

# SOA Gateway Statistics

This section describes the statistics available in the SOA Gateway.

The existing functionality helps determine the level of work flow activity through a server system. The infrastructure is also in place to enable other forms of statistics gathering such as performance monitoring, number of users, driver specific data and so on.

Work flow monitoring is set off by default. When on it will provide Key Performance Indicators (KPIs) on well know operations list, get, add, delete, update, invoke, select, selectCount, selectNext and selectEnd.

The KPIs gathered for each operation are as follows:

- total\_count

This is the number of times the operation has been called. If successful then the round trip time is added to total\_time. If applicable high\_time or low\_time may be updated. If the operation is unsuccessful then errors\_occurred is incremented.

- total\_time

This is the accumulated round trip time for the operation.

- high\_time

This is the highest round trip time for the operation.

- low\_time

This is the lowest round trip time for the operation.

- errors\_occurred

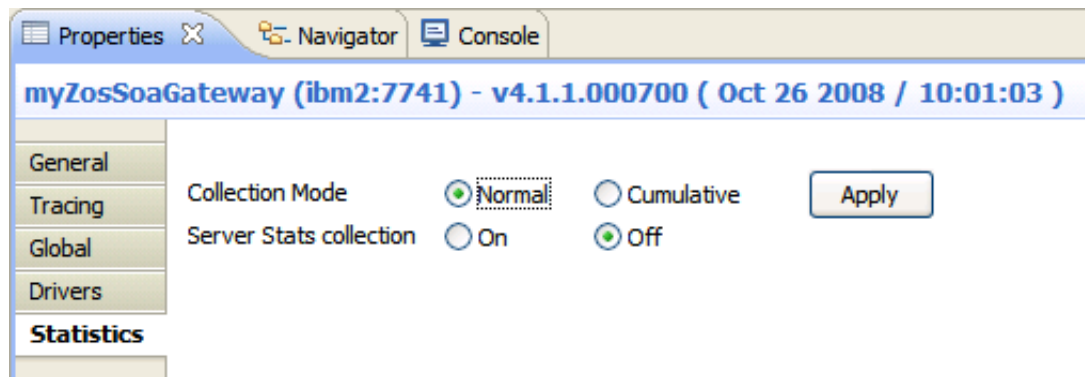
This is the number of times the operation has failed.

---

## Setup

Setup is carried out using the SOA Gateway Control Centre.

Click on the server to be monitored in the list of SOA Gateway Servers. Then click on the Statistics tab within the Properties View as shown below:



Statistics can be collected in 2 modes, Normal and Cumulative.

## Normal

Statistics are gathered from when the SOA Gateway Server is started to when it is stopped. On stopping the server a file is written with the KPIs at that point. The name of the file has the format `soag_normal_statistics_yyyy_mm_dd_hh_mm_ss_millisecc.xml` e.g.

`soag_normal_statistics_2008_10_09_12_56_11_364.xml`

## Cumulative

Statistics gathering continues over server stop/starts until the mode is set to Normal. When the server is stopped the data is written to the file `soag_cumulative_statistics.xml`. On restart the same file is read and used to initialise the statistics.

## Monitoring

Statistics monitoring can be carried out on an ad hoc basis, again using the SOA Gateway Control Centre.

Open/click on the SOA Gateway Statistics View:

Type	Name	Requests	Errors	Total time	Time high	Time low
Request	get	9	0	0.125	0.061	0.007
Request	add	1112	0	22.562	1.267	0.008
Request	delete	1113	0	20.038	0.935	0.006
Request	update	1	0	0.205	0.205	0.205
Request	invoke	0	0	0.000	0.000	0.000
Request	select	2	0	0.183	0.164	0.020
Request	selectCount	0	0	0.000	0.000	0.000
Request	selectNext	3	0	0.006	0.003	0.001

Pick the server from the Server listbox. The Name can be wildcarded or set to an individual method. Then click on the Show Statistics button to refresh the data.

The statistics can be reset at any time by selecting the Reset Statistics button.

## Reports

The format of the normal and cumulative report is identical. Here is an extract from a normal file:

```

<Statistics>.
  <Group id="Server">.
    <Type id="Request">.
      <Name id="list">.
        <total_count>518</total_count>.
        <errors_ocurred>1</errors_ocurred>.
        <total_time>67.990</total_time>.
        <high_time>43.977</high_time>.
        <low_time>0.006</low_time>.
      </Name>.
      <Name id="get">.
        <total_count>9</total_count>.
        <errors_ocurred>0</errors_ocurred>.
        <total_time>0.125</total_time>.
        <high_time>0.061</high_time>.
        <low_time>0.007</low_time>.
      </Name>.
      <Name id="add">.
        <total_count>1112</total_count>.
        <errors_ocurred>0</errors_ocurred>.
        <total_time>22.562</total_time>.
        <high_time>1.267</high_time>.
        <low_time>0.008</low_time>.
      </Name>.
      <Name id="delete">.
        <total_count>1113</total_count>.
        <errors_ocurred>0</errors_ocurred>.
        <total_time>20.038</total_time>.
        <high_time>0.935</high_time>.
        <low_time>0.006</low_time>.
      </Name>.
    </Type>.
  </Group>.
</Statistics>.

