Key Metrics for MicroStrategy Administrators

Program Document

Created by InfoCepts

Contents

Introduction

The document contains some of the Key Metrics that prove to be critical while monitoring the health of MicroStrategy Environment. These metrics belong to various categories like System Health, Application Health, License Compliance, Failures and Object Usage.

These Metrics can be used in three different ways:

1. As Ad-hoc queries to retrieve and analyze data
2. To retrieve data using automated scripts
3. To create a data warehouse and reports can be built for analysis using MicroStrategy

NOTE:

SQL queries in this document have been written for MicroStrategy 9.3 Metadata and Statistics tables implemented in Microsoft SQL Server. The queries can be modified syntactically for other databases.

In this document, you will find a database function "MSTRUID" used in multiple places. This is a user defined function that converts OBJECT ID in MicroStrategy Metadata to match OBJECT ID in MicroStrategy Statistics database.

For more details on what this function does and why you need to do this, please see http://www.bryanbrandow.com/2011/07/changes-to-object-ids-in-92.html

MSTRUID Function Definition:

```
CREATE FUNCTION <METADATA DATABASE>.[DBO].[MSTRUID](@UUID2 UNIQUEIDENTIFIER)
RETURNS VARCHAR(32)
AS BEGIN
RETURN CAST(
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 1,8) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 15,4) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 10,4) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 27,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 25,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 22,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 20,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 35,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 33,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 31,2) +
SUBSTRING(CAST(@UUID2 AS VARCHAR(36)), 29,2)
AS VARCHAR(32))
END
```

1. Top Longest Running Reports and Documents by Project

Helps Administrators to identify long running reports that are candidates for performance tuning and optimizations.

The queries must be run against both Statistics & Metadata Database

```
--TOP 3 REPORTS IN LAST WEEK--
SELECT TOP 3 REPORTID AS LONGEST_RUNNING_REPORT_ID
,Z.OBJECT_NAME "LONGEST_RUNNING_REPORT_NAME"
, 'REPORT' AS OBJECT_TYPE
,PROJECT_ID
,(EXECFINISHTIME-EXECSTARTTIME) AS EXECUTION_TIME
FROM <STATS DATABASE>.DBO.IS_REPORT_STATS Y
,<METADATA DATABASE>.DBO.DSSMDOBJINFO Z
WHERE(<METADATA DATABASE>.DBO.MSTRUID(Z.OBJECT_ID))= Y.REPORTID
AND Z.SUBTYPE<>776
AND DATEDIFF(DAY,Y.DAY_ID,SYSDATETIME())<=7
   ORDER BY (EXECFINISHTIME-EXECSTARTTIME) DESC

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A WEEK (7 DAYS), USERS CAN MODIFY THE
SAME ACCORDING TO THEIR REQUIREMENT*/

--TOP 3 DOCUMENTS IN LAST WEEK--
SELECT TOP 3 DOCUMENTID AS LONGEST_RUNNING_DOC_ID
,Z.OBJECT_NAME "LONGEST_RUNNING DOC_NAME"
,'DOCUMENT' AS OBJECT_TYPE
,PROJECT_ID
,(FINISHTIME-STARTTIME) AS EXECUTION_TIME
FROM <STATS DATABASE>.DBO.IS_DOCUMENT_STATS Y
,<METADATA DATABASE>.DBO.DSSMDOBJINFO Z
WHERE (<METADATA DATABASE>.DBO.MSTRUID(Z.OBJECT_ID))= Y.DOCUMENTID
AND DATEDIFF(DAY,Y.DAY_ID,SYSDATETIME())<=7
   ORDER BY (FINISHTIME-STARTTIME) DESC
/*THE SQL IS DESIGNED FOR AN INTERVAL OF A WEEK (7 DAYS), USERS CAN MODIFY THE
SAME ACCORDING TO THEIR REQUIREMENT*/
```

	LONGEST_RUNNING_REPORT_ID	LONGEST_RUNNING_REPORT_NAME	OBJECT_TYPE	PROJECT_ID	EXECUTION_TIME
1	BE26F4C648922F39ECFDCB8447CD5371	test report	REPORT	D1027876-507B-4A99-BFB6-AFFC26EBFB8E	6004012
2	7C782FF74D570388F56215A6C2FB9F71	test 2 – freeform	REPORT	D1027876-507B-4A99-BFB6-AFFC26EBFB8E	2467643
3	05B202B9999F4C1AB960DA6208CADF3D	Blank Report	REPORT	3EC4843D-D8C2-4E86-AA36-0B7D0E515191	2025068

	LONGEST_RUNNING_DOC_ID	LONGEST_RUNNING DOC_NAME	OBJECT_TYPE	PROJECT_ID	EXECUTION_TIME
1	860785944135B3DCC4E0A78C7ED7E2E8	System Health History_v2(history inside)	DOCUMENT	A8F35AD9-337F-4B97-9171-749664714BB3	3626714
2	51C633B54ECDF0B5893A5EA14A2E836D	Store Sales	DOCUMENT	E3549528-D975-4920-A5FD-F6A5EC8BFDEE	1332470
3	B59C26F04A61BE8DEBDACB96EB554D50	Impact Analysis	DOCUMENT	A8F35AD9-337F-4B97-9171-749664714BB3	941809

2. Top Users using Reports

Provides a list of the users who frequently execute reports in a certain time interval

The queries must be run against both Statistics & Metadata Database

```
SELECT A.OBJECT_NAME AS USER_NAME, SUM(B.NO_OF_JOBS_EXECUTED) "JOBS_EXECUTED"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO A,
(SELECT A11.USERID  AS  "USERID",
```

```
COUNT(REPORTID)  "NO_OF_JOBS_EXECUTED"
FROM <STATS DATABASE>.DBO.IS_REPORT_STATS A11, <STATS
DATABASE>.DBO.IS_SESSION_STATS A22
WHERE A11.SESSIONID=A22.SESSIONID
AND DATEDIFF(DAY,A11.DAY_ID,SYSDATETIME())<=1
GROUP BY A11.USERID,A22.SESSIONID) B
WHERE (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID))= B.USERID
GROUP BY A.OBJECT_NAME
ORDER BY 2 DESC

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

	USER_NAME	JOBS_EXECUTED
1	Rishabh	1790
2	Swati	1108
3	Ram Reddy	796
4	Vizeh	184
5	Administrator	86
6	Bhushan	58
7	Gaurav Kolarkar	10

3. Intelligence Server Scheduled Reports

Provides a list of the reports that are executed as per a defined schedule.

The query must be run against Metadata Database

```
SELECT C.OBJECT_NAME AS "PROJECT_NAME" ,A.OBJECT_NAME AS "REPORT_NAME"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO A
 , <METADATA DATABASE>.DBO.DSSMDOBJINFO C
 ,(SELECT DISTINCT REPORTID
  , PROJECTID
  FROM <STATS DATABASE>.DBO.IS_REPORT_STATS
  WHERE SCHEDULEINDICATOR='1'
  AND DATEDIFF(DAY,DAY_ID,SYSDATETIME())<=1) B
WHERE (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID))= B.REPORTID
AND (<METADATA DATABASE>.DBO.MSTRUID(C.OBJECT_ID))= B.PROJECTID
ORDER BY 2

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

Results | Messages

	PROJECT_NAME	REPORT_NAME
1	iManage_InfaDS	Service Availability
2	iManage_InfaDS	System Health Check - Grid
3	iManage_InfaDS	Disk Usage
4	iManage_InfaDS	RAM Usage
5	iManage_InfaDS	System Health Check Disk Usage- Grid

4. Top Hit Cubes Count

Provides a list of the cubes that are most hit by the reports.

