

# Azure Log Analytics and the Kusto Query Language

Collect, Query and Analyze Your Data

# Agenda

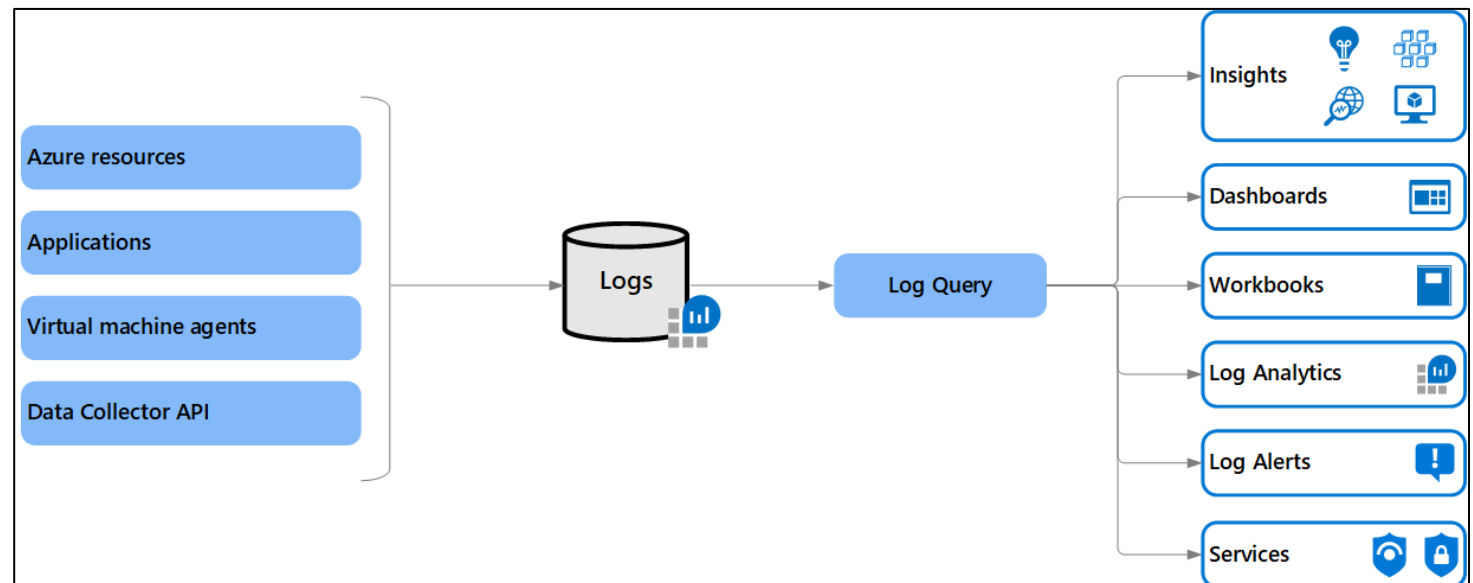
---

- Azure Monitor Logs
- Log Analytics Workspace
- Data Collection
- Log Data Structure
- Kusto Query Language



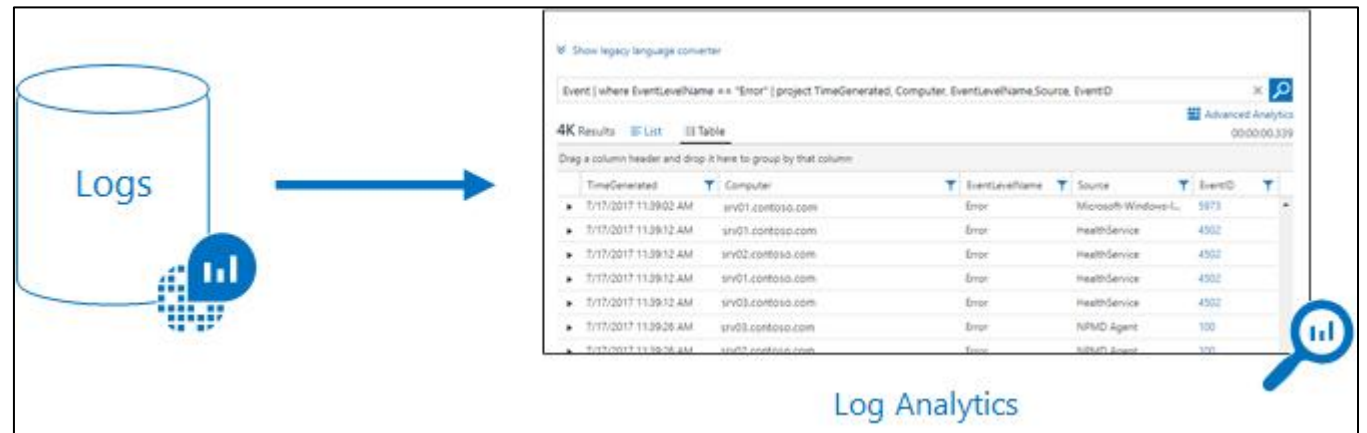
# Azure Monitor Logs

- Collect log and performance data
  - From multiple sources
- Act on logged information
  - Analyze
  - Alert
  - Visualize
  - Get insights
  - Retrieve
  - Export



