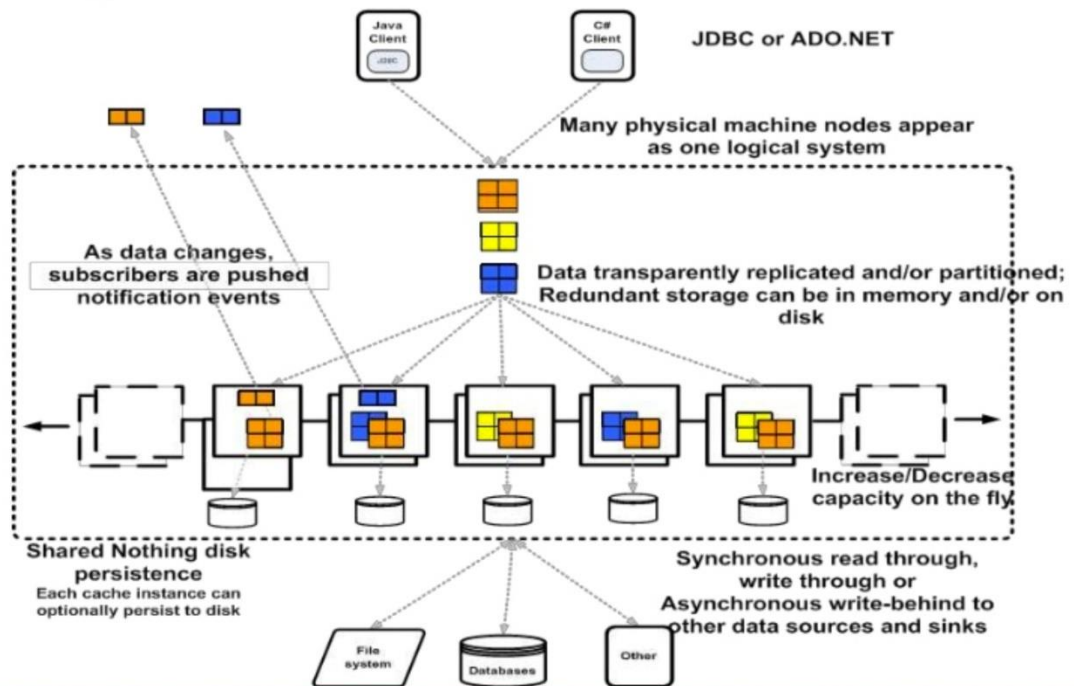


Apache Geode

Geode High Level Architecture



Locator:

The locator is a GemFire process that tells new, connecting members where running members are located and provides load balancing for server use.

You can run locators as peer locators, server locators, or both:

- Peer locators give joining members connection information to members already running in the locator's cluster.
- Server locators give clients connection information to servers running in the locator's cluster. Server locators also monitor server load and send clients to the least-loaded servers.

By default, locators run as peer and server locators.

You can run the locator standalone or embedded within another GemFire process. Running your locators standalone provides the highest reliability and availability of the locator service as a whole.

=>**Embedded (colocated) locator**. Manage a colocated locator at member startup or through the APIs:

Use the `gemfire.properties start-locator` setting to start the locator automatically inside your Geode member. The locator stops automatically when the member exits.

=>Standard Locator

Note:

- ✓ Locators use the cluster configuration service to save configurations that apply to all cluster members, or to members of a specified group.
- ✓ The configurations are saved in the Locator's directory and are propagated to all locators in a cluster.
- ✓ You organize your data in the cache into *data regions*
- ✓ store your data into your regions in key/value pairs called *data entries*.
- ✓ The cache also provides features like transactions, data querying, disk storage management, and logging.
- ✓ With larger datasets, we can better scale the system by configuring Geode to split a region up into separate partitions, or buckets.
- ✓ *Replicated Region* **maintains copies of its data on more than one Server**.
- ✓ A server is a configurable member of the *Cluster* that runs as a long-lived process and is responsible for managing data *Regions*.
- ✓ By default, Geode clients and servers use port 40404 and multicast to discover each other.
- ✓ *Locators* provide both discovery and load balancing services.
- ✓ A *partitioned* region spreads the data among cache members. After the system is configured, client applications can access the distributed data in regions without knowledge of the underlying system architecture.
- ✓ Every locator in the cluster with `--enable-cluster-configuration` set to true keeps a record of all cluster-level and group-level configuration settings.