The queries must be run against both Statistics & Metadata Database

```
SELECT B.OBJECT_NAME AS "CUBE_NAME"
,C.HIT_COUNT
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO B
,(SELECT DISTINCT A.CUBEREPORTGUID
  ,COUNT (A.CUBEREPORTGUID) AS "HIT_COUNT"
  FROM <STATS DATABASE>.DBO.IS_CUBE_REP_STATS A
  WHERE A.CUBEINSTANCEID IS NOT NULL
  AND DATEDIFF(DAY,A.DAY_ID,SYSDATETIME())<=7
  GROUP BY A.CUBEREPORTGUID) C
WHERE (<METADATA DATABASE>.DBO.MSTRUID(B.OBJECT_ID))= C.CUBEREPORTGUID
ORDER BY C.HIT_COUNT DESC

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A WEEK (7 DAYS), USERS CAN MODIFY THE
SAME ACCORDING TO THEIR REQUIREMENT*/
```

	CUBE_NAME	HIT_COUNT
1	1234	25
2	test cube import	13
3	Cube converted from user	5
4	import data for Q2 2012	3
5	KVAT - Test Intelligent Cube - Category Sub Cat ...	2
6	test	2
7	Test	2
8	Test1	2
9	test1	1

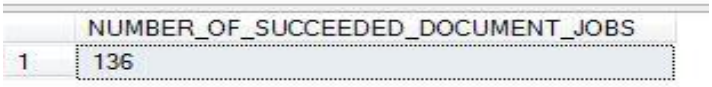
5. Count of Jobs Run (Succeeded/Failed)

Provides a count of the Report, Document Jobs that have succeeded/failed.

The query must be run against Statistics Database.

```
--COUNT OF FAILED REPORT JOBS WITHIN 24 HOURS--
SELECT COUNT(DISTINCT A.JOBID) AS "NUMBER_OF_FAILED_REPORT_JOBS"
FROM <STATS DATABASE NAME>.DBO.IS_REPORT_STATS A
WHERE DATEDIFF(HH,DAY_ID,SYSDATETIME())<=24
AND JOBSTATUS=4

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/


--COUNT OF SUCCEEDED REPORT JOBS WITHIN 24 HOURS--
SELECT COUNT(DISTINCT A.JOBID) AS "NUMBER_OF_SUCCEEDED_REPORT_JOBS"
FROM <STATS DATABASE>.DBO.IS_REPORT_STATS A
WHERE DATEDIFF(HH,DAY_ID,SYSDATETIME())<=24
AND JOBSTATUS=3

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/


--COUNT OF FAILED DOCUMENT JOBS WITHIN 24 HOURS--
SELECT COUNT(DISTINCT A.JOBID) AS "NUMBER_OF_FAILED_DOCUMENT_JOBS"
FROM <STATS DATABASE>.DBO.IS_DOCUMENT_STATS A
WHERE DATEDIFF(HH,DAY_ID,SYSDATETIME())<=24
AND EXECSTATUS=4

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/


--COUNT OF SUCCEEDED DOCUMENT JOBS WITHIN 24 HOURS--
SELECT COUNT(DISTINCT A.JOBID) AS "NUMBER_OF_SUCCEEDED_DOCUMENT_JOBS"
FROM <STATS DATABASE>.DBO.IS_DOCUMENT_STATS A
WHERE DATEDIFF(HH,DAY_ID,SYSDATETIME())<=24
AND EXECSTATUS=3
```

```
/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```



6. Number of Jobs Currently Executing, Waiting in queue

Helps Administrators in determining the overall load on the environment at a given point in time.

This includes a command manager script file with ".scp" as extension. The script file has command(s) you want to execute in this case it is "LIST ALL JOBS".

```
<scriptfilename> -> LIST ALL JOBS;
```

The code below is used to access command prompt utility of Microstrategy Command Manager.
Keys:

-n Project Source

-u Username

-p Password

-f Script file path (.scp file)

-o Output text file name

-xml Converted xml output of the output file

Save the code below with appropriate values in a batch file with ".bat" extension.

```
cmdmgr -n <source> -u <username> -p <password> -f <scriptfilename> -o <Output
text file name> -xml <Converted xml output filename>
```

```
<?xml version="1.0"?>
- <CommandManagerResults>
  - <ListJobs>
    - <Row>
        <JobId>27561</JobId>
        <Owner>Nirav Prasad</Owner>
        <JobStatus>Waiting For Autoprompt</JobStatus>
        <Description>Running report New Report</Description>
        <CreationTime>October 10, 2012 12:13:24 AM PDT</CreationTime>
        <ProjectId>E35495284920D975A5F6FDA5EEFD8BEC</ProjectId>
        <Project>MicroStrategy Tutorial</Project>
        <JobDuration>11</JobDuration>
      </Row>
    </ListJobs>
  </CommandManagerResults>
```

7. Average Running/ Wait time for Reports by hour of day

Provides average waiting & queue time statistics for reports by hour. Average waiting time for the jobs can be used to optimize and tune MicroStrategy BI System.

The query must be run against Statistics Database.

```
SELECT HOUR_ID
,AVG(DATEDIFF(SECOND,STARTTIME,FINISHTIME))AS AVG_RUNNING_IN_SEC
,CONVERT(DECIMAL(8,2),AVG(QUEUETIME)) AS AVG_QUEUETIME_IN_SEC
FROM <STATS DATABASE>.DBO.IS_REP_STEP_STATS
WHERE DATEDIFF(DAY,DAY_ID,SYSDATETIME())<=1
GROUP BY HOUR_ID
ORDER BY HOUR_ID

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

	HOUR_ID	AVG_RUNNING_IN_SEC	AVG_QUEUETIME_IN_SEC
1	2013020110	3	3797.85
2	2013020111	5	1384.73
3	2013020112	3	2537.39
4	2013020116	5	1106.73
5	2013020117	6	1357.16
6	2013020118	4	1670.31

8. Number of Jobs by Connection Type

Provides a count of the Jobs with respect to those executed on Desktops and Web. This helps Administrators to know the usage of corresponding connection type.

The query must be run against Statistics Database.

```
SELECT DISTINCT COUNT(JOBID) AS  "NO_OF_JOBS",
CASE WHEN CONVERT(VARCHAR(10),B.EVENTSOURCE)='1' THEN 'DESKTOP'
WHEN CONVERT(VARCHAR(10),B.EVENTSOURCE)='6' THEN 'WEB'
```

```
ELSE '0' END AS CONNECTION_SOURCE
FROM <STATS DATABASE>.DBO.IS_REPORT_STATS A,
<STATS DATABASE>.DBO.IS_SESSION_STATS B
WHERE
A.SESSIONID=B.SESSIONID
AND
B.EVENTSOURCE IN (1,6)
AND
DATEDIFF(DAY,A.DAY_ID,SYSDATETIME())<=7
GROUP BY B.EVENTSOURCE

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A WEEK (7 DAYS), USERS CAN MODIFY THE
SAME ACCORDING TO THEIR REQUIREMENT*/
```

	NO_OF_JOBS	CONNECTION_SOURCE
1	5844	DESKTOP
2	8677	WEB

9. Reports Dependent on Cubes

Provides a list of reports that hit cubes for execution.