# Log Analytics

- Analyze log data with queries
- Kusto query language
  - Simple log queries
  - Advanced functionality
    - Aggregations
    - Joins
    - Smart analytics



- Configure sources to send data
  - Azure resources
    - Create diagnostic settings
  - Collect data from VMs
    - Enable VM Insights
  - Collect more events and performance data
    - Configure data sources
- Data collected is stored in Log Analytics workspace

- Container where data is
  - Collected
  - Aggregated
  - Analyzed
  - Presented
- Manage sets of data
  - Collected from your entire IT infrastructure
    - Cloud
    - On-prem

- Provides
  - Geographic location for data storage
  - Data isolation
    - Access rights
  - Scope for configuration settings
    - Pricing tier
    - Retention
    - Data capping

# Pricing

- Data ingestion

- Capacity reservations
  - Fixed predictable fee

Capacity	Price
100 GB per day	€233,45 per day
200 GB per day	€438,30 per day
300 GB per day	€643,16 per day
400 GB per day	€838,49 per day
500 GB per day	€1030,24 per day
1000 GB per day	€2024,75 per day

- Pay-As-You-Go
  - Billed per GB of data uploaded to service

	Free units	Price
Data ingestion	5 GB per month	€2,761 per GB



- Data Retention
  - Data is retained for
    - 31 days
      - No extra charge
    - 90 days
      - If Sentinel is enabled
    - 90 days
      - For Application Insights data


Feature	Free Units	Price
Data retention	31 days	€0,121 per GB per month

# Create Workspace

- Azure Portal
  - Create new resource
  - Log analytics workspace
    - Name
    - Resource group
    - Region

## Create Log Analytics workspace

[Basics](#) [Pricing tier](#) [Tags](#) [Review + Create](#)

 A Log Analytics workspace is the basic management unit of Azure Monitor Logs. There are specific considerations you should take when creating a new Log Analytics workspace. [Learn more](#)

×

With Azure Monitor Logs you can easily store, retain, and query data collected from your monitored resources in Azure and other environments for valuable insights. A Log Analytics workspace is the logical storage unit where your log data is collected and stored.

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Visual Studio Premium met MSDN

▼

Resource group \* ⓘ

▼

[Create new](#)

### Instance details

Name \* ⓘ

Region \* ⓘ

North Europe

▼

# Data Collection - VMs

---

- Agent must be deployed
  - Connected to log analytics workspace
- 2 agents
  - Azure Monitor Agent
  - Log Analytics Agent
    - Legacy

# Azure Monitor Agent

---

- On-prem VM
  - Install Azure Arc agent
- Create data collection rule set
  - Supported systems
    - Azure VMs
    - Azure Arc enabled VMs
  - Define what to collect
    - Logs
    - Performance counters



# Data Collection – Azure Resources

- Go to Azure resource
  - Monitoring – Diagnostic settings
    - Add diagnostic setting
      - Select Log categories
      - Select Metrics
      - Destination details
        - Log Analytics workspace

### Diagnostic setting

Save Discard Delete Feedback

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)

Diagnostic setting name \* saels0 ✓

#### Logs

Categories

☒ StorageRead

☒ StorageWrite

☒ StorageDelete

#### Metrics

☒ Transaction

#### Destination details

☒ Send to Log Analytics workspace

Subscription  
Visual Studio Premium met MSDN

Log Analytics workspace  
asels1 ( westeurope )

