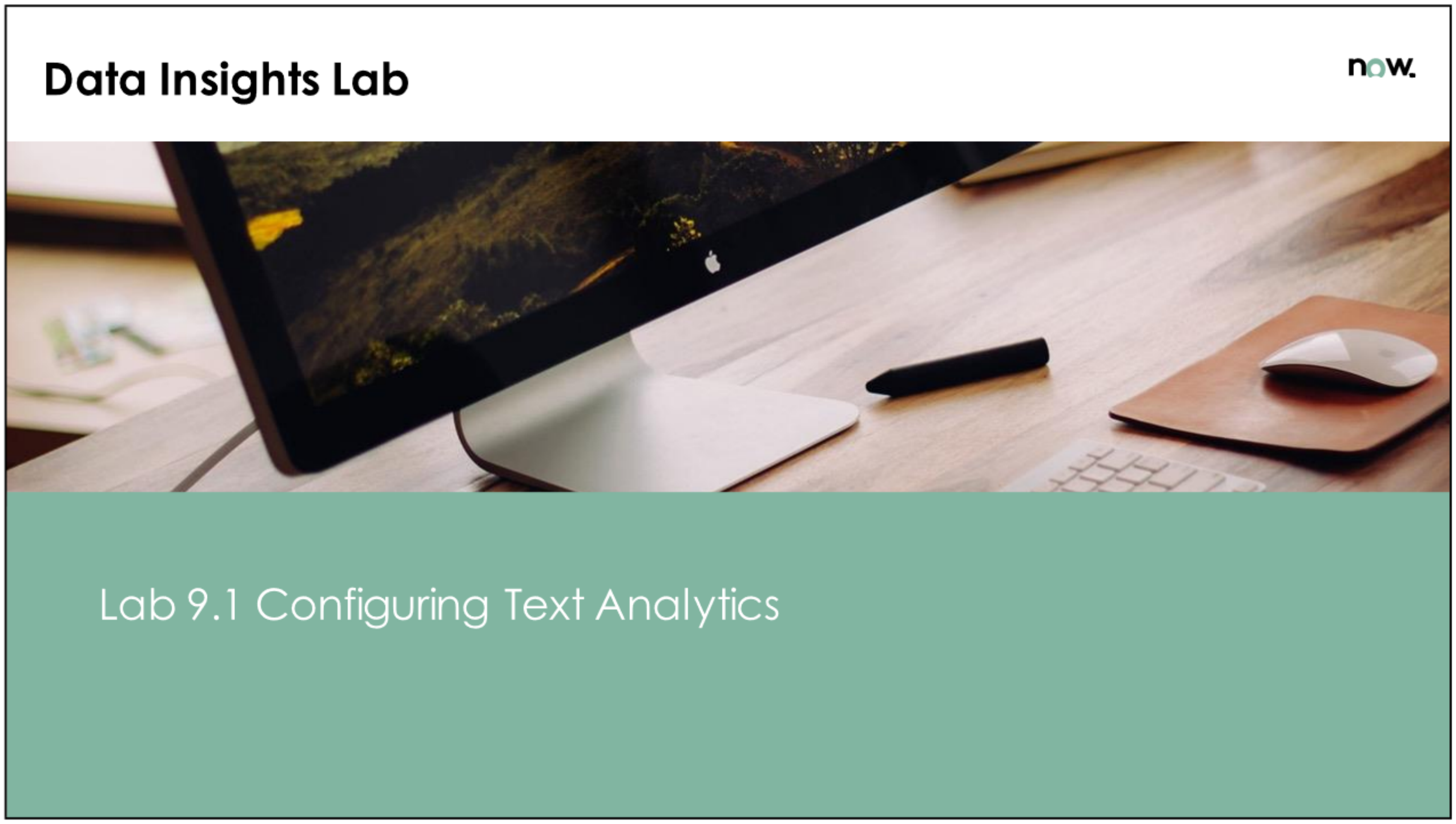
pa_viewer

- View Text Widgets on a Dashboard
- Search for keywords and phrases

The new Performance Analytics role **pa_analyst** has been introduced that can:

- Define Stop words
- Select/save/delete Keywords
- Manage Phrases

The pa_analyst is contained by the pa_power_user and pa_admin roles. This role would be assigned to individual fulfillers or groups whose expertise includes keywords, phrases, and stop words for word clouds.