The queries must be run against both Statistics & Metadata Database

```
SELECT DISTINCT Y.OBJECT_NAME AS "CUBE_NAME"
,X.OBJECT_NAME AS "REPORT_NAME"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO X, <METADATA
DATABASE>.DBO.DSSMDOBJINFO Y,
(SELECT A.DAY_ID
 ,A.REPORTID
 ,A.CUBEINSTANCEID
 ,B.CUBEREPORTGUID
 FROM <STATS DATABASE>.DBO.IS_REPORT_STATS A
 ,<STATS DATABASE>.DBO.IS_CUBE_REP_STATS B
 WHERE A.CUBEINSTANCEID IS NOT NULL
AND A.CUBEINSTANCEID=B.CUBEINSTANCEID
AND A.REPORTID <> B.CUBEREPORTGUID ) W
WHERE (<METADATA DATABASE>.DBO.MSTRUID(X.OBJECT_ID))= W.REPORTID
AND (<METADATA DATABASE>.DBO.MSTRUID(Y.OBJECT_ID))= W.CUBEREPORTGUID
 AND DATEDIFF(DAY,W.DAY_ID,SYSDATETIME())<=1

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

CUBE_NAME	REPORT_NAME
my test cube	Query Grid-Date Time
my test cube	License Summary1
my test cube	Session with Performance degradation
my test cube	Availability Service CY vs LY
my test cube	Target
my test cube	System Health Check - Disk Usage - Grid
my test cube	TX_User
my test cube	New Report 2
my test cube	Pie Charts
my test cube	System Health Check - Graph

10. Top Used Tables

Determines the tables that are frequently hit by reports. This information helps a Database Administrator while addressing database outages.

The queries must be run against both Statistics & Metadata Database

```
SELECT TOP 5 C.OBJECT_NAME AS TABLE_NAME
,C.TIMES_HIT
FROM(SELECT DISTINCT B.TABLEID
    ,A.OBJECT_NAME
    ,COUNT(B.TABLEID) AS "TIMES_HIT"
    FROM <STATS DATABASE>.DBO.IS_REP_COL_STATS B
   JOIN <METADATA DATABASE>.DBO.DSSMDOBJINFO A
   ON (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID))= B.TABLEID
   WHERE DATEDIFF(DAY,B.DAY_ID,SYSDATETIME())<=1
   GROUP BY B.TABLEID, A.OBJECT_NAME) C
ORDER BY C.TIMES_HIT DESC


/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

	TABLE_NAME	TIMES_HIT
1	VW_LU_SERVER_DETAILS	13599
2	VW_FA_AGG_SYSTEM_HEALTH_STATS_CURR_HOUR	10069
3	VW_LU_DAY	6002
4	VW_LU_SERVICE_DRIVE_DETAILS	5547
5	VW_FA_SYS_LICENSE_AUDIT_RPT	5468

11. Number of Caches loaded and Total Memory Consumption

Helps in determining the memory usage by caches.

The query must be run against Statistics Database.

```
SELECT DISTINCT(
SELECT SUM(COUNTER_VALUE) AS TOT_COUNTER_VAL FROM <STATS
DATABASE>.DBO.IS_PERF_MON_STATS
```

```
WHERE EVENT_TIME = (SELECT MAX(EVENT_TIME) FROM <STATS
DATABASE>.DBO.IS_PERF_MON_STATS)
AND
(COUNTER_NAME IN ('NUMBER OF REPORT CACHES',
'NUMBER OF LOCAL INTELLIGENT CUBE CACHES','NUMBER OF DOCUMENT CACHES IN MEMORY'
,'NUMBER OF LOCAL DOCUMENT CACHES','NUMBER OF INTELLIGENT CUBE CACHES IN
MEMORY')
)) "NO_OF_CACHE_LOADED",(
SELECT SUM(COUNTER_VALUE) AS TOT_COUNTER_VAL FROM <STATS
DATABASE>.DBO.IS_PERF_MON_STATS
WHERE EVENT_TIME = (SELECT MAX(EVENT_TIME) FROM <STATS
DATABASE>.DBO.IS_PERF_MON_STATS)
AND
(COUNTER_NAME IN ('TOTAL LOCAL REPORT CACHE SIZE (MB)'
,'TOTAL LOCAL DOCUMENT CACHE SIZE (MB)','TOTAL LOCAL CUBE CACHE SIZE (MB)')
)) AS "TOTAL_MEMORY_CONSUMED(MB)" FROM <STATS DATABASE>.DBO.IS_PERF_MON_STATS
WHERE EVENT_TIME = (SELECT MAX(EVENT_TIME) FROM <STATS
DATABASE>.DBO.IS_PERF_MON_STATS)

/*THE SQL IS DESIGNED FOR CURRENT STATISTICS*/
```

	NO_OF_CACHE_LOADED	TOTAL_MEMORY_CONSUMED(MB)
1	47	1

12. Average Memory Consumption by Cubes

Helps in determining the memory usage by the cubes against which reports are executed.

The queries must be run against both Statistics & Metadata Database

```
SELECT DISTINCT MD.OBJECT_ID AS CUBE_ID
, MD.OBJECT_NAME AS CUBE_NAME,MD.PROJECT_ID
, CONVERT(DECIMAL(8,2),B.CUBE_SIZE) AS CUBE_SIZE_KB
,B.HIT_COUNT
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO MD,(SELECT DISTINCT A.PROJECTID
 ,A.CUBEREPORTGUID
 ,COUNT(A.CUBEREPORTGUID) AS "HIT_COUNT"
 ,AVG(ISNULL(A.CUBEKBSIZE,0)) AS "CUBE_SIZE"
 FROM <STATS DATABASE>.DBO.IS_CUBE_REP_STATS A
 WHERE A.CUBEINSTANCEID IS NOT NULL
 AND DATEDIFF(DAY,A.DAY_ID,SYSDATETIME())<=1
 GROUP BY A.PROJECTID, A.CUBEREPORTGUID ) B
WHERE (<METADATA DATABASE>.DBO.MSTRUID(MD.OBJECT_ID))=B.CUBEREPORTGUID
ORDER BY CUBE_SIZE_KB DESC

/*THE SQL IS DESIGNED FOR AN INTERVAL OF A DAY, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

	CUBE_ID	CUBE_NAME	PROJECT_ID	CUBE_SIZE_KB	HIT_COUNT
1	80C0F38E-9926-4A43-AA24-A40183A58715	KVAT - Test Intelligent Cube - Category Sub Cat ...	E3549528-D975-4920-A5FD-F6A5EC8BFDEE	34.50	2
2	A12CA86B-7545-4BB6-9058-AC4D997591E0	Intelligent_cube_Hist	A8F35AD9-337F-4B97-9171-749664714BB3	34.00	4
3	33390924-FA67-4291-B594-DE6922E7ABC1	Intelligent_cube_current	A8F35AD9-337F-4B97-9171-749664714BB3	34.00	4
4	AEFA1BEF-7AD4-47A5-A7E1-A307E6B90687	1234	E3549528-D975-4920-A5FD-F6A5EC8BFDEE	22.50	2
5	67D5EE61-A77F-403C-921E-9C139D473427	Test1	E3549528-D975-4920-A5FD-F6A5EC8BFDEE	12.50	2
6	1983B6CE-E2A0-44D9-B2D5-A21F5A7721A3	Test	E3549528-D975-4920-A5FD-F6A5EC8BFDEE	12.50	2
7	EC344D6B-4A00-4FC6-8C93-2C545D495AC3	Cube converted from user	E3549528-D975-4920-A5FD-F6A5EC8BFDEE	0.00	1

13. Report jobs that failed in the last 2 hours

Provides alerts about the reports that have failed recently.

The queries must be run against both Statistics & Metadata Database