Regions:

- All cached data is organized into data regions and you do all of your data puts, gets, and querying activities against them.
- Each region's attributes define how the data in the region is stored, distributed, and managed. Data regions can be distributed, partitioned among system members, or local to the member.

A Geode server is a process that runs as a long-lived, configurable member of a client/server system.

The Geode server is used primarily for hosting long-lived data regions and for running standard Geode processes such as the server in a client/server configuration.

You can start and stop servers using the following methods:

- The `gfsh` tool allows you to manage Geode server processes from the command line.
- You can also start, stop and manage the Geode servers through the `org.apache.geode.distributed.ServerLauncher` API. The `ServerLauncher` API can only be used for Geode Servers that were started with `gfsh` or with the `ServerLauncher` class itself. See the JavaDocs for additional specifics on using the `ServerLauncher` API.

Start the Server

The startup syntax for Geode servers in `gfsh` is:

```
start server --name=value [--assign-buckets(=value)] [--bind-address=value]
  [--cache-xml-file=value] [--classpath=value] [--disable-default-server(=value)]
  [--disable-exit-when-out-of-memory(=value)] [--enable-time-statistics(=value)]
  [--force(=value)] [--include-system-classpath(=value)] [--properties-file=value]
  [--security-properties-file=value]
  [--group=value] [--locators=value] [--locator-wait-time=value] [--log-
level=value]
  [--mcast-address=value] [--mcast-port=value] [--memcached-port=value]
  [--memcached-protocol=value] [--rebalance(=value)] [--server-bind-address=value]
  [--server-port=value] [--spring-xml-location=value]
  [--statistic-archive-file=value] [--dir=value] [--initial-heap=value]
  [--max-heap=value] [--use-cluster-configuration(=value)] [--J=value(,value)*]
  [--critical-heap-percentage=value] [--critical-off-heap-percentage=value]
  [--eviction-heap-percentage=value] [--eviction-off-heap-percentage=value]
  [--hostname-for-clients=value] [--max-connections=value]
  [--message-time-to-live=value] [--max-message-count=value] [--max-threads=value]
  [--socket-buffer-size=value] [--lock-memory=value] [--off-heap-memory-size=value]
```

Note: When both `--max-heap` and `--initial-heap` are specified during server startup, additional GC parameters are specified internally by Geode's Resource Manager.

Example:

```
gfsh>start server --name=server1 --mcast-port=10338 \
--cache-xml-file=../ServerConfigs/cache.xml --server-port=40404
```

```
gfsh>start server --name=server2 --mcast-port=10338 \
--cache-xml-file=../ServerConfigs/cache.xml --server-port=40405

gfsh>start server --name=server1 \
--properties-file=D:\gfseserver\gemfire.properties
```

Check Server Status:

```
gfsh>status server --name=server1
gfsh>status server --pid=2484
```

Stop the Server:

```
gfsh>stop server --name=server1

create region --name=region1 --group=group1 --type=REPLICATE
start server --name=server2 --group=group1 --server-port=40405
start locator --name=locator1 --port=10335
shutdown --include-locators=true //Shutdown cluster
list regions
describe region --name=region1
show missing-disk-stores //to findout missing diskstore files
shutdown --include-locators=true --time-out=60 // to shutdown all members including
locators with graceful shutdown
put --key=abc --value=def --region=region
get --key=abc --region=region
destroy region --name=region
stop locator --name=locator
stop server --name=server2
status locator --pid=2986
put --region=regionA --key="1" --value="one"
query --query="select * from /regionA"
export cluster-configuration --zip-file-name=/Users/username/myClConfig.zip
```

gemfire.properties:

ou use the `gemfire.properties` settings to join a distributed system and configure system member behavior. Distributed system members include applications, the cache server, the locator, and other Geode processes.