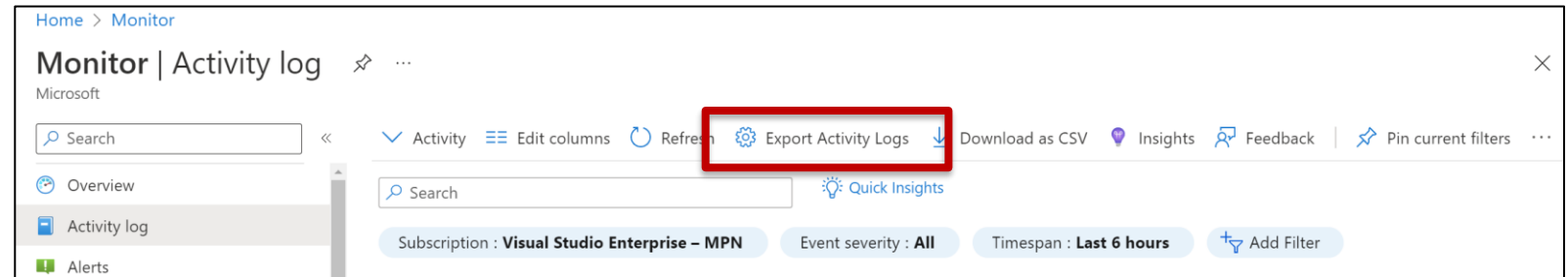
☐ Archive to a storage account

☐ Stream to an event hub

☐ Send to partner solution

# Data Collection – Activity Log

- Azure Monitor
  - Activity log
    - Export Activity Logs



### Diagnostic setting

Save Discard Delete Feedback

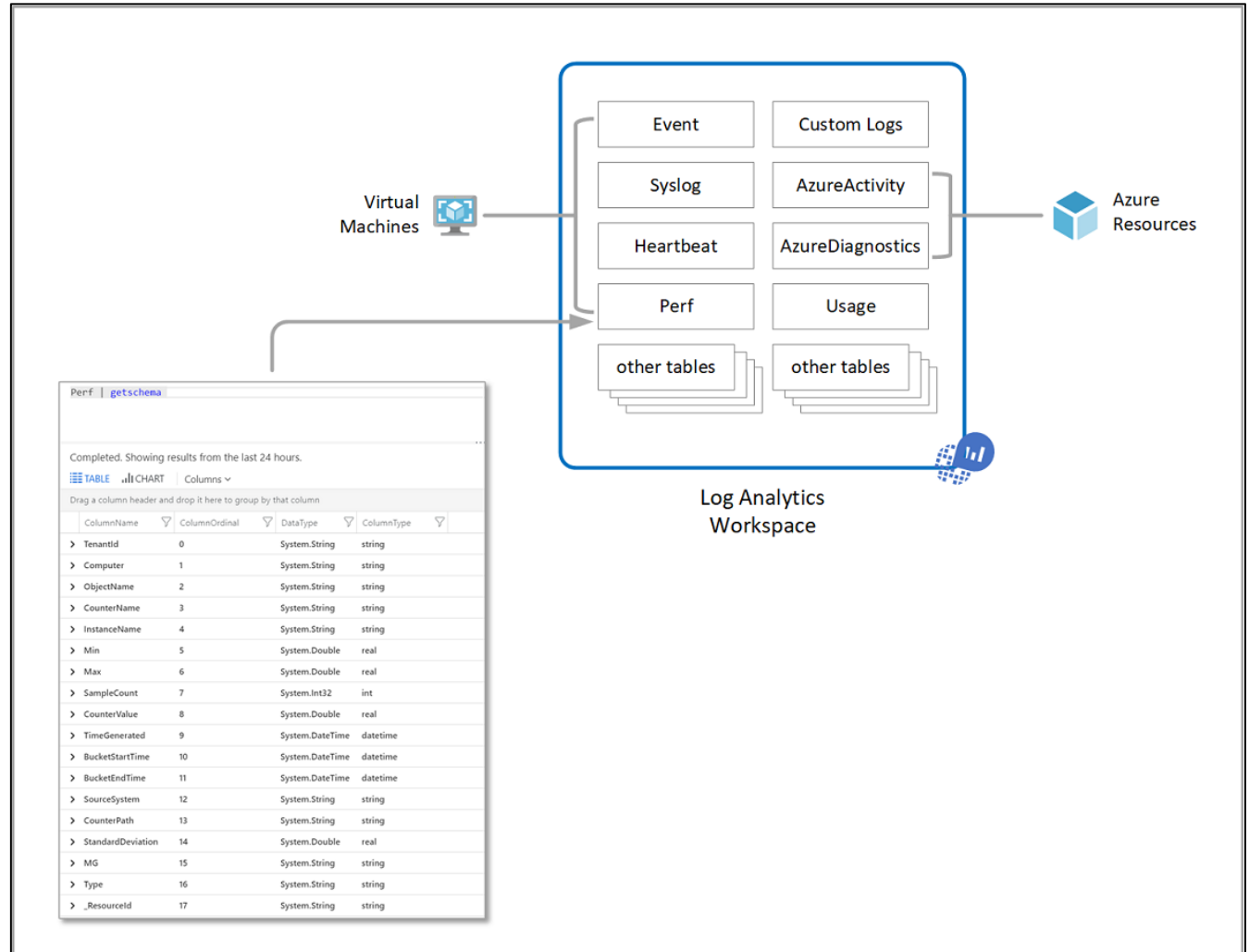
A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a subscription, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. [Learn more about the different log categories and contents of those logs](#)