```
SELECT A14.OBJECT_UNAME PROJECT_NAME,
 A13.OBJECT_UNAME REPORT_NAME,
 A12.OBJECT_UNAME  USER_NAME,
 A11.SERVERMACHINE  SERVERMACHINE,
 A11.ERRORMESSAGE  ERRORMESSAGE,
 MAX(A11.RECORDTIME)  RECORDTIME,
 MAX(A11.REQUESTRECTIME)  REQUEST_RECORDTIME,
 SUM(A11.FINALRESULTSIZE)  FINALRESULTSIZE
FROM <STATS DATABASE>.DBO.IS_REPORT_STATS A11,
<METADATA DATABASE>.DBO.DSSMDOBJINFO A12,
<METADATA DATABASE>.DBO.DSSMDOBJINFO A13,
<METADATA DATABASE>.DBO.DSSMDOBJINFO A14
WHERE (<METADATA DATABASE>.DBO.MSTRUID(A12.OBJECT_ID))= A11.USERID
AND
(<METADATA DATABASE>.DBO.MSTRUID(A13.OBJECT_ID))= A11.REPORTID
 AND
(<METADATA DATABASE>.DBO.MSTRUID(A14.OBJECT_ID))= A11.PROJECTID
 AND (A11.JOBERRORCODE <> 0
 AND A11.ERRORMESSAGE NOT LIKE '%CANCELED%'
 AND A11.ERRORMESSAGE NOT LIKE '%ROLLED BACK BY CLIENT%'
AND DATEDIFF(HH,A11.RECORDTIME,SYSDATETIME())<2 )
GROUP BY A14.OBJECT_UNAME  ,
 A13.OBJECT_UNAME  ,
 A12.OBJECT_UNAME  ,
 A11.SERVERMACHINE ,
    A11.ERRORMESSAGE
/*THE SQL IS DESIGNED FOR AN INTERVAL OF A TWO HOURS, USERS CAN MODIFY THE SAME
ACCORDING TO THEIR REQUIREMENT*/
```

	PROJECT_NAME	REPORT_NAME	USER_NAME	SERVERMACHINE	ERRORMESSAGE	RECORDTIME	REQUEST_RECORDTIME	FINALRESULTSIZE
1	IMANAGE_INFADS	DETAILED REPORT 7 DAYS	BHUSHAN	INFO-SAGGROUP:34952	(SQL Generation CompleteQueryEngine encountered ...	2013-02-01 02:39:54.180	2013-02-01 16:09:41.000	0

14. Count of Configuration Object by Type

Provides the number of configuration objects in the environment that includes number of database connections, number of users and number of schedules.

The query must be run against Metadata Database.

```
SELECT(
SELECT COUNT (*)
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE='34'
AND SUBTYPE='8704') "NO_OF_USERS",
(SELECT COUNT (*)
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE='29'
AND SUBTYPE='7424') "NO_OF_DB_INSTANCES" ,
(
SELECT COUNT (*)
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE='51'
AND SUBTYPE='13056') "NO_OF_SCHEDULES"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE='33'
AND SUBTYPE='8448'
```

	NO_OF_USERS	NO_OF_DB_INSTANCES	NO_OF_SCHEDULES
1	14	22	10

15. Number of Objects by Project

Provides the count of objects created for each project.

The query must be run against Metadata Database.

```
SELECT  C.OBJECT_NAME AS PROJECT_NAME
,PB.PUBLIC_OBJECT_COUNT
,SC.SCHEMA_OBJECT_COUNT
FROM
(SELECT PROJECT_ID
,COUNT(OBJECT_ID) AS "SCHEMA_OBJECT_COUNT"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE IN (14,12,13,11,15,43)
AND SUBTYPE IN (3585,3072,3328,2816,3840,11009)
GROUP BY PROJECT_ID) SC,
(SELECT PROJECT_ID
,COUNT(OBJECT_ID) AS "PUBLIC_OBJECT_COUNT"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE IN (3,55,6,47,1,56,1,4,10,39,2)
AND SUBTYPE IN
(14081,1536,12032,257,14336,256,1024,2556,2544,9984,512,768,769,770,774,777,776)
GROUP BY PROJECT_ID) PB
,<METADATA DATABASE>.DBO.DSSMDOBJINFO C
WHERE SC.PROJECT_ID=PB.PROJECT_ID
AND C.OBJECT_ID=SC.PROJECT_ID
ORDER BY 1
```

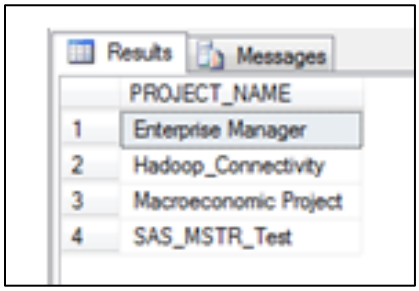
	PROJECT_NAME	PUBLIC_OBJECT_COUNT	SCHEMA_OBJECT_COUNT
1	Big Data - Cloudera	111	484
2	BIG Data -- Infa DS	102	352
3	Enterprise Manager	1141	709
4	Hadoop_Connectivity	90	306
5	iManage POC	110	362
6	iManage_InfaDS	410	532
7	Macroeconomic Project	430	363
8	MicroStrategy Tutorial	2034	799
9	SAS_MSTR_Test	97	322

16. Unused Projects

Provides a list of projects that have not been used.

The queries must be run against both Statistics & Metadata Database

```
SELECT DISTINCT OBJECT_NAME AS "PROJECT_NAME"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE=32
AND (<METADATA DATABASE>.DBO.MSTRUID(OBJECT_ID)) NOT IN
(SELECT DISTINCT PROJECTID
FROM <STATS DATABASE>.DBO.IS_PROJ_SESS_STATS
GROUP BY PROJECTID
HAVING DATEDIFF(DAY,MAX(CONNECTTIME),SYSDATETIME())<183)

/*THE SQL IS DESIGNED FOR AN INTERVAL OF 182 DAYS i.e. 6 MONTHS, USERS CAN
MODIFY THE SAME ACCORDING TO THEIR REQUIREMENT*/
```



17. List of Unused Reports by Project

Helps Administrators to archive / clean up the reports that are not being used in a project.

The queries must be run against both Statistics & Metadata Database

```
SELECT B.OBJECT_NAME AS "PROJECT_NAME"
,A.OBJECT_ID AS "REPORTID"
,A.OBJECT_NAME AS "UNUSED_REPORT"
,A.DESCRIPTION
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO A
,<METADATA DATABASE>.DBO.DSSMDOBJINFO B
WHERE A.SUBTYPE IN (768,769,770,774,777)
AND (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID)) NOT IN
```