Data Insights Lab

Lab 9.1 Configuring Text Analytics

Text Analytics

Lab 9.1

⌛15m

Lab Objectives

Incidents are sometimes initially assigned to the wrong team which leads to resolution delays. The Customer Success Manager at Cloud Dimensions requires a deeper insight into open incidents that ignores the standard categorization of Priority, Assignments, and Category, and focuses on the Description itself.

Text analytics on the Incident Description is needed to detect frequently recurring words and patterns that can provide additional info about common issues.

This lab will show you how to do perform word pattern analysis of Incident Descriptions using Text Analytics Indicators

A. Text Analytics Requirements

Indicator and Field	Indicator Source	Words and phrases to ignore
Number of Open Incidents – Short description	Incidents.Open	a, an, has, have, our, my, the, not working, is broken, not accessible, is down, mail server, password reset, response time

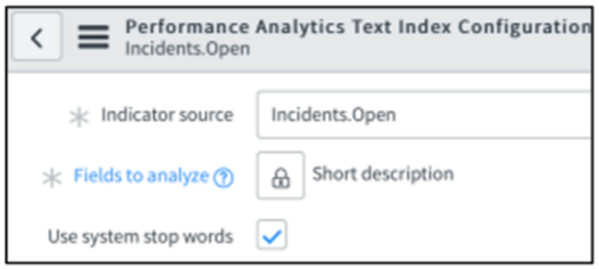
B. Text Analytics Configuration

Text Index Configuration

1. Navigate to **Performance Analytics >Text Analytics > Setup**.

2. Click **New** to add a new **Performance Analytics Text Index Configuration** as follows:

Indicator source: **Incidents.Open**
Fields to analyze: **Short description**
Use system stop words: **checked**

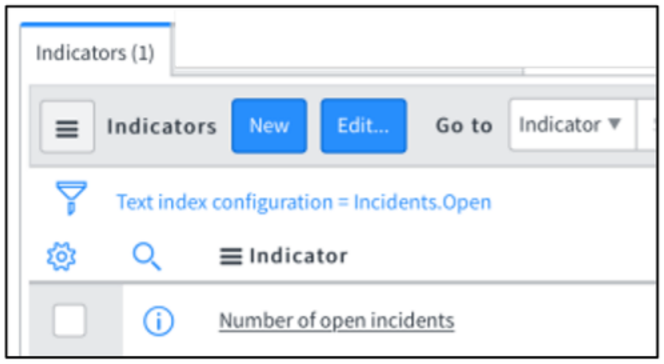


Performance Analytics Text Index Configuration
Incidents.Open

* Indicator source: Incidents.Open
* Fields to analyze: Short description
Use system stop words: ☑

3. **Save** the configuration.

4. Click **Edit…** and add the **Number of open incidents** indicator as shown:



Text index configuration = Incidents.Open

Number of open incidents

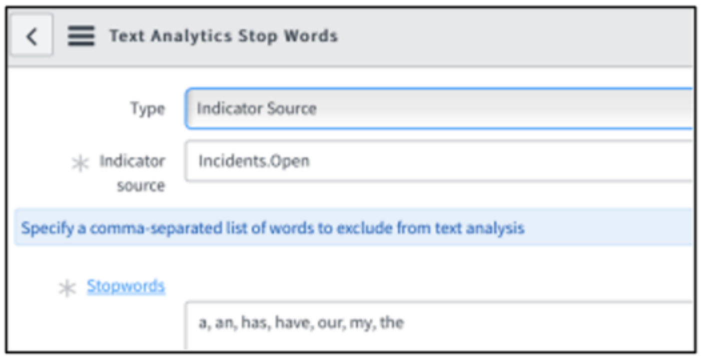
5. Confirm that the **Run historical collection** button is now visible but **do not run collection yet!**

Stop Words Configuration

1. Navigate to **Performance Analytics > Text Analytics > Stop Words**.

2. Click **New** to create the following configuration:

   - Type: **Indicator Source**
   - Indicator source: **Incidents.Open**
   - Stopwords: **a, an, has, have, our, my, the**