Diagnostic setting name \*

Logs	Destination details
<p>Categories</p> <ul style="list-style-type: none"><li><input checked="" type="checkbox"/> Administrative</li><li><input checked="" type="checkbox"/> Security</li><li><input checked="" type="checkbox"/> ServiceHealth</li><li><input checked="" type="checkbox"/> Alert</li><li><input checked="" type="checkbox"/> Recommendation</li><li><input checked="" type="checkbox"/> Policy</li><li><input checked="" type="checkbox"/> Autoscale</li><li><input checked="" type="checkbox"/> ResourceHealth</li></ul>	<ul style="list-style-type: none"><li><input checked="" type="checkbox"/> Send to Log Analytics workspace</li><li>Subscription: <input type="text" value="Visual Studio Premium met MSDN"/></li><li>Log Analytics workspace: <input type="text" value="asels1 ( westeurope )"/></li><li><input type="checkbox"/> Archive to a storage account</li><li><input type="checkbox"/> Stream to an event hub</li><li><input type="checkbox"/> Send to partner solution</li></ul>

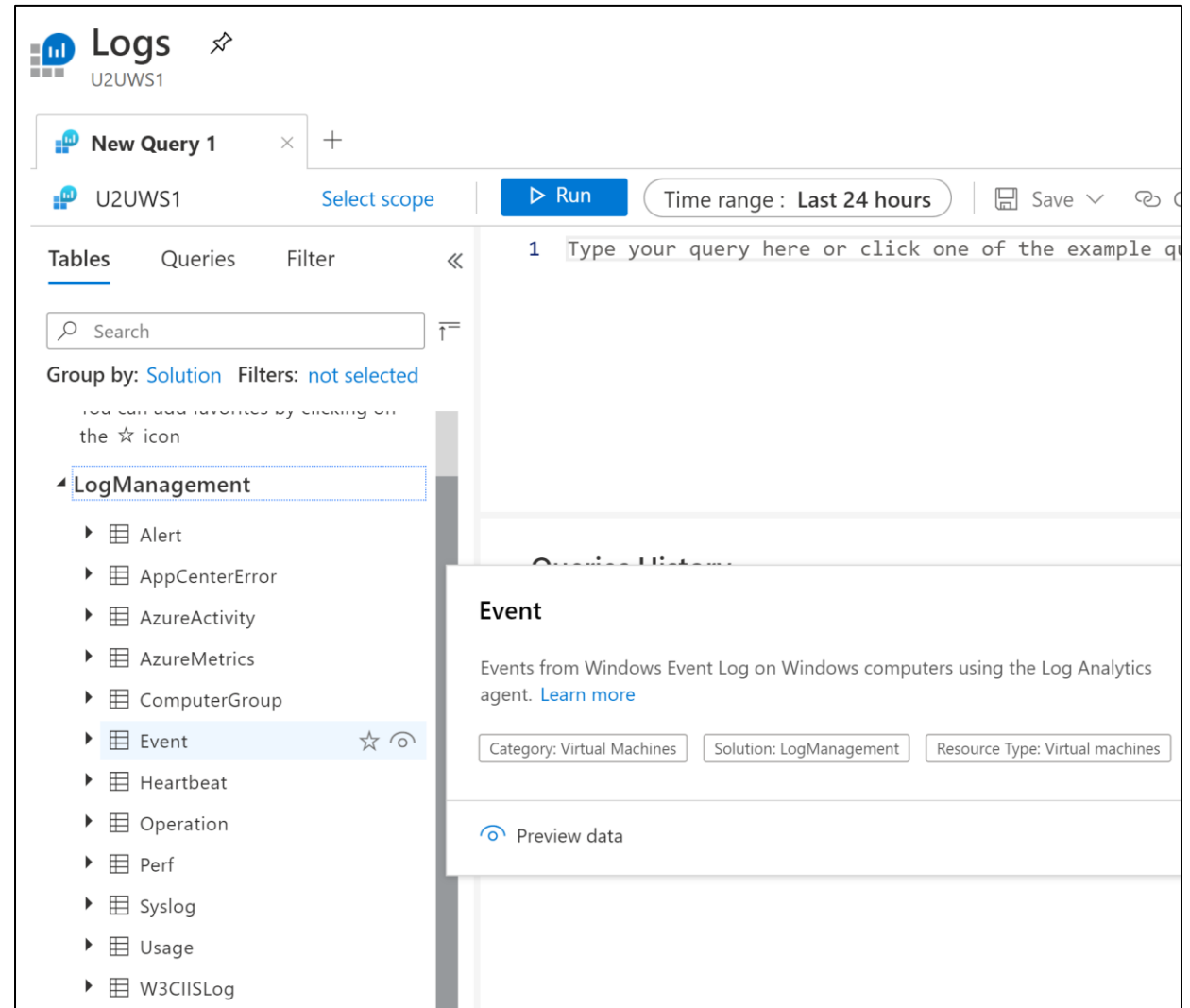
# Data Structure

- Workspace contains tables
  - Organized into columns
    - With multiple rows of data



# Tables

- Data is stored in tables
  - Syslog
  - Heartbeat
  - Event
  - Operation
  - ...
- Explore tables
  - Log Analytics workspace
    - Logs
      - Table pane





# Kusto Query Language

- Tool to
  - Explore data and discover patterns
  - Identify anomalies
- Uses schema entities organized in a hierarchy similar to SQL
  - Databases
  - Tables
  - Columns
- Kusto query
  - Read-only request to
    - Process data
    - Return results
  - Easy to read and author
  - Case-sensitive



# Kusto Query Language

---

- KQL is used with
  - Azure Monitor
  - Azure Data Explorer
- Some differences
  - Operators not supported by Azure Monitor
  - Operators specific to Azure Monitor