```
(SELECT DISTINCT REPORTID FROM <STATS DATABASE>.DBO.IS_REPORT_STATS
WHERE DATEDIFF(DAY,DAY_ID,SYSDATETIME())<183)
AND A.PROJECT_ID=B.OBJECT_ID
GROUP BY B.OBJECT_NAME, A.OBJECT_ID,A.OBJECT_NAME,A.DESCRIPTION

/*THE SQL IS DESIGNED FOR AN INTERVAL OF 182 DAYS i.e. 6 MONTHS, USERS CAN
MODIFY THE SAME ACCORDING TO THEIR REQUIREMENT*/
```

	PROJECT_NAME	REPORTID	UNUSED_REPORT	DESCRIPTION
1	Big Data - Cloudera	2916A19A-4D55-4CDB-AA1A-0B421A49CB96	test 1	
2	Big Data - Cloudera	69C647C6-46B2-4DFC-857E-0FFED85995FC	test 1	
3	Big Data - Cloudera	429DFDDD-405E-A7BC-F995-8691754F63B9	Report Builder	This is a fully prompted report that can be used as ...
4	Big Data - Cloudera	EFDD8F72-4AD6-4614-C337-95B3755E0EB5	Report Wizard	The report wizard allows you to build new reports b...
5	Big Data - Cloudera	904DFFFA-D428-4205-81F9-A057802DA9AC	Blank Query Builder Report	
6	Big Data - Cloudera	BE26F4C6-2F39-4892-84CB-FDEC7153CD47	test report	
7	BIG Data -- Infa DS	3B674070-BF32-4B35-93BA-7A0776AB1B5F	test report	
8	BIG Data -- Infa DS	429DFDDD-405E-A7BC-F995-8691754F63B9	Report Builder	This is a fully prompted report that can be used as ...
9	BIG Data -- Infa DS	EFDD8F72-4AD6-4614-C337-95B3755E0EB5	Report Wizard	The report wizard allows you to build new reports b...
10	BIG Data -- Infa DS	904DFFFA-D428-4205-81F9-A057802DA9AC	Blank Query Builder Report	
11	BIG Data -- Infa DS	D43115F7-15CC-4098-873F-CABD231D3D12	US state Rev-Qty Daily	
12	Enterprise Manager	8D5E8050-EB32-11D5-A6A2-0010A4E3AEB2	Execution Cycle Breakdown w/o Queue Time_depreca...	Provides the time breakdown over time of the four s...
13	Enterprise Manager	5CBA094A-2F5A-11D5-90E5-00C04F5FAA4F	11. Daily Session Concurrency Analysis	Provides an analysis with various metrics on sessio...
14	Enterprise Manager	5CBA0968-2F5A-11D5-90E5-00C04F5FAA4F	12. Session Duration Analysis	Provides an analysis with various metrics on user s...
15	Enterprise Manager	5CBA0986-2F5A-11D5-90E5-00C04F5FAA4F	13. Daily User Connection Concurrency Analysis	Provides an analysis with various metrics on user c...
16	Enterprise Manager	5CBA09AB-2F5A-11D5-90E5-00C04F5FAA4F	10. Concurrency by Hour of Day	Provides number of active users and number of acti...
17	Enterprise Manager	5C8F2C19-2EAA-11D5-90E5-00C04F5FAA4F	50.1 Attribute Form Properties_deprecated	Lists the properties of all attribute forms in all monito...
18	Enterprise Manager	5C8F2B33-2EAA-11D5-90E5-00C04F5FAA4F	72. Summary of Schema Objects by Project	New in V8.0: Provides counts of each type of Sche...

18. List of Inactive Users

Provides a list of users who haven't logged into MicroStrategy since last six months.

The queries must be run against both Statistics & Metadata Database

```
SELECT A.LOGIN
FROM <METADATA DATABASE>.DBO.DSSMDUSRACCT A
WHERE (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID))NOT IN
(SELECT B.USERID FROM <STATS DATABASE>.DBO.IS_PROJ_SESS_STATS B
WHERE DATEDIFF(DAY,B.DAY_ID,SYSDATETIME())<183)
  AND A.ISGROUP=0

/*THE SQL IS DESIGNED FOR AN INTERVAL OF 182 DAYS i.e. 6 MONTHS, USERS CAN
MODIFY THE SAME ACCORDING TO THEIR REQUIREMENT*/
```

	LOGIN
1	SSHARMA
2	AMAGRAWAL

19. List of Unused Cubes

Provides a list of cubes which exist in the environment but are not hit by any reports.

The queries must be run against both Statistics & Metadata Database

```
SELECT A.OBJECT_ID As "UNUSED_CUBEID"
,A.OBJECT_NAME AS "UNUSED_CUBE_NAME"
,A.DESCRIPTION
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO A
WHERE A.SUBTYPE=776
AND (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID)) NOT IN(
SELECT DISTINCT (B.CUBEREPORTGUID)
FROM <STATS DATABASE>.DBO.IS_CUBE_REP_STATS B
WHERE DATEDIFF(DAY,B.DAY_ID,SYSDATETIME())<183)
GROUP BY A.OBJECT_ID, A.OBJECT_NAME, A.DESCRIPTION
ORDER BY 2

/*THE SQL IS DESIGNED FOR AN INTERVAL OF 182 DAYS i.e. 6 MONTHS, USERS CAN
MODIFY THE SAME ACCORDING TO THEIR REQUIREMENT*/
```

	UNUSED_CUBEID	UNUSED_CUBE_NAME	DESCRIPTION
1	CC02C5C2-803A-4AE2-A5ED-14BFD4598103	Actual vs Forecast Performance Cube	
2	42FF415D-2846-4E16-AD4F-7DC84EBF268B	Cube Memphis Discounts	This in-memory intelligence cube displays discou...
3	20BF307F-FC3F-4DAF-8F5D-3D9D28859065	cube_1	
4	4B057254-9B11-4338-AE31-FD9141DDEABE	cube_2	
5	5B7CEAB5-0C3F-47AD-9743-C169ABFDA709	cube_temp	
6	B86C42F4-76AC-4C97-9A4A-612EB7C223E8	cubee	
7	3CA239F1-46E4-4CED-AD0C-0E1CF065C3E9	D_Activity Cube	
8	0EEBCE3E-4954-4400-B9D4-314C324A443B	DATA_IMPORT_CSV_FILE	This MicroStrategy Intelligent Cube is imported fr...
9	628C1D58-FD6D-4794-A079-3F35A06BC44D	DATA_IMPORT_EXCEL_FILE	This MicroStrategy Intelligent Cube is imported fr...
10	BF11980E-4828-40F1-B8F4-EDF663478582	DATA_IMPORT_SQL_STATEMENT	This MicroStrategy Intelligent Cube is imported u...
11	00843D10-426B-4759-9CD8-99D43EC3609C	IC test	
12	ED99260A-6367-42CF-9E0D-9FDB7FA598ED	import data for Q2 2012	
13	1501C7F4-E6B6-4AEF-B1BD-E16F2DF14E95	Intelligent Cube - All project languages	This intelligent cube has been enabled to suppor...
14	8CCD8D9D-A4C5-4051-A619-C733D8EE0D59	Intelligent Cube - Drilling outside the cube is disa...	This Intelligent Cube is set so drilling outside the ...
15	5731DB3F-9679-4496-B78A-E871A6632D60	Intelligent Cube - Dynamic Sourcing	This Intelligent Cube has been enabled to suppo...
16	6137E096-D84F-4C68-AA16-781009224C69	Intelligent Cube - Time, Products, Geography - S...	This Intelligent Cube includes Time (Year to Mon...
17	26BB1684-5FF2-4B4F-B6D5-00F754848372	my test cube	
18	6C204A56-DEB3-4286-98CB-CAE2C0722276	Profit and Revenue Cube	

20. List of Objects in "My Reports" Folder

Helps in determining the custom reports created by a user. This information is useful during migration and upgrades, and can be used to determine the global utilization of these reports.

The query must be run against Metadata Database.

