Performance Benchmark and Capacity Planning

Intellicus Enterprise Reporting and BI Platform

intellicus

Enterprise Reporting and
Business Insights Platform

Dated: March 2014

Acknowledgements

Contents

Introduction

As a business enterprise, you will buy software after making sure that it has features meeting all of your enterprise reporting needs.

You also expect the software to perform as per pre-defined parameters of normal and extreme usage conditions.

This document provides information related to:

- Factors that affect Intellicus performance

- Intellicus capacity limits

- Benchmark results

- Tips on capacity planning and

- Performance tuning parameters.

UI Capacity limits, gives an idea of how Intellicus portal reacts when a number of users hit the same application page.

Benchmark results and tips on capacity planning will assist you plan the hardware configuration for deployment of Intellicus.

Make use of performance tuning parameters information to set properties of Intellicus Server and get the performance levels that meet your needs.

Factors affecting the performance

Performance of Intellicus depends on multiple factors. Some of which are Intellicus internal factors, whereas others are external ones. You can control effect of internal factors by setting properties of Intellicus Report Server as well as Portal.

Volume of data needed for the report

A report may need to process millions of rows of data before it can be viewed or delivered. Large volume of data means longer time to:

- Fetch and transfer data from database server to report server.

- Process the data to construct report pages.

- Transfer of report pages from report server to browser.

Size of a row, number of rows

Report data is fetched from the database using a SQL. Depending on the nature of report, its record-set may be of large size. This may happen due to factors like number of fields, field size and type (for example, image field). Row size may contribute significantly in performance when a report has large size of record-set. However, this factor will be insignificant for a report having low volume of data.

Number of concurrent requests

Report Server allots one thread to each request. If number of requests is more than that of available threads, such requests are queued until a thread is released. More number of concurrent requests means more waiting time for queued requests. How sooner a queued request is allotted a thread will depend on request type and data volume that running threads needs to process.

Database

Intellicus depends on database for report data as well as meta-data. For almost every user request, it has to query the repository for meta-data and query the business database for report data. Business database may be receiving queries from application other than Intellicus also.

Factors like server hardware, connectivity and database schema will affect database response time. And this will in turn, affect Intellicus response time.

Network

Intellicus supports distributed deployment of its server (report server) and client (portal) components. Users spread across wide geographical locations may access Intellicus functionalities over corporate intranet or the Internet. In this case network will become a factor.

Network Speed

Network resources are used to:

- Fetch data from database server to Report server

- Transfer reports from Report server to portal

- Transfer reports from portal to browser

Bandwidth consumed also depends on the volume of data that has to travel and so becomes a concern for a report having millions of rows.

Firewall

When a firewall is implemented all the content received are checked for pre-set parameters in order to filter the content. Filtering process takes some time and in turn effects all over performance of Intellicus.

Processor

Intellicus Report Server is a processor-intensive application. Depending on the nature of deployment and usage of Intellicus, it may have to serve a significant number of requests every second. The processor handles:

- Report execution requests.

- Repository update and administration tasks.

- Complex mathematical and logical operations for each of the report execution requests.

RAM

Intellicus Report Server needs RAM for activities like:

To keep data read from the result-set.

This depends on data fetch size, which decides number of records fetched from the database at a time.

For internal sorting of records

User can dynamically sort records even after the report is executed. Since data has already been fetched from database, sorting is carried out in RAM.

To keep a minimal chunk of report pages

Report pages are kept in RAM before they are sent to the portal (and in turn, users). For a report having large number of pages, a minimum number of pages are kept in RAM. Remaining report-pages are kept on disk. As and when a user requests for specific pages, they are picked up from the disk and swapped with those kept in RAM.

Portal's RAM needs

Intellicus Portal needs RAM for activities like

- Retaining session details

- Streaming report pages or report files received from report server to browser.

The amount of RAM required at a time will depend on specific Number of logged in users, tasks they are requesting to Intellicus, etc.

If an application's memory requirement increases, operating system may use swapping so that application can continue running. In worst cases, application may stop responding.

Free hard disk space

Before generating final report output, Intellicus creates intermediate files. When a report is requested in output formats like PDF or MS Word, Intellicus needs to create temporary files on report server's local disk.

Amount of required free disk space will depend on:

- Number of reports having high volume of data

- Report data cache settings

- Frequency of report generation

- Number and Validity period of published reports, etc.

Operating System

Operating systems have their respective capacities and limits of memory usage and process management, support for web server and web browser, etc.

Intellicus is available for operating systems that support 32-bit architecture as well as 64-bit architecture.

Number of user threads

Intellicus maintains separate threads for activities like report generation, scheduler, dashboards, internal activities, etc.

Each report server activity is allotted one thread. Requests received after all the threads are consumed, are queued until a thread is freed.

Delivery Option

Intellicus supports multiple report delivery options. A Report Delivery Option *Publish* is relatively faster than other formats.

Report output / delivery option that needs external resources or application may be slower.

- **E-mail:** Number of reports that can be e-mailed in a set time will depend on the capacity of the mail server.

- **Upload (FTP):** Time taken for uploading a report will depend on available bandwidth as well as availability of the FTP server.

- **View:** Viewing a report in a specific output format takes one more cycle of processing. In this cycle, the report is converted from intermediate format to the requested format, for example, PDF.

Operation / Action

Response time of a request will depend on the type of operation / action requested, for instance, a repository update, report generation, etc. Report generation actions generally take more time than repository update actions.

Database connection Pool

Each report generation request needs one connection with respective database. Having less number of open connections then required will keep query requests on hold until connections are freed and allotted to the queued requests.

Report

Execution of reports with extraordinary characteristics may take longer then usual. For example, reports having:

- Comparatively complex queries

- Very large volume of data

- Large number of Cross-tabs

- Large number of charts, or charts having many series

- Sub reports within sub reports

Capacity Limits

This section of the document provides capacity limits Information that will assist you in comparing Intellicus capacity limits with your enterprise reporting needs (existing needs and future needs).

Machine configuration

Since the objective is to find limits, we selected a lower-end machine. Intellicus in production environment will have much better results.

For example, if a report runs in one minute on a lower-end machine, it will run significantly faster on a higher-end report server.

- **Processor:** Intel(R) Core(TM) i5-3470CPU @3.20 GHz

- **Report Server RAM:** 8 GB

- **Web Server RAM:** 2 GB

- **Platform:** Windows 64 Bit

- **Web Server:** Jakarta Tomcat

Report Performance Capacity Limits

This section depicts report execution related limitations. Small sized and big sized reports were executed to receive output in multiple output formats like MS Excel, PDF, RAW TEXT, HTML and iHTML.

Maximum Concurrent Users

- Small Reports: 40 Concurrent reports   OR

- Big Reports: 10 Concurrent reports

Small sized reports mean a report having around 30 pages and row size in range of 100 – 150 characters. All the reports having larger row size or number of pages are treated as big sized reports.

If the requests exceed the above-mentioned limitations, then the response time and throughput time may drastically increase as requests may pileup in the queue.

Maximum Data that can be generated in Excel format

This test is to fetch huge number of rows, render report in multiple pages and then generate Excel by checking single-sheet option and unzipped format.

In this test, maximum available RAM for the report server was 512 MB.

- Row size: Normal (100-120 characters per row in 5 columns)

- Maximum rows: 600,000

- Throughput Time: 20 minutes

- Request concurrency: 1

Behaviour after crossing above limits

- Increase in the number of rows can cause report server to throw "out of memory" error.

- Increase in the number of concurrent requests can cause report server to throw "out of memory" error.

> **Note:** When number of rows being inserted in a sheet exceeds 65000, next rows are shifted to another sheet in the same workbook.

Maximum pages in a report that can be generated in PDF format

This test is to fetch huge number of rows and render the report in PDF.

RAM: 2 GB
Concurrency: 1 user

Single Page PDF output

- Maximum number of records: 826

Multi-page PDF output

Pages	Rows	Time
123,846	2,500,000	1 hr 40 mins
248,846	5,000,000	3 hr 26 mins
498,846	10,000,000	6 hr 33 mins

Behaviour after crossing above limits

- Increase in number of rows will result in generation of output having blank rows.

- Increase in number of rows can result in Out of Memory error.

Maximum report pages that can be generated to HTML format

This test is to fetch huge number of rows, render report in multiple pages, generate HTML and send it to the browser for viewing. In this test, maximum available RAM for the report server was increased to 1 GB. Report page size was A4.

- Report Size: 498,839 pages

- Request concurrency: 1

- Response Time: 5 seconds

- Throughput time: 6 hours 26 min minutes

- Row count: 10,000,000

When the same test was conducted with single page HTML, results were different:

- Report Size: 6,000 records

- Request concurrency: 1

- Throughput time: 1 minute

Important: Report server could render above 30000 records, but due to limitation on browser, the output was not as expected after it rendered 6000 records.

Maximum Data that can be generated to Raw Text format

This test is to fetch huge number of rows, render report in Raw Text format, compress and stream it to browser for downloading. There were 100 to 120 characters per row.

Test was repeated twice. Table displays number of rows and time taken to complete each execution.

Compressed Output (zipped)

Number of Rows	Time taken	File size (Zipped format)
10,000,000	10 minutes 20 seconds	188 MB
20,000,000	20 minutes 19 seconds	375 MB

Compressed Output (Un-zipped)

Number of Rows	Time taken	File size (Unzipped format)
10,000,000	12 minutes	1.25 GB
20,000,000	24 minutes	2.59 GB

Behaviour after crossing above limits

- Increase in the number of rows can cause report server to throw "out of memory" error.

- Increase in the number of concurrent requests can cause report server to throw "out of memory" error.

In addition to the report execution and output format related information, this section of the document also covers information on report design and preview related limits.

Report Design and Preview: General observations

Number of groupings in a report

This refers to the level of groupings (nested sections) in which the report is divided. Up to 15 levels of groupings have been executed without noticing any significant effect in execution speeds.

Each grouping level added in the report increases the complexity. This remains quite insignificant when up to 15 groupings are set for a small report. Crossing this level may result in noticeable degradation of performance.

Fields on parameter input form

This refers to a form (HTML page) that is dynamically designed and displayed in the browser when a report needs run time parameter values from the user.

For screen resolution of 1024 x 768 pixels, the page will be able to display up to 50 parameters. However, the actual limit will also depend on the type of parameters on the form.

Depending on the type of parameters and particularly the number of dropdown boxes, page may take longer to load. Based on the browser used, some of the dropdown boxes may not have all the values.

Length of value specified in parameter

This refers to the size of the parameter. A parameter can hold multiple values depending on its type. For example, a combo type parameter. Length of values specified in parameter means total number of characters required to store all the values for a parameter.

This limit value is set during parameter design.
Maximum limit allowed is **32000 characters**.

Intellicus portal UI Capacity Limits

In addition to report execution capacities, Intellicus portal UI presents some limits related to information display.

This section of the document provides information about maximum limits / practical limits of following items in Intellicus portal:

- Users

- Report design

- Report Deployment

Page loading times of Organization and Users related pages

Organization page

- Number of Organizations: 500

- Concurrent users: 100

- Time taken to load the page: 26 seconds

User / Role page

- Number of Organizations: 100

- Number of users: 1000

- Number of roles: 10

- Concurrent users: 100

- Time taken to load the page: 21 seconds

Entity Access Rights page

- Number of Categories: 100

- Number of reports in selected category: 100

- Concurrent users: 100

- Time taken to load the page: 60 seconds

Users

We created users from Intellicus portal as well as using a test-API and checked page loading time on following portal pages:

- Navigation > Administration > Manage Users > User/Role

- Report collaboration comment section while viewing a report in HTML

All the above pages responded in the similar fashion. Following table summarises the page loading times.

Number of users loaded on management page	Page Loading time
100	4 seconds
500	5 seconds
1000	5 seconds

When more than 4000 users were created through a test-API, the user list did not get populated on portal pages for over 15 minutes.

Upon checking of logs, it was observed that Intellicus Report Server had transferred the information to portal and portal had in turn transferred the information to browser. But browser could not display it.

Report Design

Report design has section limitation information about:

- Report page dimensions

- Query Objects

- Ad hoc Reports

- Parameter Objects

- Parameter Value Groups

Report page dimensions

Mostly, reports having standard page dimensions are designed. We wanted to find out the maximum page width and length that Intellicus is able to support.

We started by designing a report having A4 page-size and went on increasing the page size. For each cycle, we previewed the report in Desktop studio. With increase in page-width, we went on increasing fields on the report.

During testing, we designed a report page having width of 280 inches and placed 245 controls in width.

It was observed that Desktop Studio does not support a page-width of more than 280 inches. If you try setting a page wider than 280 inches, canvas will be extended to the value specified, but Desktop studio will not allow placement of controls on canvas. Upon previewing such a report, you will receive "parsing error".

Query Objects

Query Objects are other most used entities in Intellicus; we observed the timings being taken by different screens to list the QO when multiple users hit it simultaneously.

Here is a table that shows the time taken to load different pages. The Count of Query Objects used for the listing purpose was 300 in a category.

- Repository > Report Objects in a Category

- Navigation > Administration > Manage Users >Entity Access Rights > Query Objects

- Navigation > Design > Ad hoc Report Wizard

To observe the behaviour when information within a QO is larger than normal, we conducted tests with increase in complexity of the QO. Here are the test results:

Fields in the QO	Fields having fetch data True	Number of Mandatory filters	Page loading time
10	1	1	2 seconds
20	2	2	5 seconds
30	3	3	10 seconds
40	4	4	13 seconds
60	4	4	18 seconds

Loading of 500 Query Objects from different tabs for 3 hrs. with 100 concurrent users:

Query Object listing from Repository tab (500)	29 secs
Query Object listing from Entity Access Rights page (500)	40 secs
Query Object Listing in Object Selector of Ad hoc Wizard	7 secs

Ad hoc Reports

Intellicus portal has following limits on Ad hoc report wizard.

- Width of field (as drop-down box): 300 characters

- Number of filters: 100

- Number of groups: 20

- Number of totals (group summary fields): 100

- Number of Sort levels: 100

- Number of highlights one can set: 100

Parameter Objects

Parameter page displays a list of parameter objects and details of the selected parameter object.