```

The statistic reports are written to the configuration folder where the SOA Gateway is installed.

## XML

The statistics are reported in XML and there are many programs which can consume this format e.g. here is the result of importing the above report into Excel:

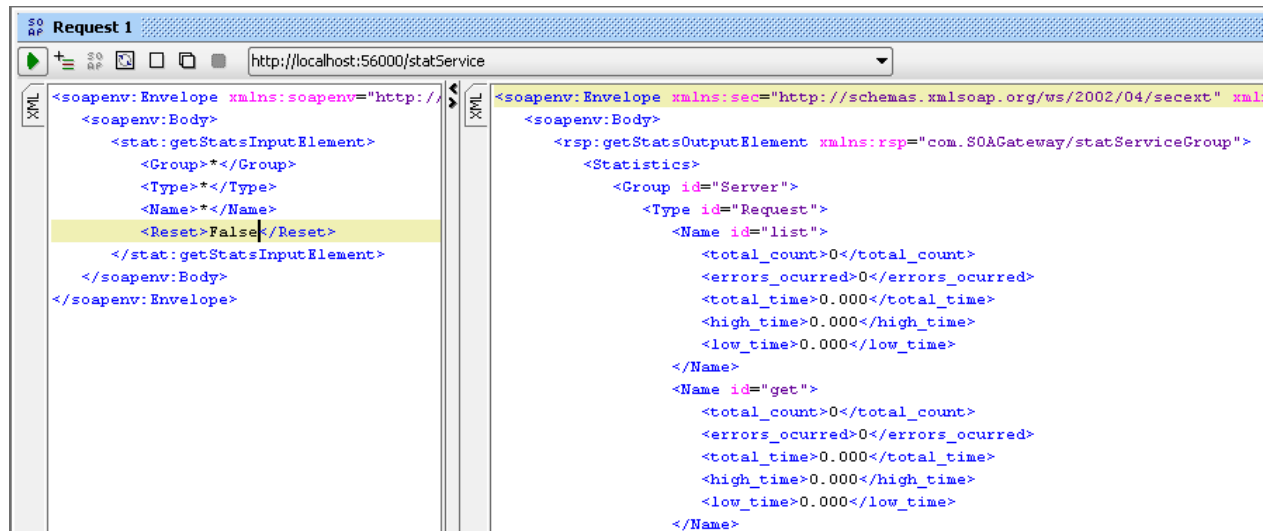
id	id2	id3	total count	errors occurred	total time	high time	low time
Server	Request	list	518	1	67.99	43.977	0.006
Server	Request	get	9	0	0.125	0.061	0.007
Server	Request	add	1112	0	22.562	1.267	0.008
Server	Request	delete	1113	0	20.038	0.935	0.006
Server	Request	update	1	0	0.205	0.205	0.205
Server	Request	invoke	0	0	0	0	0
Server	Request	select	2	0	0.183	0.164	0.02
Server	Request	selectCount	0	0	0	0	0
Server	Request	selectNext	3	0	0.006	0.003	0.001
Server	Request	selectEnd	1	0	0.001	0.001	0.001

## WSDL

Access to statistics is also made available via the statService web service as described by the WSDL <http://server:port/statService?WSDL>.