```
SELECT C.OBJECT_NAME AS "PARENT_USER"
,D.OBJECT_NAME AS "PROJECT_NAME"
,A.OBJECT_NAME
,B.OBJECT_NAME AS "FOLDER_NAME"
```

```
FROM
(SELECT * FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE PARENT_ID IN
(SELECT OBJECT_ID
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO
WHERE OBJECT_TYPE=8
AND SUBTYPE=2048
AND OBJECT_NAME LIKE 'MY REPORTS')) A,
<METADATA DATABASE>.DBO.DSSMDOBJINFO B
,<METADATA DATABASE>.DBO.DSSMDOBJINFO C
,<METADATA DATABASE>.DBO.DSSMDOBJINFO D
WHERE A.PARENT_ID=B.OBJECT_ID
AND B.PARENT_ID=C.OBJECT_ID
AND C.PROJECT_ID=D.OBJECT_ID
ORDER BY 1
```

	PARENT_USER	PROJECT_NAME	OBJECT_NAME	FOLDER_NAME
1	Administrator	MicroStrategy Tutorial	Profit	My Reports
2	Administrator (Administrator)	Big Data - Cloudera	test1	My Reports
3	Administrator (Administrator)	Big Data - Cloudera	test	My Reports
4	Administrator (Administrator)	Big Data - Cloudera	test cube import	My Reports
5	Administrator (Administrator)	BIG Data – Infa DS	my test cube	My Reports
6	Administrator (Administrator)	BIG Data – Infa DS	D_Activity Cube	My Reports
7	Administrator (Administrator)	BIG Data – Infa DS	V_sample cube	My Reports
8	Administrator (Administrator)	Macroeconomic Project	Threshold Report	My Reports
9	Gaurav Kolarkar (gaurav.kolarkar)	Enterprise Manager	Gaurav Kolarkar-Test	My Reports
10	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	License Distribution by Users for Department	My Reports
11	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	License Distribution by Department	My Reports
12	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	My_stats	My Reports
13	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	avg_cpu	My Reports
14	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	License Summary	My Reports
15	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	Intelligent_cube_Hist	My Reports
16	Gaurav Kolarkar (gaurav.kolarkar)	iManage_InfaDS	Intelligent_cube_current	My Reports
17	Gaurav Kolarkar (gaurav.kolarkar)	MicroStrategy Tutorial	New Report	My Reports
18	Gaurav Kolarkar (gaurav.kolarkar)	MicroStrategy Tutorial	test	My Reports

21. Number of Owned and Allocated Licenses by Product

Provides information on licenses owned and allocated by product. It helps the Administrators to determine the utilization of MicroStrategy licenses.

We can access the License Manager by Batch script. In this batch Script we call the console utility of Microstrategy License Manger. Paste the below code in ".bat" file.

```
@echo off
cd C:\Program Files (x86)\Common Files\MicroStrategy
malicmgr -audit -n <<source>> -u <<username>>  -p <<password>> -o
License_Output_File.html
```

Product	Total Named Users	Enabled Named Users	Disabled Named Users
Intelligence Server	94	15	79
Web Reporter	65	15	50
Web Analyst	69	15	54
Web Professional	59	14	45
Web MMT (eTrainer)	26	10	16
Office	47	9	38
Mobile Server	52	13	39
MultiSource	42	10	32
Distribution Services	69	15	54
Transaction Services	35	13	22
Command Manager	21	6	15
Object Manager	26	10	16
Desktop Analyst	55	14	41
Desktop Designer	54	14	40
Architect	35	14	21
Integrity Manager	26	10	16
OLAP Services	71	15	56
Report Services	71	15	56
Users without license association	21	0	21

22. Duration for which a user hasn't logged in MicroStrategy

Helps Administrators to identify inactive Users. If a User hasn't logged in for a defined duration, the Administrator can investigate and revoke the license of the User, if necessary.

The queries must be run against both Statistics & Metadata Database

```
SELECT A.OBJECT_ID "USER_ID"
,A.OBJECT_NAME "USERNAME"
,A.CREATE_TIME "CREATION_TIME"
,A.MOD_TIME "MODIFIED_TIME"
,B.LAST_LOGIN_TIME
,DATEDIFF(DAY,b.Last_Login_Time,SYSDATETIME()) AS "DAYS_SINCE_LOGIN"
FROM <METADATA DATABASE>.DBO.DSSMDOBJINFO A
JOIN
(SELECT DISTINCT USERID, MAX (CONNECTTIME)"LAST_LOGIN_TIME"
FROM <STATS DATABASE>.DBO.IS_SESSION_STATS
GROUP BY USERID) B
ON (<METADATA DATABASE>.DBO.MSTRUID(A.OBJECT_ID)) = B.USERID
```

	USER_ID	USERNAME	CREATION_TIME	MODIFIED_TIME	LAST_LOGIN_TIME	DAYS_SINCE_LOGIN
1	9CA3C2C2-94E2-4FA9-B885-001482DF29D4	Gaurav Kolarkar	2012-09-17 05:21:42.273	2012-12-26 11:31:45.957	2013-01-30 19:27:47.000	2
2	54F3D260-8965-11D2-8E9A-006008960167	Administrator	2011-09-26 05:57:29.410	2012-09-17 16:22:23.777	2013-01-31 18:30:01.000	1
3	4C98D120-39B4-4CD1-A053-2DD95CC97648	Rishabh	2012-12-24 09:47:01.417	2012-12-24 09:47:32.947	2013-02-01 16:28:17.000	0
4	2603D75F-B853-444E-A08E-3692C95CBCC7	Beta Eval	2012-09-07 04:17:10.650	2013-01-16 10:33:14.057	2013-01-16 16:04:50.000	16
5	19A64422-BF13-4EE7-A733-3E22C45C75D8	Vizeh	2012-12-07 05:27:38.030	2012-12-07 05:27:58.907	2013-02-01 12:19:11.000	0
6	DCBFC7A0-E033-4260-85A0-4B107AAAD6AA	Shivam	2012-11-09 06:56:10.803	2012-11-09 06:56:23.337	2012-11-09 16:37:29.000	84
7	77939EF7-FD03-4C26-8EA5-592640C0D3B5	Bhushan	2012-11-26 12:23:21.037	2012-12-14 09:37:03.937	2013-02-01 17:23:42.000	0
8	64EAF25C-C44B-4252-B007-726AF1E40669	New User (1)	2012-11-06 09:15:47.680	2012-11-06 09:16:00.413	2012-11-06 14:51:01.000	87
9	3FD486C5-63DF-49B9-847B-7766ED6EF9B6	Swati	2012-12-24 09:46:16.807	2012-12-24 09:46:48.447	2013-02-01 16:11:43.000	0
10	F1BE1BEE-D027-47C0-A07E-99B37DD9F01F	Copy of Manish	2012-12-06 09:53:22.267	2012-12-14 09:37:18.937	2013-01-14 17:01:23.000	18
11	F173E761-83BE-49D7-AB2E-9EFFCEA92BCB	Rohit Singh	2012-09-17 05:21:21.413	2012-10-04 07:12:42.153	2013-02-01 11:58:57.000	0
12	22F94970-0D5C-4A2F-919C-E45499BF2B0D	Nirav Prasad	2012-09-27 18:58:18.233	2012-10-04 06:33:59.073	2012-12-16 13:18:51.000	47
13	F1C3AD24-8493-4EFC-AA41-F51EF2B9ED1B	Ram Reddy	2012-11-26 12:25:14.867	2013-01-30 09:28:18.913	2013-02-01 16:29:54.000	0

23. List of objects changed by users

Helps to identify the reports that were changed by MicroStrategy Administrators or the Users directly in the Production environment.

The query must be run against Metadata Database.