We created multiple parameter objects and tried opening the pages where they are listed. These tests gave us information about the time taken to open a page.

Tests-results were derived on following pages:

- Repository Menu > Parameter Objects in a category.

- Navigation > Administration > Manage Users >Entity Access Rights > Parameter Object

- Repository > Report Objects > Parameter value groups

- Navigation > Personalization > My Preferences

Different Screens were tested for the loading time taken for the Parameter objects. The Scenario was constant listing being done by 100 Users for a time period of 3 hours. Here are the results.

Number of Parameter Objects: 250.

Parameter Object from Entity **Access rights**	10 secs
Parameter Object listing from **Repository Menu**	3 secs
Parameter Object Listing from **Parameter Value Group**	5 secs
Parameter Object listing from **My Preferences**	10 secs

Parameter Value Groups

Parameter value groups are created on Parameter Value Groups page (Navigation > Repository > Report Objects > Parameter Value Groups).

We created 500 groups, (having 3 values in each group) for a PO, where average length of values of PO was 80 characters.

Page loading time was tested on following pages:

- Parameter Value groups page

- My Preferences

Page loading times were nearly same for both the pages and so we combined them in a common table.

Number of Value-groups	Page loading time
20	2 seconds
50	4 seconds
100	5 seconds
200	9 seconds

Reports / Categories

Reports are organised within categories. We conducted tests to find out time taken to open the page. We started by creating 100 categories and continued the testing up to 5000 categories. The table shows the time it took to construct the bar where category names appear. This is done from Manage Folder and Reports tab.

No of Categories	Time taken
100	2 seconds
500	8 seconds
1000	15 seconds
4000	30 seconds
5000	39 seconds

With help of a test-API, we deployed up to 1000 reports within a category and tried opening following pages:

- Repository > Manage reports and folders

- Report Listing from Explorer

- Report List (Report list for category) from Navigation Menu.

- Navigation > Administration > Manage Users > Entity Access right

Table Showing result with **Concurrency = 1**

No. of reports	Manage reports And Folders	Report Listing-From Explorer (in Sec)	Report list for a category
100	2 sec	4 sec	7 secs
500	8 sec	7 sec	14 secs
1000	15 sec	18 sec	35 secs

Same Scenarios were taken for **Concurrency of 100 users** to see how the application responds under load.

Number of reports: 100

Report Listing from Manage reports and Folders	2 secs
Report Listing from Explorer	10 secs
Report list for a category	6 secs
Report Listing from Entity Access rights	21 secs
Category listing from Manage folder page (100)	9 secs
Category Listing from Reports Menu	14 secs

Viewing Published report from secondary location

Report Server keeps all RPG files at a single configured location. But as the number of report outputs and thus memory usage increases disk space management becomes a problem, thus to solve this problem secondary RPG location is introduced.

Viewing of report is HTML format from secondary location and the result is measured for the 1st response time.

RPG folder size	Time taken (Primary location)	Time taken (Secondary location)
2.20 GB	3 seconds	4 seconds
5.0 GB	3 seconds	5 seconds
10.70 GB	4 seconds	6 seconds

Executing Flash Chart on Dashboard

Ad hoc Report is having 68 lac records having chart type as bar, 1-field on X-Axis and 6-fields on Y-Axis. In the template file set the property of Chart Provider as "Am Chart". Now set this report on dashboard.

Viewing of Dashboard having single report, 68 lac records in chart and the response is measured for single iteration concurrent users hit.

S.No.	No. Virtual Users	Minimum Response time (mm:ss:mss)	Average Response time (mm:ss:mss)	Maximum Response time(mm:ss:mss)
1	400 (h2 db) 400 (orcl db)	01:06:879 00:36:046	02:24:789 03:06: 279	03:43:794 03:46: 271
2	600 (h2 db) 600 (orcl db)	0:48:342 01: 31: 783	03:13:937 02:57:048	04:54:557 04:46:273
3	900 (h2 db) 900 (orcl db) 1100(orcl db)	01: 34: 936 01: 08: 464 01: 53: 304	05: 18: 710 04: 16: 734 05: 56: 687	09: 11: 123 06: 05: 694 09: 03: 319

On increasing number of users, the response time increases.

Executing Image Chart on Dashboard

Ad hoc Report is having 68 lac records having chart type as bar, 1-field on X-Axis and 6-fields on Y-Axis. In the template file set the property of Chart Provider as "R Chart". Now set this report on dashboard.

Viewing of Dashboard having single report, 68 lac records in chart and the response is measured for single iteration concurrent users hit.

S.No.	No. Virtual Users	Minimum Response time (mm:ss:mss)	Average Response time (mm:ss:mss)	Maximum Response time(mm:ss:mss)
1	400	00: 26: 109	04: 10: 365	05: 04: 979
2	600	00: 27:170	06: 21: 787	07: 58: 448
3	900	01: 16: 248	23: 49: 475	34: 27: 503

intellicus
Enterprise Reporting and
Business Insights Platform

> **Note:** pre-request for dashboard scenario should have the following thread configured:

To execute such a scenario you need to configure the application as mentioned below.

Jakarta: In server.xml set maxThreads="2000"

Report Server properties:

- (a)   Database Connection TimeOut (seconds) = 1400
- (b)   Exec Threads = 1500
- (c)   Service Threads = 1500
- (d)   Dashboard Threads = 1500
- (e)   Data Source Fetch Size (rows per fetch) = 5000

Report Client properties: set

- (a)   Report Server Time Out (seconds) = 7200
- (b)   HTML Viewer Time Out (seconds) = 1200
- (c)   Report Server Chunk Time Out (seconds) =7200

Database Repository connection: set
- (a)   Initial Connection(s) = 900
- (b)   Incremental Size = 100
- (c)   Max. Connections = 1300

Report Server

Load that Intellicus Report Server can take depends on factors like:

- License

- Operating System

- RAM allotted to Intellicus

To run heavy loads, we recommend going for clustering and load balancing.

> **Note:** The number of reports that can be run will essentially depend on the License.

OLAP on File System

The time taken to build a cube with 5 dimensions and 5 measures is shown below:

Number of Records (lacs)	10	25
Cube Build Time (mins)	20	68

Benchmark tests and results

Benchmark aims to provide us with facts and figures of resource usage and timings under various report execution scenarios.

Analysis of benchmark tests results will help you compare your actual usage requirements with scenario results.

Scenarios

- Data volume v/s concurrent users

- Data volume v/s output types

- Chart report: Flash Chart v/s Image Chart

- Complexity: Sub-reports, Groupings and scripting

- Reports that contain images

- iHTML Output Format (Expanded v/s Fetch on demand)

- Resource consumption and release

- OLAP Report Execution

- SMART Report Execution

Test results provides following figures:

- Response Time (first page available in time)

- Throughput time (complete report available in time)

- CPU usage (% of total CPU)

- RAM usage (% of total RAM assigned to engine)

Machine configuration

Intellicus Report Server and Intellicus portal was deployed on machines having standard server configuration:

Configuration parameter	Intellicus	Database Server	Browser
Processor	Intel(R) Core(TM) i5-3470CPU @3.20 GHz, 3.20 GHz	Intel(R) Xeon(TM) 3.2 GHz (2 processors)	Intel(R) Pentium(R) D CPU 2.8 GHz , 2.8 GHz
RAM	8 GB system RAM, 4 GB for Server, 2 GB for Portal	8 GB	4 GB
Operating System	Windows Server 2012	Windows 7 Enterprise	Windows 7 Enterprise
Database	-	Oracle Database 11g	-
Web Server	Jakarta Tomcat	-	-
JRE	6	-	-
System Type	64-bit	64-bit	64-bit

Important: The results in different environments, may vary depending on the factors specified earlier in this document.

Scenario 1: Data volume v/s concurrent users

This scenario is aimed to observe change in Intellicus performance when number of users is increased along with data volume.

Ad hoc report having a record-size of approximately 200 characters, spread across 10 columns was designed and executed in HTML output format.

Scenario was repeated multiple times to get performance results of:

- **Users:** 10, 25, 50

- **Data volume – records (pages):** 1000 records (78 Pages), 10000 records (800 Pages).

Response time (1st chunk)

Chart indicates that with increase in user concurrency response time does not increase in the same order. Same is the case observed when tests were repeated with 10000-record report.

Response time - 1st chunk (Records v/s users)

	10	25	50
1000	35.561	36.299	43.205
10000	40.192	62.668	73.741
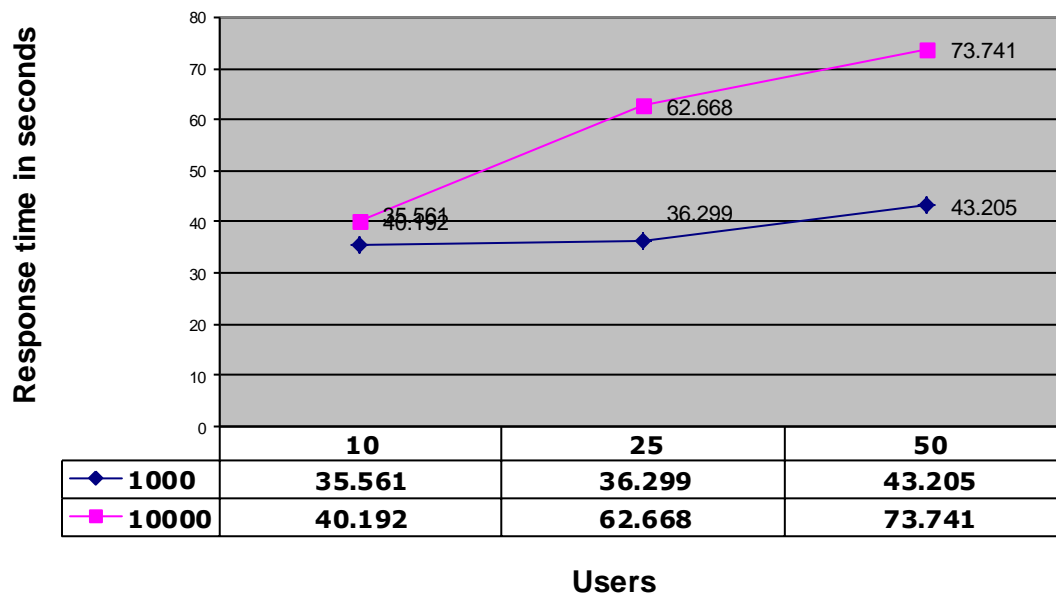
Users

Maximum CPU Usage

The CPU usage increases upon concurrency increase for same amount of data. CPU usage also increases for larger data (consumes CPU for more time) for same concurrency.

Maximum CPU Usage (Records v/s users)

	10	25	50
1000	27.5	48.5	71.125
10000	51.5	60	83
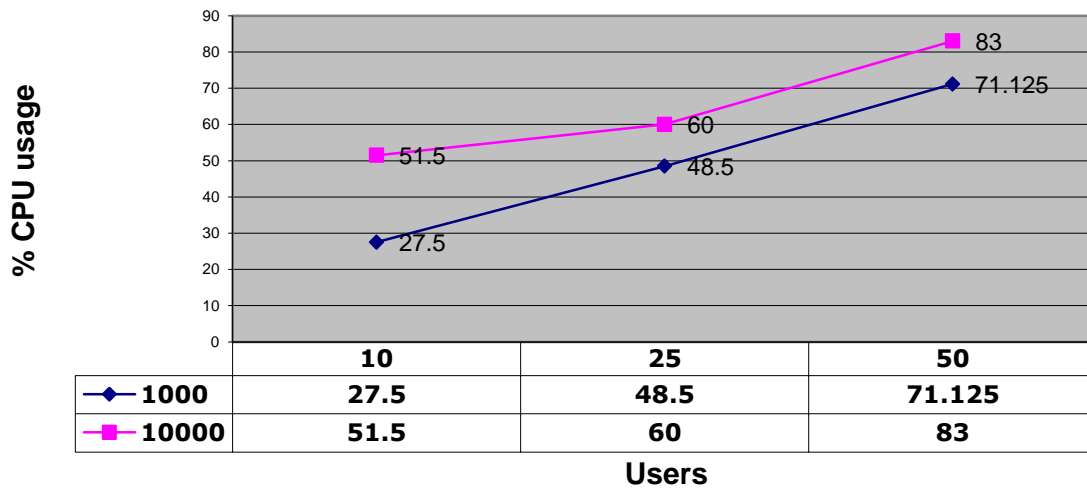
Users

Maximum RAM Usage

The MAX heap size is set to 2GB, Report server reaches certain limit in using and releases memory after the load is reduced.

Maximum Memory Usage (Records v/s users)

Scenario 2: Data volume v/s output types

Intellicus offers multiple options for report output and delivery.

When a report is requested, Intellicus processes data and prepares intermediate format. This intermediate format is converted in the requested generic output format.
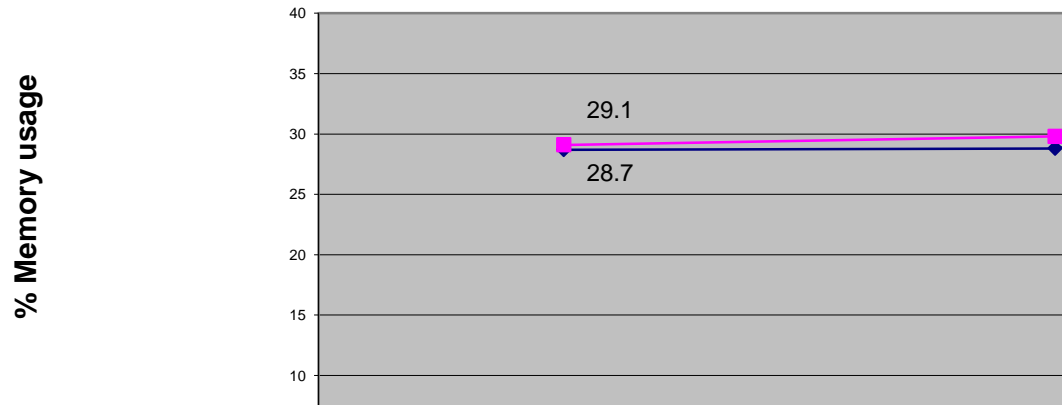
This scenario aims to get the resource usage figures and **throughput time** for various output formats. (Response time for HTML format)

Scenario repeated multiple times to get performance results of the following:

- **Output type:** PDF, CSV, XLS, RAW TEXT, HTML

- **Data volume:** 1000 and 10000 records.

