

Article

[David Underhill](#) · Jul 12, 2019 2m read

[Open Exchange](#)

Basic Database Metrics example

This is a self contained class that can be run from the InterSystems Task Scheduler which records peak usage details for databases and licenses built up throughout the day and retaining 30 days history.

To schedule the task to run every hour:

```
d ##class(Metrics.Task).Schedule()
```

You can also specify your own start time, stop time, and run interval:

```
d ##class(Metrics.Task).Schedule(startTime, stopTime, intervalMins)
```

Metrics are stored in ^Metrics in the namespace that the class resides in/is run from.

```
^Metrics("SystemMetrics", "Databases", {database}, +$h, "usage", {metric})={value}
```

```
"Available"      = lowest peak space available in the database (KB).
"Used"           = peak space used in the database (KB).
"Size"           = peak size of the database (KB).
"Free"           = lowest peak percentage free space of the database (%).
"MaxSize"        = peak maximum size parameter of the database (KB).
"DiskFreeSpace" = peak space available on the disk/volume for the database
(KB).
"DiskSize"       = peak total size of the disk/volume for the database
(KB).
"MaxFree"        = lowest peak percentage free space for the database based
on MaxSize or DiskFreeSpace if not set (%).
```

```
^Metrics("SystemMetrics", "License", +$h, {metric})
"Customer"       = customer name on the license key.
"Expires"        = expiry date on the license key.
"Total"          = peak total units on the license key.
"Used"           = peak license units in use.
"Available"      = lowest peak license units available.
```

I am sure there are improvements to be made but hopefully a good starting point.

[#Analytics #Databases #Monitoring #System Administration #Caché #InterSystems IRIS #Open Exchange](#)
[Check the related application on InterSystems Open Exchange](#)

Source URL: <https://community.intersystems.com/post/basic-database-metrics-example>