```
SELECT DISTINCT B.OBJECT_NAME
,A.USER_ID
,C.OBJECT_NAME CHANGED_BY
,D.CREATE_TIME
,D.MOD_TIME
FROM <METADATA DATABASE>.DBO.DSSMDJRNINFO A
JOIN <METADATA DATABASE>.DBO.DSSMDJRNOBJD B
ON A.TRANSACTION_ID=B.TRANSACTION_ID
JOIN <METADATA DATABASE>.DBO.DSSMDOBJINFO C
ON A.USER_ID = C.OBJECT_ID
JOIN <METADATA DATABASE>.DBO.DSSMDOBJINFO D
ON B.OBJECT_NAME=D.OBJECT_NAME
AND DATEDIFF (DAY, D.MOD_TIME, SYSDATETIME ()) <=7
/*THE SQL IS DESIGNED FOR AN INTERVAL OF A WEEK (7 DAYS), USERS CAN MODIFY THE
SAME ACCORDING TO THEIR REQUIREMENT*/
```

	OBJECT_NAME	USER_ID	CHANGED_BY	CREATE_TIME	MOD_TIME
1	1	22F94970-0D5C-4A2F-919C-E45499BF2B0D	Nirav Prasad	2013-01-31 13:03:57.463	2013-01-31 13:03:57.463
2	1	F1C3AD24-8493-4EFC-AA41-F51EF2B9ED1B	Ram Reddy	2013-01-31 13:03:57.463	2013-01-31 13:03:57.463
3	Archive	77939EF7-FD03-4C26-8EA5-592640C0D3B5	Bhushan	2013-01-31 11:20:44.803	2013-01-31 11:21:02.040
4	Archive	3FD486C5-63DF-49B9-847B-7766ED6EF9B6	Swati	2013-01-31 11:20:44.803	2013-01-31 11:21:02.040
5	Check	77939EF7-FD03-4C26-8EA5-592640C0D3B5	Bhushan	2013-01-30 07:13:36.467	2013-01-30 07:13:36.467
6	Columns	54F3D260-8965-11D2-8E9A-006008960167	Administrator	2012-10-09 09:38:27.987	2013-02-01 06:53:25.003
7	Cpu Consumption	22F94970-0D5C-4A2F-919C-E45499BF2B0D	Nirav Prasad	2013-01-31 09:27:27.577	2013-01-31 09:27:27.577
8	CPU Consumption	F1C3AD24-8493-4EFC-AA41-F51EF2B9ED1B	Ram Reddy	2013-01-31 09:27:27.577	2013-01-31 09:27:27.577
9	Current	54F3D260-8965-11D2-8E9A-006008960167	Administrator	2012-10-22 10:10:58.120	2013-01-31 07:34:30.130
10	Current	54F3D260-8965-11D2-8E9A-006008960167	Administrator	2012-10-22 10:14:14.597	2013-01-31 09:27:28.203
11	Current	F173E761-83BE-49D7-AB2E-9EFFCEA92BCB	Rohit Singh	2012-10-22 10:10:58.120	2013-01-31 07:34:30.130
12	Current	F173E761-83BE-49D7-AB2E-9EFFCEA92BCB	Rohit Singh	2012-10-22 10:14:14.597	2013-01-31 09:27:28.203
13	Current	F1C3AD24-8493-4EFC-AA41-F51EF2B9ED1B	Ram Reddy	2012-10-22 10:10:58.120	2013-01-31 07:34:30.130
14	Current	F1C3AD24-8493-4EFC-AA41-F51EF2B9ED1B	Ram Reddy	2012-10-22 10:14:14.597	2013-01-31 09:27:28.203
15	Dashboard	54F3D260-8965-11D2-8E9A-006008960167	Administrator	2012-12-24 10:32:57.140	2013-01-30 09:17:00.543

24. RAM and CPU Utilization for Intelligence Server and Webserver

Provides current utilization and helps Administrators in server capacity planning.

For Calculating the CPU & RAM performance we have used the performance monitor utility of windows. In a configuration file we give parameters on which performance logging is to be done. Then we create a Perf Mon Counter on those parameters.

Intelligence Server Configuration File. Paste the code in any file any save it with (.conf) extension.

```
"\Process (MSTRSvr2_64)\ID Process"
"\Process (MSTRSvr2_64)\% Processor Time"
"\Process (MSTRSvr2_64)\Private Bytes"
"\Process (MSTRSvr2_64)\Virtual Bytes"
```

Intelligence Server Counter. Paste code in a batch file with appropriate values.

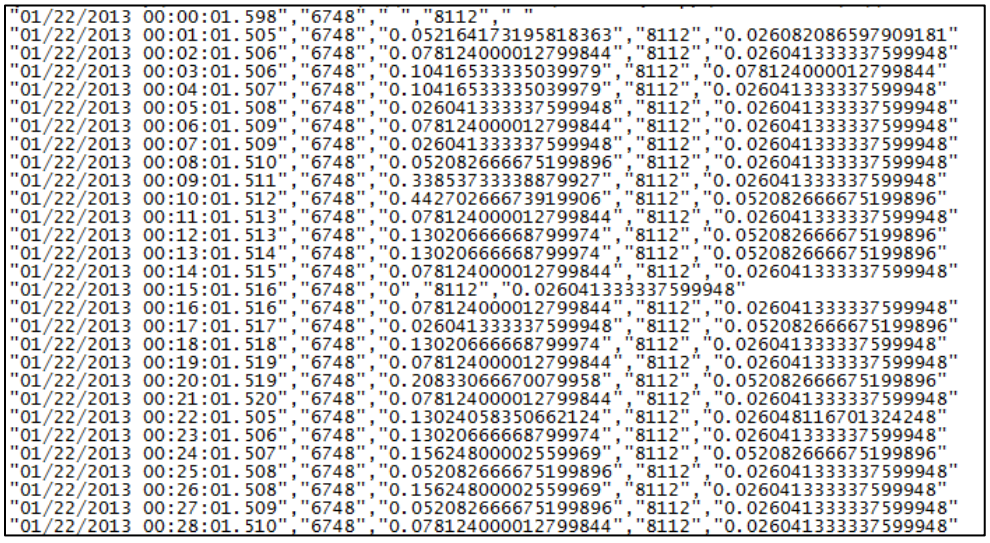
```
logman create counter MSTRiServStats -f <out file format> -si <<time interval>>
--v -o "<OutFile Name with Location>" -cf "<<Intelligence Server Configuration
File>>"
logman.exe START MSTRiServStats
```

Webserver Configuration File. Paste the code in any file any save it with (.conf) extension.