Record size was 200 characters per row spread across 10 columns.

1000 rows data generated 77 pages report.
10000 rows data generated 769 pages report.

Report Execution time

Report execution time - (Records v/s output types)

	PDF	CSV	XLS	RAW	HTML
1000	4.175	2.503	3.425	0.766	1.177
10000	57.797	45.395	60.386	1.495	58.077

Response time in seconds

Output types

Maximum CPU Usage

Maximum CPU Usage (Records v/s output types)

	PDF	CSV	XLS	RAW	HTML
1000	79	63	67	16	27
10000	85	71	100	49	61
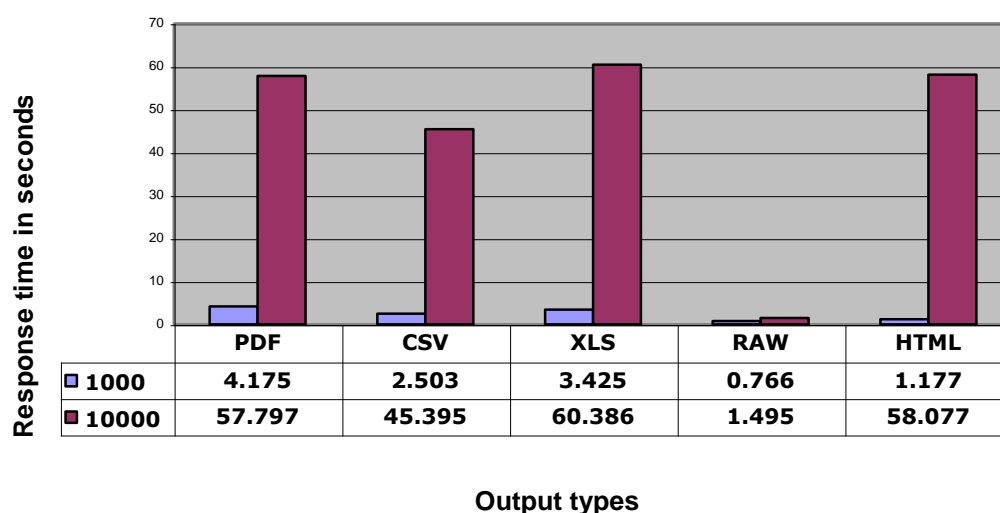
Output types

Maximum RAM Usage

Maximum RAM Usage (Records v/s Output types)

	PDF	CSV	XLS	RAW	HTML
1000	**28.8**	**28.8**	**28.8**	**29.1**	**28.7**
10000	**31.3**	**29.4**	**99**	**29.1**	**29.1**

Output types
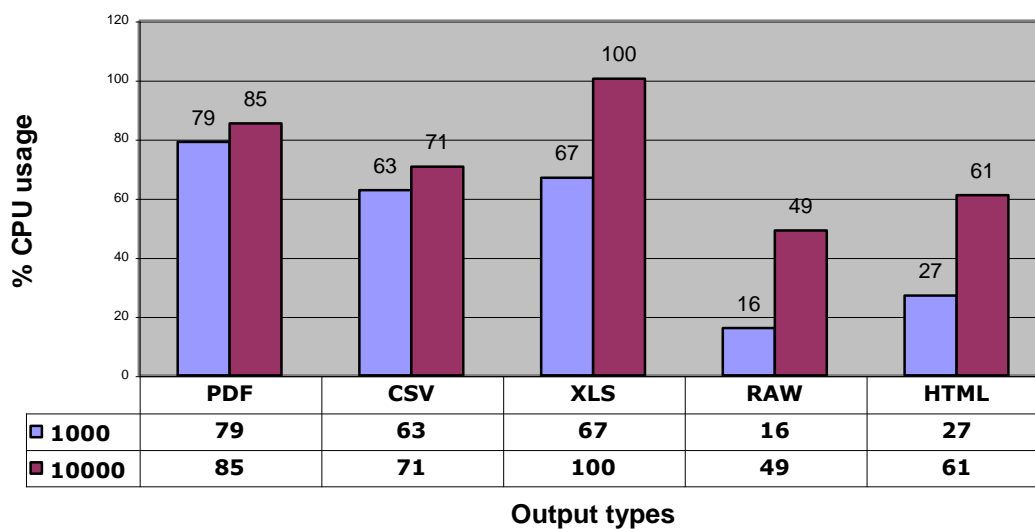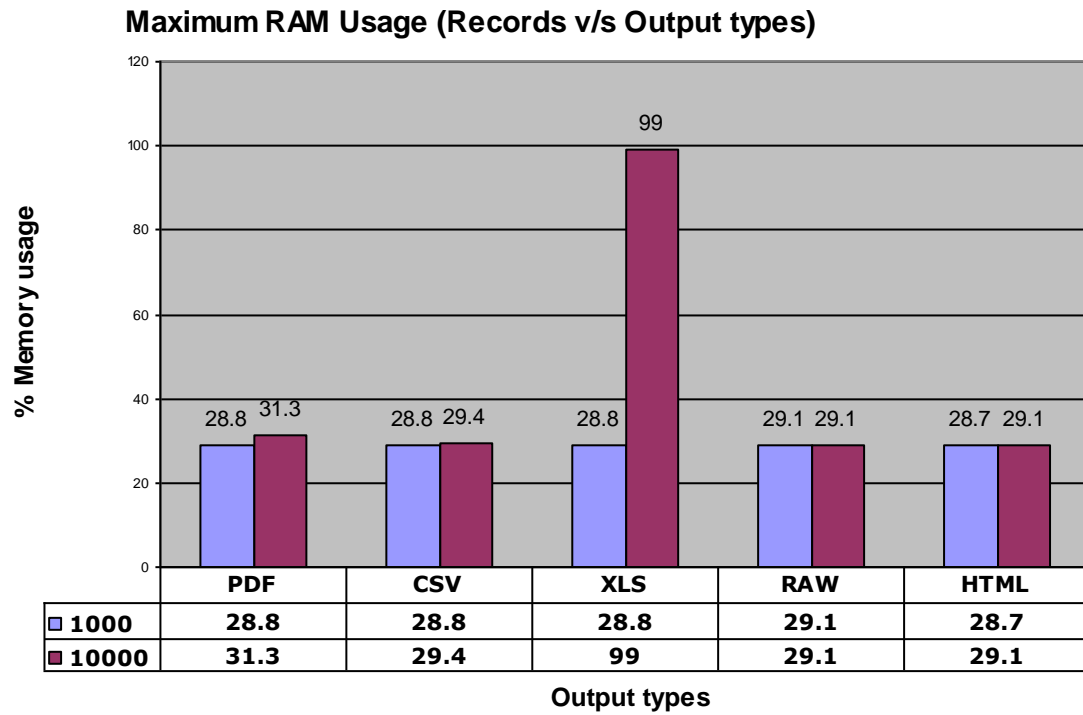
Scenario 3: Chart report: Flash v/s Image

Report contains information generally represented in the form of table and charts.

Chart is plotted after processing all the required data. Depending on placement of the chart, it may require processing all the records in the record-set before the chart is actually plotted.

This Scenario aims to figure out the difference in performance when user executes the same report with two different chart providers.

Here the chart is generated after processing all the 500 rows.

Report Execution Time

Response time - (Flash chart v/s Image chart)

	10	25	50
Image chart	0.955	1.148	2.069
Flash Chart	1.048	1.728	2.639

Users
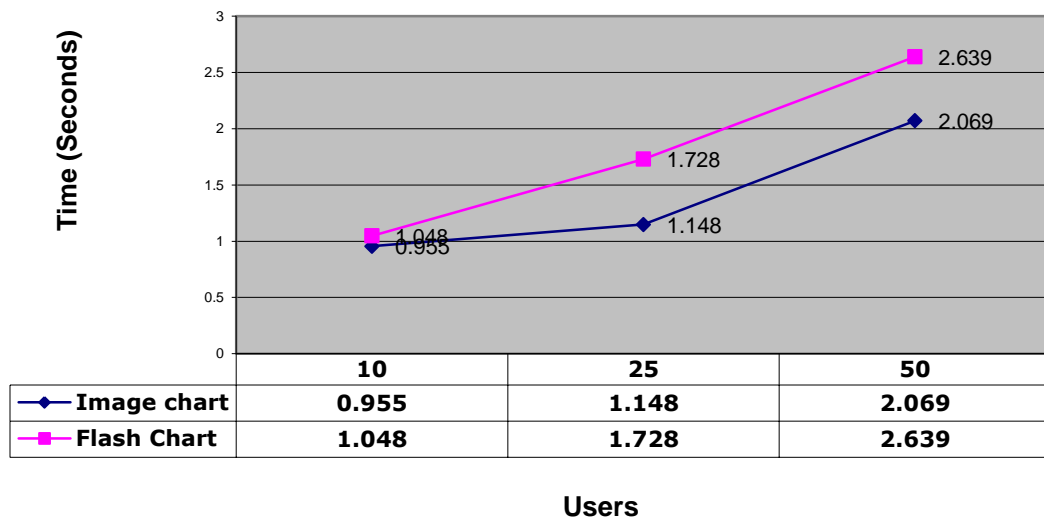
Maximum CPU Usage

Maximum CPU usage - (Flash Chart v/s Image Chart)

	10	25	50
Image Chart	25	36.625	42
Flash Chart	46	55	60

Users
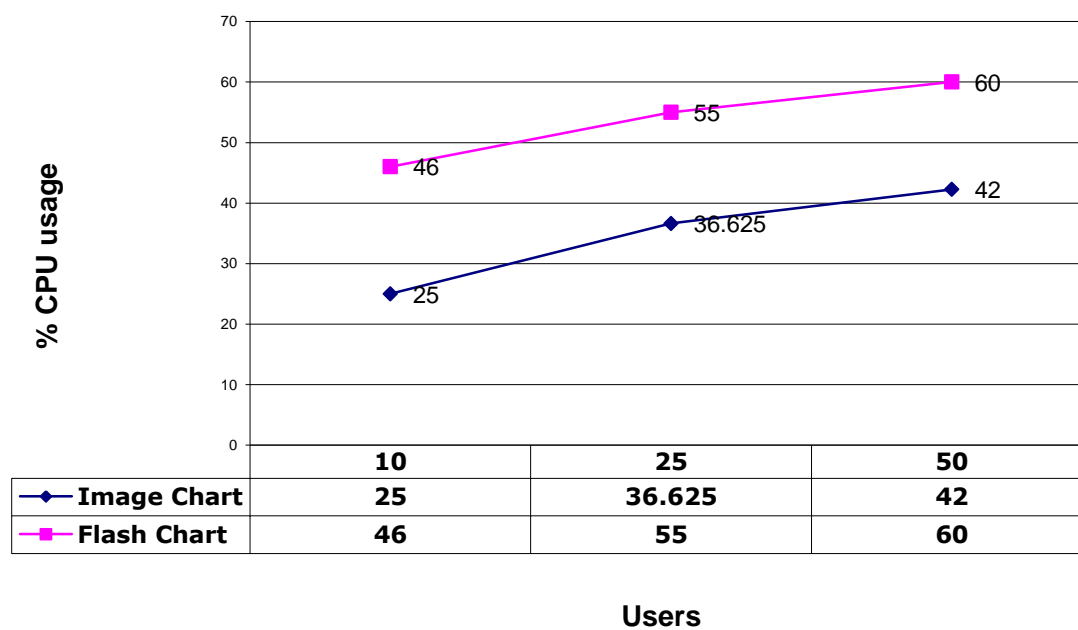
Maximum RAM Usage

Maximum Memory usage - Flash Chart v/s Image Chart)

% Memory usage	10	25	50
Image Chat	12	12.7	13.8
Flash Chart	13.3	15.6	21.5

Users

intellicus
Enterprise Reporting and
Business Insights Platform
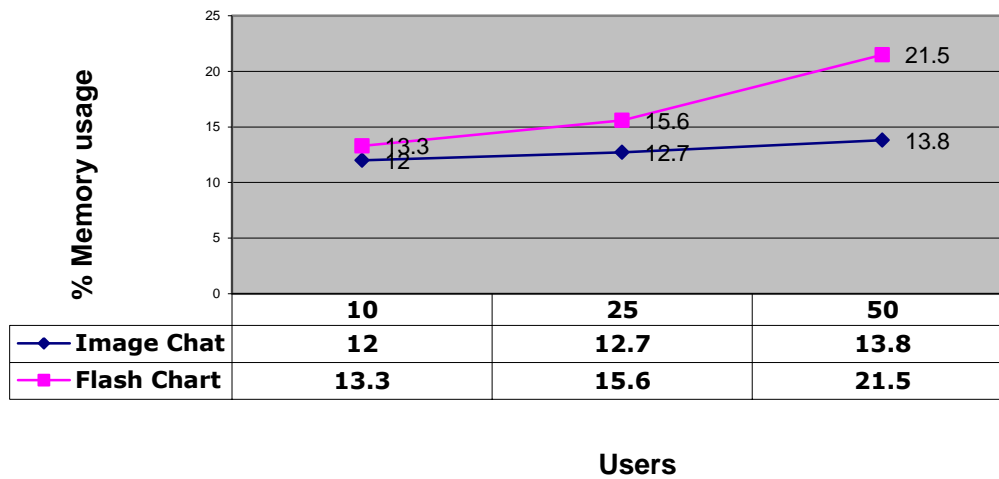
Scenario 4: Sub-reports

Execution of a report having sub-report(s) may affect resource usage of report server.

When a sub-report control is encountered in a report, execution of parent report is put on hold and execution of sub-report is started. A sub-report adds to the complexity in report execution.

A sub-report consumes a separate data connection than its parent report.

More levels of sub-report may consume more RAM. CPU usage is less likely to get effected.