- Practice writing KQL statements in a demo environment
  - <https://aka.ms/lademo>

# Kusto Query Language (KQL)

---

- Powerful query language
  - Join data from multiple tables
  - Aggregate large sets of data
  - Perform complex operations with minimal code

# Sample Queries

- Retrieve all records from a table
- Retrieve records from table
  - Filter records
  - Summarize
  - Visualize results in chart
- Retrieve data from multiple tables
  - Using a join
  - Analyze combined results

Events

```
SecurityEvent  
| where TimeGenerated > ago(7d)  
| where EventID == 4625  
| summarize count() by Computer, bin(TimeGenerated, 1h)  
| render timechart
```

```
app("ContosoRetailWeb").requests  
| summarize count() by bin(timestamp,1hr)  
| join kind= inner (Perf  
    | summarize avg(CounterValue)  
    by bin(TimeGenerated,1hr))  
on $left.timestamp == $right.TimeGenerated
```

# Run a Query

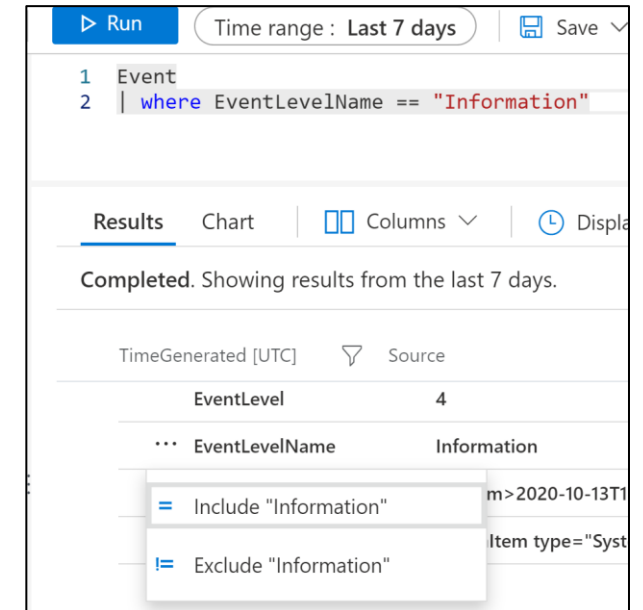
- Place cursor in query window
  - Click Run
  - Shift+Enter
- Use the Time range control
  - Default
    - Last 24 hours

The screenshot displays the 'U2UWS1 | Logs' interface, a Log Analytics workspace. The top navigation bar includes a 'New Query 1\*' tab, a 'U2UWS1' scope selector, a 'Run' button, and a 'Time range : Last 7 days' control. Below the navigation bar, the 'Tables' tab is active, showing a search bar and a 'Group by: Solution' filter. The 'Queries' tab is also visible. The 'Results' section shows a table of log data with columns for TimeGenerated [UTC], Source, EventLog, Computer, and EventLevel. The table contains two rows of data, both from 10/12/2020 at 1:30:41.317 PM and 1:30:41.363 PM, with sources 'Service Control Manager' and 'Microsoft-Windows-Security-SPP' respectively. The 'Results' section also includes a 'Completed. Showing results from the last 7 days.' message and a 'Group columns' toggle.

