

Computation of Service Availability Metrics in Gridview

Digamber Sonvane, Rajesh Kalmady, Phool Chand, Kislay Bhatt, Kumar Vaibhav
Computer Division, BARC, India

| Version | Date | Comments |
|---------|------------|--|
| 0.4 | 27/08/2008 | Removed all references to old algorithm and added more details. |
| 0.3 | 24/06/2008 | Added a clarification that Reliability is undefined (Not Applicable) for a period when status is Scheduled Down and/or Unknown for the entire period. |
| 0.2 | 15/11/2007 | Introduced a new status value 'DONTCARE', when no critical test is defined for a service. For Availability computation 'DONTCARE' is considered as good as 'UP'. Change made in Section 5.1, 5.2, 5.3 and 5.4. Added Section 5.5 |
| 0.1 | 11/09/2007 | First Draft |

Gridview's service availability module computes status, availability and reliability metrics of grid entities such as sites, services and individual service instances. In addition to the above, it also computes aggregate availabilities such as for all Tier 1/0 sites and central services such as FTS, RB, LFC and so on. These computed metrics are displayed in the Gridview frontend in the form of graphs and charts, with the ability for a user to drill-down from the availability of a site to individual test results that contributed to the computed figure.

1. Terms and Definitions:

This section describes certain terms which occur frequently in this document.

- a) **Service Instance:** A service instance is a single resource in the grid, such as a particular compute element or a storage element in some site. It should be noted that the service instance refers to the service endpoint rather than to a physical machine - it is quite possible that the same physical server run multiple service instances.
- b) **Service:** A service is a set of similar service instances. For example, multiple Computing Elements in a site together make up the CE service for the site. A service can be made up of one or more service instances.
- c) **Site:** A site in the grid is a collection of several services such as CE, SE, SRM and so on. A site can be made up of one or more services. For example, the CERN site consists of CE, SE, SRM and sBDII services.
- d) **Site services and central services:** Site services are those services that are deemed to be run by a site for the site's use. Examples of site services are CE, SE, SRM and sBDII. On the other hand, central services are deemed to be independent to a site in the sense that they could be used by other sites too. Examples of central services are FTS, LFC, RB and BDII.
- e) **Status:** Status of a service instance, service or a site is the status of that entity at a given point in time. Possible status values are
 - 'UP' : The service instance, service or site is working
 - 'DOWN' : It is not working
 - 'SCHEDULED DOWN' : It is under maintenance
 - 'UNKNOWN' : The status cannot be computed because test results are not available
 - 'DONTCARE' : The status need not be computed (see 3.4)
 - 'DEGRADED': One or more (but not all) redundant service instances of a service are not working.

2. Availability and Reliability

Availability: Availability of a service instance, service or a site over a given period is defined as the fraction of time the same was UP during the given period.

Reliability: Reliability of a service instance, service or a site over a given period is defined as the fraction of time the same was UP (Availability), divided by the scheduled availability (1 – Scheduled Downtime – Unknown Interval) for that period.

$$\begin{aligned}\text{Reliability} &= \text{Availability} / \text{Scheduled Availability} \\ &= \text{Availability} / (1 - \text{Scheduled Downtime} - \text{Unknown})\end{aligned}$$

where

Availability is the fraction of the time the service was UP (Tests Passed)

Scheduled Downtime is the fraction of the time the service was Scheduled Down

Unknown Interval is the fraction of the time for which the Test Results were not known (that is, validity of earlier test result expired and new test result did not arrive yet).

The concept of availability and reliability can be illustrated in the following example:

In a particular hour, a service instance was under maintenance (Scheduled Down) for 15 minutes, with ‘unknown’ status for 15 minutes, ‘UP’ for 15 minutes and ‘DOWN’ for 15 minutes. Then for the above service:

$$\text{Availability} = (\text{UP time})/(\text{total time}) = 15/60 = 25\%$$

$$\begin{aligned}\text{Reliability} &= (\text{UP time})/(\text{total}-\text{scheduled_down}-\text{unknown}) = 15/(60-15-15) \\ &= 50 \%\end{aligned}$$

In other words, reliability is the ratio of the UP time of a service to the time the service was scheduled to be UP. For computing the amount of time the service was scheduled to be UP, the time for scheduled downtime and unknown period are deducted from the total time.

It follows from this that if the status of a service instance, service or a site is Scheduled Down and/or Unknown for the entire period (Scheduled Downtime + Unknown = 1), then Reliability is undefined (Not Applicable) for that period. For example, in a particular hour, if the service is scheduled down for 30 minutes and of unknown status for 30 minutes,

$$\text{Availability} = 0/60 = 0\%$$

$$\text{Reliability} = 0/(60-30-30) = 0/0 \text{ (undefined)}$$

Availability and Reliability figures can be computed for different periodicities such as hourly, daily, weekly and monthly.