Type: Indicator Source

Indicator source: Incidents.Open

Specify a comma-separated list of words to exclude from text analysis

Stopwords: a, an, has, have, our, my, the

3. Click **Submit**.

Note: *The above words will be removed from the Index during collection.*

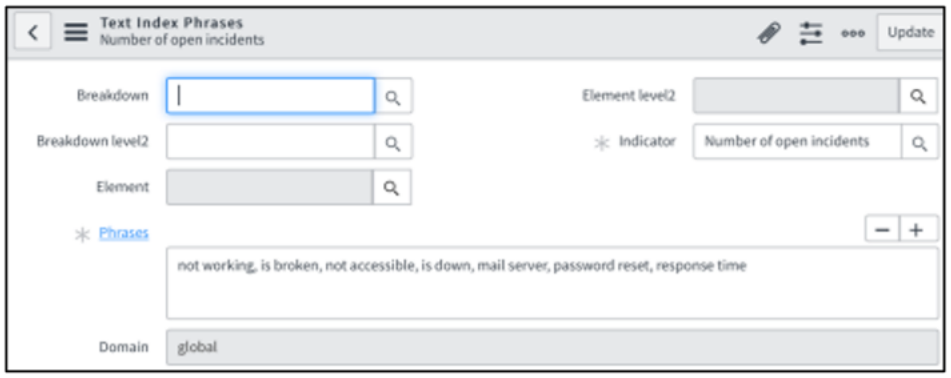
Phrases Configuration

1. Navigate to **Performance Analytics > Text Analytics > Phrases.**

2. Click **New** to create the following Text Index Phrases configuration:

   - Indicator: **Number of open incidents**

- Phrases: **not working, is broken, not accessible, is down, mail server, password reset, response time**



Text Index Phrases
Number of open incidents

Breakdown:
Breakdown level2:
Element:
Element level2:
* Indicator: Number of open incidents
* Phrases: not working, is broken, not accessible, is down, mail server, password reset, response time
Domain: global

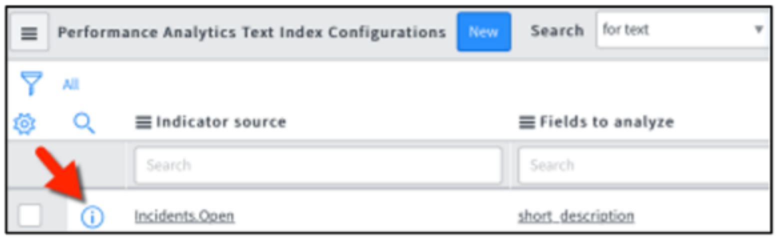
3. Click **Submit**.

> ***Note:*** *The above phrases will be available to search for in the Word Cloud.*

Text Analytics Collection

1. Navigate to **Performance Analytics >Text Analytics > Setup**.

2. Click the **Info** icon to open the Text Index Configuration.



Performance Analytics Text Index Configurations

Search: for text

Indicator source	Fields to analyze
Incidents.Open	short_description

3. Click **Run historical collection** to trigger a one-time/1-month **Text indexes only** collection for the Number of open incidents indicator.

> ***Note:*** *The application responds with a Job confirmation message.*

4. Navigate to **Performance Analytics > Data Collector > Jobs**.

5. View **All** collection jobs.

6. Review the **Temporary job for collecting text indexes for Incidents.Open** job.
   - Note the value of the **Collect** property.
   - Note that no new scores have been inserted as the job only parses text indexes.

C. Text Analytics Visualization

1. Navigate to the **Customer Success Advocate** dashboard.

2. **Clear all** Breakdown Source filters.