TimeGenerated [UTC]	Source	EventLog	Computer	EventLevel
10/12/2020, 1:30:41.317 PM	Service Control Manager	System	DC1.u2ucourse.com	4
10/12/2020, 1:30:41.363 PM	Microsoft-Windows-Security-SPP	Application	DC1.u2ucourse.com	4

# Filter Results

- Restrict table elements
  - From results
    - Select a property
    - Click 3 dots (...)
      - Include/exclude
  - In query
    - Where Propertyname == 'Value'
- Filter results
  - Select Filter icon next to column heading



The screenshot shows the u2u interface with a table of results. The table has columns: TimeGenerated [UTC], Source, EventLog, Computer, EventLevel, EventLevelName, and ParameterXml. A filter menu is open over the 'EventLevelName' column, showing options: 'Is equal to', 'information', 'And', 'Is equal to', and 'Filter'. The table shows results from the last 7 days.

TimeGenerated [UTC]	Source	EventLog	Computer	EventLevel	EventLevelName	ParameterXml
> 10/12/2020, 1:30:41.317 PM	Service Control Manager	System	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:30:41.363 PM	Microsoft-Windows-Security-SPP	Application	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:31:11.440 PM	Microsoft-Windows-Security-SPP	Application	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:31:11.440 PM	Service Control Manager	System	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:32:41.330 PM	Service Control Manager	System	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:32:41.393 PM	Microsoft-Windows-Security-SPP	Application	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:33:11.483 PM	Microsoft-Windows-Security-SPP	Application	DC1.u2ucourse.com	4	Information	
> 10/12/2020, 1:33:11.483 PM	Service Control Manager	System	DC1.u2ucourse.com	4	Information	

# Sort or Group Columns

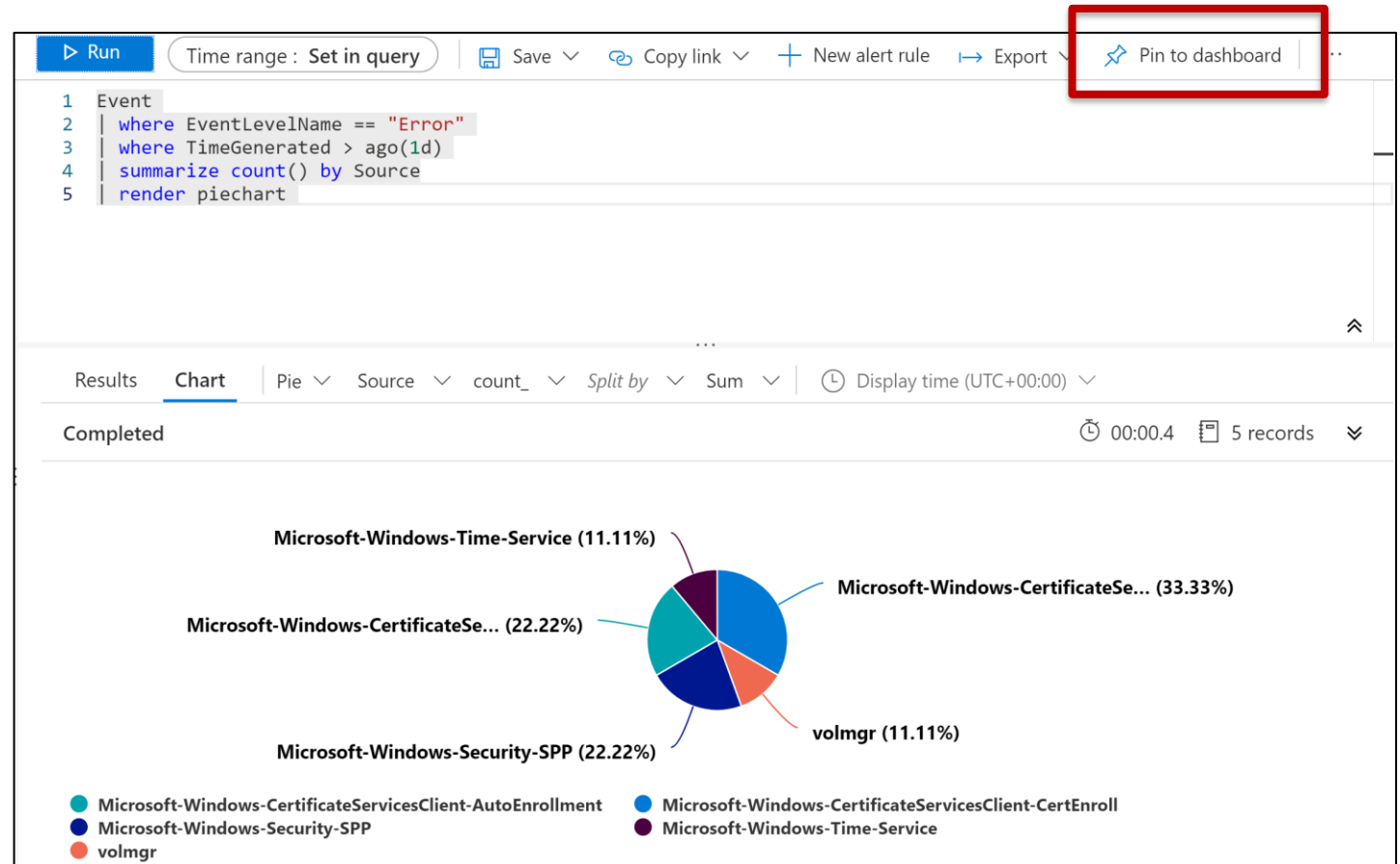
- Sort results
  - Click column header
- Group by
  - Drag column header to bar above results table

