

# How to select Nodes (and Interfaces) for reporting in opReports 3

In opReports 3.0 the various methods for selecting what to report on have been consolidated and simplified.

This page describes what mechanisms are available, and how to control them in the GUI, with opreports-cli and for report schedules.

- [Which selection mechanisms are supported by what report types?](#)
- [Choosing the selection mechanism for scheduled reports](#)
- [The Simplest Choice: Everything](#)
- [Nodes that belong to a specific group](#)
- [Nodes whose name matches a regular expression](#)
- [Nodes and Interfaces whose names/descriptions match regular expressions](#)
- [Groups, Nodes and Interfaces whose names/descriptions match regular expressions](#)
- [Groups, Nodes and Interfaces whose names/descriptions match regular expressions for separate report for each group](#)
- [Explicitly listed Nodes](#)
- [Nodes listed in a file](#)
- [Nodes and specific Interfaces, listed in a file](#)
- [Nodes, specific Interfaces and Types, listed in a file](#)
  - [Plain Text Format](#)
  - [CSV Format \(3.0.16 and newer\)](#)
- [Burst Billing Customer List](#)
- [Nodes and Interfaces that are part of an opCharts Business Service](#)
  - [Configuration](#)
  - [Usage](#)
- [Related Topics](#)
  - [How to select Nodes \(and Interfaces\) for reporting in opReports 4](#)

## Which selection mechanisms are supported by what report types?

Join Paul McClendon, an Opmantek Support Engineer, as he demonstrates quickly and easily you can generate reports using opReports.

[Certain reports](#) do have specific requirements which are shown in the table below.

In general, however, providing more precision than necessary is allowed: For example, you can use a node and interfaces (and even types) list file for all reports; for reports where interfaces are not relevant, the extra information will simply be discarded and just the listed nodes will be used. Reports that don't need the summary type will simply use the nodes (and possibly the interfaces).

On the other hand, providing *insufficiently* precise information to reports that require it will result in an error message (e.g. trying to run a summary report with just group=X).

Report Type	Selection Mechanisms	Notes
Node	single node only	The node report supports a single node only. If your selection contains more nodes, then the report is created for the first listed node.
Node Health	all, interfaces irrelevant	Interface selections are not relevant for this report.
Node Availability	all, interfaces irrelevant new in opReports 3.0.8	Interface selections are not relevant for this report.
Grouped (Node) Availability	all, interfaces irrelevant new in opReports 3.1.4	Interface selections are not relevant for this report.
WAN	all interfaces ignored before 3.1.4	Before version 3.1.4 the WAN report ignored interface selections. It operates on all active or any selected interfaces of the selected nodes whose net type is "wan".
WAN Utilisation Distribution and WAN Utilisation Distribution Summary	all new in opReports 3.1.8	It operates on all active or any selected interfaces of the selected nodes whose net type is "wan".
QoS	all interfaces ignored before 3.1.4	Before version 3.1.4 the QoS report ignored interface selections. It operates on all active or any selected interfaces of the selected nodes', if QoS is configured for the interface.

CoS	all	For each selected node this report covers all active or any selected interfaces that have Class of Service configured.
Uptime	all, interfaces irrelevant	Interface selections are not relevant for this report.
Response Time	all, interfaces irrelevant	Interface selections are not relevant for this report.
Interface Capacity	all	
Grouped Interface Capacity	all new in opReports 3.1.8	
Interface Utilisation	all	Interface selections are honored but the "type" component (in selections by node+interface+type) are ignored as not relevant.
Interface Unicast Packets	all new in opReports 3.1.8	Interface selections are honored but the "type" component (in selections by node+interface+type) are ignored as not relevant.
CPU	all, interfaces irrelevant	Interface selections are not relevant for this report.
Free Memory	all, interfaces irrelevant	Interface selections are not relevant for this report.
Memory Pool	all, interfaces irrelevant	Interface selections are not relevant for this report.
Memory Buffer	all, interfaces irrelevant	Interface selections are not relevant for this report.
Monitored Services	all, interfaces irrelevant	Interface selections are not relevant for this report.
Traffic Usage	all	Interface selections are honored but the "type" component (in selections by node+interface+type) are ignored as not relevant.
Traffic Summary	node+interface+type	This report <b>strictly</b> requires this specific selection format.
Traffic Snapshot	opCharts Business Services, selected per page	This report type <b>strictly</b> requires lists of Business Services for each of its (multiple) page definitions.
Configuration Summary	all, interfaces irrelevant	Interface selections are not relevant for this report
Burst Billing Summary Report	<b>strictly</b> Burst Billing Customer List File	List file must contain one node plus one interface plus one client number/service number plus contracted upstream plus contracted downstream in this order. Interface can be listed by index or by interface description.
Burst Billing Report	<b>strictly</b> Burst Billing Customer List File	List file must contain one node plus one interface plus one client number/service number plus contracted upstream plus contracted downstream in this order. Interface can be listed by index or by interface description.
Customer Exceeds Bandwidth Report	<b>strictly</b> Burst Billing Customer List File	List file must contain one node plus one interface plus one client number/service number plus contracted upstream plus contracted downstream in this order. Interface can be listed by index or by interface description.