2. Contributing factors to availability computation:

The following is a list of different factors which contribute to the computation of the availability and reliability of service instances, services and sites.

- a. SAM test results:

The SAM[1] framework is responsible for scheduling and submitting tests on individual resources of the grid. The results of these tests are gathered by the SAM submission framework and archived in the SAM/Gridview database.

b. Validity of test results

Each SAM test result is deemed to be valid for a specific period of time after which the result is invalidated. SAM tests will have to be submitted frequently enough so that a fresh result is available before the earlier result gets invalidated. By default, each SAM test result is deemed to be valid for a VO specific timeout period.

c. Critical tests for services

Each VO defines a set of critical tests for every service which supports the VO. This is defined by the VO admin through the FCR (Freedom of Choice for Resources)[0] interface. Only those tests that are deemed to be critical by the VO are considered for availability computation.

d. Node Downtimes

Every site reports scheduled/unscheduled downtimes for individual service instances or services in the GOCDB[3] interface. These downtime periods are taken into account by the Gridview algorithm when computing reliability metrics. Test results during periods of downtimes are not considered for computing status.

e. Service Scope : Site Service or Central Service

Services in the grid are classified as site services and central services. Site services include services such as CE, SE, SRM and Site BDII which are run by sites. Central services include services such as LFC, FTS, RB which are run centrally and used by users from other sites too. The availability of a site is derived from the availabilities of site services only.

f. Service instances of a site

Sites may contain multiple instances of a service, e.g. multiple CEs and SEs. The availability of a particular service in a site is derived from the availability figures of individual service instances.

g. VOs which the site service instances support

Each service instance on the grid is defined to support one or more VOs; this information is published in the BDII, and is used to compute VO-specific availability information for every service instance in the grid.

3. Features of the algorithm for Computation of Service Availability Metrics

The algorithm for computation of service availability metrics is implemented in the Gridview service availability summarizer. This algorithm has the following major features:

3.1 Computation of status on a continuous time scale

The status of every service instance in every site in the grid is computed on a continuous time scale for each supported VO using SAM test results. Each VO defines a set of critical tests for a service, which means that every critical test needs to run

successfully on the service instance in order to consider its status as UP. Test results are deemed to be valid for a specific duration defined by the VO. The VOs supported by a service instance is published in the site's BDII.

The status of individual services (service types) such as CE and SE are computed by combining the statuses of all redundant service instances in an operation akin to a logical OR. The status of a site is computed by logically ANDing the status of the site services run by the site.

3.2 Handling of downtime information:

Scheduled downtimes of services and sites, which are published in GOCDB are considered while computing the service status and availability. The status of a service instance, for which a downtime is published in the GOCDB, is explicitly marked as SCHEDULED DOWN and any SAM test results during this period are ignored. In fact, this is a pre-condition to compute reliability as per the approved definition. Also, if one of the critical services of the site is scheduled down, the entire site is considered scheduled down. If scheduled downtime extends or shrinks, the site administrator can update it in GOCDB and the updated interval will be considered, provided that the change is made before the interval has elapsed.

3.3 'Unknown' status and how to handle it:

As described in section 3.1, SAM test results are taken into account to compute the status of a service instance. This will result in the status being computed as 'UP' if all tests are successful, 'DOWN' if any test has failed or 'SCHEDULED DOWN' if the downtime is published in the GOCDB. What if the test results are unavailable? This may happen due to problems in the SAM framework or due to tests not being submitted to a service instance before the earlier test result expires. In such cases, the algorithm sets the status of the service instance to 'UNKNOWN'. The status of a service and a site could also be unknown for similar reasons.

3.4 'DONTCARE' status:

There are services for which certain VOs have not defined any critical tests. Such services are considered as having status 'DONTCARE', meaning that the status of the service is not to be computed. The DONTCARE status is similar to the UP status in availability calculations.

4. Metrics being computed

The following metrics are being computed:

1. Individual Service Instance Status, Availability and Reliability
2. Individual Service Status, Availability and Reliability
3. Individual Site Status, Availability and Reliability
4. Aggregate Tier-1/0 Availability
5. Central Services Status, Availability and Reliability

5. Computation of Service Status, Availability and Reliability Metrics

The status of service instances, services and sites (service instance status, service status and site status) is computed on a continuous time scale for each supported VO using SAM test results. Availability metrics for service instances, services and sites are computed on an hourly basis for each supported VO. These hourly metrics are later used to generate metrics for different periodicities such as daily and monthly.