## SOAP

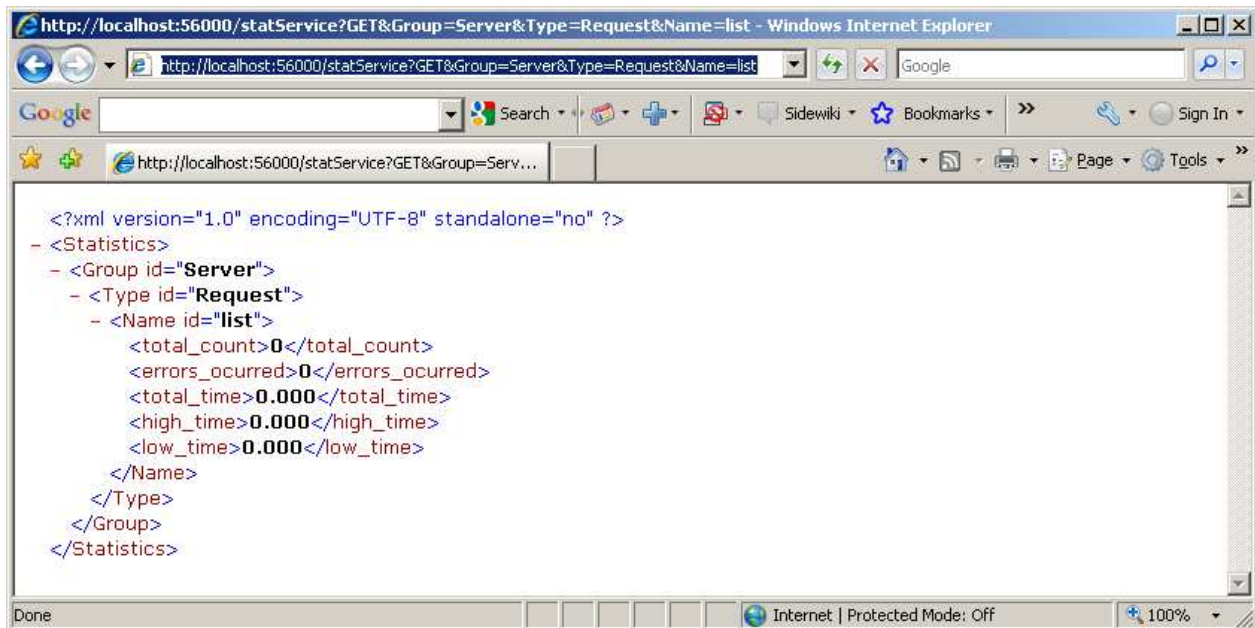
Statistics can be retrieved via a SOAP request using the statService web service. Using a WSDL consumer such as soapUI, as shown below, a request can be made using Group, Type and Name. As with the Control Centre, the '\*' construct enables the entire set of statistics to be returned or a subset. The Reset option can be set to True or False as required. The default value is False if omitted.



## URL

The statistics can also be retrieved via a URL-based request in a browser such as Internet Explorer. The statService service exposes one method, GET. As with previous requests this can be restricted by Group, Type and Name. Again the '\*' construct enables the entire set of statistics to be returned or a subset by specifying a value. The option is also provided to reset the current set of statistics. The example below restricts the search to the list method.

<http://localhost:56000/statService?GET&Group=Server&Type=Request&Name=list>



## Example

Let us assume that the server has been started for the first time and that statistics collection has been set on for the first time. An Adabas database is available and a web service created to access it. Both queries below are issued via a browser such as IE. The LIST successfully retrieves the data for employees whose ids start with 200. The GET query fails because the syntax is incorrect.

