The statistics used are obtained from a CTG gateway via the getStatsByStatId() call which returns a set of statistic name/value pairs.  For the gateway as a whole (GD) and for various sub-categories such as CICS connections (CS) and web services (WS) a request count and average response time are provided.   There are two varieties for these statistics distinguished by a naming convention: L statistics are cumulative for the day and I statistics apply to a customer specified interval.  For the gateway as a whole, the statistics used are GD\_LALLREQ and GD\_LAVGRESP.

The problem is that neither the current day statistics nor the interval statistics fit the time interval that we want to collect the metrics.  This mismatch occurs in two contexts:

* Users may seek reporting for the whole on a specific regular time interval, such as an hour interval.  This creates a coordination issue even when the interval is the same as the CTG statistics interval because the statistics are obtained via a specific request.  Interval totals are produced but are only written to SMF.  They are not available to an external monitor which may be waiting for them.  Intervals that do not match the hard wired CTG interval must be deduced from the statistics provided at the interval start and end.
* Users may seek reporting on smaller intervals in order to gauge shorter term effects of external events.  In order to do this, they want to compare statistics from the time they first look to the statistics obtained at a later time.  The point is that the user needs to be able to determine the times.  The current statistics are not very usable for such comparisons.

We would like to be able to accommodate a variety of reporting intervals by performing arithmetic to fit the data to the desired time intervals.  The problem with this approach derives from the fact that the average response times, reported in milliseconds, are not granular enough to compute a usable total response.  As computers get faster and response times decrease to fewer and fewer milliseconds, the relative error introduced in multiplying average response by number of requests has increased.

Two changes to CTG statistics reporting have been identified that could resolve this issue:

* A new set of statistics reporting "total response time" could eliminate the uncertainty inherent in a multiplication.  This appears to be the least disruptive change to the CTG statistics reporting.
* The granularity of the average response time statistics could be increased from milliseconds to TOD clock units.  This would enable much more precise computation, but would present the data in units much further removed from observed response times.