

# SIMPHERA License Usage Analyzer

## Introduction

SIMPHERA logs actions requiring a license with the help of the *Billing Service*. The license usage analyzer reads and aggregates the log entries to provide administrators with an overview of when and how many licenses are used simultaneously.

The license usage analyzer is provided as an executable command-line tool in a separate container and can be executed via a `kubectl run usageanalyzer` command from any host that has access to the SIMPHERA cluster. The easiest way to execute the `kubectl` command is to call the PowerShell script `Run-UsageAnalyzer.ps1` which extracts the required information about accessing the cluster from the Helm configuration.

## Prerequisites

To execute the usage analyzer on any host the system must fulfill the following prerequisites:

- `kubectl` must be installed on the host.
- To use the PowerShell script, you must also have `helm` and PowerShell 7 installed.
- The user Kubernetes configuration file must point to your cluster.

## Using the PowerShell Script

The PowerShell script `Run-UsageAnalyzer.ps1` needs two arguments in order to access the cluster and start the container which provides the `usageanalyzer`: \* Namespace: The Kubernetes namespace the Helm chart has been deployed to. \* HelmRelease: The name of the SIMPHERA Helm release.

Further arguments to the script will be passed on to the `usageanalyzer`. Since the arguments for the `usageanalyzer` also start with a hyphen, they must be enclosed in quotes so that they are not to evaluated by PowerShell.

## Example

```
Run-UsageAnalyzer.ps1 -Namespace <mysimphera> -HelmRelease <mysimphera>  
<arguments to the usageanalyzer>
```

# Controlling the Output of the Usage Analyzer

The main purpose of the `usageanalyzer` is to determine the actual license demand (concurrent license usage). For this purpose, the `usageanalyzer` offers a number of reports: From detailed listings of individual user actions to long-term aggregated and anonymized license usage counts.

All reports issued by `usageanalyzer` can be configured by command line parameters in terms of the time period to be considered, as well as limited to specific users or licenses.

## Time Period to be Considered

Time specifications must be made in ISO 8601 format (i.e. in the form `YYYY-MM-DD[Thh:mm]`), for example `2021-11-30T08:15`. Time periods can be specified in weeks, days, hours and minutes. For simplicity the valid format is (in deviation from ISO 8601): `[Ww][Dd][Hh][Mm]`, for example `4d12h` (4 days and 12 hours).

`-f:<start time>` The start time of the report (i.e. from which the data should be analyzed). This time must be in the past.

`-t:<end time>` The end time of the report (i.e. up to which the data should be analyzed). This time must be in the past and after the start time.

`-p:<time period>` The time period of the report (i.e. the period between start and end). This parameter must be specified after `-f:` and `-t:`, if any.

Note that no more than two of the three parameters mentioned above may be specified, as they are mutually redundant. The variety of time specifications possible with this is only to conveniently cover different use cases.

`-i:<time interval>` The time intervals in which the license usage is broken down for the detailed report.

The following examples show some useful parameter combinations and what time period they specify. To simplify the explanation we assume that in the examples `P` is specified in minutes.

`-f:<T1> -t:<T2>` The time period from `T1` to `T2`

`-f:<T>` The time period from `T` to now

`-p:<P>` The time period of the past `P` minutes before now

`-t:<T> -p:<P>` The time period of `P` minutes before `T`

`-f:<T> -p:<P>` The time period of `P` minutes after `T`

If you do not specify both start and end time or a specific period, then `usageanalyzer` analyzes the past 7 days.

## Limiting Reports to Specific Users and Licenses

In general, `usageanalyzer` analyzes all license usage entries within the specified time period. To limit the analysis and the report to only one specific user or one specific license, you need to specify these by the following parameters.

`-u:<user>` User specification. Note that this is typically the token of the user issued by the authentication system, not the user's real name.

`-l:<license>` Name of the license.