The screenshot shows a web-based data interface. At the top, there are tabs for 'Results' (selected) and 'Chart'. To the right of the tabs are controls for 'Columns' (a dropdown menu), 'Display time (UTC+00:00)' (a dropdown menu), and a 'Group columns' toggle switch which is currently turned on. Below these controls, a status bar indicates 'Completed. Showing results from the last 7 days.' on the left, and a clock icon with '00:03.4' and a document icon with '5,105 records' on the right. Below the status bar is a horizontal bar with two active filters: '↑ Computer' and '↑ EventLog', each with a close button (X). Below this bar is a table with columns: 'TimeGenerated [UTC]', 'Source', 'EventLog', 'Computer', and 'EventLevel'. The table is grouped under 'Computer: DC1.u2ucourse.com' and 'EventLog: Application'. The table contains two rows of data:

	TimeGenerated [UTC]	Source	EventLog	Computer	EventLevel
▼ Computer: DC1.u2ucourse.com					
▼ EventLog: Application					
>	10/26/2020, 8:16:15.457 AM	Microsoft-Windows-Security-SPP	Application	DC1.u2ucourse.com	4
>	10/26/2020, 8:16:19.703 AM	Microsoft-Windows-CertificationAuthority	Application	DC1.u2ucourse.com	2

# View and Modify Charts

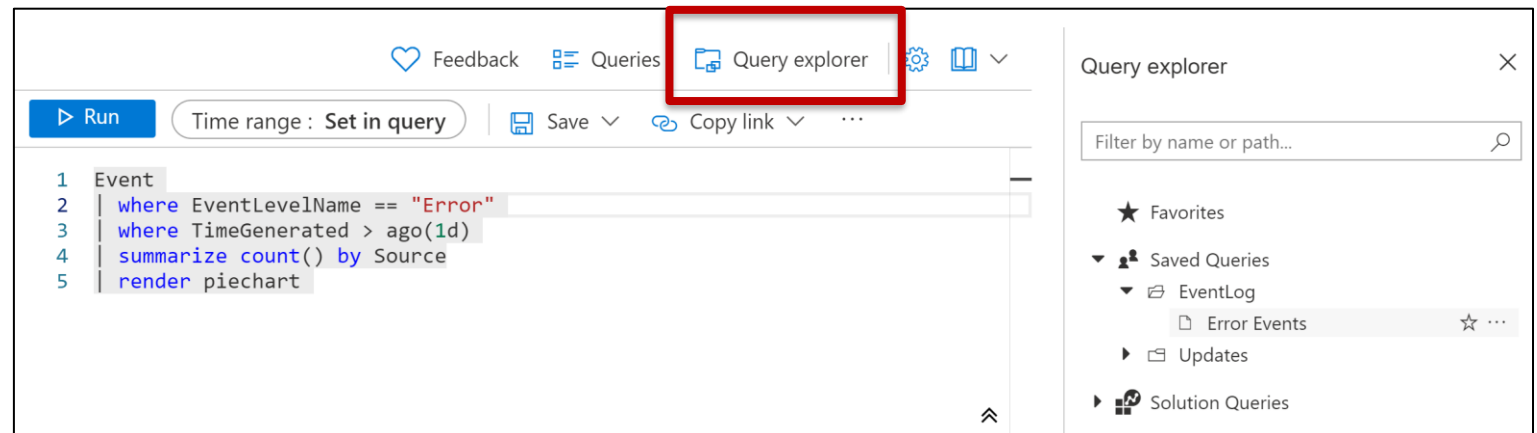
- Graphic views
  - Chart
  - Stacked Bar
  - Stacked column
  - Pie
- Pin results to dashboard