```
"\Process(w3wp)\ID Process"
"\Process (w3wp)\% Processor Time"
"\Process (w3wp)\Private Bytes"
"\Process (w3wp)\Virtual Bytes"
```

Webserver Counter. Paste code in a batch file with appropriate values.

```
logman create counter MSTRWebServStats -f <out file format> -si <<time
interval>> --v -o "<OutFile Name with Location>" -cf "<<WebServer Configuration
File>>"
logman.exe START MSTRWebServStats
```



```
"01/22/2013 00:00:01.598","6748"," ","8112"," "
"01/22/2013 00:01:01.505","6748","0.052164173195818363","8112","0.026082086597909181"
"01/22/2013 00:02:01.506","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:03:01.506","6748","0.10416533335039979","8112","0.078124000012799844"
"01/22/2013 00:04:01.507","6748","0.10416533335039979","8112","0.026041333337599948"
"01/22/2013 00:05:01.508","6748","0.026041333337599948","8112","0.026041333337599948"
"01/22/2013 00:06:01.509","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:07:01.509","6748","0.026041333337599948","8112","0.026041333337599948"
"01/22/2013 00:08:01.510","6748","0.052082666675199896","8112","0.026041333337599948"
"01/22/2013 00:09:01.511","6748","0.33853733338879927","8112","0.026041333337599948"
"01/22/2013 00:10:01.512","6748","0.44270266673919906","8112","0.052082666675199896"
"01/22/2013 00:11:01.513","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:12:01.513","6748","0.13020666668799974","8112","0.052082666675199896"
"01/22/2013 00:13:01.514","6748","0.13020666668799974","8112","0.052082666675199896"
"01/22/2013 00:14:01.515","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:15:01.516","6748","0","8112","0.026041333337599948"
"01/22/2013 00:16:01.516","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:17:01.517","6748","0.026041333337599948","8112","0.052082666675199896"
"01/22/2013 00:18:01.518","6748","0.13020666668799974","8112","0.026041333337599948"
"01/22/2013 00:19:01.519","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:20:01.519","6748","0.20833066670079958","8112","0.052082666675199896"
"01/22/2013 00:21:01.520","6748","0.078124000012799844","8112","0.026041333337599948"
"01/22/2013 00:22:01.505","6748","0.13024058350662124","8112","0.026048116701324248"
"01/22/2013 00:23:01.506","6748","0.13020666668799974","8112","0.026041333337599948"
"01/22/2013 00:24:01.507","6748","0.15624800002559969","8112","0.052082666675199896"
"01/22/2013 00:25:01.508","6748","0.052082666675199896","8112","0.026041333337599948"
"01/22/2013 00:26:01.508","6748","0.15624800002559969","8112","0.026041333337599948"
"01/22/2013 00:27:01.509","6748","0.052082666675199896","8112","0.026041333337599948"
"01/22/2013 00:28:01.510","6748","0.078124000012799844","8112","0.026041333337599948"
```

25. Availability of Intelligence Server and Web Server

Provides Administrators with the Uptime of the Intelligence Server and Web Server.

Intelligence Server Availability is checked by querying the Intelligence Server machine host using IP and Port Number. This IP and Port Number is saved in a textfile separated by space. Now by Installing “Portquery” client we can query any machine for activity on a particular port number. You can Download Portquery easily and install it. Copy and paste the code below in a “.bat” file with appropriate values.

```
@echo off
set serverList= <File with Server IP and Port Number>
set query_output=< Text Output File>
set output_file= Server_Availability_Output_File.txt
FOR /f "tokens=1, 2 delims= " %%a IN ('type "%serverList%"') DO (
setlocal enabledelayedexpansion
set server=%%a
set port=%%b
portqry.exe -n !server! -e !port! -y -l "%query_output%"
 FINDSTR /C:"NOT LISTENING" "%query_output%"
 IF NOT ERRORLEVEL 1 (
  ECHO! server!; MSTR_Intelligence Server; Unavailable; %date% %time% >>
"%output_file%"
 ) ELSE (
  ECHO! server!; MSTR_Intelligence Server; Available; %date% %time% >>
"%output_file%"
 )
 endlocal
)
```

Paste the code below in “.vbs” Vb Script file with appropriate values.

```
Dim strWebsite
strWebsite = "<MicroStrategy Home Page>"
If PingSite( strWebsite ) Then
    WScript.Echo "MSTR_Webserver; Available;" & Date &" "& Time
Else
     WScript.Echo "MSTR_Webserver; Unavailable;" & Date &" "& Time
End If
Function PingSite ( myWebsite )
```

```
    Dim intStatus, objHTTP
    Set objHTTP = CreateObject ("WinHttp.WinHttpRequest.5.1")
    objHTTP.Open "GET", "http://" & myWebsite & "/", False
    objHTTP.SetRequestHeader "User-Agent", "Mozilla/4.0 (compatible; MyApp 1.0;
Windows NT 5.1)"
    On Error Resume Next
    objHTTP.Send
    intStatus = objHTTP.Status
    On Error Goto 0
    If intStatus = 200 Then
        PingSite = True
    Else
        PingSite = False
    End If
    Set objHTTP = Nothing
End Function
```

```
10.10.10.115;Availble;21-11-2012 19:00:11.44
10.10.10.115;Availble;21-11-2012 19:01:11.34
10.10.10.115;Availble;21-11-2012 19:02:15.23
10.10.10.115;Availble;21-11-2012 19:03:15.21
10.10.10.115;Availble;21-11-2012 19:04:11.35
10.10.10.115;Availble;21-11-2012 19:05:11.38
10.10.10.115;Availble;21-11-2012 19:06:11.38
10.10.10.115;Availble;21-11-2012 19:07:11.35
10.10.10.115;Availble;21-11-2012 19:08:11.34
10.10.10.115;Availble;21-11-2012 19:09:11.36
10.10.10.115;Availble;21-11-2012 19:10:11.37
10.10.10.115;Availble;21-11-2012 19:11:11.35
10.10.10.115;Availble;21-11-2012 19:12:11.36
10.10.10.115;Availble;21-11-2012 19:13:11.35
10.10.10.115;Availble;21-11-2012 19:14:11.36
10.10.10.115;Availble;21-11-2012 19:15:11.36
10.10.10.115;Availble;21-11-2012 19:16:11.35
10.10.10.115;Availble;21-11-2012 19:17:11.35
10.10.10.115;Availble;21-11-2012 19:18:11.36
10.10.10.115;Availble;21-11-2012 19:19:11.36
10.10.10.115;Availble;21-11-2012 19:20:11.35
10.10.10.115;Availble;21-11-2012 19:21:11.35
10.10.10.115;Availble;21-11-2012 19:22:11.33
10.10.10.115;Availble;21-11-2012 19:23:11.36
10.10.10.115;Availble;21-11-2012 19:24:11.36
10.10.10.115;Availble;21-11-2012 19:25:11.35
10.10.10.115;Availble;21-11-2012 19:26:11.35
10.10.10.115;Availble;21-11-2012 19:27:11.35
```