BIRT Reports Review for Jasper Reports

Last updated: 03/25/2018

Objective:

To determine the data sets used in BIRT reports for porting to Jaspersoft reports.

BIRT Projects:

Two BIRT projects in SVN apparently.

1. http://geneva/groundwork-professional/trunk/foundation/misc/web-application/reportserver/reports/
   Latest SVN log
   r18669 | ashanmugam | 2011-11-28 16:29:17 -0800 (Mon, 28 Nov 2011)

2. http://geneva/groundwork-professional/branches/BIRT_2_5_UPGRADE/reportserver/reports/
   Latest SVN log
   r15363 | hchanekar | 2010-03-19 05:11:19 -0700 (Fri, 19 Mar 2010)

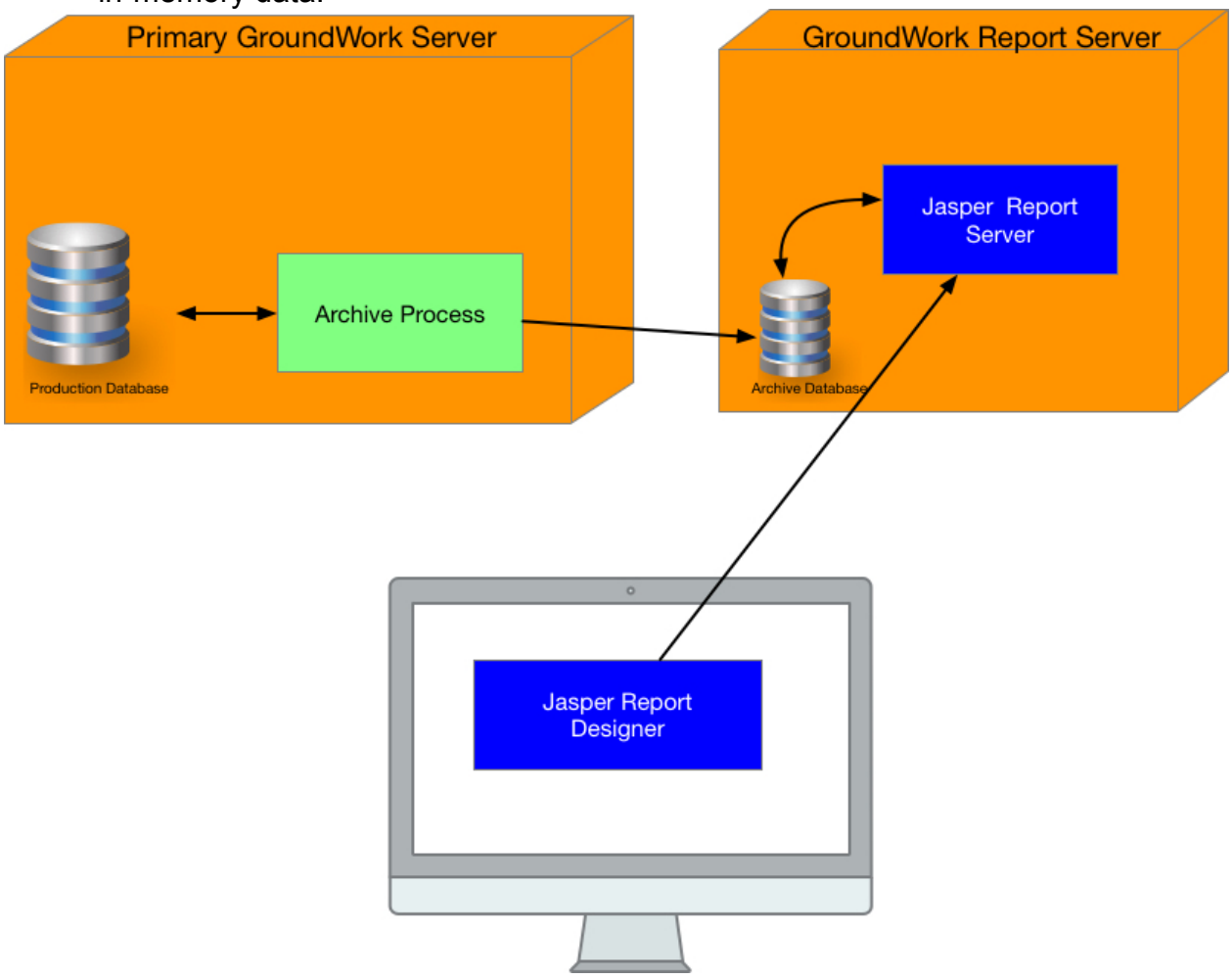
Note: apparently the second project is older based on the SVN log history. Besides, it is missing the Eclipse ***.project*** file and also pointing to MySQL database on some reports.

Requirements:

Based on the document provided from Tomas, the Jasper reports should just work with the ***archive_gwcollagedb*** database, which could probably guide to assess feasibility on porting BIRT to Jasper as well as checking if data in ***archive_gwcollagedb*** is sufficient to generate all reports delivered in BIRT currently.

Note:

1. ***dashboard*** database is used from BIRT***?***
2. Per David, some of the SOAP web service as data source is to get real-time and in-memory data.



DataSource and DataSet Definition per report:

1. Status (2 reports)

- ***gw-host-status.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***
2. ***http://localhost:8080/foundation-webapp/services/wshost?wsdl***
3. ***http://localhost:8080/foundation-webapp/services/wsservice?wsdl***
4. ***http://localhost:8080/foundation-webapp/services/wsstatistics?wsdl***

DataSet: ***getSimpleHost***

http://localhost:8080/foundation-webapp/services/wshost ***Parameter***		
hostName	deep	

DataSet: ***getStatistics***

http://localhost:8080/foundation-webapp/services/wsstatistics ***Parameter***		
statisticQueryType (TOTAL_FOR_SERVICE_BY_HOSTNAME)	value	applicationType

DataSet: ***getServicesForHost***

http://localhost:8080/foundation-webapp/services/wsservice   ***Parameter***   hostName

- ***gw-hostgroup-status.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***
2. ***http://localhost:8080/foundation-webapp/services/wsstatistics?wsdl***

DataSet: ***getHostStatistics***

http://localhost:8080/foundation-webapp/services/wsstatistics ***Parameter***		
statisticQueryType (TOTAL_FOR_SERVICE_BY_HOSTNAME)	value	applicationType

DataSet: ***getServiceStatistics***

http://localhost:8080/foundation-webapp/services/wsstatistics ***Parameter***		
statisticQueryType (TOTAL_FOR_SERVICE_BY_HOSTNAME)	**value**	applicationType

Note: both DataSet are the same web service call, not a typo. Looks like the Soap request def. are the same as well.