3. Create a new tab and name it **Incident Description Analytics**.

4. Add a new Performance Analytics Widget of type **Text** as follows:

   Name: **Incident Descriptions**

   Type: **Text**

   Visualization: **Word Cloud**

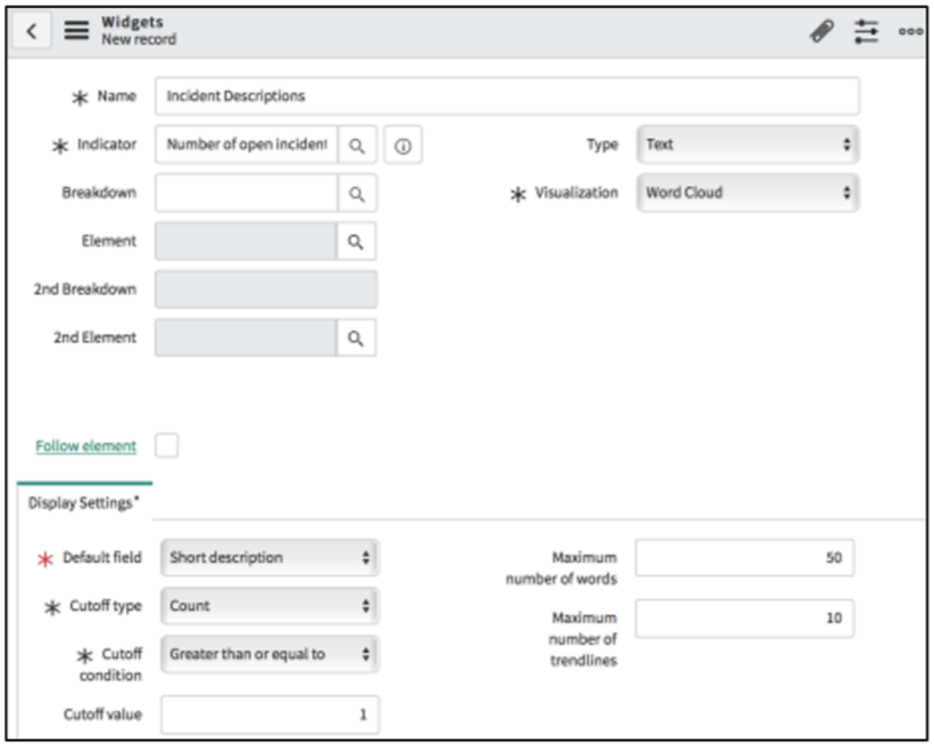
   Indicator: **Number of Open Incidents**

   Default Field: **Short description**

   Cut-off value: **1**

   Number of words in Word Cloud: **50**

   Number of words in Trend-line: **10**

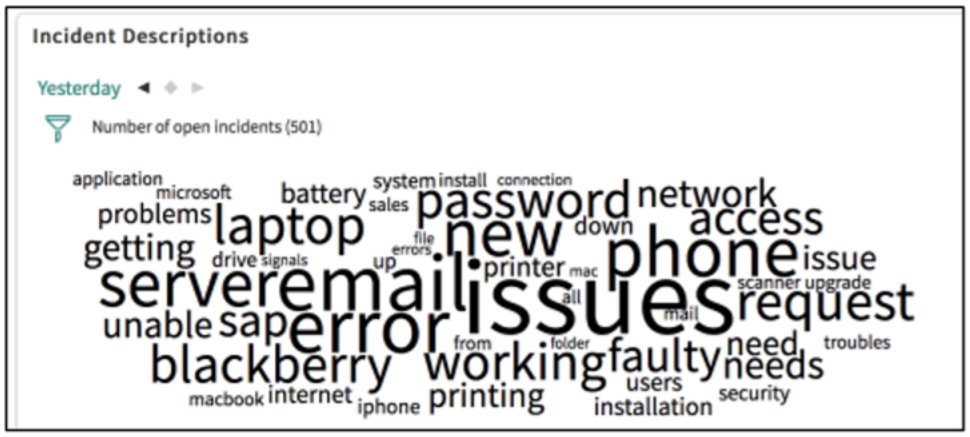


* Name	Incident Descriptions		
* Indicator	Number of open incident	Type	Text
Breakdown		* Visualization	Word Cloud
Element			
2nd Breakdown			
2nd Element			
Follow element	☐		

Display Settings*

* Default field	Short description	Maximum number of words	50
* Cutoff type	Count	Maximum number of trendlines	10
* Cutoff condition	Greater than or equal to		
Cutoff value	1		

