



Vocabulary

The following alphabetized list comprises commonly used Grid technology terms either developed by GridSystems or used within the Fura product and are explained according to this framework. In other words, these explanations should be interpreted as descriptive and operational definitions within our technology, not as holistic descriptions:

- **Advanced Distributed Execution:** this Service is included to guarantee backward compatibility with our previous version, allowing those applications that had been grid-enabled for InnerGrid V, as well as IG V Connectors, will run on Fura.
- **Agent:** a Grid Agent is the basic type of grid computing resource; it is lightweight and is usually installed in standard PCs. It receives the Grid commands from the Server, it executes them, and it sends back the results.
- **Agent Base:** for compatibility purposes, this Service is the interface that guarantees that the calls IG V Connectors make to Agents will work on Fura.
- **Agent Broker:** this Grid Service is responsible for facilitating and managing the connection between Agents and the Server.
- **Allocation Mode:** allocation modes help divide and distribute the task assignments to the resources in the Grid. This mode specifies which group or groups will work on a task. A *Universal* mode means that a task is not assigned to one group in particular and can be executed by resources of any group that accepts tasks assigned to “others”. *Priority* mode means that a task is assigned to one or more groups but may be executed by other resources for as long as they can accept tasks. And *Exclusive* mode means that a task will only be executed by the resources of one or more specific groups.
- **Basic Ranges:** this Service allows an automatic splitting of tasks in smaller microtasks, which will become the pieces of computation sent to the computing resources.
- **Computing Resource Manager:** this Service registers and manages the computing resources present in the grid.
- **Exit codes:** defined as numeric ranges, they represent the possible values with which an execution may end.
- **File & User System:** This Grid Service provides Authorization, Basic Policies, and an Internal File System. The File System provides a mean for keeping the state of the Grid Services that depend on the File & User System, or of the clients of these Grid Services (including clients of the File & User System). This is analogous to the File System’s role of an operating system, which offers a service for storing data to both the remaining modules of the operating system and the software applications running on the operating system. This Grid Service is one of the fundamental keystones to use Fura for robust Data Grid projects and applications.
- **Filters:** Computing resource platforms details, configuration specifications, and keywords are recorded by the Server and can be queried later in order to select just a subset of the existing resources. The querying conditions are known as filters.
- **Groups:** A Group represents a set of Grid resources that are to be managed together from the point of view of policy management. Working with groups allows controlling task execution because tasks are assigned to groups (all, some, or one group) to be executed. One resource can belong to several groups for as



long as the timeframes the groups are active do not overlap; one resource cannot belong to two different groups at exactly the same time.

- **Iteration:** The parametric execution implies an iterative execution through a range of values for some application parameters. Some applications are able to traverse a range of values during one execution while others can only deal with one value per execution.
- **Keywords:** It is a manual filtering mechanism. A keyword serves as a manual flag to distinguish a resource by something that usually falls outside of the characteristics of their platform or grid configuration. This keyword can be about anything the user may consider necessary, such as the owner, the location, the software installed, etc.
- **LRP (Long Running Process):** Applications that take a long time to boot or start-up when contrasted with the time they spend executing work.
- **Messaging:** This Grid Service is a mailing service that can be used by other Grid Services. Currently, messaging is implemented so that users are notified via e-mail when a task finishes and when a scheduled task begins its execution. It works as a reporting mechanism.
- **Microtasks:** Each of the independent runs into which a task can be divided is called a microtask.
- **Modules:** A program that has been Grid-enabled is a Module. In other words, a module is the set of instructions (or piece of code) put into a Grid Server to distribute. Parameters and ranges definitions, input formats, file templates, file filters, and exit codes are among the metadata contained in a module. Modules also describe where all the required files are, including the code binaries (per-platform), launchers, templates, and input files. Modules do not imply any action by themselves; they are just metadata and binaries.
- **Module Manager:** it allows the grid-enabling of applications by providing the Grid with the metadata necessary for the execution of a certain application (input parameters, input files, output, exit codes, meaning of the parameters, invocation on different operating systems, etc.)
- **Monitoring and Metering:** This Service stores different average values and summaries regarding the Grid usage, and can be used to develop billing systems.
- **Parameters:** Parameters are individual input values.
- **Platforms:** Platforms classify grid computing resources based on the computer types on which they run. The division is based on distinctions in the Operating system, CPU architecture, and keyword. The basic platforms (differentiated by the OS type) can be fine-tuned into many more to better fit user requirements.
- **Portal:** It is the web-based graphical interface to the Grid Server functionalities and objects.
- **Post Processing:** It is the processing or manipulation of the result files obtained by the computing resource after the normal execution of a task.
- **Priority:** It is the weight a running task will have relative to other tasks running at the same time. Standard priorities range from *Very High* to *Very Low*. There are two special priorities: *Emergency*, which hardwires all the priority mechanisms to get a task done as fast as possible, and *Background*, to run tasks only when no other tasks could use the resources.