Results of this scenario aim to get figures of CPU usage, RAM usage and execution times.

Scenario was repeated for sub-report level 1 and sub-report level 2. For each sub-report level scenario was repeated for 10, 25 and 50 concurrent users.

- **Sub-report level 1:** One sub-report in a report.

- **Sub-report level 2:** Report has a sub-report. The sub-report also has a sub-report.

Report Execution time

There is not much increase in response time with increase in levels. However, response time does increase in linear fashion with increase in number of concurrent users.

Maximum CPU usage - Sub-report levels v/s users

	10	25	50
Level 1	1.119	3.067	10.346
Level 2	2.507	6.261	21.373

Users

Maximum CPU Usage

Maximum CPU usage - Sub-report levels v/s users

	10	25	50
◆ Level 1	57.625	68.25	71.5
■ Level 2	62.375	69.52625	75.25
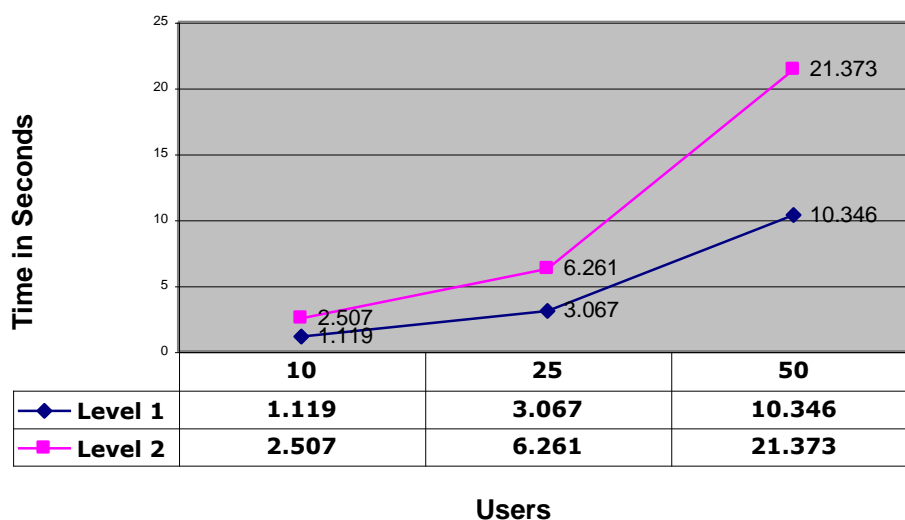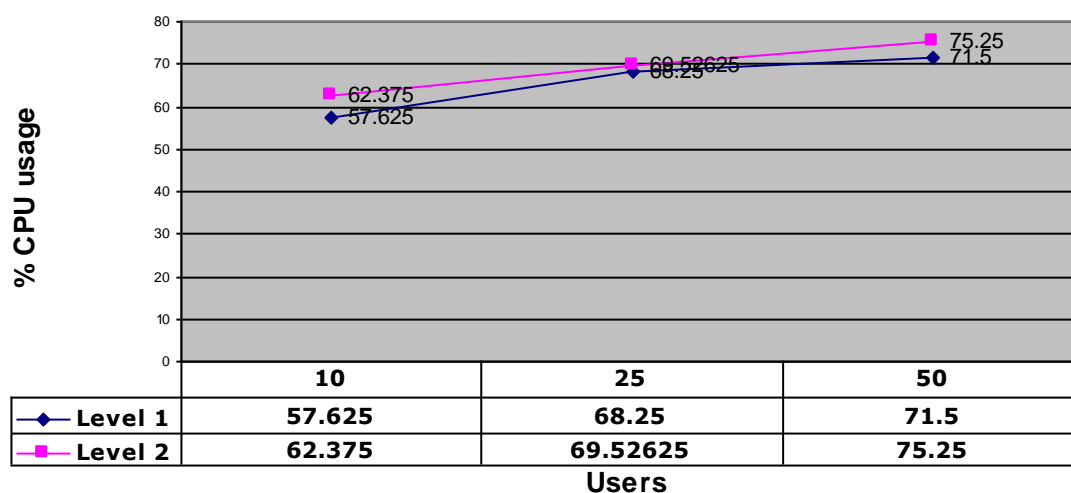
Users

CPU Usage shows slight increase with increase in number of users. From level 1 to level 2, it shows marginal increase.

Maximum RAM Usage

Maximum Memory usage - Sub-report levels v/s users

	10	25	50
◆ Level 1	27.2	29.8	29.8
■ Level 2	28.8	29.8	32.8
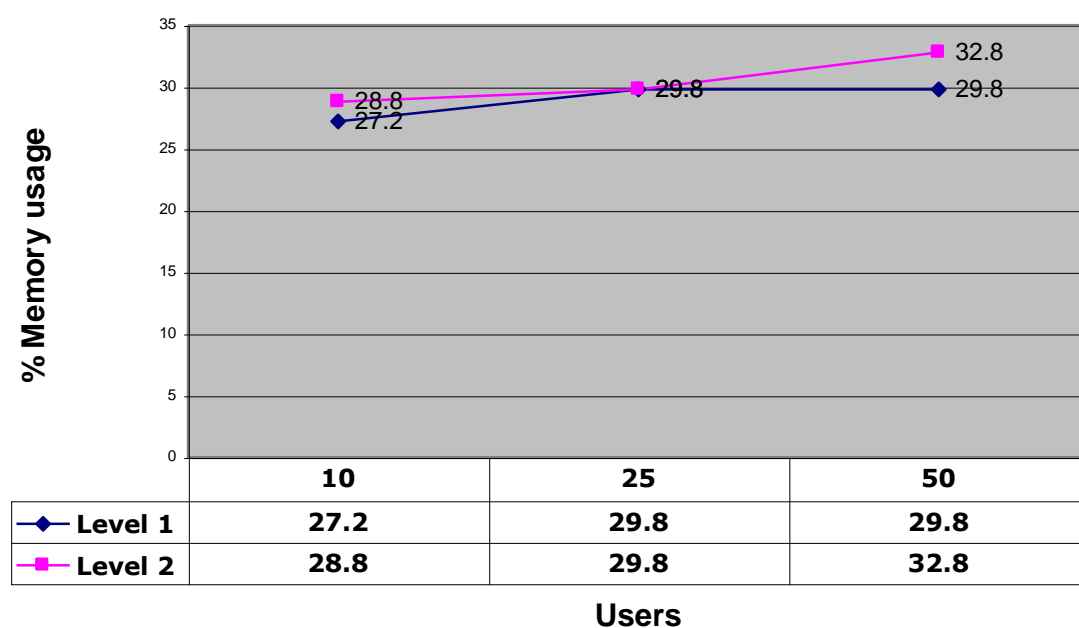
Users

When a report has one sub report, RAM usage increases in linear fashion with increase in concurrent users, whereas it remains almost constant for level 2, irrespective of number of concurrent users.

Scenario 5: Logo images

Reports generally have company logo (image).

Image can be used on a report as dynamic data coming from database or static image inserted during design.

This scenario is designed to get performance numbers of executing reports having a static image.

Result of this scenario will provide figures of report execution time, CPU usage and RAM usage for every combination of users and number of pages.

Report execution time

Report Execution time - (report having images)

	10	25	50
◆ **1000 Rows**	1.398	5.435	16.288
■ **10000 Rows**	23.259	331.665	418.085

Users
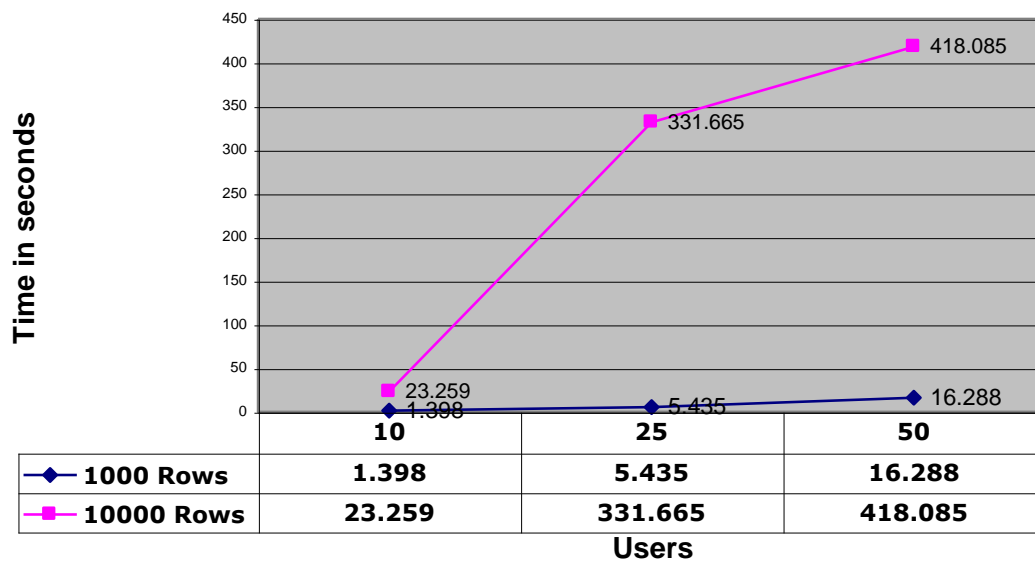
Maximum CPU usage

CPU usage - (report having images)

	10	25	50
1000	54.5	61	64.875
10000	66.25	68.375	71.75

Users
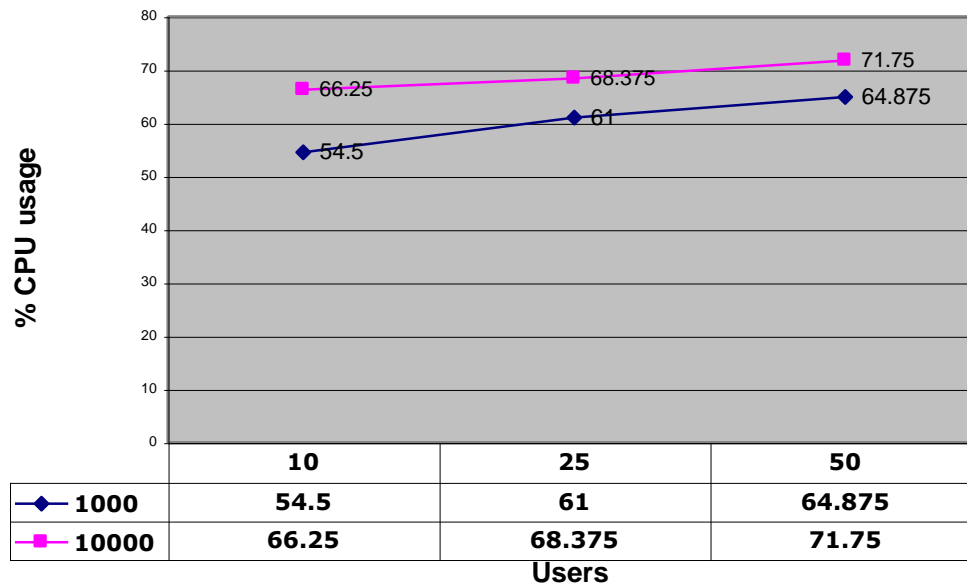
Maximum Memory usage

Memory usage - (report having images)

	10	25	50
◆ 1000	29.3	29.2	29.6
■ 10000	29.7	29.6	29.6

Users
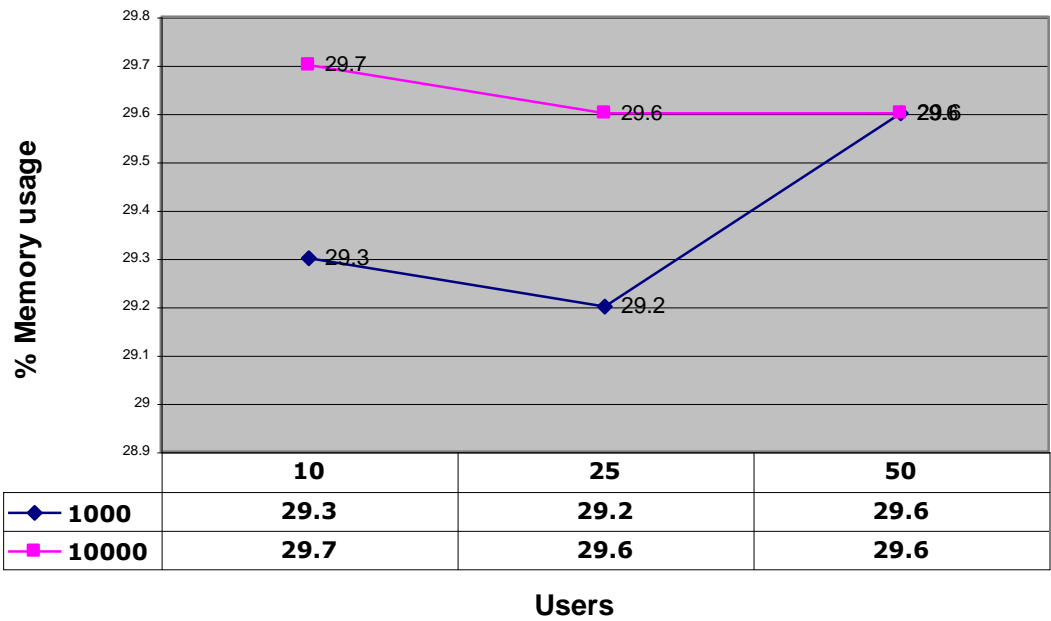
Scenario 6: Report Complexity through scripting

Scripting adds a considerable degree of flexibility in report designing in Intellicus. Some of Intellicus clients use scripting abundantly to dynamically decide what to hide and what to show on a report.

Scripting if used abundantly in a report may add significant levels of complexity for report server. This may have effect on report performance. This scenario is aimed to provide information on effects of scripting on report performance.

Scenario is repeated for different number of concurrent users. For each number of concurrent users, the scenario is repeated for reports having different set of records.