# Manage Queries

- Save queries
- Load queries
  - Query explorer
  - Mark query as favorite
- Export and share queries
  - Export to
    - .csv
    - PowerBI
  - Share a link to a query



# Write New Query

- Table-based query
  - Find tables in schema pane
  - Defines clear scope
  - Improves query performance
- Search-based query
  - Find records that meet criteria
  - Less structured
  - Slower
  - Search across multiple tables

```
Event  
| take 10
```

```
search "Hyper-V"  
| take 10
```

```
search in (Event) "Hyper-V"  
| take 10
```

# Take - Sort - Top

- Take
  - Return specific number of arbitrary records
- Sort
  - Sort by preferred column
  - Sorts entire table
    - Could return many results
    - Could take long time
- Top
  - Get latest records
  - Sorts table on server side
    - Returns only top records

```
Event  
| take 10
```

```
Event  
| sort by Source asc
```

```
Event  
| top 10 by TimeGenerated
```

- Where
  - Filter data by a specific condition
  - Limit query results

- Expressions

Expression	Description	Example
==	Check equality (case-sensitive)	Level == 8
=~	Check equality (case-insensitive)	EventSourceName =~ "microsoft"
!=, <>	Check inequality	Level != 4
and, or	Combine conditions	Level == 16 or CommandLine != ""

```
Event  
| where EventLog == "Application"
```

```
Event  
| where EventLog == "Application" and EventLevel == 1
```

# Specify Time Range

- Time picker
  - Default 24 hours
- Time filter in query
  - Place time filter immediately after table name
  - Time units
    - Days (d)
    - Minutes (m)
    - Seconds (s)

```
Event  
| where TimeGenerated > ago(30m)
```

# Select and Compute Columns

- Project

- Select specific columns to include in results
- Rename columns
- Define new columns
- Variations
  - Project-away
  - Project-keep
  - Project-rename
  - Project-reorder

```
Event  
| take 50  
| project TimeGenerated, Computer, EventLog, EventLevel
```

```
Event  
| take 50  
| project TimeGenerated, Computer, LogName=EventLog
```

- Extend

- Keep all original columns in results
- Define additional ones
  - Calculated columns

```
Event  
| take 50  
| extend LogName=EventLog
```

# Aggregate Groups of Rows

- Summarize

- Identify groups of records
- Based on columns
- Apply aggregations
  - Count
    - Return number of results in each group
  - Dcount
    - Returns estimate for distinct values
  - Avg
    - Perform mathematical calculations
  - Max, min
  - Sum

```
Perf
| where TimeGenerated > ago(1h)
| summarize by ObjectName
```

```
Perf
| where TimeGenerated > ago(1h)
| summarize count() by ObjectName
```

```
Perf
| where TimeGenerated > ago(1h)
| summarize count() by ObjectName, CounterName
```

```
Perf
| where TimeGenerated > ago(1h)
| summarize avg(CounterValue) by Computer, CounterName
```

# Summarize by a Time Column

- Group results based on time
  - Summarize by TimeGenerated
    - Creates groups for every millisecond over time range
  - Break range into manageable units
    - Bin
- Example
  - Analyze perf records for free memory on computer
    - Available Mbytes
  - Calculate average value of each 1-hour period over last 7 days

```
Perf
| where TimeGenerated < ago(7d)
| where Computer == "DC1.u2ucourse.com"
| where CounterName == "Available MBytes"
| summarize avg(CounterValue) by bin(TimeGenerated, 1h)
| render timechart
```



# Visualizations

- Render
  - Generate visualization of query results
  - Supported options
    - Areachart
    - Barchart
    - Columnchart
    - Piechart
    - Scatterchart
    - Timechart

```
SecurityEvent  
| summarize count() by Account  
| render barchart
```

```
SecurityEvent  
| summarize count() by bin(TimeGenerated, 1d)  
| render timechart
```

# Joins

- Analyze data from multiple tables in same query
  - Merge rows of two datasets
  - Match values of specified columns
- Join flavors
  - Innerunique (default)
    - Inner join with left side deduplication
  - Inner
  - Fullouter
- Join datasets with different key

```
Table1  
| join ( Table2 )  
on $left.key1 == $right.key2
```

```
Heartbeat  
| where OSType == "Windows"  
| project ComputerIP, Computer, OSType  
| join (  
    Event  
    | where EventLevelName == "Error"  
    | project EventLevelName, Computer, Message  
) on Computer
```