2. Availability (4 reports)

- ***gw-host-availability.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***
2. ***jdbc:postgresql://localhost:5432/dashboard***

DataSet: ***host-service-chart***

SQL details under Availability of DataSet Definitions Details. ***Parameter***		
HostName FromDate ToDate	HostName2 FromDate2 ToDate2	HostName3 FromDate3 ToDate3
HostName4 FromDate4 ToDate4	HostName5 FromDate5 ToDate5	HostName6 FromDate6 ToDate6

DataSet: ***host-chart***

SQL details under Availability of DataSet Definitions Details. ***Parameter***			
HostName FromDate ToDate	HostName2 FromDate2 ToDate2	HostName3 FromDate3 ToDate3	HostName4 FromDate4 ToDate4

DataSet: ***gw-service-availability***

SQL details under Availability of DataSet Definitions Details. ***Parameter***		
HostName	FromDate	ToDate

DataSet: ***gw-host-availability***

SQL details under Availability of DataSet Definitions Details. ***Parameter***		
HostName	FromDate	ToDate

- ***gw-host-state-transition.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***
2. ***http://localhost:8080/foundation-webapp/services/wsevent***
3. ***http://localhost:8080/foundation-webapp/services/wshost***
4. ***http://localhost:8080/foundation-webapp/services/wsservice***

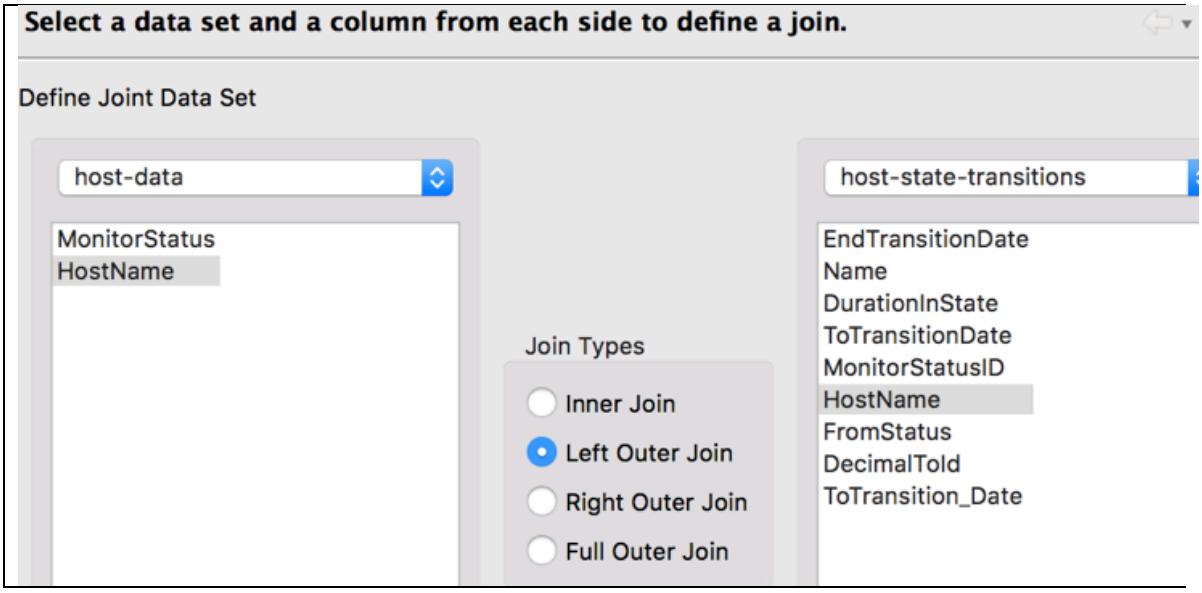
DataSet: ***host-state-transitions***

http://localhost:8080/foundation-webapp/services/wsevent ***Parameter***		
hostName	startDate	endDate

DataSet: ***service-state-transitions***

http://localhost:8080/foundation-webapp/services/wsevent ***Parameter***		
hostName	serviceName	startDate
endDate		

DataSet: ***JoinedDataSet***



DataSet: ***host-data***

http://localhost:8080/foundation-webapp/services/wshost ***Parameter***		
hostName	deep	

- ***gw-hostgroup-availability.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/dashboard***
2. j***dbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***

DataSet: ***hostgroup-service-chart***

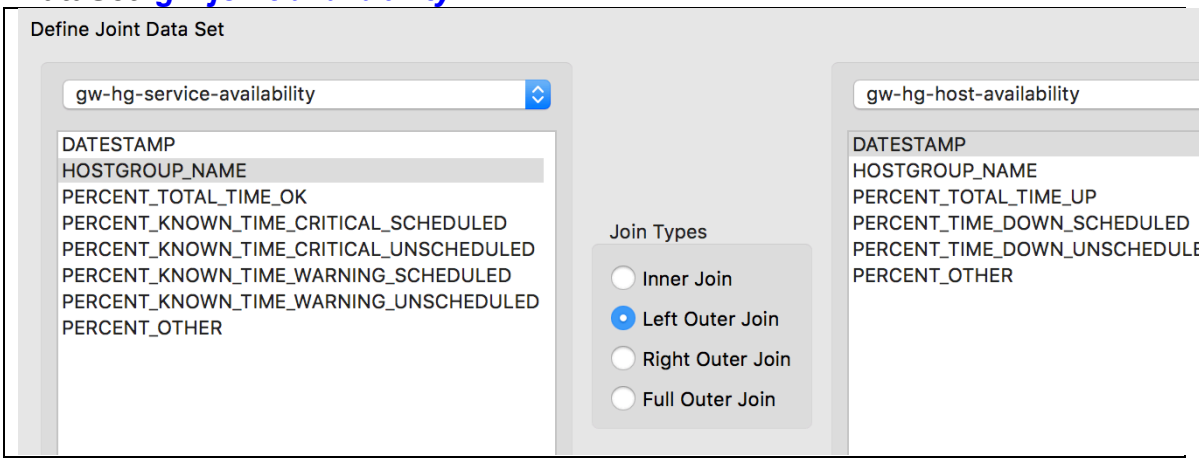
SQL details under Availability of DataSet Definitions Details. ***Parameter***		
HostName	HostName2	HostName3

FromDate ToDate	FromDate2 ToDate2	FromDate3 ToDate3
HostName4 FromDate4 ToDate4	HostName5 FromDate5 ToDate5	HostName6 FromDate6 ToDate6

DataSet: ***hostgroup-host-chart***

SQL details under Availability of DataSet Definitions Details. ***Parameter***			
HostName FromDate ToDate	HostName2 FromDate2 ToDate2	HostName3 FromDate3 ToDate3	HostName4 FromDate4 ToDate4

DataSet: ***gw-joint-availability***



DataSet: ***gw-hg-service-availability***

SQL details under Availability of DataSet Definitions Details. ***Parameter***		
HostGroup	FromDate	ToDate

DataSet: ***gw-hg-host-availability***

SQL details under Availability of DataSet Definitions Details. ***Parameter***		
HostGroup	FromDate	ToDate