Report execution time

Report Execution time - Report with Scripting

	10	25	50
1000 Rows	1.478	3.1	10.759
10000 Rows	73.2	157.8	253
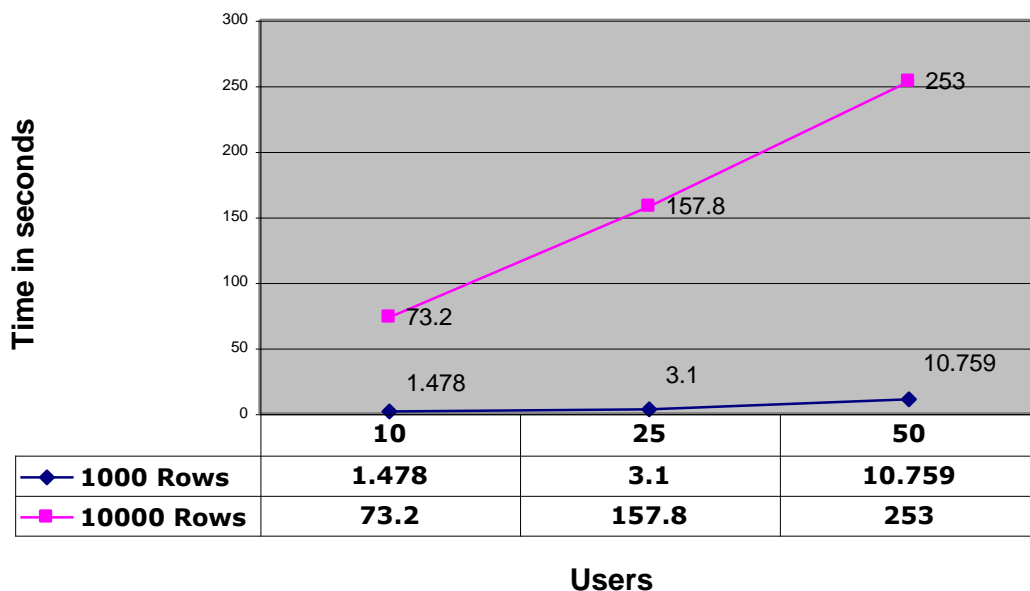
Users

Maximum CPU usage

Maximum CPU usage - Report with Scripting

	10	25	50
◆ 1000 Rows	45.25	62.5	72
■ 10000 Rows	55.25	67.875	81
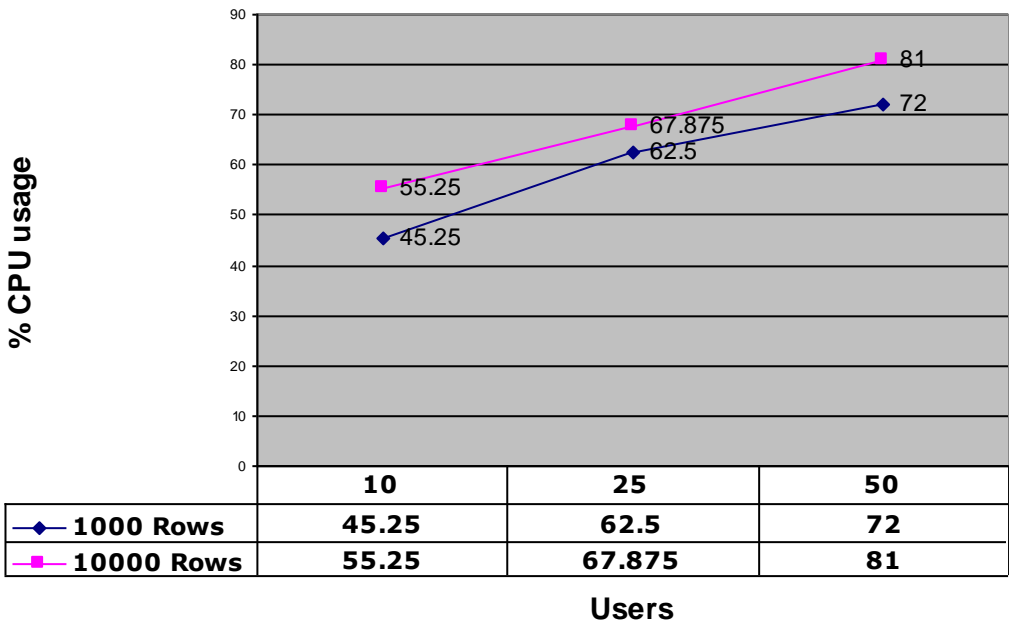
Users

Maximum RAM usage

Data volume v/s users - Report with scripting

	10	25	50
1000 Rows	45	67	82
10000 Rows	76	92	95

Users
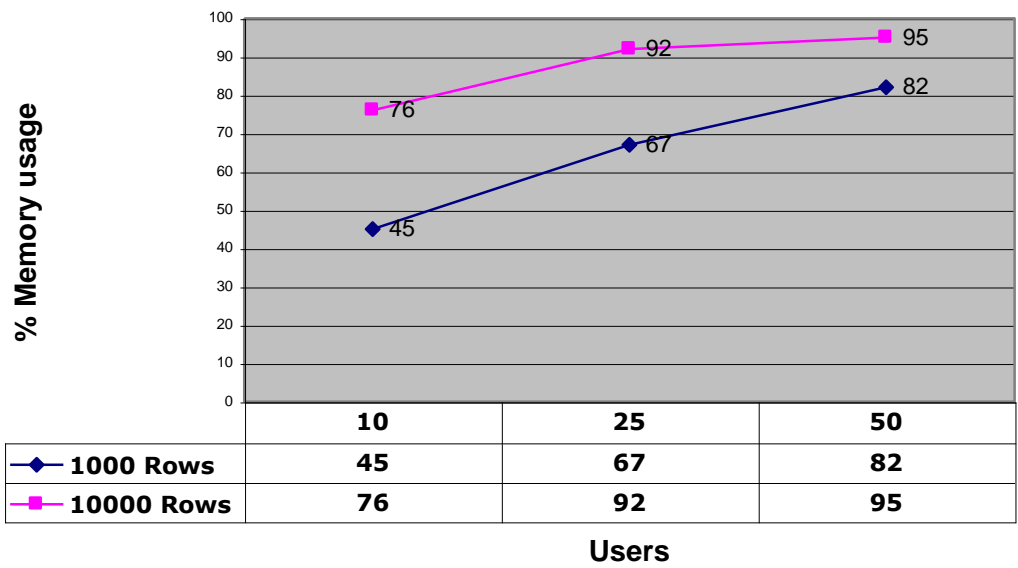
Scenario 7: Report having groupings

When a report has groupings, Intellicus report server needs to carry out additional tasks. This scenario aims to find out how does data groupings affect report performance.

When a report includes groupings, report server needs to make extra efforts like rendering group headers and footers; maintaining and rendering summary information (like totals) in the desired section. These efforts may affect all over performance.

When data in a report is grouped, stake-holders of that report would like to view all the data related to a group together on a page. To achieve this, Intellicus has a property KeepGrpTogether. Setting this property to true will make sure all the data of a group appears on a page.

This scenario is designed to get the figures of report execution time, CPU usage and RAM usage when a report is run with KeepGrpTogether property set to true and False. This will give user an understanding of the impact of switching this property ON and OFF.

A tabular report having 4 levels of groupings was designed for this scenario and executed for different set of concurrent users.

Report execution time

Report execution time - (report having groups)
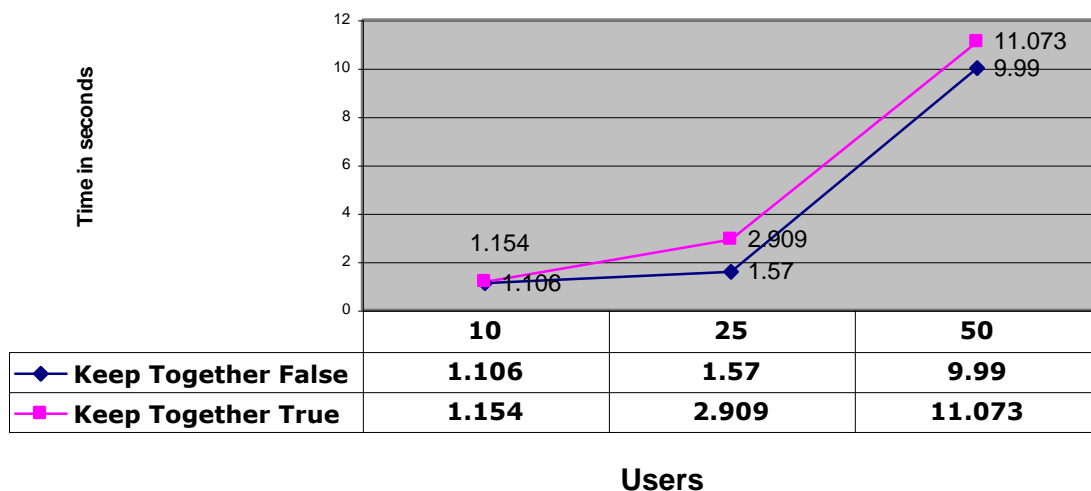

	10	25	50
Keep Together False	1.106	1.57	9.99
Keep Together True	1.154	2.909	11.073

Users

Maximum CPU usage

Maximum CPU usage - data volume v/s users (report having groups)
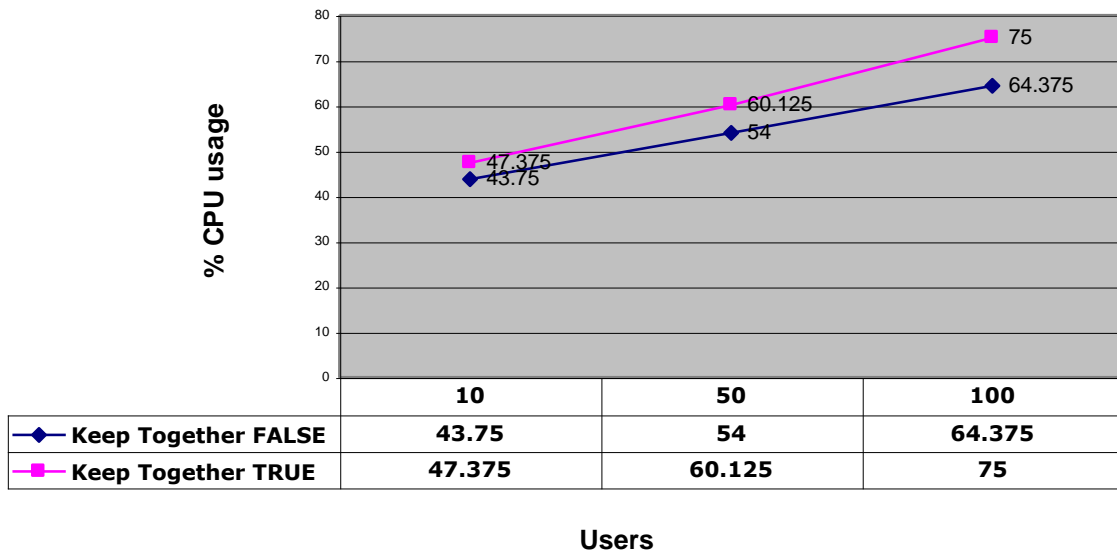
	10	50	100
◆ Keep Together FALSE	**43.75**	**54**	**64.375**
■ Keep Together TRUE	**47.375**	**60.125**	**75**

Users

Maximum RAM usage

Maximum RAM usage - data volume v/s users
(report having groups)

	10	50	100
Keep Together FALSE	12.6	30	52
Keep Together TRUE	12.8	33	67
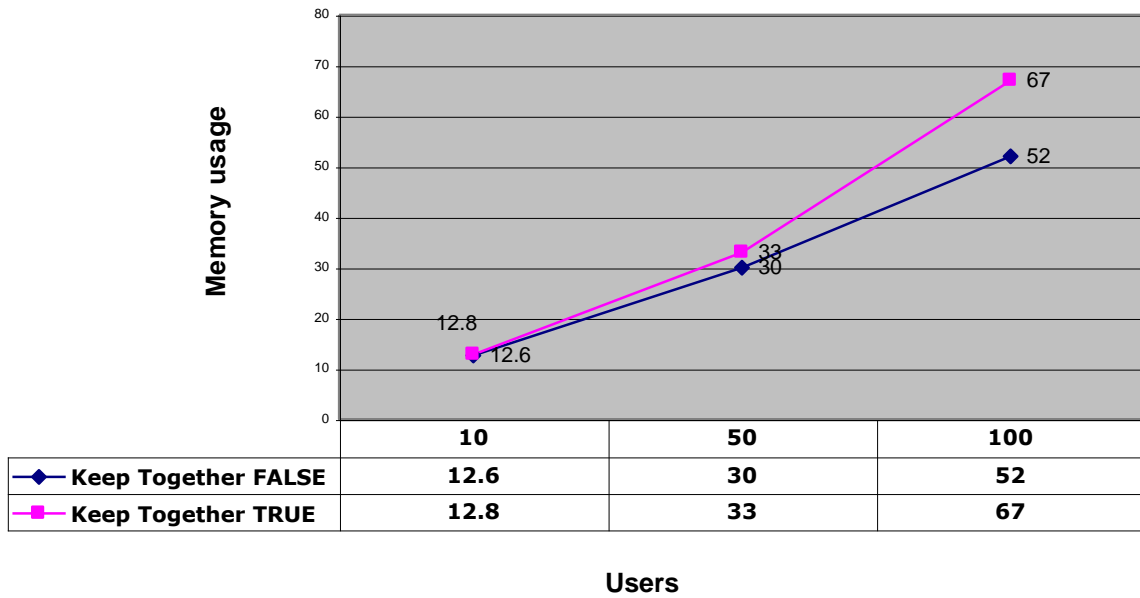
Users

Scenario 8: Resource usage trend for Standard Reports having 1000 records

The volume of traffic that multi-user software like Intellicus has to handle varies depending on the number of user logged, type of users logged in and the kind of operations they expect Intellicus to do for them.

Intellicus has to acquire right amount of resources to handle the traffic when traffic increases and also go on releasing unutilised resources when volume of traffic is reduced.

To get figures of resources are used by Intellicus in:

- Increasing traffic

- Decreasing traffic

The scenario provides results of % CPU usage and RAM Usage when variable load (10 users to 100 users) is given for variable time (from 15 min to 60 min).