## Choosing the selection mechanism for scheduled reports

For scheduled reports your schedule must contain a property named `sources`, with one of the following values: "everything", "group\_each\_regexp", "group\_regexp", "node\_regexp", "node\_group", "nodes", "node\_list", "node\_intf\_list", "node\_intf\_type\_list", or "business\_services" (in 3.0.14 an newer). Those mechanisms are described below.

If you use the opReports schedule editing GUI then this property will be managed on your behalf.

## The Simplest Choice: Everything

If you do not make an explicit selection, then opReports will work on all active nodes (and all their active interfaces, for report types that handle interfaces).

In the GUI this choice is shown as "All Active Nodes".

## Nodes that belong to a specific group

In NMIS every node belongs to precisely one group, and this concept therefore applies to opReports as well.

With `opreport-cli` you have to give the argument `group=<groupname>`. In a report schedule this is expressed using the property `node_group`. In the GUI this choice is presented as "by Group".

There are two "wildcard" groups available:

- Group "All" is equivalent to the default choice, all active nodes. This wildcard should not be used as we will likely retire it in a future version of opReports.
- Group "Each" is **available for scheduled reports only, excluding 'once only' scheduled reports, and causes the generation of a separate report for each of the known groups.**

## Nodes whose name matches a regular expression

In the GUI this choice is called "by Regular Expression for Nodes", `opreports-cli` uses the command line argument `node_regexp=<regular expression>`, and for scheduled reports you'd specify this with the property `node_regexp`. The node regular expression is evaluated at report creation time.

The regular expression syntax is Perl's standard, described in detail in this [Perl Regexp Tutorial](#).

## Nodes and Interfaces whose names/descriptions match regular expressions

This feature was added to opReports in version 3.1.4.

Nodes must match the regular expression given for the node name, but interface descriptions must also match a separate regular expression. Only those interfaces are selected where both regular expressions match. However, for reports where interfaces are not relevant, interfaces are disregarded.

The regular expression for interfaces is applied to both the interface's `ifDescr` and `Description` properties in parallel, and a match for either or both selects the interface. (The NMIS GUI presents `ifDescr` as "Name" or "Name (ifDescr)". Depending on the device and its modelling `ifDescr` may or may not be adjustable, but `Description` can be set easily within NMIS.)

In the GUI this option is called "by Regular Expression for Nodes and Interfaces". `opreports-cli` requires that you supply both `node_regexp=<regular expression>` **and** `node_intf_regexp=<regular expression>` arguments. The report schedules use the same property names as the `opreports-cli` parameters.

Both regular expressions are evaluated at report creation time.

## Groups, Nodes and Interfaces whose names/descriptions match regular expressions

This feature was added to opReports in version 3.1.8.

Groups must match the regular expression given for the group name **AND** Nodes must match the regular expression given for the node name **AND** Interface descriptions must also match a separate regular expression.

Only those interfaces are selected where all three regular expressions match. However, for reports where interfaces are not relevant, interfaces are disregarded.

The regular expression for interfaces is applied to both the interface's `ifDescr` and `Description` properties in parallel, and a match for either or both selects the interface. (The NMIS GUI presents `ifDescr` as "Name" or "Name (ifDescr)". Depending on the device and its modelling `ifDescr` may or may not be adjustable, but `Description` can be set easily within NMIS.)

In the GUI this option is called "by Regular Expression for Groups Nodes and Interfaces". `opreports-cli` requires that you supply `group_regexp=<regular expression>` **AND** `node_regexp=<regular expression>` **AND** `node_intf_regexp=<regular expression>` arguments. The report schedules use the same property names as the `opreports-cli` parameters.