You can place any security-related (properties that begin with `security-*`) configuration properties in `gemfire.properties` into a separate `gfsecurity.properties`.

Note: You can also define provider-specific properties (“ssl” properties) in `gfsecurity.properties`

Setting	Definition	Default
ack-severe-alert-threshold	Number of seconds the distributed system will wait after the <code>ack-wait-threshold</code> for a message to be acknowledged before it issues an alert at severe level. A value of zero disables this feature.	0
ack-wait-threshold	<p>Number of seconds a distributed message can wait for acknowledgment before it sends an alert to signal that something might be wrong with the system member that is unresponsive.</p> <p>The waiter continues to wait. The alerts are logged in the system member’s log as warnings.</p> <p>Valid values are in the range 0...2147483647</p>	15
archive-disk-space-limit	Maximum size (in megabytes) of all inactive statistic archive files combined. If this limit is exceeded, inactive archive files are deleted, oldest first, until the total size is within the limit. If set to zero, disk space use is unlimited.	0
archive-file-size-limit	The maximum size (in megabytes) of a single statistic archive file. Once this limit is exceeded, a new statistic archive file is created, and the current archive file becomes inactive. If set to zero, file size is unlimited.	0
async-distribution-timeout	<p>The number of milliseconds a process that is publishing to this process should attempt to distribute a cache operation before switching over to asynchronous messaging for this process. The switch to asynchronous messaging lasts until this process catches up, departs, or some specified limit is reached, such as <code>async-queue-timeout</code> or <code>async-max-queue-size</code>.</p> <p>To enable asynchronous messaging, the value must be set above zero. Valid values are in the range 0...60000.</p> <p>**Note:**</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p>	0

Setting	Definition	Default
async-max-queue-size	<p>Affects non-conflated asynchronous queues for members that publish to this member. This is the maximum size the queue can reach (in megabytes) before the publisher asks this member to leave the distributed system.</p> <p>Valid values are in the range 0..1024.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p>	8
async-queue-timeout	<p>Affects asynchronous queues for members that publish to this member. This is the maximum milliseconds the publisher should wait with no distribution to this member before it asks this member to leave the distributed system. Used for handling slow receivers.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p>	60000
bind-address	<p>Relevant only for multi-homed hosts - machines with multiple network interface cards. Specifies the adapter card the cache binds to for peer-to-peer communication. Also specifies the default location for Geode servers to listen on, which is used unless overridden by the <code>server-bind-address</code>. An empty string causes the member to listen on the default card for the machine. This is a machine-wide attribute used for system member and client/server communication. It has no effect on locator location, unless the locator is embedded in a member process.</p> <p>Specify the IP address, not the hostname, because each network card may not have a unique hostname. An empty string (the default) causes the member to listen on the default card for the machine.</p>	<i>not set</i>
cache-xml-file	Declarative initialization file for the member's cache.	cache.xml
cluster-configuration-dir	This property specifies the directory in which the cluster configuration related disk-store and artifacts are stored. This property is only applicable to dedicated locators that have "enable-cluster-configuration" set to true.	<i>not set</i>

Setting	Definition	Default
conflate-events	Used only by clients in a client/server installation. This is a client-side property that is passed to the server. Affects subscription queue conflation in this client's servers. Specifies whether to conflate (true setting), not conflate (false), or to use the server's conflation setting (server).	server
conserve-sockets	Specifies whether sockets are shared by the system member's threads. If true, threads share, and a minimum number of sockets are used to connect to the distributed system. If false, every application thread has its own sockets for distribution purposes. You can override this setting for individual threads inside your application. Where possible, it is better to set conserve-sockets to true and enable the use of specific extra sockets in the application code if needed. WAN deployments increase the messaging demands on a Geode system. To avoid hangs related to WAN messaging, always set <code>conserve-sockets=false</code> for Geode members that participate in a WAN deployment.	true
delta-propagation	Specifies whether to distribute the deltas for entry updates, instead of the full values, between clients and servers and between peers.	true
deploy-working-dir	Working directory used when deploying JAR application files to distributed system members. This directory can be local and unique to the member or a shared resource. See Deploying Application JARs to Apache Geode Members for more information.	.(current directory)
disable-auto-reconnect	By default, a Geode member (both locators and servers) will attempt to reconnect and reinitialize the cache after it has been forced out of the distributed system by a network partition event or has otherwise been shunned by other members. Use this property to turn off the autoreconnect behavior. See Handling Forced Cache Disconnection Using Autoreconnect for more details.	false
disable-tcp	Boolean indicating whether to disable the use of TCP/IP sockets for inter-cache point-to-point messaging. If disabled, the cache uses datagram (UDP) sockets.	false
distributed-system-id	Identifier used to distinguish messages from different distributed systems. This is required for Portable Data eXchange (PDX) data serialization. Set <code>distributed-system-id</code> to different values for different systems in a multi-site (WAN) configuration, and to different values for production vs. development environments. This setting must be the same for every member of a given distributed system and unique to each distributed	-1