Outcomes indicate that the resource usage increases with increase in users (report execution) and decreases proportionately with decreases in traffic.

CPU Usage

	15 Min (10 users)	30 Min (25 users)	45 Min (50 users)	60 min (10 users)
◆ CPU Usage	43.75	46.25	48.5	32

intellicus
Enterprise Reporting and
Business Insights Platform
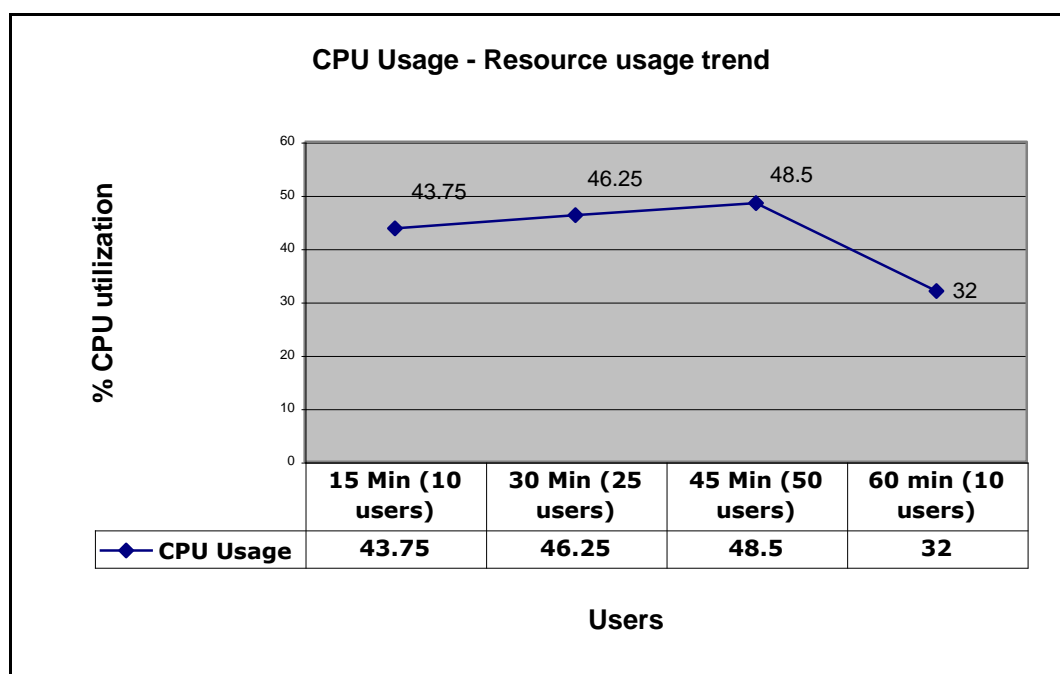
RAM usage

RAM usage - Resource usage trend

	15 Min (10 users)	30 Min (50 users)	45 Min (100 users)	60 min (10 users)
◆ RAM Usage	29.3	29.3	29.5	29.6

Users
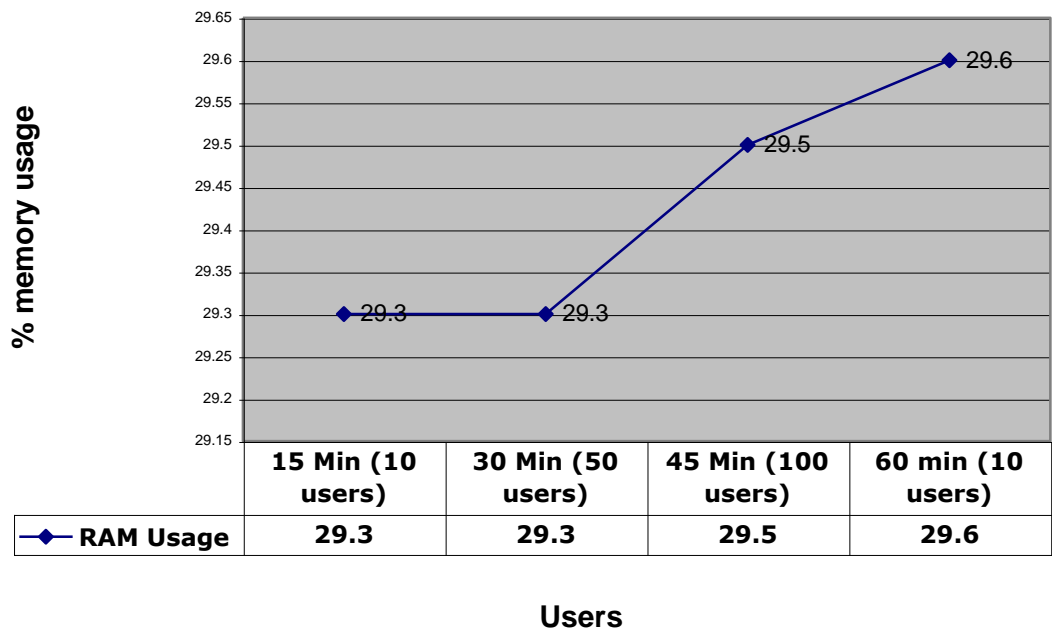
Scenario 9: iHTML Format (Expanded v/s Fetch on demand)

Intellicus introduced a new report output format - iHTML.

In this Format user can use any HTML page as a template for its report and embed some of the components like GRID and Charts on the HTML Page.

Performance of such kind of reports depends on the type of components being used in the HTML and the components of the Intellicus embedded in the report.
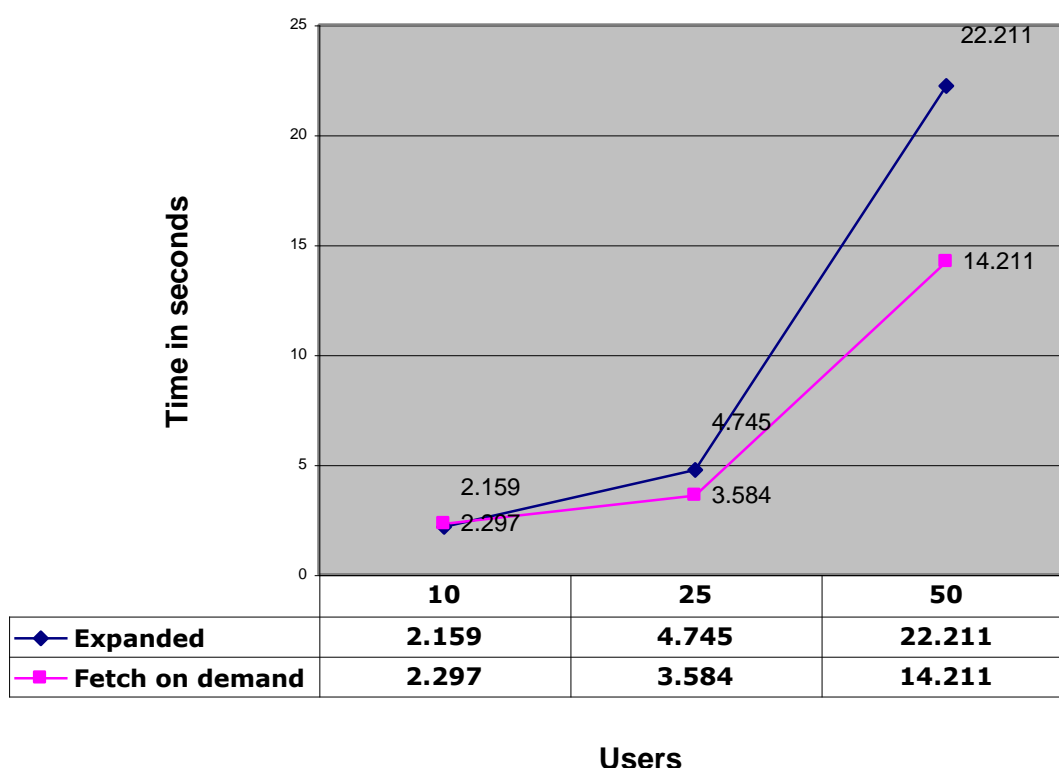
These kind of reports are not used for a huge record set. Number of records for this scenario is set to 1000.

The Scenario here is showing the comparison between two modes of iHTML report execution: **Expanded** mode is the one in which all the fetched records are shown at once and **Fetch on Demand** mode initially fetches records to fit on the size of the control or a single page and then more records are fetched as per user demand.

Report execution time

Report Execution time - iHTML Format

	10	25	50
Expanded	2.159	4.745	22.211
Fetch on demand	2.297	3.584	14.211

Users

intellicus
Enterprise Reporting and
Business Insights Platform

Maximum CPU usage

Maximum CPU usage - iHTML format

	10	50	100
◆ Expanded	51.125	52.75	53.125
■ Fetch on demand	40.25	49	52.5
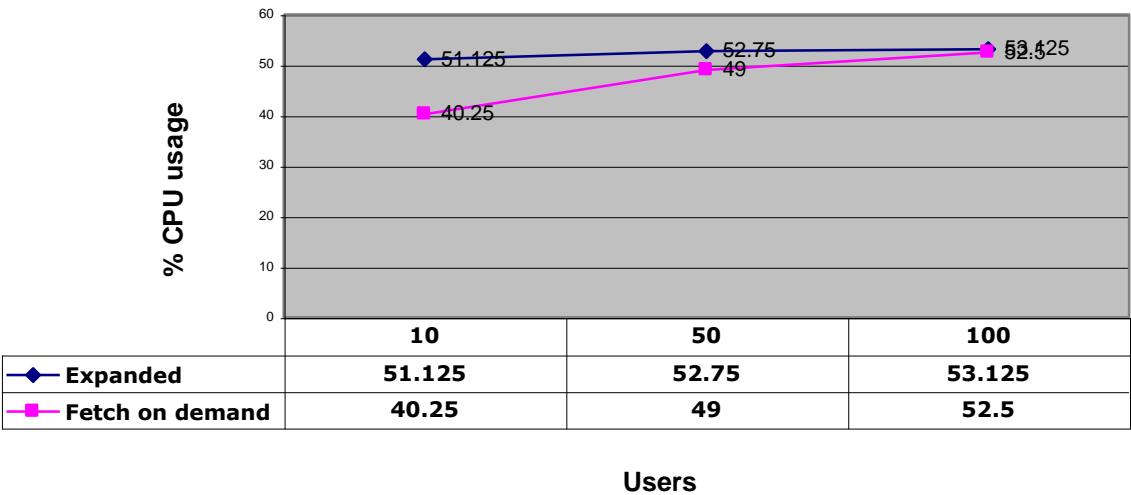
Users

Maximum RAM usage

Maximum RAM usage - iHTML format

	10	50	100
Expanded	12.5	15.1	16.9
Fetch on demand	16.8	16.8	16.9
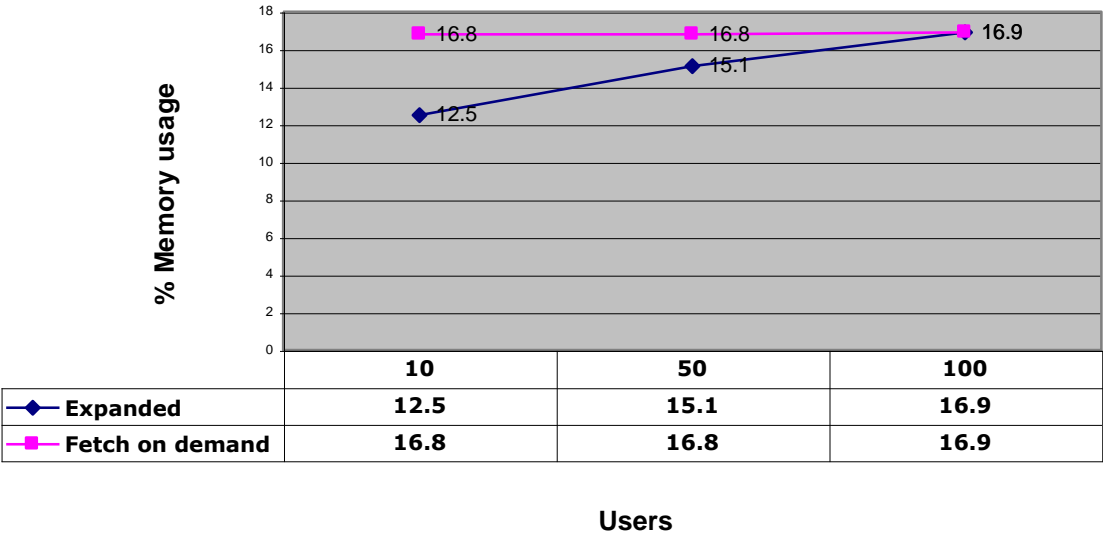
Users

Scenario 10: OLAP Report Execution- Grid Only

This scenario is aimed to observe change in report execution time of OLAP report when number of users is increased along with cube dimensions.

Cube having 25 lac records with a record-size of approximately 200 characters, spread across 10 columns was built and report executed in OLAP Viewer.

Scenario was repeated multiple times to get performance results of:

- **Users:** 10, 25, 50

- **Dimensions/ Measures:** 3 dimensions/ 3 measures, 5 dimensions/ 5 measures

Cube was built over local CSV connection and report executed in OLAP Viewer – Report contains Grid.