- **Ranges:** Ranges are an input-value variation that covers an interval. There are three types of ranges: BigDecimal (for decimal-base numbers with arbitrary precision and increments); BigInteger (for integer numbers in base 2 (binary), 8 (octal), 10 (decimal), or 16 (hexadecimal) with arbitrary increments); and File (when a file is processed in each iteration).
- **Object Access Control:** Access to dynamic Grid object (modules, tasks, and groups) is controlled by a security mechanism. Only the creator of the object has all the possible permissions (owner, read, modify, and delete) granted. In the case of groups, the permissions are two: owner and execution. The creator must grant all other users and/or roles one or more of the permissions if they need to access that object.
- **Partition policy:** each iteration range has an iteration or partition policy. A module can combine different ranges with different iteration policies. There are three types of policies: **one by one** (a single value is processed in each microtask execution); **iterated in resource** (each resource receives a collection of values and the information needed to perform the iterations); and **iterated in application** (the executable file can treat the given collection of values within a single invocation).
- **Resource Groups:** For compatibility purposes, this Grid Service guarantees that those calls IG V grid-enabled applications, as well as IG V Connectors, made to groups of resources will work in Fura.
- **Restricted Usage:** A resource is set to be restricted if the timeframe when it can execute grid tasks is restricted to those hours when the group/s it belongs to is active. It is unrestricted if it can accept Grid work all the time, having priority the work assigned to its group/s.
- **Roles:** Roles define, limit, and group users' access to resources and web services.
- **Server:** A Grid Server is the central piece of software in the Grid computing. The Server coordinates, controls, distributes, recollects, and verifies all the Grid work.
- **Subordinate Server Broker:** this Grid Service is responsible for managing the communications and work assignments between a main grid Server and a subordinate Grid Server (those with Server Executor installed).
- **Server Clustering:** This Service supports the functionalities necessary for a multi-Server Grid environment with a master Server and back-up Servers, which allows full tolerance and the guaranteed continuation of the activities in the Grid when a Server is lost for whatever reason.
- **Subordinate Server Executor:** this Grid Service converts a computing resource where it is installed as a subordinate or secondary Grid Server that can receive microtask assignments from another Grid Server (for as long as it has the Subordinate Server Broker installed) and assign it, in turn, to the computing resources that depend on it.
- **Service Provisioning:** This Service can be used to start and monitor LRP applications. In case of an application failure or malfunction, this service would re-launch the application automatically. The provisioning mechanism can be put in place through Agent keywords.
- **SSL:** Security Socket Layer protocol is used to encrypt and secure communications at the transport level between the Server and Portal, the Server and Agents, or between the Portal and browser.



- **Tasks:** Tasks are modules taken into action; tasks are a way to invoke modules. Additionally, tasks determine the execution parameters, which machines will take part in the process, whether there will be memory or CPU limitations, etc.
- **Task Submission:** it allows the creation, edition, and execution of tasks based on those grid-enabled applications defined by the Module Manager. Part of this component is in charge of dispatching the computable items (microtasks) to the computing resources. It is also able to handle caching requirements and monitoring executions in order to assure a proper quality of service.
- **Users:** Access to the Grid functionalities and objects is only granted with a valid username and password, which is in no way related to the username and password of the underlying operating system.