All three regular expressions are evaluated at report creation time.

## Groups, Nodes and Interfaces whose names/descriptions match regular expressions for separate report for each group

This feature was added to opReports in version 3.1.8.

**This option causes the generation of a separate report for each of the known groups. This option is available for scheduled reports only, excluding 'once only' scheduled reports.**

Groups must match the regular expression given for the group name **AND**  
Nodes must match the regular expression given for the node name **AND**  
Interface descriptions must also match a separate regular expression.

Only those interfaces are selected where all three regular expressions match.  
However, for reports where interfaces are not relevant, interfaces are disregarded.

The regular expression for interfaces is applied to both the interface's `ifDescr` and `Description` properties in parallel, and a match for either or both selects the interface.

(The NMIS GUI presents `ifDescr` as "Name" or "Name (ifDescr)". Depending on the device and its modelling `ifDescr` may or may not be adjustable, but `Description` can be set easily within NMIS.)

In the GUI this option is called "by Regular Expression for a Nodes and Interfaces Report per Group".

The report schedules requires that you supply `group_each_regexp=<regular expression> AND node_regexp=<regular expression> AND node_intf_regexp=<regular expression>`.

All three regular expressions are evaluated at report creation time.

## Explicitly listed Nodes

In the GUI this choice is called "Pick from Node List".

To use this mechanism with `opreports-cli`, you have to list each node you want in a separate `nodes=<nodename>` argument.

In a report schedule the property `nodes` would have to be set to an *array* of node names, like in this example:

```
"nodes" : [
  "ASGARD",
  "midgard"
],
```

## Nodes listed in a file

`opReports` expects a node list file to contain one node name per line. Whitespace before or after the node name is removed.

In the GUI this choice is called "from Node List File (Upload)", and you need to select a suitable file for uploading.

For `opreports-cli`, the command line argument `node_list=<path to node list file>` would be used. In that case the node list file must already reside on the `opReports` server.

In a report schedule definition, you'd use the property `node_list`, with the path to the list file as value:

```
"node_list" : "/tmp/my_list_of_lotsa_nodes.txt"
```

## Nodes and specific Interfaces, listed in a file

Certain reports allow a more precise selection of nodes and just some of their interfaces. This is implemented using a list file.

In the GUI you'd select "from Node and Interfaces List File (Upload)" and upload the file of choice. For `opreports-cli` you use `node_intf_list=<path to listfile>`, and for a scheduled report you would set the property `node_intf_list` with the value being the path to the list file.

The node and interface list can be in one of two formats, JSON or plain text:

- JSON: it must be a valid JSON document, consisting of a hash of the node name as key, and the value being a list of the interfaces in question.

```
{ "testnode": [ "eth0" ], "othernode" : [ 1, 2, "Tunnel20" ] }
```

- Plain Text: a text file, one entry per line.  
Each entry must start with the node name, followed by one or more TAB characters, and one or more interfaces (again separated by TAB characters). If you list a node on multiple lines then all listed interfaces will be combined into a single list. Blank lines and lines starting with the "#" sign are treated as comments and are ignored.

```
testnode 2 14 eth0
othernode Dialer1
testnode 17
```

For both JSON and Plain Text formats, interfaces can be identified by the numeric SNMP interface index, or by the SNMP ifDescr property.

## Nodes, specific Interfaces and Types, listed in a file



Careful not to confuse this option with 'Nodes and specific Interfaces, listed in a file' described above. If the 'Type' option is not necessary this is not the appropriate feature.

Certain reports offer a refinement of the above, with the added notion of a "Type" for grouping of Node+Interface into particular reporting classes.



"Type" is completely arbitrary. This value can be anything the user would like. This value is not referenced anywhere else in opReports or any other OMK application.

This is implemented again using a list file, but with a very specific format - which has a few inherent limitations; Starting with version 3.0.16, opReports also supports CSV as format for this type of input.

The relevant GUI choice is called "from Node, Interfaces and Type List File (Upload)", for opreports-cli the parameter is `node_intf_type_list=<path to listfile>`, and in a report schedule the controlling property is `node_intf_type_list` (value again the path to the list file).

## Plain Text Format

Please note: as of version 3.0.16 it's recommended that you use CSV as a safer alternative to this format.