Response time

Response time - OLAP Report

	10	25	50
3 Dim	2.137	3.645	5.412
5 Dim	3.703	6.964	15.487

Users

Maximum CPU Usage

Maximum CPU Usage (OLAP Report)

	10	25	50
3 Dim	17.543	36.126	43
5 Dim	50	50	60

Users

Maximum RAM Usage

Maximum Memory Usage (OLAP Report)

	10	25	50
3 Dim	36.19	37.62	41.76
5 Dim	68.20	67.70	70.90

Users

% Memory usage

80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00

68.20   67.70   70.90

36.19   37.62   41.76
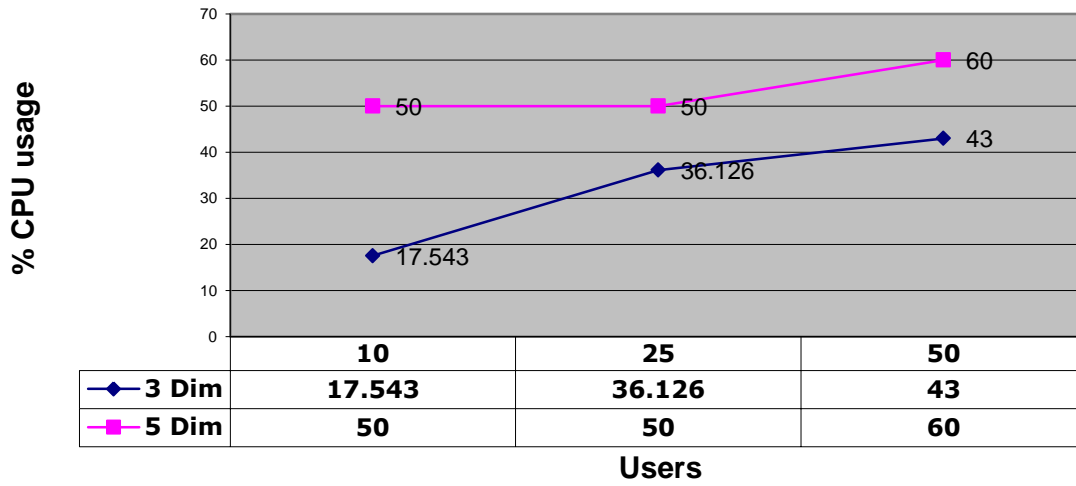
Scenario 11: OLAP Report Execution- Grid and Chart

This scenario is aimed to observe change in report execution time of OLAP report when number of users is increased along with data volume.

Cube having different number of records with a record-size of approximately 200 characters, spread across 10 columns was built and report executed in OLAP Viewer.
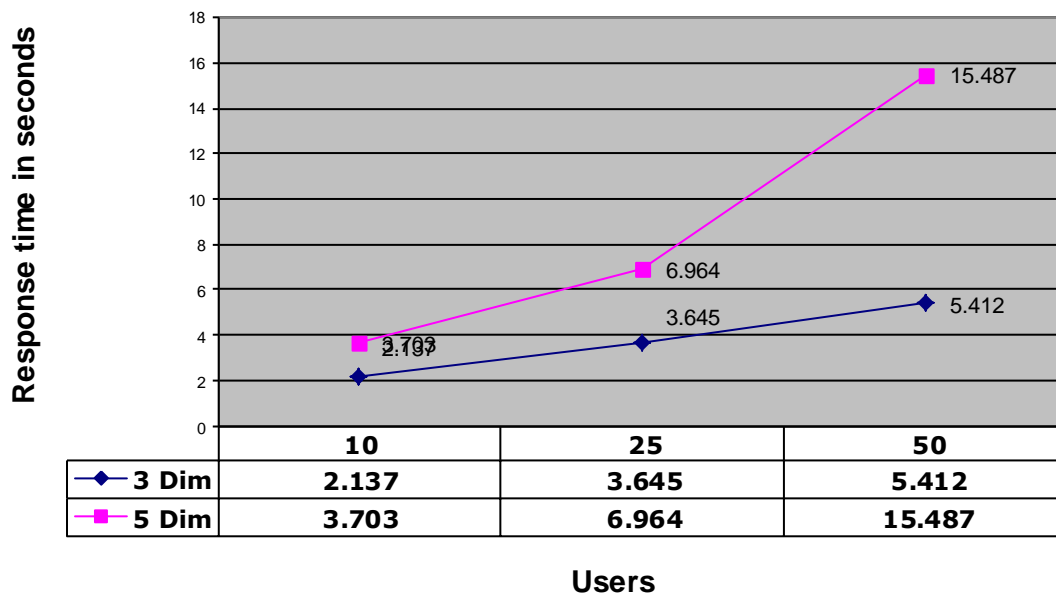
Scenario was repeated multiple times to get performance results of:

- **Users:** 10, 25, 50

- **Number of Records:** 10 lacs, 25 lacs

Cube was built over local CSV connection and report executed in OLAP Viewer – Report contains Grid and Chart.

Response time

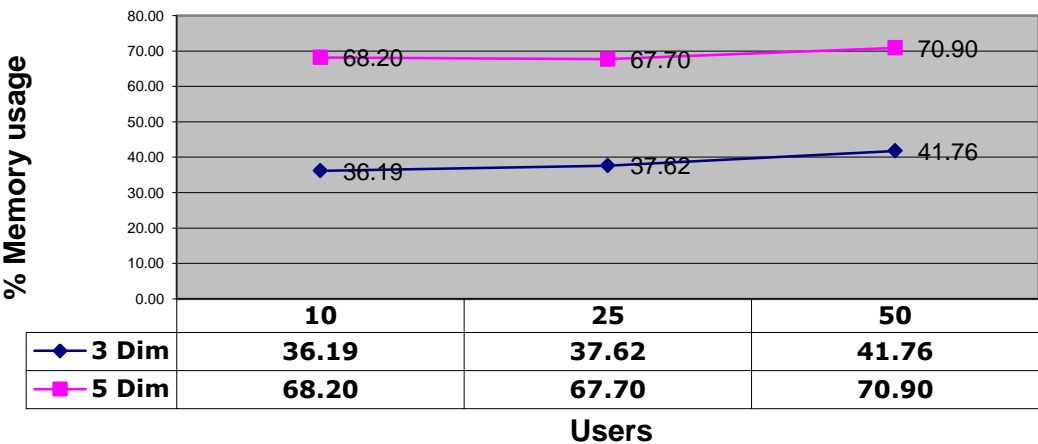**Response time - OLAP Report (Grid & Chart)**
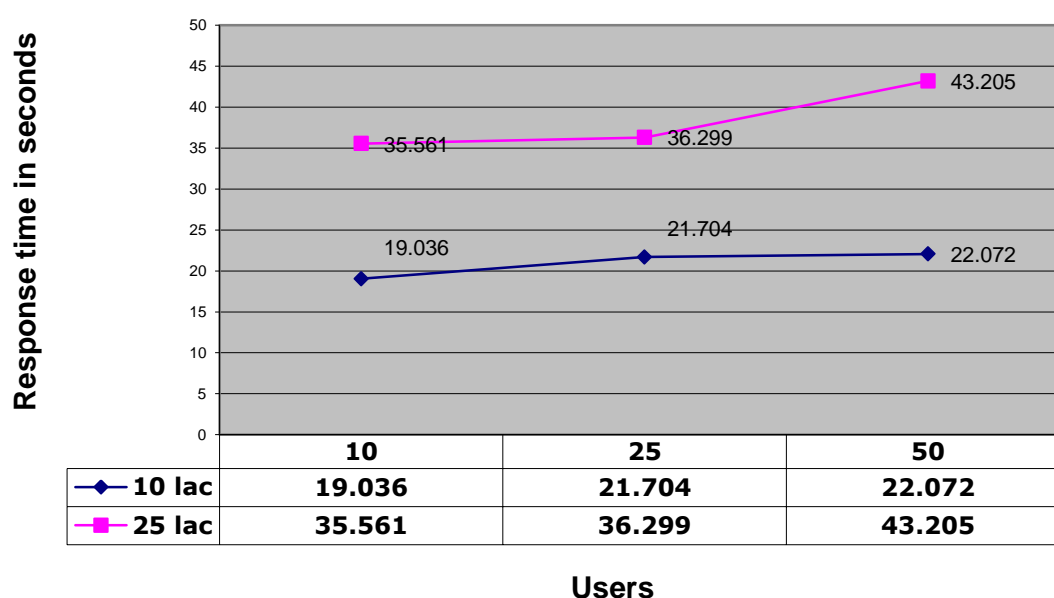
	10	25	50
10 lac	19.036	21.704	22.072
25 lac	35.561	36.299	43.205

Users

Maximum CPU Usage

Maximum CPU Usage (OLAP Report- Grid & Chart)

	10	25	50
10 lac	16.666	38.024	55.555
25 lac	22.272	52.987	80

Users

Maximum RAM Usage

Maximum Memory Usage (OLAP Report- Grid & Chart)

	10	25	50
10 lac	61.500	61.500	61.800
25 lac	65.500	65.900	68.200
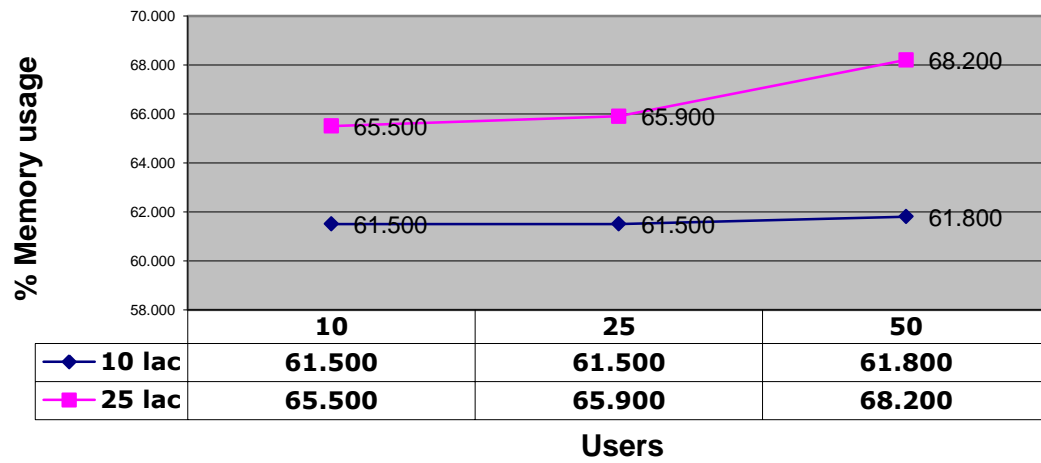
Users

Scenario 12: SMART Report Execution- Grid, Chart and Matrix

Intellicus' Ad hoc Visualizer reports are saved in SMART format.

This scenario is aimed to observe change in report execution time of SMART report when number of users is increased along with data volume.

SMART report having a grid of 10 fields and 1000 records, 3D Bar Chart and a matrix having 2 fields in Row, 2 fields in Column and 2 fields as Summary fields was designed and executed.

Scenario was repeated multiple times to get performance results of:

- **Users:** 10, 25, 50

- **Data volume – records:** 1000 records, 10000 records.

Response time

Response time - SMART Report
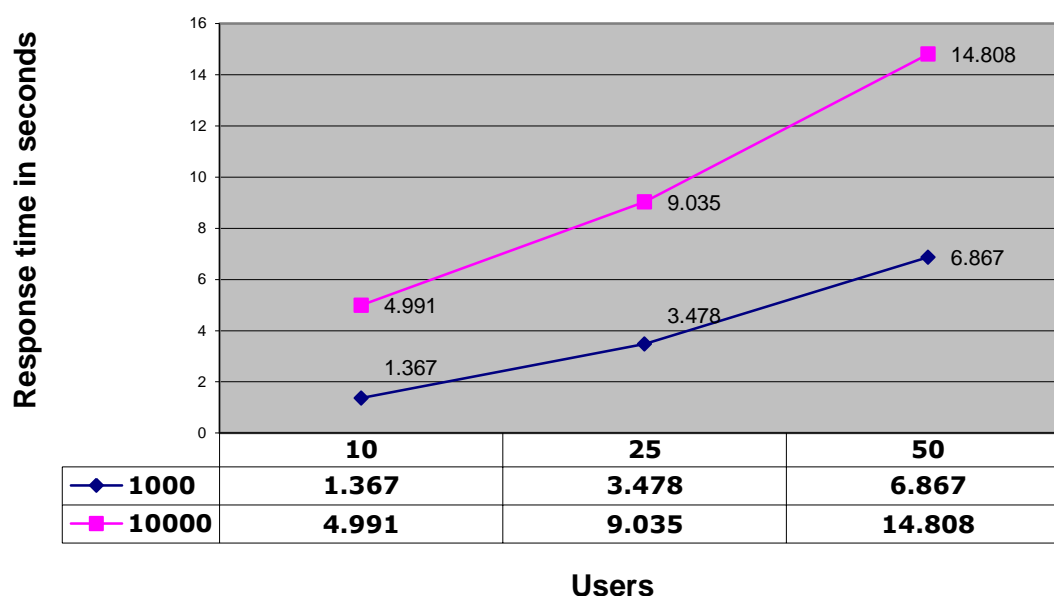
	10	25	50
1000	1.367	3.478	6.867
10000	4.991	9.035	14.808

Users

Maximum CPU Usage

Maximum CPU Usage (SMART Report)

	10	25	50
1000	33.333	47.226	55.75
10000	50	50	60.064

Users

Maximum RAM Usage

Maximum Memory Usage (SMART Report)