Setting	Definition	Default
	system within a WAN installation. Valid values are integers in the range -1...255. -1 means no setting.	
durable-client-id	Used only for clients in a client/server installation. If set, this indicates that the client is durable and identifies the client. The ID is used by servers to reestablish any messaging that was interrupted by client downtime.	<i>not set</i>
durable-client-timeout	Used only for clients in a client/server installation. Number of seconds this client can remain disconnected from its server and have the server continue to accumulate durable events for it.	300
enable-network-partition-detection	Boolean instructing the system to detect and handle splits in the distributed system, typically caused by a partitioning of the network (split brain) where the distributed system is running. You must set this property to the same value across all your distributed system members. In addition, this property must be set to <code>true</code> if you are using persistent regions and configure your regions to use DISTRIBUTED_ACK or GLOBAL scope to avoid potential data conflicts.	true
enable-cluster-configuration	A value of “true” causes the creation of cluster configuration on dedicated locators. The cluster configuration service on dedicated locator(s) with this property set to “true” would serve the configuration to new members joining the distributed system and also save the configuration changes caused by the <code>gfsh</code> commands. This property is only applicable to dedicated locators..	true
enable-time-statistics	Boolean instructing the system to track time-based statistics for the distributed system and caching. Disabled by default for performance reasons and not recommended for production environments. You must also configure <code>statistics-sampling-enabled</code> to true and specify a <code>statistics-archive-file</code> .	false
enforce-unique-host	Whether partitioned regions will put redundant copies of the same data in different members running on the same physical machine. By default, Geode tries to put redundant copies on different machines, but it will put them on the same machine if no other machines are available. Setting this property to true prevents this and requires different machines for redundant copies.	false
groups	Defines the list of groups that this member belongs to. Use commas to separate group names. Note that anything defined by the roles gemfire property will also be considered a group. See Using Member Groups for more information.	<i>not set</i>

Setting	Definition	Default
http-service-bind-address	If set, then the Geode member binds the embedded HTTP service to the specified address. If this property is not set but the HTTP service is enabled using <code>http-service-port</code> , then Geode binds the HTTP service to the member's local address. Used by the Geode Pulse Web application and the developer REST API service.	<i>not set</i>
http-service-port	If non-zero, then Geode starts an embedded HTTP service that listens on this port. The HTTP service is used to host the Geode Pulse Web application and the development REST API service. If you are hosting the Pulse web app on your own Web server and are not using the development REST API service, then disable this embedded HTTP service by setting this property to zero. Ignored if <code>jmx-manager</code> and <code>start-dev-rest-api</code> are both set to false.	7070
jmx-manager	If true then this member is willing to be a JMX Manager. All the other JMX Manager properties will be used when it does become a manager. If this property is false then all other <code>jmx-manager-*</code> properties are ignored.	false (except on locator)
jmx-manager-bind-address	By default the jmx-manager (when configured with a port) will listen on all the local host's addresses. You can use this property to configure what IP address or host name the JMX Manager will listen on for non-HTTP connections. Ignored if JMX Manager is false or <code>jmx-manager-port</code> is zero.	<i>not set</i>
jmx-manager-hostname-for-clients	Lets you control what hostname will be given to clients that ask the locator for the location of a JMX Manager. By default the IP address that the jmx-manager reports is used. But for clients on a different network this property allows you to configure a different hostname that will be given to clients. Ignored if <code>jmx-manager</code> is false or <code>jmx-manager-port</code> is zero.	<i>not set</i>
jmx-manager-http-port	<i>Deprecated.</i> Use <code>http-service-port</code> instead.	7070
jmx-manager-port	The port this JMX Manager will listen to for client connections. If this property is set to zero then Geode will not allow remote client connections but you can alternatively use the standard system properties supported by the JVM for configuring access from remote JMX clients. Ignored if <code>jmx-manager</code> is false.	1099
jmx-manager-start	If true then this member will start a jmx manager when it creates a cache. Management tools like gfsh can be configured to connect to the jmx-manager. In most cases you should not set this because a jmx manager will automatically be started when needed on a member that sets "jmx-manager" to true. Ignored if jmx-manager is false.	false