5.1 Computation of Continuous Service Instance Status: ‘ServiceInstanceStatus’

This metric indicates the status of a particular instance of a service, for example, a specific CE, over a period of time. This status is determined by using results of all the critical SAM tests that were run on the particular service instance. The criticality of SAM tests and validity period of the result is defined by each VO. Site Administrators can announce the Scheduled Downtimes for service instances in GOCDB. From all these input parameters, five possible status values are derived for the service instance.

- **SCHEDULED DOWN:** The service has been marked as under scheduled downtime. Test results are ignored in this case.
- **DONTCARE:** No critical test is defined for the service
- **UP:** All critical tests have passed
- **DOWN:** At least one critical test has failed
- **UNKNOWN:** No critical test is failed and result of at least one critical test is not available

5.2 Computation of Continuous Service Status: ‘ServiceStatus’

This metric is computed by combining status of all instances of a particular service type (computed in 5.1), for example, all individual CEs in a site, at a given point in time. Here six possible status values are derived.

- **UP:** All instances of the service are UP
- **DEGRADED:** One or more (but not all) instances are UP
- **DONTCARE:** No critical test is defined for the service
- **SCHEDULED DOWN:** No service instance is UP or DONTCARE and at least one is ‘SCHEDULED DOWN’
- **DOWN:** No instance is ‘UP’ or ‘SCHEDULED DOWN’ or ‘DONTCARE’ and at least one is DOWN
- **UNKNOWN:** All instances are of ‘UNKNOWN’ status.

For computing ‘ServiceStatus’ of site level services, only instances belonging to a particular site are considered, whereas for computing ‘ServiceStatus’ of central services, all instances of the service (irrespective of site) are considered.

5.3 Computation of Continuous Site Status: ‘SiteStatus’

This metric is computed by combining the status of all site level services (computed in 5.2) provided by the site (that is, registered by the site in the Information System, either GOCDB or BDII), viz, CE, SE, SRM and sBDII, at a given point in time. The following six status values are derived.

- DOWN: At least one service is DOWN.
- SCHEDULED DOWN: No service is DOWN and at least one is 'SCHEDULED DOWN'
- UNKNOWN: No service is DOWN or 'SCHEDULED DOWN' and at least one is 'UNKNOWN'
- DEGRADED: No service is DOWN, 'SCHEDULED DOWN' or UNKNOWN and at least one service is DEGRADED
- UP: All services are UP.
- DONTCARE: All services are DONTCARE.

5.4 Hourly Availability and Reliability Computation

- The Hourly Availability (status UP, DONTCARE and DEGRADED), Scheduled Downtime and Unknown Interval for a service instance, a service and a site are computed on an hourly basis from the respective status information for that hour.
- The Hourly Reliability of a service instance, a service and a site is computed on an hourly basis from the Availability, Scheduled Downtime and Unknown Interval for that hour.

5.5 Computation of availability and reliability for higher periodicities

- The Daily, Weekly and Monthly Availability, Scheduled Downtime and Unknown Interval figures are computed from the corresponding Hourly figures by averaging over the required time periods.
- The Daily, Weekly and Monthly Reliability figures are computed directly from the Availability, Scheduled Downtime and Unknown Interval figures over the corresponding periods and **not** by averaging the Hourly Reliability figures.

This may look strange but it can be easily seen from the example below that it is not possible to obtain reliability figures for higher periodicities by averaging hourly reliabilities.

Consider a service which has the following figures for a day of 24 hours:

- Hours 00-12 : 100 % UP, 0% down, 0% scheduled down in each hour
 - o For these 12 hours, hourly availability = 100 %, hourly reliability = 100%
- Hours 13-24 : 0% UP, 10% down, 90% scheduled down in each hour
 - o For these 12 hours, hourly availability = 0%, hourly reliability = 0%

It can be seen that the service was up for 12 hours, down for 1.2 hours and scheduled down for 10.8 hours in the whole period of 24 hours.

Daily availability for the service for this day = $12/24 = 50 \%$

Daily reliability for the service for this day = $12 / (24 - 10.8) = 12/13.2 = 91\%$

If we compute daily reliability by averaging the hourly reliabilities for the 24 hours, we will obtain a daily reliability figure of $(100*12 + 0*12) / 24 = 50 \%$!

6. References:

1. SAM Framework <https://twiki.cern.ch/twiki/bin/view/LCG/SAMOverview>
2. Freedom of Choice for Resources (FCR):
<https://lcg-fcr.cern.ch:8443/fcr/help.cgi?user=1>
3. GOCDB Topology Database: <https://goc.gridops.org/>