5. Click **Submit** and confirm the correct display of a Word Cloud widget:


Incident Descriptions

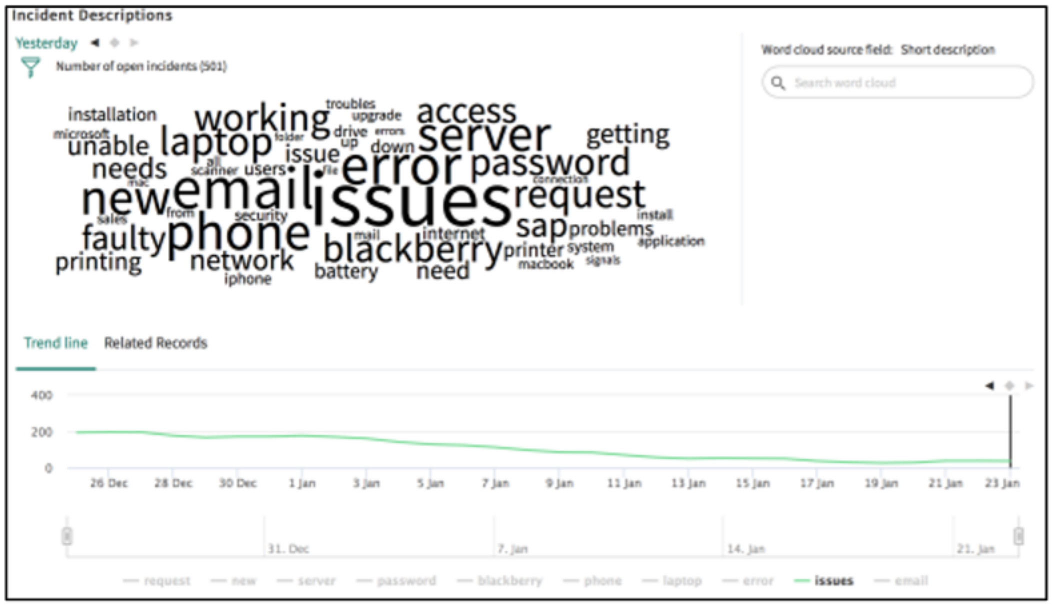
Yesterday

Number of open incidents (501)

D. Text Analysis

1. Review the **Trend line** under the Word Cloud.

   - How many trend lines are shown?
   - Which is the most frequently encountered word in Incident descriptions?



Incident Descriptions

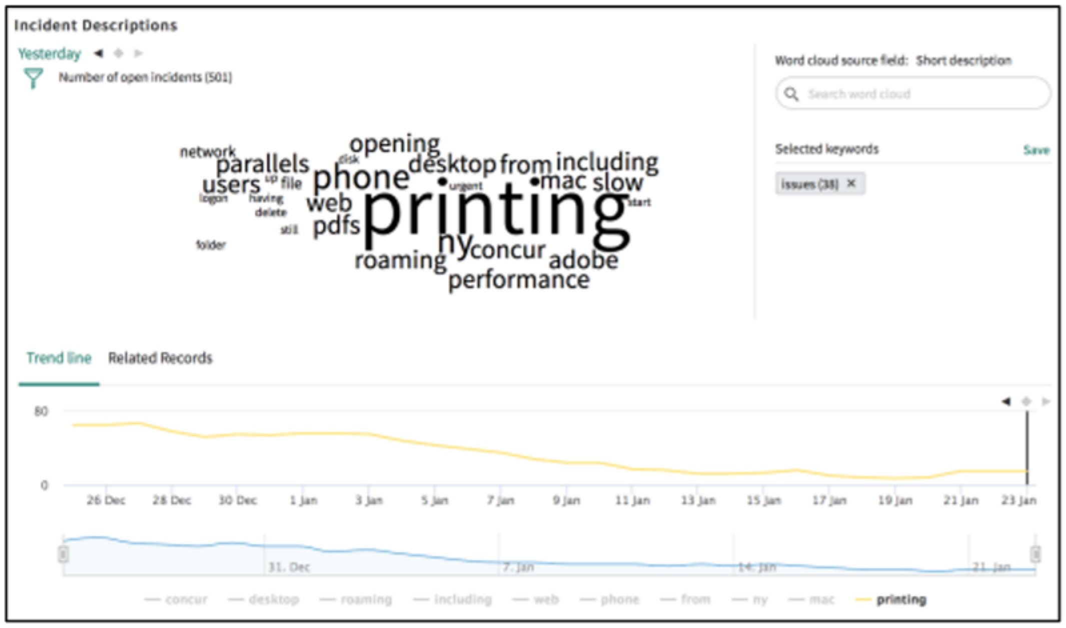
Yesterday

Number of open incidents (501)

Word cloud source field: Short description

2. Select the **most common** key word in the Word Cloud. (**issues** in our example.)

> ***Note:*** *The list is now filtered to the records that have "issues" in their description. The Word cloud contains the most frequently encountered words in the respective descriptions.*



Incident Descriptions

Yesterday

Number of open incidents (501)