Setting	Definition	Default
jmx-manager-update-rate	The rate, in milliseconds, at which this member will push updates to any JMX Managers. Currently this value should be greater than or equal to the statistic-sample-rate. Setting this value too high will cause stale values to be seen by gfsh and Geode Pulse.	2000
load-cluster-configuration-from-dir	Setting this property to “true” causes loading of cluster configuration from “cluster_config” directory in the locator. This property is only applicable to dedicated locators that have “enable-cluster-configuration” set to true.	false
locator-wait-time	The number of seconds that a member should wait for a locator to start if a locator is not available when attempting to join the distributed system. Use this setting when you are starting locators and peers all at once. This timeout allows peers to wait for the locators to finish starting up before attempting to join the distributed system.	0
locators	<p>The list of locators used by system members. The list must be configured consistently for every member of the distributed system. If the list is empty, locators are not used.</p> <p>For each locator, provide a host name and/or address (separated by ‘@’, if you use both), followed by a port number in brackets. Examples:</p> <pre>locators=address1[port1],address2[port2] locators=hostName1@address1[port1],hostName2@address2[port2] locators=hostName1[port1],hostName2[port2]</pre> <p>Note:</p> <p>On multi-homed hosts, this last notation will use the default address. If you use bind addresses for your locators, explicitly specify the addresses in the locators list—do not use just the hostname.</p> <p>If you have values specified for the <code>locators</code> property, the <code>mcast-port</code> property defaults to 0.</p> <p>Note:</p> <p>If you specify invalid DNS hostnames in this property, any locators or servers started with <code>gfsh</code> will not produce log files. Make sure you provide valid DNS hostnames before starting the locator or server with <code>gfsh</code>.</p>	<i>not set</i>
lock-memory	When <code>true</code> , locks heap and off-heap memory into RAM to prevent the operating system from paging the memory out to disk.	false

Setting	Definition	Default
log-disk-space-limit	Maximum size in megabytes of all inactive log files combined. If this limit is exceeded, inactive log files are deleted, oldest first, until the total size is within the limit. If set to zero, disk space use is unlimited.	0
log-file	<p>File to which a running system member writes log messages. If set to null, the default is used.</p> <p>Each member type has its own default output:</p> <p>application: standard out</p> <p>locator: <locator_name>.log</p> <p>server: <server_name>.log</p>	null
log-file-size-limit	Maximum size in megabytes of a log file before it is closed and logging rolls on to a new (child) log file. If set to 0, log rolling is disabled.	0
log-level	<p>Level of detail of the messages written to the system member's log. Setting log-level to one of the ordered levels causes all messages of that level and greater severity to be printed.</p> <p>Valid values from lowest to highest are fine, config, info, warning, error, severe, and none.</p>	config
max-wait-time-reconnect	Maximum number of milliseconds to wait for the distributed system to reconnect on each reconnect attempt.	60000
mcast-address	<p>Address used to discover other members of the distributed system. Only used if mcast-port is non-zero. This attribute must be consistent across the distributed system. Select different multicast addresses and different ports for different distributed systems. Do not just use different addresses. Some operating systems may not keep communication separate between systems that use unique addresses but the same port number.</p> <p>This default multicast address was assigned by IANA (http://www.iana.org/assignments/multicast-addresses). Consult the IANA chart when selecting another multicast address to use with Geode.</p> <p>**Note:**</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication. If multicast is enabled,</p>	<p>239.192.81 for IPv4 (the default IP version)</p> <p>FF38::1234 IPv6</p>