❖ ***gw-service-state-transitions.rptdesign***

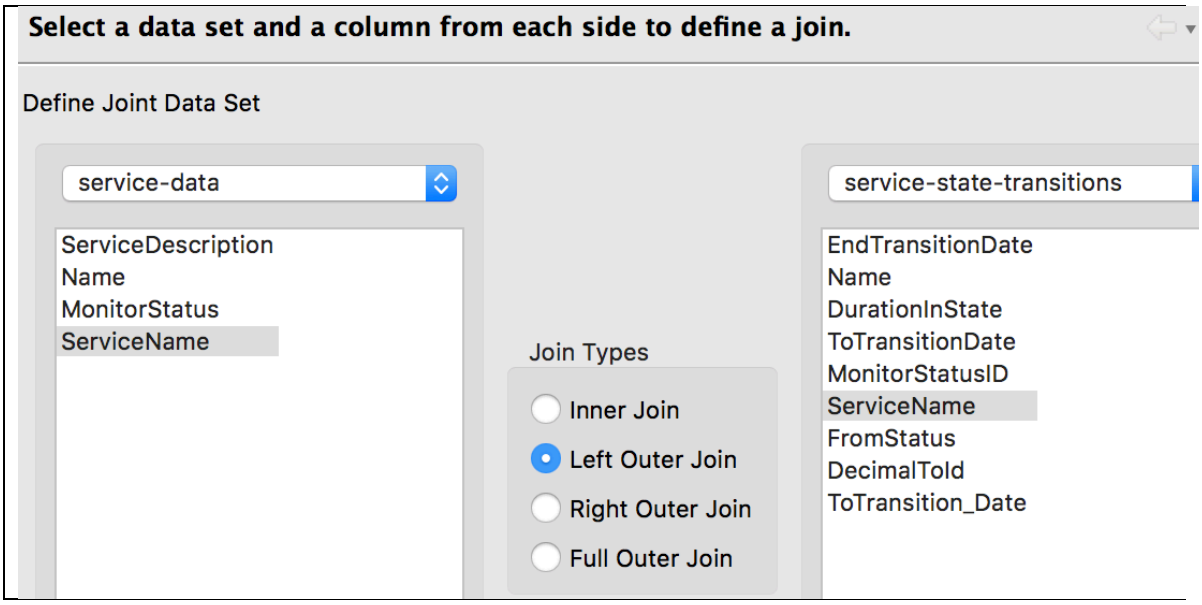
DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***
2. ***http://localhost:8080/foundation-webapp/services/wsevent***
3. ***http://localhost:8080/foundation-webapp/services/wshost***
4. ***http://localhost:8080/foundation-webapp/services/wsservice***

DataSet: *HostChooser*

See General under DataSet Definitions Details.

DataSet: *JoinedDataSet*



DataSet: *service-state-transitions*

http://localhost:8080/foundation-webapp/services/wsevent		
Parameter		
hostName	serviceName	startDate
endDate		

DataSet: *service-data*

SQL details under Availability of DataSet Definitions Details.		
Parameter		
serviceDescription	hostName	

3. Events (2 reports)

❖ *gw-event-history-overview.rptdesign*

DataSource:

1. ***http://localhost:8080/foundation-webapp/services/wscommon?wsdl***

DataSet: ***GWEventStatisticsMonitorStatus***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Body>
      <m:performEntityQuery xmlns:m="urn:fws">
         <entityType xsi:type="xsd:string">LOG_MESSAGE_STATISTICS</entityType>
         <filter>
            <StringProperty>
               <name>StatisticType</name>
               <value>MonitorStatus</value>
            </StringProperty>
            <Operator>EQ</Operator>
         </filter>
         <sort></sort>
         <firstResult xsi:type="xsd:int">-1</firstResult>
         <maxResults xsi:type="xsd:int">-1</maxResults>
      </m:performEntityQuery>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

DataSet: ***GWEventStatisticsOperationStatus***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Body>
      <m:performEntityQuery xmlns:m="urn:fws">
         <entityType xsi:type="xsd:string">LOG_MESSAGE_STATISTICS</entityType>
         <filter>
            <StringProperty>
               <name>StatisticType</name>
               <value>OperationStatus</value>
            </StringProperty>
            <Operator>EQ</Operator>
         </filter>
         <sort></sort>
         <firstResult xsi:type="xsd:int">-1</firstResult>
         <maxResults xsi:type="xsd:int">-1</maxResults>
      </m:performEntityQuery>
```

```
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

DataSet: ***GWEventStatisticsSeverity***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope
        xmlns:xsd="http://www.w3.org/2001/XMLSchema"
        xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
        <SOAP-ENV:Body>
           <m:performEntityQuery xmlns:m="urn:fws">
                <entityType xsi:type="xsd:string">LOG_MESSAGE_STATISTICS</entityType>
                     <filter>
                        <StringProperty>
                                <name>StatisticType</name>
                                <value>Severity</value>
                        </StringProperty>
                        <Operator>EQ</Operator>
                     </filter>
                <sort></sort>
                <firstResult xsi:type="xsd:int">-1</firstResult>
                <maxResults xsi:type="xsd:int">-1</maxResults>
           </m:performEntityQuery>
        </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

- ***gw-event-history.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1***
2. ***http://localhost:8080/foundation-webapp/services/wscommon?wsdl***
3. ***http://localhost:8080/foundation-webapp/services/wsevent?wsdl***
4. ***http://localhost:8080/foundation-webapp/services/wshost?wsdl***
5. ***http://localhost:8080/foundation-webapp/services/wshostgroup?wsdl***

DataSet: ***getApplicationTypes***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:fws">
   <soapenv:Header />
   <soapenv:Body>
      <urn:getAttributeData>
         <type>APPLICATION_TYPES</type>
      </urn:getAttributeData>
   </soapenv:Body>
</soapenv:Envelope>
```

DataSet: ***GWEventStatisticsMonitorStatus***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl ***Parameter***		
entityType (LOG_MESSAGE_STATICTICS)	StatisticType (MonitorStatus)	Query
Value	ApplicationType	StartDate
EndDate		

DataSet: ***GWEventStatisticsOperationStatus***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl (Sam SOAP Request as above) ***Parameter***		
entityType (LOG_MESSAGE_STATICTICS)	StatisticType (MonitorStatus)	Query
Value	ApplicationType	StartDate
EndDate		

DataSet: ***GWEventStatisticsSeverity***

http://localhost:8080/foundation-webapp/services/wscommon?wsdl (Same SOAP request as above) ***Parameter***		
entityType (LOG_MESSAGE_STATICTICS)	StatisticType (MonitorStatus)	Query
Value	ApplicationType	StartDate
EndDate		

DataSet: ***getEventList***

http://localhost:8080/foundation-webapp/services/wsevent?wsdl

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Body>
      <m:getEventsByString xmlns:m="urn:fws">
                  <type xsi:type="xsd:string">&?type?&</type>
                  <value xsi:type="xsd:string">&?value?&</value>
                  <appType xsi:type="xsd:string">&?appType?&</appType>
                  <startRange xsi:type="xsd:string">&?startRange?&</startRange>
                  <endRange xsi:type="xsd:string">&?endRange?&</endRange>
                  <sortOrder xsi:type="xsd:string">&?sortOrder?&</sortOrder>
                  <sortField xsi:type="xsd:string">&?sortField?&</sortField>
                  <firstResult xsi:type="xsd:string">&?firstResult?&</firstResult>
                  <maxResults xsi:type="xsd:string">&?maxResults?&</maxResults>
      </m:getEventsByString>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4. Performance (5 reports)

❖ *gw-epr-host-multi-variable.rptreport*

DataSource:

1. jdbc:postgresql://localhost:5432/gwcollagedb

DataSet: ***HostList***

Refer to details under General of DataSet Defintions Details.

DataSet: ***LogPerformanceData***

6 parameters are defined and bound to the SQL, refer the SQL under Performance of DataSet Defintions Details.

❖ *gw-epr-host.rptdesign*

DataSource:

1. jdbc:postgresql://localhost:5432/gwcollagedb

DataSet: ***HostChooser***

Refer to details under General of DataSet Defintions Details.

DataSet: ***host-performance***

SQL details under Availability of DataSet Definitions Details. ***Parameter***		
PerformanceData	*Host*	*DateStart*
DateEnd		

DataSet: ***host-service-performance***

SQL details under Availability of DataSet Definitions Details. Parameter		
PerformanceData	Host	DateStart
DateEnd		

❖ *gw-epr-hostgroup-multi-variable.rptdesign*

DataSource:

1. jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1

DataSet: ***HostGroupList***

SELECT NAME FROM hostgroup [where Name IN (<list>)] ORDER BY NAME;

DataSet: HostGroupList_Right

SELECT NAME FROM hostgroup

DataSet: ***LogPerformanceData***

Define 6 parameters bound in SQL, see SQL under Performance of DataSet Definitions Details.

DataSet: ***PerformanceDataTable***

Defined 6 parameters bound in SQL, see SQL under Performance of DataSet Definitions Details.

❖ *gw-epr-hostgroup-topfive.rptdesign*

DataSource:

1. jdbc:postgresql://localhost:5432/gwcollagedb?prepareThreshold=1

DataSet: ***HostGroupChooser***