Word cloud source field: Short description

Selected keywords

issues (38) ×

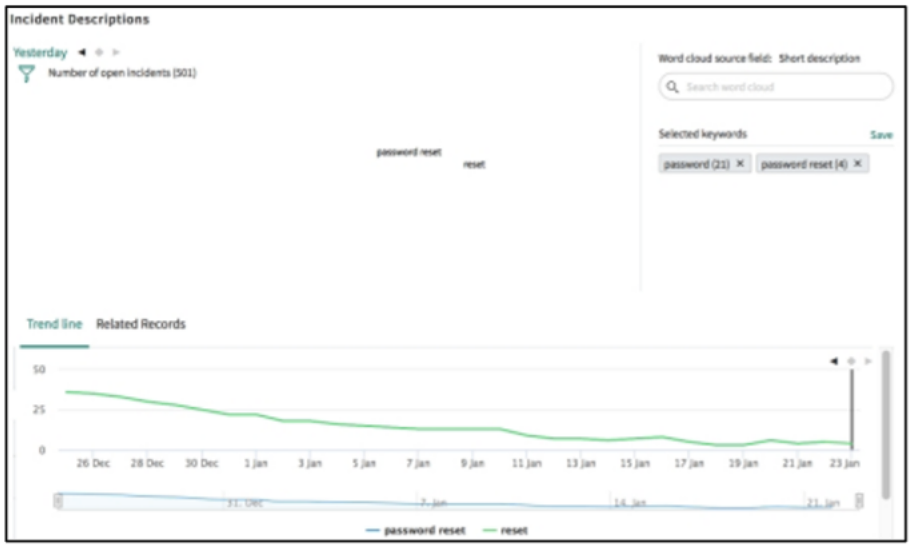
- Which is the most recurring word for Incidents with **issues** in their description?

- What can you tell about the nature of reported incidents based on this analysis?

3. Clear all **key words**.

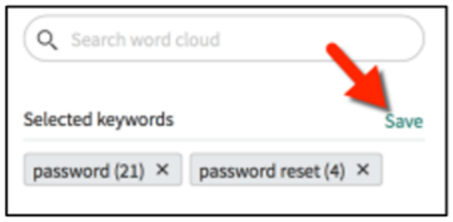
4. Search for the word **password**.

5. Without removing the **password** keyword, search for **password reset**.



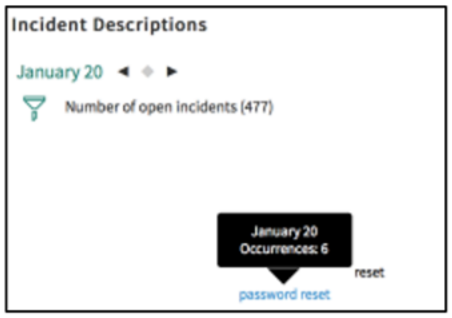
6. Navigate to the **Related Records** list to view the incident records that have **password reset** in their short description.

7. **Save** the selected keywords.



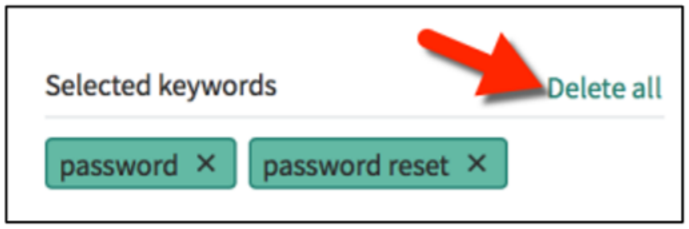
__Tip:__ Once saved, keywords (and phrases) are automatically applied to the Word Cloud.

8. Navigate back a few days using the **Go to Previous Day** arrow. The Word Cloud content is automatically filtered to apply the key words saved earlier.