Setting	Definition	Default
	distributed regions use it for most communication. Partitioned regions only use multicast for a few purposes, and mainly use either TCP or UDP unicast.	
mcast-flow-control	<p>Tuning property for flow-of-control protocol for unicast and multicast no-ack UDP messaging. Compound property made up of three settings separated by commas: byteAllowance, rechargeThreshold, and rechargeBlockMs.</p> <p>Valid values range from these minimums: 10000,0.1,500 to these maximums: no_maximum ,0.5,60000.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication, generally between distributed regions.</p>	1048576,05000
mcast-port	<p>Port used, along with the mcast-address, for multicast communication with other members of the distributed system. If zero, multicast is disabled.</p> <p>Note:</p> <p>Select different multicast addresses and ports for different distributed systems. Do not just use different addresses. Some operating systems may not keep communication separate between systems that use unique addresses but the same port number.</p> <p>Valid values are in the range 0..65535.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p> <p>If you have values specified for the <code>locators</code> property, the <code>mcast-port</code> property defaults to 0.</p>	10334
mcast-recv-buffer-size	Size of the socket buffer used for incoming multicast transmissions. You should set this high if there will be high volumes of messages.	1048576

Setting	Definition	Default
	<p>Valid values are in the range 2048.. OS_maximum.</p> <p>Note:</p> <p>The default setting is higher than the default OS maximum buffer size on Unix, which should be increased to at least 1 megabyte to provide high-volume messaging on Unix systems.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p>	
mcast-send-buffer-size	<p>The size of the socket buffer used for outgoing multicast transmissions.</p> <p>Valid values are in the range 2048.. OS_maximum.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p>	65535
mcast-ttl	<p>How far multicast messaging goes in your network. Lower settings may improve system performance. A setting of 0 constrains multicast messaging to the machine.</p> <p>Note:</p> <p>This setting controls only peer-to-peer communication and does not apply to client/server or multi-site communication.</p>	32
member-timeout	<p>Geode uses the <code>member-timeout</code> server configuration, specified in milliseconds, to detect the abnormal termination of members. The configuration setting is used in two ways: 1) First it is used during the UDP heartbeat detection process. When a member detects that a heartbeat datagram is missing from the member that it is monitoring after the time interval of 2 * the value of <code>member-timeout</code>, the detecting member attempts to form a TCP/IP stream-socket connection with the monitored</p>	5000

Setting	Definition	Default
	<p>member as described in the next case. 2) The property is then used again during the TCP/IP stream-socket connection. If the suspected process does not respond to the <i>are you alive</i> datagram within the time period specified in <code>member-timeout</code>, the membership coordinator sends out a new membership view that notes the member's failure.</p> <p>Valid values are in the range 1000..600000.</p>	
membership-port-range	<p>The range of ports available for unicast UDP messaging and for TCP failure detection. This is specified as two integers separated by a hyphen. Different members can use different ranges.</p> <p>Geode randomly chooses at least two unique integers from this range for the member, one for UDP unicast messaging and the other for TCP failure detection messaging. If tcp-port is configured to 0, it will also randomly select a port from this range for TCP sockets used for peer-to-peer communication only.</p> <p>Therefore, the specified range must include at least three available port numbers (UDP, FD_SOCKET, and TCP DirectChannel).</p> <p>The system uniquely identifies the member using the combined host IP address and UDP port number.</p> <p>You may want to restrict the range of ports that Geode uses so the product can run in an environment where routers only allow traffic on certain ports.</p>	1024-6553
memcached-port	If specified and is non-zero, sets the port number for an embedded Gemcached server and starts the Gemcached server.	0
memcached-protocol	Sets the protocol used by an embedded Gemcached server. Valid values are <code>BINARY</code> and <code>ASCII</code> . If you omit this property, the ASCII protocol is used.	ASCII
name	Symbolic name used to identify this system member.	<i>not set</i>
off-heap-memory-size	<p>Specifies the size of off-heap memory in megabytes (m) or gigabytes (g). For example:</p> <pre>off-heap-memory-size=4096m off-heap-memory-size=120g</pre>	<i>not set</i>