```
SELECT NAME FROM hostgroup ORDER  BY NAME
```

DataSet: ***misc-graph-labels***

SQL details under Availability of DataSet Definitions Details.

Parameter

PerformanceName	HostGroup	DateStart
DateEnd		

DataSet: ***host-counts***

```
SELECT     Count(h.hostid) AS numhosts, a.numhostswithdata
FROM       hostgroup hg
INNER JOIN hostgroupcollection hgc
ON         hg.hostgroupid = hgc.hostgroupid
INNER JOIN host h
ON         hgc.hostid = h.hostid
INNER JOIN
           (
             SELECT     1    AS recid, Count(DISTINCT h.hostid) AS numhostswithdata
             FROM       performancedatalabel pdl
             INNER JOIN logperformancedata lpd
             ON         lpd.performancedatalabelid = pdl.performancedatalabelid
             INNER JOIN servicestatus ss
             ON         lpd.servicestatusid = ss.servicestatusid
             INNER JOIN host h
             ON         ss.hostid = h.hostid
             INNER JOIN hostgroupcollection hgc
             ON         h.hostid = hgc.hostid
             INNER JOIN hostgroup hg
             ON         hgc.hostgroupid = hg.hostgroupid
             WHERE      hg.NAME = ?
             AND        pdl.performancename = ? ) AS a
WHERE      hg.NAME = ?
GROUP BY   a.numhostswithdata
```

DataSet: ***hostgroup-host-info-limit-5***

?

DataSet: ***hostgroup-host-info***

```
SELECT
      Avg(lpd.average) AS Average,
      Max(lpd.maximum) AS Maximum,
      Min(lpd.minimum) AS Minimum,
      h.hostname
FROM   performancedatalabel pdl
      INNER JOIN logperformancedata lpd
            ON lpd.performancedatalabelid = pdl.performancedatalabelid
      INNER JOIN servicestatus ss
            ON lpd.servicestatusid = ss.servicestatusid
      INNER JOIN host h
            ON ss.hostid = h.hostid
      INNER JOIN hostgroupcollection hgc
            ON h.hostid = hgc.hostid
      INNER JOIN hostgroup hg
            ON hgc.hostgroupid = hg.hostgroupid
WHERE  pdl.performancename = ?
      AND hg.NAME = ?
      AND lpd.lastchecktime >= ?
      AND lpd.lastchecktime <= ?
GROUP  BY h.hostname
```

- ***gw-epr-hostgroup.rptdesign***

DataSource:

1. ***jdbc:postgresql://localhost:5432/gwcollagedb***

DataSet: ***HostGroupChooser***

```
SELECT NAME FROM hostgroup ORDER BY NAME
```

DataSet: ***host-counts***

```
SELECT     Count(h.hostid) AS numhosts, a.numhostswithdata
FROM       hostgroup hg
INNER JOIN hostgroupcollection hgc
ON         hg.hostgroupid = hgc.hostgroupid
INNER JOIN host h
ON         hgc.hostid = h.hostid
INNER JOIN
          (
            SELECT     1                    AS recid,
                       Count(DISTINCT h.hostid) AS numhostswithdata
            FROM       performancedatalabel pdl
            INNER JOIN logperformancedata lpd
            ON         lpd.performancedatalabelid = pdl.performancedatalabelid
            INNER JOIN servicestatus ss
            ON         lpd.servicestatusid = ss.servicestatusid
            INNER JOIN host h
            ON         ss.hostid = h.hostid
            INNER JOIN hostgroupcollection hgc
            ON         h.hostid = hgc.hostid
            INNER JOIN hostgroup hg
```

```
            ON        hgc.hostgroupid = hg.hostgroupid
            WHERE     hg.NAME = ?
            AND       pdl.performancename = ? ) AS a
WHERE     hg.NAME = ?
GROUP BY  a.numhostswithdata
```

DataSet: ***hostgroup-performance***

```
SELECT
      Cast(lpd.lastchecktime AS DATE) AS LastCheckTime,
      Min(lpd.minimum)              AS Minimum,
      Avg(lpd.average)              AS Average,
      Max(lpd.maximum)              AS Maximum,
      pdl.performancename,
      hg.NAME                       AS HostGroup
FROM   performancedatalabel pdl
      INNER JOIN logperformancedata lpd
            ON lpd.performancedatalabelid = pdl.performancedatalabelid
      INNER JOIN servicestatus ss
            ON lpd.servicestatusid = ss.servicestatusid
      INNER JOIN host h
            ON ss.hostid = h.hostid
      INNER JOIN hostgroupcollection hgc
            ON h.hostid = hgc.hostid
      INNER JOIN hostgroup hg
            ON hgc.hostgroupid = hg.hostgroupid
WHERE  pdl.performancename = ?
      AND hg.NAME = ?
      AND lpd.lastchecktime >= ?
      AND lpd.lastchecktime <= ?
GROUP  BY Dayofyear(lpd.lastchecktime),
         pdl.performancename,
         hg.hostgroupid;
```

DataSet: ***hostgroup-host-info***

```
SELECT
      Avg(lpd.average) AS Average,
      Max(lpd.maximum) AS Maximum,
      Min(lpd.minimum) AS Minimum,
      h.hostname
FROM   performancedatalabel pdl
      INNER JOIN logperformancedata lpd
            ON lpd.performancedatalabelid = pdl.performancedatalabelid
      INNER JOIN servicestatus ss
            ON lpd.servicestatusid = ss.servicestatusid
      INNER JOIN host h
            ON ss.hostid = h.hostid
      INNER JOIN hostgroupcollection hgc
            ON h.hostid = hgc.hostid
      INNER JOIN hostgroup hg
            ON hgc.hostgroupid = hg.hostgroupid
WHERE  pdl.performancename = ?
      AND hg.NAME = ?
      AND lpd.lastchecktime >= ?
      AND lpd.lastchecktime <= ?
GROUP  BY h.hostname
```

DataSet Definitions Details:

General

1. HostChooser / HostList

```
SELECT DISTINCT hostname FROM host
   [, HostGroup, HostGroupCollection
    WHERE
       Host.HostID = HostGroupCollection.HostID
    AND
       HostGroup.HostGroupID = HostGroupCollection.HostGroupID
    AND
       HostGroup.Name IN (<List>)
   ]
ORDER BY hostname;
```

2. HostGroupChooser

```
SELECT NAME FROM hostgroup ORDER BY NAME
```

3. ServiceChooser

```
SELECT servicestatus.servicedescription
FROM   servicestatus,
       host
WHERE  host.hostid = servicestatus.hostid
       AND host.hostname = ?;
```

Availability

1. gw-host-availability.rptdesign: host-service-chart