`http://localhost:56005/Adabas_Driver_1_9?GET=20020000`

`http://localhost:56005/Adabas_Driver_1_9?LIST&AA=200*`

Now get the statistics so far by issuing a GET on statService as shown below. Note that the entry for the get request does not show any effect on the times. Note that for list, for the moment, all times are identical. The time is 0.203 or 203 milliseconds. This is the round trip time spent in the SOA Gateway Server.

```

<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <Statistics>
- <Group id="Server">
- <Type id="Request">
- <Name id="list">
  <total_count>1</total_count>
  <errors_ocurred>0</errors_ocurred>
  <total_time>0.203</total_time>
  <high_time>0.203</high_time>
  <low_time>0.203</low_time>
</Name>
- <Name id="get">
  <total_count>1</total_count>
  <errors_ocurred>1</errors_ocurred>
  <total_time>0.000</total_time>
  <high_time>0.000</high_time>
  <low_time>0.000</low_time>
</Name>
- <Name id="add">
  <total_count>0</total_count>
  <errors_ocurred>0</errors_ocurred>
  <total_time>0.000</total_time>
  <high_time>0.000</high_time>
  <low_time>0.000</low_time>
</Name>
- <Name id="delete">
  <total_count>0</total_count>
  <errors_ocurred>0</errors_ocurred>

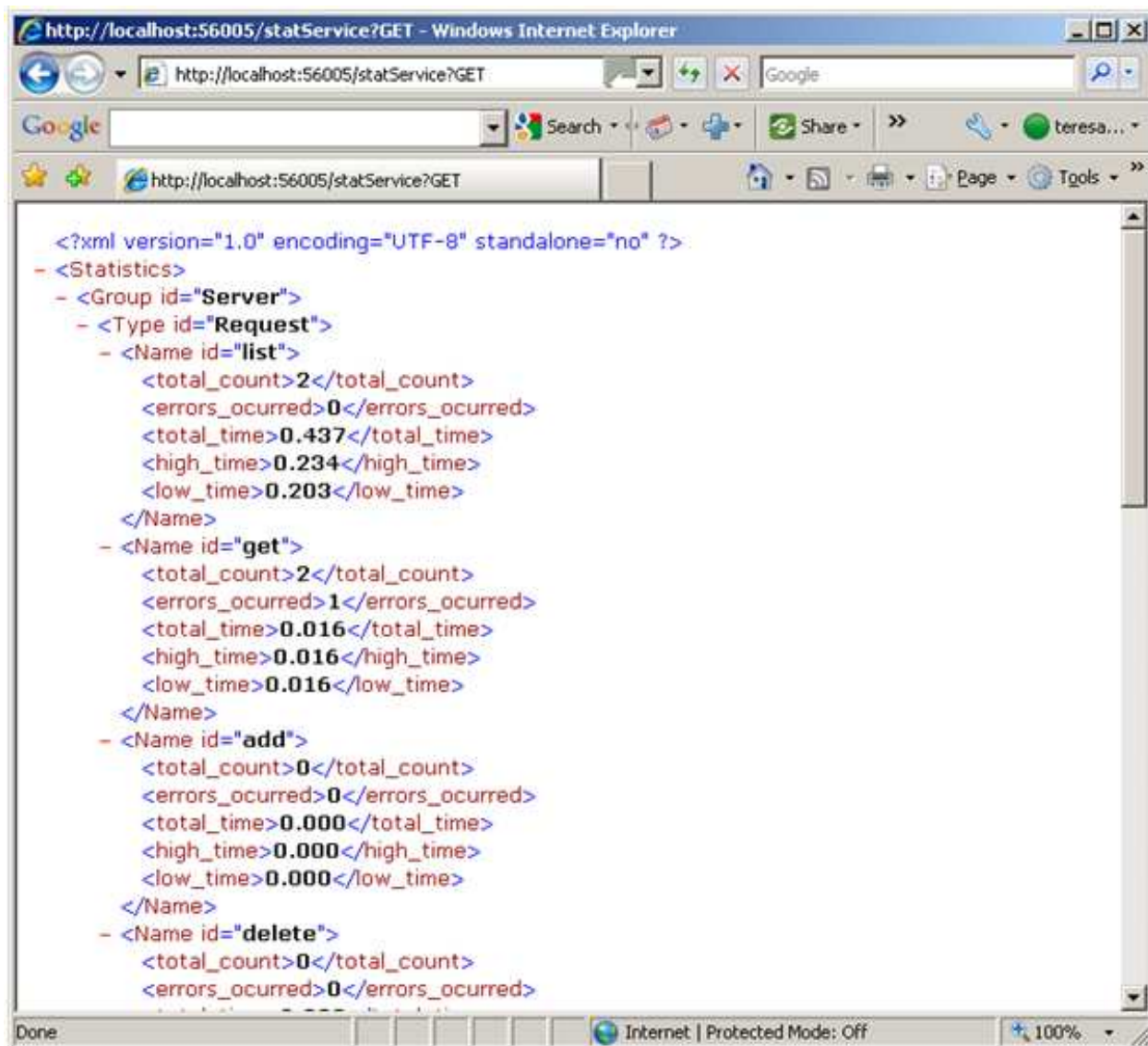
```

Let us issue the LIST and GET again, this time correcting the GET syntax and changing the LIST to retrieve ids starting with 20:

[http://localhost:56005/Adabas\\_Driver\\_1\\_9?GET&AA=20020000](http://localhost:56005/Adabas_Driver_1_9?GET&AA=20020000)

[http://localhost:56005/Adabas\\_Driver\\_1\\_9?LIST&&AA=20\\*](http://localhost:56005/Adabas_Driver_1_9?LIST&&AA=20*)

Note the now successful entry for the get operation. Note also that for list the high time is now 0.234 as the query took slightly longer. As there have been only 2 list requests so far the total time equals the sum of low and high times.



```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <Statistics>
- <Group id="Server">
- <Type id="Request">
- <Name id="list">
  <total_count>2</total_count>
  <errors_ocurred>0</errors_ocurred>
  <total_time>0.437</total_time>
  <high_time>0.234</high_time>
  <low_time>0.203</low_time>
</Name>
- <Name id="get">
  <total_count>2</total_count>
  <errors_ocurred>1</errors_ocurred>
  <total_time>0.016</total_time>
  <high_time>0.016</high_time>
  <low_time>0.016</low_time>
</Name>
- <Name id="add">
  <total_count>0</total_count>
  <errors_ocurred>0</errors_ocurred>
  <total_time>0.000</total_time>
  <high_time>0.000</high_time>
  <low_time>0.000</low_time>
</Name>
- <Name id="delete">
  <total_count>0</total_count>
  <errors_ocurred>0</errors_ocurred>
```