## Selecting the Report Types

There are 4 different report types available:

- A list of all user tokens and licenses logged within the specified time period, named '*Info*'
- An aggregated maximum license usage, named '*Usage*'
- A detailed license usage list, separated by user and action performed, named '*Detailed*'
- An anonymized compliance report to pass on to dSPACE for specific license agreements, listing the maximum license requirement per hour, named '*dSPACE*'

To generate any of these reports, you must specify the report name with the `-r` parameter. For example `-r:Detailed`. By default, if you don't request a specific report, the '*Info*' list is printed.

## Contents and Examples of the Report Types

By default, the reports '*Usage*', '*Detailed*' and '*Info*' are output in a human-readable, line-oriented format. In order to further process the reports, they can also be output in JSON format. To switch to JSON format, specify the `-j` parameter in addition to the report type.

### Detailed

Example (prints the detailed report from December 1st 2021 to now):

```
> Run-UsageAnalyzer.ps1 -Namespace mysimphera -HelmRelease mysimphera "-r:Detailed" "-f:2021-12-01"

SIMPHERA_VAL_SCBT from 12/01/2021 08:45 to 12/13/2021 09:45: 245 minutes
Detailed usages:
  User 07e67aa5-200d-4860-837e-45e559795419 from 12/01/2021 08:45 to
12/01/2021 09:16 performing
SIMPHERA_SCBT_CREATETESTCASE,SIMPHERA_SCBT_CREATETESTSUITE
  User 0a97c904-8f30-4edb-87ef-ed01a2ba4073 from 12/08/2021 08:10 to
```

```
12/08/2021 08:41 performing SIMPHERA_SCBT_UPDATETESTCASE
User 8ab355bc-f3f3-4c6d-8d18-1d41bf75893a from 12/08/2021 12:48 to
12/08/2021 13:18 performing SIMPHERA_SCBT_DELETETESTSUITE_API
...
```

## Usage

The '*Usage*' report aggregates all license usages in the specified time period and determines the maximum concurrent usage during this period. To split the aggregation into smaller intervals (e.g. printing the maximum concurrent usage per day), specify an interval length in minutes with the `-i` paramter. For example `-i:1440` (24 hours \* 60 minutes).

Example (prints the usage report for the entire November 2021 in JSON format):

```
> Run-UsageAnalyzer.ps1 -Namespace mysimpchera -HelmRelease mysimpchera "-
r:Usage" "-f:2021-11-01" "-t:2021-12-01" "-j"

{
  "Licenses": [
    {
      "Usage": 3,
      "Available": 5,
      "Name": "SIMPHERA_SIM_CORE",
      "Start": "2021-11-01T00:00:00",
      "End": "2021-12-01T00:00:00"
    },
    {
      "Usage": 3,
      "Available": 3,
      "Name": "SIMPHERA_VAL_SCBT",
      "Start": "2021-11-01T00:00:00",
      "End": "2021-12-01T00:00:00"
    }
  ],
  "Name": "Maximum license usage",
  "Start": "2021-11-01T00:00:00",
  "End": "2021-12-01T00:00:00"
}
```

## dSPACE

The '*dSPACE*' report uses a fixed XML output format and cannot be modified, except for the time period which can be specified by any pair of the three parameters `-f`, `-t` and `-p`.

Example (prints the dSPACE XML report for the past 13 weeks):

```
> Run-UsageAnalyzer.ps1 -Namespace mysimpheera -HelmRelease mysimpheera "-  
r:dSPACE" "-p:13w"
```

```
<LicenseStats>  
...  
<Day Date="2021-11-25">  
  <Time Hour="13">  
    <License Id="SIMPHERA_SIM_CORE" Quantity="1/5" />  
  </Time>  
</Day>  
<Day Date="2021-11-26">  
  <Time Hour="8">  
    <License Id="SIMPHERA_SIM_CORE" Quantity="3/5" />  
  </Time>  
  <Time Hour="9">  
    <License Id="SIMPHERA_SIM_CORE" Quantity="2/5" />  
  </Time>  
  <Time Hour="10">  
    <License Id="SIMPHERA_SIM_CORE" Quantity="1/5" />  
  </Time>  
</Day>  
...  
</LicenseStats>
```