```
SELECT  *
FROM    (
        SELECT    h.host_name,
                  h.service_name,
                  h.datestamp,
                  '% total                time ok'    AS state_name,
                  h1.percent_total_time_ok AS                time
        FROM      service_availability h
        INNER JOIN
                  (
                        SELECT host_name,
                              service_name,
                              percent_total_time_ok,
                              datestamp
                        FROM   service_availability) h1
        on        h1.host_name = h.host_name
        AND       h1.service_name = h.service_name
        AND       h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND       h.datestamp >= ?
        AND       h.datestamp <= ?
        UNION
        SELECT    h.host_name,
                  h.service_name,
                  h.datestamp,
                  '% known                          time critical scheduled'    AS state_name,
                  h1.percent_known_time_critical_scheduled AS                          time
        FROM      service_availability h
        INNER JOIN
                  (
                        SELECT host_name,
                              service_name,
                              percent_known_time_critical_scheduled,
                              datestamp
                        FROM   service_availability) h1
        ON        h1.host_name = h.host_name
        AND       h1.service_name = h.service_name
        AND       h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND       h.datestamp >= ?
        AND       h.datestamp <= ?
        UNION
        SELECT    h.host_name,
                  h.service_name,
                  h.datestamp,
                  '% known                          time critical unscheduled'    AS state_name,
                  h1.percent_known_time_critical_unscheduled AS                          time
        FROM      service_availability h
        INNER JOIN
                  (
                        SELECT host_name,
                              service_name,
```

```
                    percent_known_time_critical_unscheduled,
                    datestamp
                FROM   service_availability) h1
ON        h1.host_name = h.host_name
AND       h1.service_name = h.service_name
AND       h1.datestamp = h.datestamp
WHERE      h.host_name=?
AND       h.datestamp >= ?
AND       h.datestamp <= ?
UNION
SELECT     h.host_name,
          h.service_name,
          h.datestamp,
          '% known time warning scheduled' AS state_name,
          h1.percent_known_time_warning_scheduled AS time
FROM       service_availability h
INNER JOIN
          (
                SELECT host_name,
                    service_name,
                    percent_known_time_warning_scheduled,
                    datestamp
                FROM   service_availability) h1
ON        h1.host_name = h.host_name
AND       h1.service_name = h.service_name
AND       h1.datestamp = h.datestamp
WHERE      h.host_name=?
AND       h.datestamp >= ?
AND       h.datestamp <= ?
UNION
SELECT     h.host_name,
          h.service_name,
          h.datestamp,
          '% known time warning unscheduled' AS state_name,
          h1.percent_known_time_warning_unscheduled AS time
FROM       service_availability h
INNER JOIN
          (
                SELECT host_name,
                    service_name,
                    percent_known_time_warning_unscheduled,
                    datestamp
                FROM   service_availability) h1
ON        h1.host_name = h.host_name
AND       h1.service_name = h.service_name
AND       h1.datestamp = h.datestamp
WHERE      h.host_name=?
AND       h.datestamp >= ?
AND       h.datestamp <= ?
UNION
SELECT     h.host_name,
          h.service_name,
          h.datestamp,
          '% time other' AS state_name,
          (100.0 – (h1.percent_total_time_ok +
                h1.percent_known_time_critical_scheduled +
```

```
                    h1.percent_known_time_critical_unscheduled +
                    h1.percent_known_time_warning_scheduled +
                    h1.percent_known_time_warning_unscheduled)) AS time
        FROM      service_availability h
        INNER JOIN
                (
                    SELECT host_name,
                        service_name,
                        percent_total_time_ok,
                        percent_known_time_critical_scheduled,
                        percent_known_time_critical_unscheduled,
                        percent_known_time_warning_scheduled,
                        percent_known_time_warning_unscheduled,
                        datestamp
                    FROM   service_availability) h1
        ON       h1.host_name = h.host_name
        AND      h1.service_name = h.service_name
        AND      h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND      h.datestamp >= ?
        AND      h.datestamp <= ? ) AS ha
ORDER BY ha.datestamp, ha.service_name
```

2. **gw-host-availability.rptdesign:host-chart**

```
SELECT  *
FROM    (
        SELECT    h.host_name,
                h.datestamp,
                '% total time up' AS state_name,
                h1.percent_total_time_up AS time
        FROM      host_availability h
        INNER JOIN
                (
                    SELECT host_name,
                        percent_total_time_up,
                        datestamp
                    FROM   host_availability) h1
        ON       h1.host_name = h.host_name
        AND      h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND      h.datestamp >= ?
        AND      h.datestamp <= ?
        UNION
        SELECT    h.host_name,
                h.datestamp,
                '% time down unscheduled' AS state_name,
                h1.percent_time_down_unscheduled AS time
        FROM      host_availability h
        INNER JOIN
                (
                 SELECT host_name,
                      percent_time_down_unscheduled,
                      datestamp
                 FROM   host_availability) h1
```

```
        ON      h1.host_name = h.host_name
        AND     h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND     h.datestamp >= ?
        AND     h.datestamp <= ?
        UNION
        SELECT    h.host_name,
                h.datestamp,
                '% time down scheduled' AS state_name,
                h1.percent_time_down_scheduled AS time
        FROM      host_availability h
        INNER JOIN
                (
                 SELECT host_name,
                      percent_time_down_scheduled,
                      datestamp
                 FROM   host_availability) h1
        ON      h1.host_name = h.host_name
        AND     h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND     h.datestamp >= ?
        AND     h.datestamp <= ?
        UNION
        SELECT    h.host_name,
                h.datestamp,
                '% time other',
                (100.0 –
                   (h1.percent_total_time_up +
                    h1.percent_time_down_unscheduled +
                    h1.percent_time_down_scheduled)) AS time
        FROM      host_availability h
        INNER JOIN
                (
                 SELECT host_name,
                      percent_total_time_up,
                      percent_time_down_unscheduled,
                      percent_time_down_scheduled,
                      datestamp
                 FROM   host_availability) h1
        ON      h1.host_name = h.host_name
        AND     h1.datestamp = h.datestamp
        WHERE     h.host_name=?
        AND     h.datestamp >= ?
        AND     h.datestamp <= ? ) AS ha
ORDER BY ha.datestamp
```

3. gw-host-availability.rptdesign: gw-service-availability

```
SELECT *,
    ( 100.0 - ( percent_total_time_ok
              + percent_known_time_critical_scheduled
              + percent_known_time_critical_unscheduled
              + percent_known_time_warning_scheduled
              + percent_known_time_warning_unscheduled ) ) AS PERCENT_OTHER
FROM   service_availability
```

```
WHERE  service_availability.host_name = ?
     AND service_availability.datestamp >= ?
     AND service_availability.datestamp <= ?
ORDER  BY service_availability.datestamp,
       service_availability.service_name
```

4. **gw-host-availability.rptdesign: gw-service-availability**

```
SELECT *,
     ( 100.0 - ( percent_total_time_up
               + percent_time_down_scheduled
               + percent_time_down_unscheduled ) ) AS PERCENT_OTHER
FROM   host_availability ha
WHERE  ha.host_name = ?
     AND ha.datestamp >=?
     AND ha.datestamp <=?
ORDER  BY ha.host_name
```

5. **gw-hostgroup-availability.rptdesign: hostgroup-service-chart**