	10	25	50
◆ 1000	42.58	47.48	54.99
■ 10000	55.21	55.81	66.00

Users

intellicus
Enterprise Reporting and
Business Insights Platform
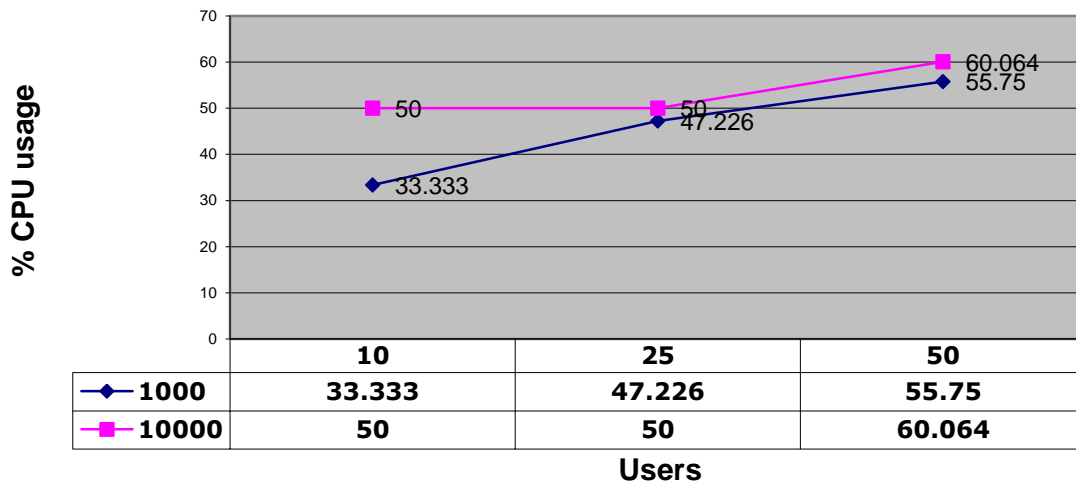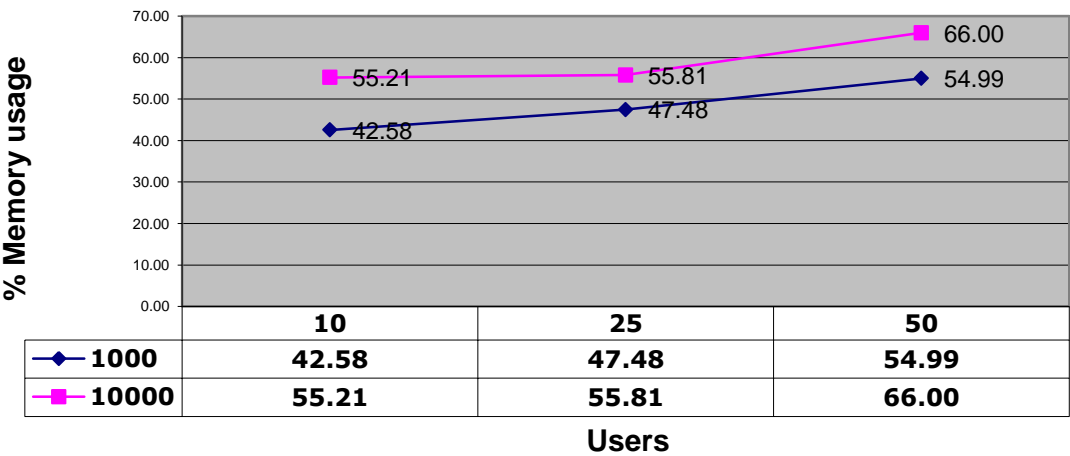
Scenario 13: SMART Report Execution- Different Chart Types

This scenario is aimed to observe trend in report execution time of SMART report when number of users is increased.

SMART report having 1500 records of different number of charts, different chart types on single chart tab was designed and executed.

Scenario was repeated multiple times to get performance results of:

- **Users:** 10, 25, 50

- **Number of Charts**: 6, 12

Response time

Response time - SMART Report of different chart types
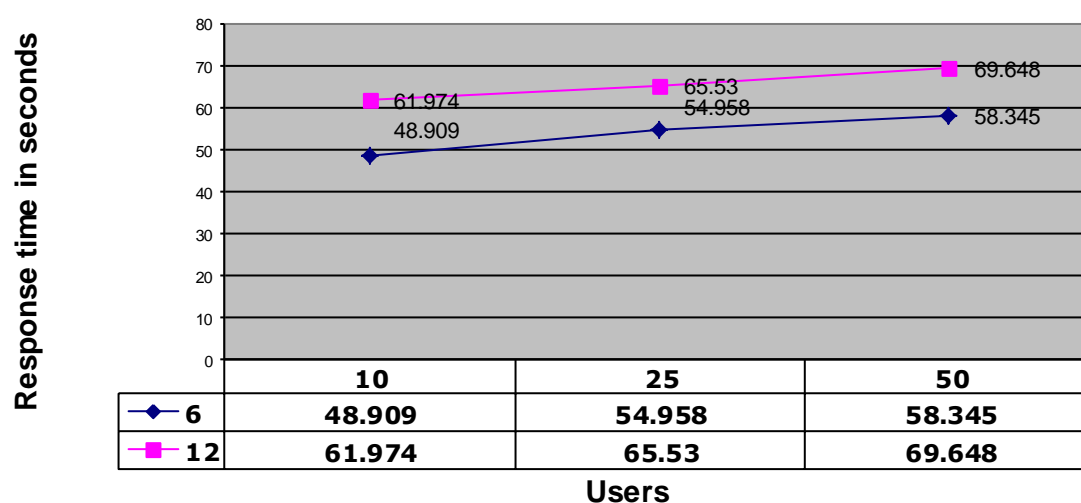
	10	25	50
6	48.909	54.958	58.345
12	61.974	65.53	69.648

Users

Maximum CPU Usage

Maximum CPU Usage (SMART Report of different chart types)

	10	25	50
6	**10.948**	**19.195**	**33.333**
12	**11.689**	**26.806**	**38.542**

Users

Maximum RAM Usage

Maximum Memory Usage (SMART Report of different chart types)

	10	25	50
6	50.761	54.857	67.868
12	56.763	66.419	77.893
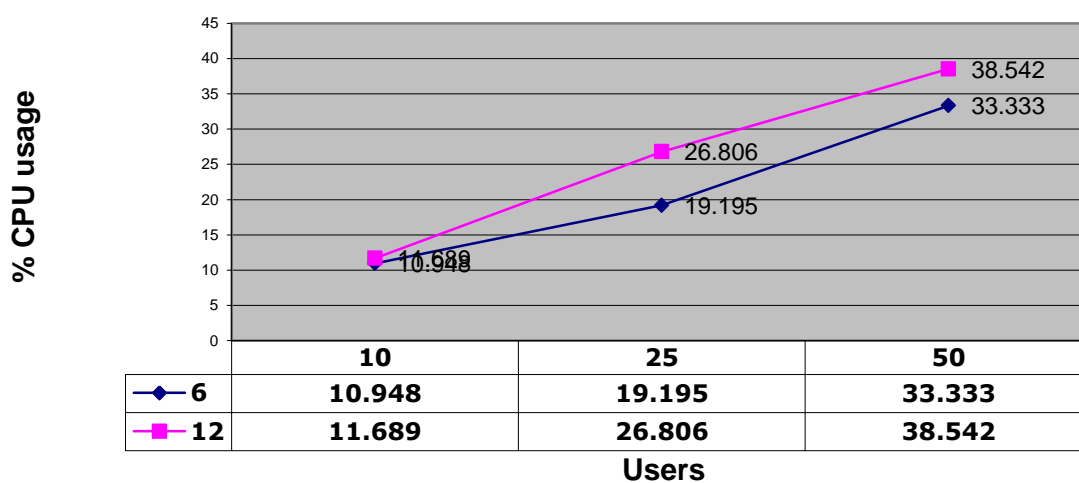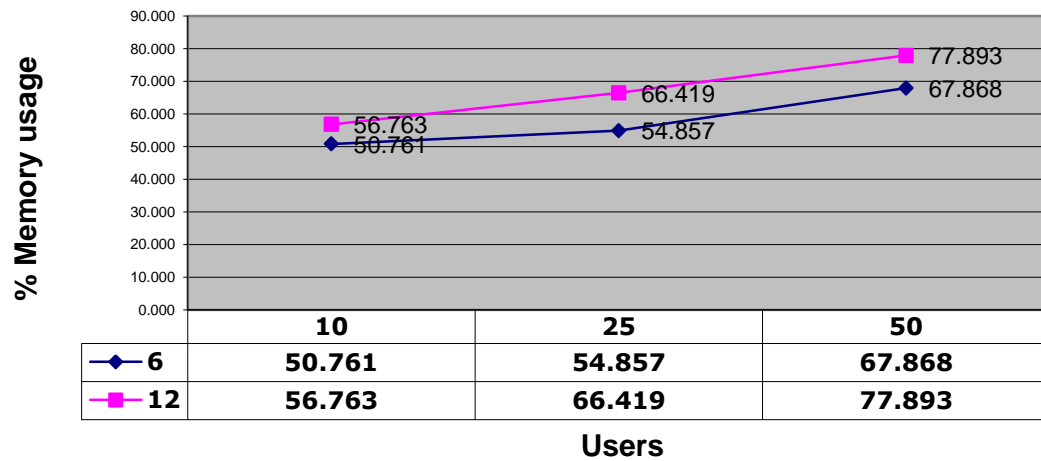
Users

Capacity Planning

Calculating Concurrency Requirement

To calculate the possible concurrent hits by your end users, the following may be useful formula:

Created Application Users

"Created applications users" is the count of users, created in your business application.

For a sample case, let us consider it as 1000.

Logged in Application Users

Logged in applications users is the count of users, logged in and working in your application at any point in time. You may take the maximum of these numbers for peak capacity planning.

We can consider logged in users as 25% of created application users. This percentage may vary depending on application type and organization.

For sample case, let us consider as 25% i.e. 250 users.

Concurrent Report Requests

Concurrent report request is the count of report generation requests fired at a given point in time, by logged in users. You may want take the maximum of these numbers for peak capacity planning.

We can consider concurrent report requests as 25% of logged in application users. This percentage may vary depending on application type.

For sample case, let us consider as 25% i.e. 65 requests.

Report Server CPU

Report Server is responsible for all major activities in Intellicus.

One of the major and CPU-intensive activities carried out by report server is report execution (as a result of user request, batch execution, or dashboard report execution). Server listens to the user request, parses a report, fetches data from database server, processes data, and generates report pages and streams to the portal. It carries out these activities for each report execution request.

Report Server also takes care of user authentication and authorization. It handles repository update activities like user management, schedules management and Report Object management.

Repository Services

One of the main activities of Report Server is Repository Services, which it performs by accessing the repository database. Because Repository Service is built-in to a report server, each report server in a cluster automatically distributes repository activity.
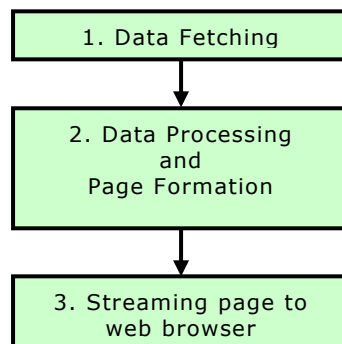
Report Execution Services

Interactive Reporting services are CPU-intensive when multiple reports are accessed and queries are processed concurrently. Interactive reports can be high or low priority requests coming from user on demand requests, or auto refresh dashboards.

Scheduled reports and reports submitted in background are also CPU-intensive but are executed at a lesser priority.

The pipelined architecture explained below, can guide you further how a reporting task is distributed with other computers in the system and how CPU sharing of this machine can be effective.

Intellicus performs the three major data reporting activities in a series of pipelined actions:

```
┌─────────────────────────┐
│    1. Data Fetching     │
└─────────────────────────┘
             │
             ▼
┌─────────────────────────┐
│   2. Data Processing    │
│           and           │
│      Page Formation     │
└─────────────────────────┘
             │
             ▼
┌─────────────────────────┐
│   3. Streaming page to  │
│       web browser       │
└─────────────────────────┘
```

Out of the three actions, report server spends its most effort and uses CPU time directly in data processing and page formation.

For a given request, during when the data is being fetched or the page is being streamed, Report Server thread yields most, the CPU for other threads or processes.

Installing the following multiple combinations on same machine is acceptable configuration:

   a) Report server and the reporting database or
   b) Report Server and the portal web server or
   c) Report server, reporting database and portal Web server,

You need to ensure that the machine has enough CPU power and also you need to consider the factor of CPU sharing due to pipelined architecture shall be limited.

Memory (RAM)

Memory is very important for both Report Server and Portal. Report Server needs RAM for the following:

- Fetch and process data set

- For parsing the report template information

- Prepare and process report pages

Intellicus Portal needs RAM for activities like

- To store details of logged in users

- To store session and object information

Small: The recommended amount of RAM allocated to Intellicus is 1GB (512 for report server +512 for web server).

Medium: The recommended amount of RAM allocated to Intellicus is 2GB (1GB for report server +1GB for web server).

Large: The recommended amount of RAM allocated to Intellicus is 4GB or more depending on scaling requirements (2GB for report server +2GB for web server).

> **64-Bit Note:** For large installs, to allocate RAM heap larger than 1 GB per process, you can run Intellicus 64-bit report server and web servers.

Disk Space

Intellicus needs disk space for Report Server, Portal and Windows Studio.

Report Server

Running report server is disk intensive because it does disk-IO for operations like:

Caching of meta-data

Intellicus connects to a database using its data connection. Meta-data of all the data connections is cached at the time or report server start.

Storing of record sets

Actual report data fetched from database is cached on report server. This improves performance since second time onward, data for the report comes from report server itself and not from database server.

Report Pages

Before being made available to the user, report pages are stored on report server. When report is published, it is stored on report server. Report server also may need to create temporary files when report view operation is requested in formats like PDF. Depending on the nature of the published report, it may continue to be on the disk, or may be deleted after the validity period is over.

Archive files

Published files can be archived and stored on report server. When an archive file is created, report server takes all the published files and creates a file that can be kept on report server itself or can be stored away from the server.

When an archive file is restored, all the contents of that archive are again stored on report server.

Logs

Logs are created on local disk. Increase in the log file size depends on the log level. Log level can be

- FATAL

- ERROR

- WARN

- INFO

- DEBUG

Log level "FATAL" will log only when report server crashes. Log level DEBUG every action taken in report server and so log file size will increase rapidly.

Portal

Intellicus portal will carry out disk IO for following activities:

Caching of meta-data

When SQL Editor is opened, portal requests and stores meta-data information of the database. This information is stored on local disk.

Report HTML files

When a user requests for a report view in HTML, HTML pages coming from report server are stored on portal's local hard disk.

Log files

Portal logs are created and saved on local disk. Its size generally remains within KBs. However, increase in log file size depends on log level as explained in the above topic.

Windows Studio

Windows Studio is used to design and store reports (report templates that will be used to get actual report having data). It needs disk space for:

IRL Files

Reports designed in Windows Studio are stored on hard disk as IRL files. IRL file size generally goes in KBs.

If an IRL has an embedded image, Windows Studio will convert it into a bitmap and embed it inside the report file. This will significantly increase the file size. In case it can go in terms of MBs depending on the image size.

Log files

Windows Studio logs are created and saved on local disk. Its size generally remains within KBs. However, increase in log file size depends on log level as explained in topic above.