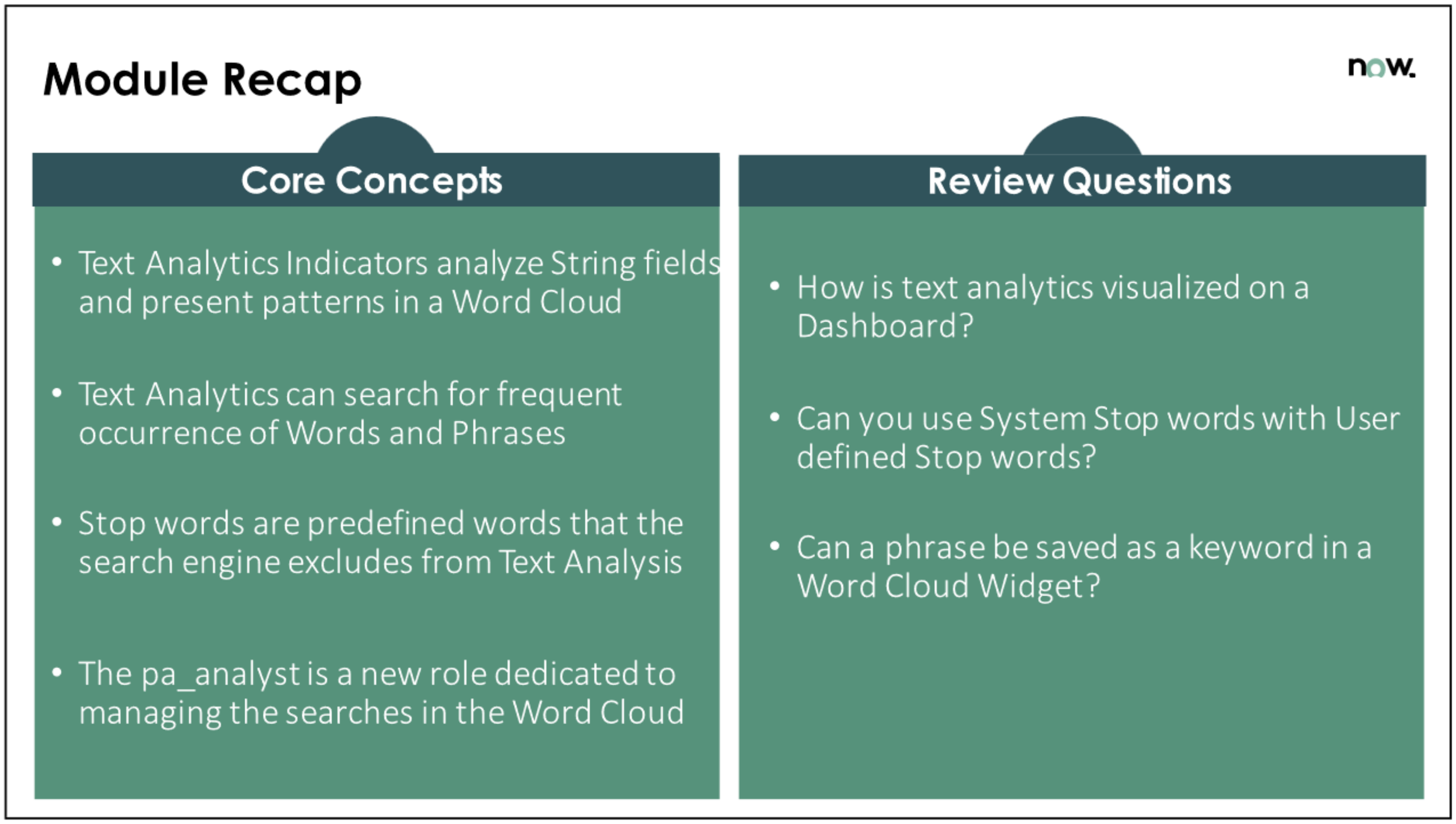


- How many occurrences of **password reset** are there on the selected day?

9. Select **Delete All** to remove the saved keywords.



You have now completed the Text Analytics lab.



Module Recap

now.

Core Concepts

- Text Analytics Indicators analyze String fields and present patterns in a Word Cloud
- Text Analytics can search for frequent occurrence of Words and Phrases
- Stop words are predefined words that the search engine excludes from Text Analysis
- The pa_analyst is a new role dedicated to managing the searches in the Word Cloud

Review Questions

- How is text analytics visualized on a Dashboard?
- Can you use System Stop words with User defined Stop words?
- Can a phrase be saved as a keyword in a Word Cloud Widget?

www.servicenow.com/services/training-and-certification

servicenow

4810 Eastgate Mall, San Diego, CA 92121, USA • (858) 720-0477 • (858) 720-0479 • www.servicenow.com

ServiceNowbelievesinformation in this publication is accurate as of its publication date. This publication could include technical inaccuracies or typographical errors. The information is subject to change without notice. Changes are periodically added to the information herein; these changes will be incorporated in new additions of the publication. ServiceNow may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time. Reproduction of this publication without prior written permission is forbidden. Then information in this publication is provided “as is”. ServiceNow makes no representations or warranties of any kind, with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.