```
SELECT *
FROM   (SELECT h.hostgroup_name,
           h.datestamp,
           '% TOTAL TIME OK'       AS STATE_NAME,
           h1.percent_total_time_ok AS Time
      FROM   hostgroup_service_availability h
           INNER JOIN (SELECT hostgroup_name,
                              percent_total_time_ok,
                              datestamp
                       FROM   hostgroup_service_availability
                       WHERE  hostgroup_name = Replace(?, ' ', '+')
                              AND datestamp >= ?
                              AND datestamp <= ?) h1
                   ON h1.hostgroup_name = h.hostgroup_name
                     AND h1.datestamp = h.datestamp
      UNION
      SELECT h.hostgroup_name,
           h.datestamp,
           '% KNOWN TIME CRITICAL - SCHEDULED'     AS STATE_NAME,
           h1.percent_known_time_critical_scheduled AS time
      FROM   hostgroup_service_availability h
           INNER JOIN (SELECT hostgroup_name,
                              percent_known_time_critical_scheduled,
                              datestamp
                       FROM   hostgroup_service_availability
                       WHERE  hostgroup_name = Replace(?, ' ', '+')
                              AND datestamp >= ?
                              AND datestamp <= ?) h1
                   ON h1.hostgroup_name = h.hostgroup_name
                     AND h1.datestamp = h.datestamp
      UNION
      SELECT h.hostgroup_name,
           h.datestamp,
           '% KNOWN TIME CRITICAL - UNSCHEDULED'     AS STATE_NAME,
```

```
        h1.percent_known_time_critical_unscheduled AS Time
FROM   hostgroup_service_availability h
       INNER JOIN (SELECT hostgroup_name,
                          percent_known_time_critical_unscheduled,
                          datestamp
                   FROM   hostgroup_service_availability
                   WHERE  hostgroup_name = Replace(?, ' ', '+')
                          AND datestamp >= ?
                          AND datestamp <= ?) h1
               ON h1.hostgroup_name = h.hostgroup_name
                  AND h1.datestamp = h.datestamp
UNION
SELECT h.hostgroup_name,
       h.datestamp,
       '% KNOWN TIME WARNING - SCHEDULED'      AS STATE_NAME,
       h1.percent_known_time_warning_scheduled AS Time
FROM   hostgroup_service_availability h
       INNER JOIN (SELECT hostgroup_name,
                          percent_known_time_warning_scheduled,
                          datestamp
                   FROM   hostgroup_service_availability
                   WHERE  hostgroup_name = Replace(?, ' ', '+')
                          AND datestamp >= ?
                          AND datestamp <= ?) h1
               ON h1.hostgroup_name = h.hostgroup_name
                  AND h1.datestamp = h.datestamp
UNION
SELECT h.hostgroup_name,
       h.datestamp,
       '% KNOWN TIME WARNING - UNSCHEDULED'      AS STATE_NAME,
       h1.percent_known_time_warning_unscheduled AS Time
FROM   hostgroup_service_availability h
       INNER JOIN (SELECT hostgroup_name,
                          percent_known_time_warning_unscheduled,
                          datestamp
                   FROM   hostgroup_service_availability
                   WHERE  hostgroup_name = Replace(?, ' ', '+')
                          AND datestamp >= ?
                          AND datestamp <= ?) h1
               ON h1.hostgroup_name = h.hostgroup_name
                  AND h1.datestamp = h.datestamp
UNION
SELECT h.hostgroup_name,
       h.datestamp,
       '% TIME OTHER'                                AS
       STATE_NAME
       ,
       ( 100.0 - ( h1.percent_total_time_ok
                   + h1.percent_known_time_critical_scheduled
                   + h1.percent_known_time_critical_unscheduled
                   + h1.percent_known_time_warning_scheduled
                   + h1.percent_known_time_warning_unscheduled ) ) AS
       Time
FROM   hostgroup_service_availability h
       INNER JOIN (SELECT hostgroup_name,
                          percent_total_time_ok,
```

```
                        percent_known_time_critical_scheduled,
                        percent_known_time_critical_unscheduled,
                        percent_known_time_warning_scheduled,
                        percent_known_time_warning_unscheduled,
                        datestamp
                FROM   hostgroup_service_availability
                WHERE  hostgroup_name = Replace(?, ' ', '+')
                    AND datestamp >= ?
                    AND datestamp <= ?) h1
            ON h1.hostgroup_name = h.hostgroup_name
              AND h1.datestamp = h.datestamp) AS ha
ORDER  BY ha.datestamp
```

6. gw-hostgroup-availability.rptdesign: hostgroup-host-chart

```
SELECT *
FROM   (SELECT h.hostgroup_name,
        h.datestamp,
        '% TOTAL TIME UP'       AS STATE_NAME,
        h1.percent_total_time_up AS Time
    FROM   hostgroup_host_availability h
        INNER JOIN (SELECT hostgroup_name,
                        percent_total_time_up,
                        datestamp
                FROM   hostgroup_host_availability
                WHERE  hostgroup_name = Replace(?, ' ', '+')
                    AND datestamp >= ?
                    AND datestamp <= ?) h1
            ON h1.hostgroup_name = h.hostgroup_name
              AND h1.datestamp = h.datestamp
    UNION
    SELECT h.hostgroup_name, h.datestamp,
        '% TIME DOWN - UNSCHEDULED'     AS STATE_NAME,
        h1.percent_time_down_unscheduled AS time
    FROM   hostgroup_host_availability h
        INNER JOIN (SELECT hostgroup_name,
                        percent_time_down_unscheduled,
                        datestamp
                FROM   hostgroup_host_availability
                WHERE  hostgroup_name = Replace(?, ' ', '+')
                    AND datestamp >= ?
                    AND datestamp <= ?) h1
            ON h1.hostgroup_name = h.hostgroup_name
              AND h1.datestamp = h.datestamp
    UNION
    SELECT h.hostgroup_name,
        h.datestamp,
        '% TIME DOWN - SCHEDULED'     AS STATE_NAME,
        h1.percent_time_down_scheduled AS Time
    FROM   hostgroup_host_availability h
        INNER JOIN (SELECT hostgroup_name,
                        percent_time_down_scheduled,
                        datestamp
                FROM   hostgroup_host_availability
                WHERE  hostgroup_name = Replace(?, ' ', '+')
```

```
                        AND datestamp >= ?
                        AND datestamp <= ?) h1
                ON h1.hostgroup_name = h.hostgroup_name
                  AND h1.datestamp = h.datestamp
      UNION
      SELECT h.hostgroup_name,
           h.datestamp,
           '% TIME OTHER',
           ( 100.0 - ( h1.percent_total_time_up
                     + h1.percent_time_down_unscheduled
                     + h1.percent_time_down_scheduled ) ) AS Time
      FROM   hostgroup_host_availability h
           INNER JOIN (SELECT hostgroup_name,
                        percent_total_time_up,
                        percent_time_down_unscheduled,
                        percent_time_down_scheduled,
                        datestamp
                    FROM   hostgroup_host_availability
                    WHERE  hostgroup_name = Replace(?, ' ', '+')
                        AND datestamp >= ?
                        AND datestamp <= ?) h1
                ON h1.hostgroup_name = h.hostgroup_name
                  AND h1.datestamp = h.datestamp)AS ha
ORDER  BY ha.datestamp
```

7. gw-hostgroup-availability.rptdesign:gw-hg-service-availability