The list file format is plain text, and each line must consist of precisely one node name, one of its interfaces and a "type" declaration. The interface must be identified by its SNMP Interface Index.

Node name and interface must be separated by "\_", and this must be separated by the "type" by one or more spaces, like in the example below:

```
nodeA_1 groupA
nodeB_41 groupB
nodeC_41_eth0 groupB
```

The example also shows an optional format variant: the node+interface stanza *may* include a trailing "`<ifdescr>`", but the interface description is not used for the selection logic: only the SNMP Interface Index is relevant. It is recommended that you do not use this flavour as it's ambiguous: interface descriptions can (and often do) include both \_ and space characters.

The "type" will be used to group all nodes and interfaces with the same type value into a group for summary reporting.

## CSV Format (3.0.16 and newer)

opReports now also supports CSV (with comma as the separator character) for this kind of input.

The lines in your file must contain at least the following four columns, in the following order:

1. The node name,
2. the SNMP Interface Index (must be present but may be empty if the Interface Description is given)
3. the Interface Description (must be present but is ignored if an SNMP Interface Index is given),
4. the "type" declaration.

Extra columns are ignored; files with fewer columns are rejected. Empty lines and comment lines (ie. lines starting with a "#" character) are ignored.

For extra convenience you may now also specify interfaces by their Interface Description instead of the Interface Index; both columns **must** be present in your input, but one of them may be blank.

The SNMP Interface Index is considered authoritative, if given. If it is not, then opReports looks for an interface with the given Interface Description. Nonexistent interfaces are skipped, and a warning message is logged.

Here is an example CSV file:

```
# comment, ignored. columns: nodename,interface index,interface description,type name
"some node","1","FastEthernet0/0","categoryA"
"not_the_greatest_name","10","Dialer1","catB"
"pleasefindme","Dialer1194","categoryA"
"iknowtheindex",12,"catB"
```

## Burst Billing Customer List

This is implemented again using a list file, but with a very specific format.

In the GUI you'd select "from Burst Billing Customer List File (Upload)" and upload the file of choice.

For `opreports-cli` you use `burst_billing_customer_list=<path to listfile>`,

and for a scheduled report you would set the property `burst_billing_customer_list` with the value being the path to the list file.

**The Burst Billing Customer List can only be in csv format:**

- Each entry must start with the **node name**, followed by an **interface** (ifDescr or ifIndex), a **client number**, **contracted upstream (mbps)** and **contracted downstream (mbps)**:

### Burst Billing Customer List in csv format

```
ASGARD,FastEthernet0/0,809-101-6781,5,45
ASGARD,2,809-101-6782,60,5
Switch-1,GigabitEthernet1/0/1,809-101-6784,4,2
Huawei_AR2240,5,12345678,20,200
Huawei_AR2240,GigabitEthernet0/0/1.243,12345679,100,10
Teldat_ATLAS60,ethernet0/0.3839,12345670,20,20
TeldatM1,10,12345671,10,10
```

**Interfaces can be identified by the numeric SNMP interface index, or by the SNMP ifDescr property.**

## Nodes and Interfaces that are part of an opCharts Business Service

If you have `opReports` version 3.0.14 and newer and `opCharts` is installed on the same system, then you can make use of Business Services to declare nodes and interfaces for reporting.

### Configuration

The following three configuration options (in `conf/opCommon.nmis`) are vital for `opReports` accessing `opCharts`:

```
# base of opCharts server url, eg http://localhost or http://localhost:80 - no slash at the end
'opreports_opcharts_url_base' => "http://127.0.0.1:8042",
'opreports_opcharts_user' => "nmis", # opreports needs a user with readonly-access
'opreports_opcharts_password' => "nm1888",
```

If you've changed the password for the default `nmis` user (or disabled it altogether), then these configuration items need to be adjusted accordingly. Once that's done you need to restart the OMK webserver (using `sudo service omkd restart`) to activate the changed configuration.

### Usage

In the GUI you will be presented with a list of known Business Service names, which supports multiple selections.

When using `opreports-cli`, the parameter is called `business_services` and you need to pass in each business service name separately, e.g. `opreports-cli.pl type=health business_services=first_service business_services=second_service`.

In a report schedule file, the property is called `business_services` and its value must be a list of business service names.

Business service memberships are expanded at report creation time.

## Related Topics

- [How to select Nodes \(and Interfaces\) for reporting in opReports 4](#)