

## Reportd

\$OPENNMS\_HOME/etc/reportd-configuration.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<reportd-configuration
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.opennms.org/xsd/config/reportd-configuration"
  storage-location="/opt/opennms/share/reports/"
  persist-reports="yes"
>
  <!--
  http://quartz.sourceforge.net/javadoc/org/quartz/CronTrigger.html
  Field Name  Allowed Values  Allowed Special Characters
  Seconds     0-59                , - * /
  Minutes     0-59                , - * /
  Hours       0-23                 , - * /
  Day-of-month 1-31                , - * ? / L W C
  Month       1-12 or JAN-DEC     , - * /
  Day-of-Week 1-7 or SUN-SAT     , - * ? / L C #
  Year (Opt)  empty, 1970-2099   , - * /
  -->
  <!--
  <report report-name="sample-report" report-template="sample-report.jrxml" report-engine="jdbc">
    <cron-schedule>0 16 * * * ? </cron-schedule>
    <recipient>indigo@springfield.local</recipient>
  </report>
  <!--
  <report report-name="Early-Morning-Report" report-template="Early-Morning-Report.jrxml" report-engine="jdbc">
    <cron-schedule>0 0 0 * * ? * </cron-schedule>
    <recipient>you@mail.domain</recipient>
  </report>
</reportd-configuration>
```

## Report API

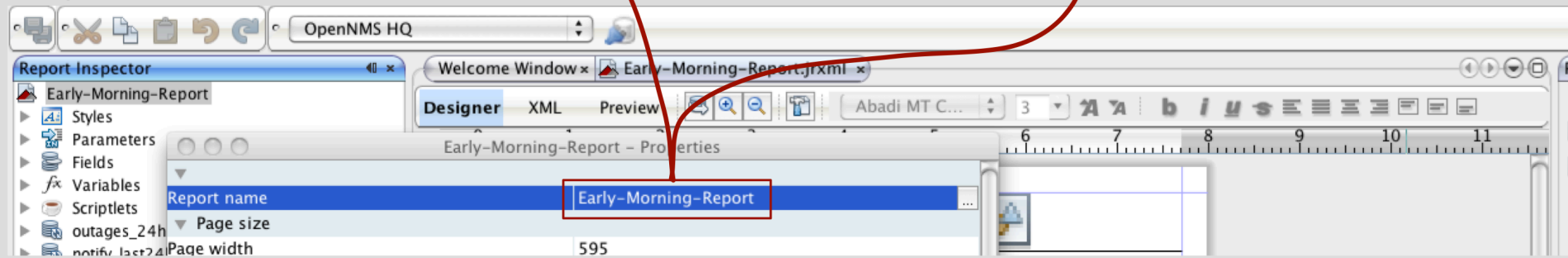
\$OPENNMS\_HOME/etc/database-reports.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<database-reports
  xmlns:this="http://xmlns.opennms.org/xsd/config/opennms-database-reports"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.opennms.org/xsd/config/opennms-database-reports">
  <report id="defaultCalendarReport" display-name="default calendar report"
    report-service="availabilityReportService" description="standard opennms report in calendar format" />
  <report id="defaultClassicReport" display-name="default classic report"
    report-service="availabilityReportService" description="standard opennms report in tabular format" />
  <report id="sample-report" display-name="sample JasperReport" online="true"
    report-service="jasperReportService" description="sample Jasper report using jdbc datasource" />
  <report id="trivial-report" display-name="trivial JasperReport" online="true"
    report-service="jasperReportService" description="trivial Jasper report using null datasource" />
  <report id="Early-Morning-Report" display-name="Early morning report" online="true"
    report-service="jasperReportService" description="Global overview of outages, notifications and events
  in last 24 hours" />
</database-reports>
```

\$OPENNMS\_HOME/etc/jasper-reports.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<jasper-reports
  xmlns:this="http://xmlns.opennms.org/xsd/config/jasper-reports"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.opennms.org/xsd/config/jasper-reports">
  <report id="sample-report" template="sample-report.jrxml" engine="jdbc" />
  <report id="trivial-report" template="trivial-report.jrxml" engine="null" />
  <report id="Early-Morning-Report" template="Early-Morning-Report.jrxml" engine="jdbc" />
</jasper-reports>
```

## iReport \$OPENNMS\_HOME/etc/report-templates



## Get all nodes with outages and services

```
SELECT
  *
FROM
  outages
JOIN
  node
ON
  (outages.nodeid=outages.nodeid)
JOIN
  service
ON
  (outages.serviceid=service.serviceid)
```

## Get all unresolved and resolved outages

```
SELECT
  CASE
    WHEN ifregainedservice = null THEN CAST('1' AS INTEGER)
    ELSE CAST('0' AS INTEGER)
  END AS unresolved,
  CASE
    WHEN ifregainedservice is not null THEN CAST('1' AS INTEGER)
    ELSE CAST('0' AS INTEGER)
  END AS resolved,
  *
FROM
  outages
```

## Get nodes by category

```
SELECT
  *
FROM
  node
JOIN
  category_node
ON
  (node.nodeid=category_node.nodeid)
JOIN
  categories
ON
  (category_node.categoryid=categories.categoryid)
```

## Get all nodes with outages, services and duration statistics

```
SELECT
  node.nodelabel,
  service.servicename,
  SUM(ifregainedservice-iflostservice) AS tot_dur,
  AVG(ifregainedservice-iflostservice) AS avg_dur,
  COUNT(*) AS outage_count
FROM
  outages
JOIN
  node
ON
  (outages.nodeid=node.nodeid)
JOIN
  service
ON
  (outages.serviceid=service.serviceid)
WHERE
  iflostservice > now()-INTERVAL '7 days' OR
  ifregainedservice is null
GROUP BY
  node.nodelabel,
  service.servicename
```

## Get notifications with user and nodes with service

```
SELECT
  node.nodelabel,
  notifications.eventuei,
  usersnotified.userid,
  usersnotified.media,
  notifications.answeredby
FROM
  notifications
JOIN
  usersnotified
ON
  (notifications.notifyid=usersnotified.notifyid)
JOIN
  node
ON
  (notifications.nodeid=node.nodeid)
LEFT JOIN
  service
ON
  (notifications.serviceid=service.serviceid)
```

## Statistics reports with data and nodeid

```
SELECT
  regexp_replace(
    substring
      (resourcereference.resourceid, 'node\\[[0-9]*\\]'),
    'node\\[[\\]]', '', 'g'
  )::INTEGER AS nodeid,
  *
FROM
  statisticsreport
JOIN
  statisticsreportdata
ON
  (statisticsreport.id=statisticsreportdata.reportid)
JOIN
  resourcereference
ON
  (statisticsreportdata.resourceid=resourcereference.id)
```