```
SELECT hgs.datestamp, hgs.hostgroup_name,
      CASE
        WHEN ( hgs.percent_total_time_ok <= 0 ) THEN 0
        ELSE ( Trunc(Cast(hgs.percent_total_time_ok AS NUMERIC), 2) )
      END AS PERCENT_TOTAL_TIME_OK,
      CASE
        WHEN ( hgs.percent_known_time_critical_scheduled <= 0 ) THEN 0
        ELSE ( Trunc(Cast(hgs.percent_known_time_critical_scheduled AS NUMERIC)
             , 2) )
      END AS PERCENT_KNOWN_TIME_CRITICAL_SCHEDULED,
      CASE
        WHEN ( hgs.percent_known_time_critical_unscheduled <= 0 ) THEN 0
        ELSE ( Trunc(Cast(hgs.percent_known_time_critical_unscheduled AS
                     NUMERIC), 2)
          )
      END AS PERCENT_KNOWN_TIME_CRITICAL_UNSCHEDULED,
      CASE
        WHEN ( hgs.percent_known_time_warning_scheduled <= 0 )THEN 0
        ELSE ( Trunc(Cast(hgs.percent_known_time_warning_scheduled AS NUMERIC),
             2) )
      END AS PERCENT_KNOWN_TIME_WARNING_SCHEDULED,
      CASE
        WHEN ( hgs.percent_known_time_warning_unscheduled <= 0 )THEN 0
        ELSE Trunc(Cast(hgs.percent_known_time_warning_unscheduled AS NUMERIC),
          2)
      END AS PERCENT_KNOWN_TIME_WARNING_UNSCHEDULED,
      CASE
        WHEN ( ( 100.0 - ( percent_total_time_ok
```

```
                    + percent_known_time_critical_scheduled
                    + percent_known_time_critical_unscheduled
                    + percent_known_time_warning_scheduled
                    + percent_known_time_warning_unscheduled ) ) <= 0 )
    THEN 0
     ELSE Trunc(Cast(100.0 - ( percent_total_time_ok
                        + percent_known_time_critical_scheduled
                        + percent_known_time_critical_unscheduled
                        + percent_known_time_warning_scheduled
                        + percent_known_time_warning_unscheduled ) AS
                NUMERIC), 2)
    END AS PERCENT_OTHER
FROM   hostgroup_service_availability hgs
WHERE  hgs.hostgroup_name = Replace(?, ' ', '+')
    AND hgs.datestamp >= ?
    AND hgs.datestamp <= ?
ORDER  BY hgs.datestamp
```

8. gw-hostgroup-availability.rptdesign:gw-hg-host-availability

```
SELECT hgh.datestamp,
    hgh.hostgroup_name,
    hgh.percent_total_time_up,
    hgh.percent_time_down_scheduled,
    hgh.percent_time_down_unscheduled,
    ( 100.0 - ( percent_total_time_up
            + percent_time_down_scheduled
            + percent_time_down_unscheduled ) ) AS PERCENT_OTHER
FROM   hostgroup_host_availability hgh
WHERE  hgh.hostgroup_name = Replace(?, ' ', '+')
    AND hgh.datestamp >= ?
    AND hgh.datestamp <= ?
ORDER  BY hgh.datestamp
```

9. gw-service-state-transitions.rptdesign:service-data

```
SELECT servicestatus.servicedescription, monitorstatus.NAME
FROM   monitorstatus, servicestatus, host
WHERE  servicestatus.monitorstatusid = monitorstatus.monitorstatusid
    AND servicestatus.servicedescription = ?
    AND servicestatus.hostid = host.hostid
    AND host.hostname = ?;
```

Event

1. gw-event-history.rptdesign: GWEventStatisticsMonitorStatus

http://localhost:8080/foundation-webapp/services/wscommon?wsdl

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Body>
      <m:performEntityQuery xmlns:m="urn:fws">
         <entityType xsi:type="xsd:string">&?entityType?&</entityType>
         <filter>
            <LeftFilter>
               <LeftFilter>
                  <StringProperty>
                     <name>StatisticType</name>
                     <value>&?StatisticType?&</value>
                  </StringProperty>
                  <Operator>EQ</Operator>
               </LeftFilter>
               <RightFilter>
                  <LeftFilter>
                     <StringProperty>
                        <name>&?Query?&</name>
                        <value>&?value?&</value>
                     </StringProperty>
                     <Operator>EQ</Operator>
                  </LeftFilter>
                  <RightFilter>
                     <StringProperty>
                        <name>ApplicationType</name>
                        <value>&?ApplicationType?&</value>
                     </StringProperty>
                     <Operator>EQ</Operator>
                  </RightFilter>
                  <Operator>AND</Operator>
               </RightFilter>
               <Operator>AND</Operator>
            </LeftFilter>
            <RightFilter>
               <LeftFilter>
                  <DateProperty>
                     <name>StartDate</name>
                     <value>&?StartDate?&</value>
                  </DateProperty>
                  <Operator>EQ</Operator>
               </LeftFilter>
               <RightFilter>
                  <DateProperty>
                     <name>EndDate</name>
                     <value>&?EndDate?&</value>
                  </DateProperty>
                  <Operator>EQ</Operator>
               </RightFilter>
```

```
                <Operator>AND</Operator>
            </RightFilter>
            <Operator>AND</Operator>
        </filter>
        <sort></sort>
        <firstResult xsi:type="xsd:int">-1</firstResult>
        <maxResults xsi:type="xsd:int">-1</maxResults>
    </m:performEntityQuery>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Performance

1. gw-epr-host-multi-variable.rptreport:LogPerformanceData

```
SELECT left1.lastchecktime AS LastCheckTime,
    left1.minimum      AS left_min,
    left1.maximum      AS left_max,
    left1.average      AS left_avg,
    right1.minimum     AS right_min,
    right1.maximum     AS right_max,
    right1.average     AS right_avg
FROM  (SELECT lastchecktime, minimum, maximum, average
     FROM  logperformancedata
     WHERE  performancedatalabelid = (SELECT performancedatalabelid
                          FROM  performancedatalabel
                          WHERE  performancename = ?)
         AND servicestatusid IN (SELECT servicestatusid
                          FROM  servicestatus
                          WHERE  hostid = (SELECT hostid
                                   FROM  host
                                   WHERE  hostname = ?)))
    AS left1,
    (SELECT lastchecktime, minimum, maximum, average
     FROM  logperformancedata
     WHERE  performancedatalabelid = (SELECT performancedatalabelid
                          FROM  performancedatalabel
                          WHERE  performancename = ?)
         AND servicestatusid IN (SELECT servicestatusid
                          FROM  servicestatus
                          WHERE  hostid = (SELECT hostid
                                   FROM  host
                                   WHERE  hostname = ?)))
    AS right1
WHERE  Date(left1.lastchecktime) = Date(right1.lastchecktime)
    AND Date(left1.lastchecktime) BETWEEN ? AND ?
```

2. gw-epr-host.rptdesign:host-performance\

```
SELECT Cast(lpd.lastchecktime AS DATE) AS LastCheckTime,
    Max(lpd.maximum)          AS Maximum,
    Min(lpd.minimum)          AS Minimum,
    Avg(lpd.average)          AS Average,
    pdl.performancename,
    h.hostname
FROM  performancedatalabel pdl
    INNER JOIN logperformancedata lpd
        ON lpd.performancedatalabelid = pdl.performancedatalabelid
    INNER JOIN servicestatus ss
        ON lpd.servicestatusid = ss.servicestatusid
    INNER JOIN host h
        ON ss.hostid = h.hostid
WHERE  pdl.performancename = ?
    AND h.hostname = ?
    AND Cast(lpd.lastchecktime AS DATE) >= Cast(? AS DATE)
    AND Cast(lpd.lastchecktime AS DATE) <= Cast(? AS DATE)
```

```
GROUP  BY lpd.lastchecktime, pdl.performancename, h.hostname;
```

3. gw-epr-host.rptdesign:host-service-performance

```
SELECT Max(lpd.maximum) AS Maximum,
       Min(lpd.minimum) AS Minimum,
       Avg(lpd.average) AS Average,
       ss.servicedescription
FROM   performancedatalabel pdl
       INNER JOIN logperformancedata lpd
               ON lpd.performancedatalabelid = pdl.performancedatalabelid
       INNER JOIN servicestatus ss
               ON lpd.servicestatusid = ss.servicestatusid
       INNER JOIN host h
               ON ss.hostid = h.hostid
WHERE  pdl.performancename = ?
       AND h.hostname = ?
       AND Cast(lpd.lastchecktime AS DATE) >= Cast(? AS DATE)
       AND Cast(lpd.lastchecktime AS DATE) <= Cast(? AS DATE)
GROUP  BY pdl.performancename, h.hostname, ss.servicedescription
```

4. *gw-epr-hostgroup-multi-variable*.rptdesign:LogPerformanceData

```
SELECT left1.lastchecktime AS LastCheckTime,
       left1.average       AS left_avg,
       right1.average      AS right_avg
FROM   (SELECT Date(lpd.lastchecktime) AS LastCheckTime,
               hg1.NAME               AS HostGroupName_Left,
               Avg(average)           AS Average
        FROM   logperformancedata lpd,
               servicestatus ss1,
               hostgroupcollection hg_col_1,
               hostgroup hg1,
               performancedatalabel pdl
        WHERE  lpd.performancedatalabelid = pdl.performancedatalabelid
               AND pdl.performancename = ?
               AND lpd.servicestatusid = ss1.servicestatusid
               AND ss1.hostid = hg_col_1.hostid
               AND hg_col_1.hostgroupid = hg1.hostgroupid
               AND hg1.NAME = ?
        GROUP  BY Date(lpd.lastchecktime),
                  hg1.NAME
        ORDER  BY Date(lpd.lastchecktime) ASC) AS left1,
       (SELECT Date(lpd.lastchecktime) AS LastCheckTime,
               hg1.NAME               AS HostGroupName_Left,
               Avg(average)           AS Average
        FROM   logperformancedata lpd,
               servicestatus ss1,
               hostgroupcollection hg_col_1,
               hostgroup hg1,
               performancedatalabel pdl
        WHERE  lpd.performancedatalabelid = pdl.performancedatalabelid
               AND pdl.performancename = ?
               AND lpd.servicestatusid = ss1.servicestatusid
```

```
            AND ss1.hostid = hg_col_1.hostid
            AND hg_col_1.hostgroupid = hg1.hostgroupid
            AND hg1.NAME = ?
      GROUP  BY Date(lpd.lastchecktime),
                hg1.NAME
      ORDER  BY Date(lpd.lastchecktime) ASC) AS right1
WHERE  Date(left1.lastchecktime) = Date(right1.lastchecktime)
     AND Date(left1.lastchecktime) BETWEEN ? AND ?
```

5. *gw-epr-hostgroup-multi-variable*.rptdesign: PerformanceDataTable

```
SELECT left1.lastchecktime  AS LastCheckTime,
     left1.hostgroupname  AS HostGroupName_Left,
     left1.hostname       AS HostName_Left,
     left1.minimum        AS left_min,
     left1.maximum        AS left_max,
     left1.average        AS left_avg,
     right1.hostgroupname AS HostGroupName_Right,
     right1.hostname      AS HostName_Right,
     right1.minimum       AS right_min,
     right1.maximum       AS right_max,
     right1.average       AS right_avg
FROM   (SELECT lpd.lastchecktime,
           hg1.NAME                    AS HostGroupName,
           (SELECT hostname
            FROM   host
            WHERE  hostid = hg_col_1.hostid) AS HostName,
           minimum,
           maximum,
           average
      FROM   logperformancedata lpd,
           servicestatus ss1,
           hostgroupcollection hg_col_1,
           hostgroup hg1
      WHERE  performancedatalabelid = (SELECT performancedatalabelid
                                       FROM   performancedatalabel
                                       WHERE  performancename = ?)
           AND lpd.servicestatusid = ss1.servicestatusid
           AND ss1.hostid = hg_col_1.hostid
           AND hg_col_1.hostgroupid = hg1.hostgroupid
           AND hg1.NAME = ?) AS left1,
     (SELECT lpd.lastchecktime,
           hg1.NAME                    AS HostGroupName,
           (SELECT hostname
            FROM   host
            WHERE  hostid = hg_col_1.hostid) AS HostName,
           minimum,
           maximum,
           average
      FROM   logperformancedata lpd,
           servicestatus ss1,
           hostgroupcollection hg_col_1,
           hostgroup hg1
      WHERE  performancedatalabelid = (SELECT performancedatalabelid
                                       FROM   performancedatalabel
```

```
                                WHERE  performancename = ?)
            AND lpd.servicestatusid = ss1.servicestatusid
            AND ss1.hostid = hg_col_1.hostid
            AND hg_col_1.hostgroupid = hg1.hostgroupid
            AND hg1.NAME = ?) AS right1
WHERE  Date(left1.lastchecktime) = Date(right1.lastchecktime)
     AND Date(left1.lastchecktime) BETWEEN ? AND ?
```

6. *gw-epr-hostgroup-topfive.rptdesign:misc-graph-labels*

```
SELECT h.hostname,
     pdl.servicedisplayname,
     pdl.performancename,
     pdl.metriclabel,
     pdl.unit
FROM   performancedatalabel pdl
     INNER JOIN logperformancedata lpd
          ON lpd.performancedatalabelid = pdl.performancedatalabelid
     INNER JOIN servicestatus ss
          ON lpd.servicestatusid = ss.servicestatusid
     INNER JOIN host h
          ON ss.hostid = h.hostid
     INNER JOIN hostgroupcollection hgc
          ON h.hostid = hgc.hostid
     INNER JOIN hostgroup hg
          ON hgc.hostgroupid = hg.hostgroupid
WHERE  pdl.performancename = ?
     AND hg.name = ?
     AND lpd.lastchecktime >= ?
     AND lpd.lastchecktime <= ?
LIMIT  1
```

7. *gw-epr-hostgroup-topfive.rptdesign:host-count*

```
 SELECT     Count(h.hostid) AS numhosts,
          a.numhostswithdata
 FROM       hostgroup hg
 INNER JOIN hostgroupcollection hgc
 ON         hg.hostgroupid = hgc.hostgroupid
 INNER JOIN host h
 ON         hgc.hostid = h.hostid
 INNER JOIN
          (
            SELECT     1                    AS recid,
                     Count(DISTINCT h.hostid) AS numhostswithdata
            FROM       performancedatalabel pdl
            INNER JOIN logperformancedata lpd
            ON         lpd.performancedatalabelid = pdl.performancedatalabelid
            INNER JOIN servicestatus ss
            ON         lpd.servicestatusid = ss.servicestatusid
            INNER JOIN host h
            ON         ss.hostid = h.hostid
            INNER JOIN hostgroupcollection hgc
            ON         h.hostid = hgc.hostid
```

```
            INNER JOIN hostgroup hg
            ON         hgc.hostgroupid = hg.hostgroupid
            WHERE      hg.NAME = ?
            AND        pdl.performancename = ? ) AS a
WHERE       hg.NAME = ?
GROUP BY    a.numhostswithdata
```