Setting	Definition	Default
redundancy-zone	Defines this member's redundancy zone. Used to separate member's into different groups for satisfying partitioned region redundancy. If this property is set, Geode will not put redundant copies of data in members with the same redundancy zone setting. See Configure High Availability for a Partitioned Region for more details.	<i>not set</i>
remote-locators	<p>Used to configure the locators that a cluster will use in order to connect to a remote site in a multi-site (WAN) configuration. To use locators in a WAN configuration, you must specify a unique distributed system ID (<code>distributed-system-id</code>) for the local cluster and remote locator(s) for the remote clusters to which you will connect.</p> <p>For each remote locator, provide a host name and/or address (separated by '@', if you use both), followed by a port number in brackets. Examples:</p> <pre>remote-locators=address1[port1],address2[port2] remote-locators=hostName1@address1[port1],hostName2@address2[port2] remote-locators=hostName1[port1],hostName2[port2]</pre>	<i>not set</i>
remove-unresponsive-client	When this property is set to true, the primary server drops unresponsive clients from all secondaries and itself. Clients are deemed unresponsive when their messaging queues become full on the server. While a client's queue is full, puts that would add to the queue block on the server.	false
security-*	Any security-related (properties that begin with <code>security-</code>) configuration properties that are normally configured in <code>gemfire.properties</code> can be moved to a separate <code>gfsecurity.properties</code> file. Placing these configuration settings in a separate file allows you to restrict access to security configuration data. This way, you can still allow read or write access for your <code>gemfire.properties</code> file.	<i>not set</i>
security-client-accessor	Deprecated. Used for authorization. Static creation method returning an <code>AccessControl</code> object, which determines authorization of client-server cache operations. This specifies the callback that should be invoked in the pre-operation phase, which is when the request for the operation is received from the client.	<i>not set</i>
security-client-accessor-pp	Deprecated. Used for authorization. The callback that should be invoked in the post-operation phase, which is when the operation has completed on the server but before the result is sent to the client. The post-operation callback is also invoked for the updates that are sent from server to client through the notification channel.	<i>not set</i>

Setting	Definition	Default
security-client-auth-init	Used for authentication. Static creation method returning an <code>AuthInitialize</code> object, which obtains credentials for peers in a distributed system. The obtained credentials should be acceptable to the <code>Authenticator</code> specified through the security-peer-authenticator property on the peers.	<i>not set</i>
security-client-authenticator	Deprecated. Used for authentication. Static creation method returning an <code>Authenticator</code> object, which is used by a peer to verify the credentials of the connecting peer.	<i>not set</i>
security-client-dhalgo	Used for authentication. For secure transmission of sensitive credentials like passwords, you can encrypt the credentials using the Diffie-Hellman key exchange algorithm. Do this by setting the security-client-dhalgo system property on the clients to the name of a valid symmetric key cipher supported by the JDK.	<i>not set</i>
security-log-file	Used with authentication. The log file for security log messages. If not specified, the member's regular log file is used.	<i>not set</i>
security-log-level	Used with authentication. Logging level detail for security log messages. Valid values from lowest to highest are fine, config, info, warning, error, severe, and none.	config
security-manager	Specifies the implementation of the <code>SecurityManager</code> interface that implements the callbacks that do authentication and authorization.	<i>not set</i>
security-peer-auth-init	Deprecated. Used with authentication. Static creation method returning an <code>AuthInitialize</code> object, which obtains credentials for peers in a distributed system. The obtained credentials should be acceptable to the <code>Authenticator</code> specified through the security-peer-authenticator property on the peers.	<i>not set</i>
security-peer-authenticator	Deprecated. Used with authentication. Static creation method returning an <code>Authenticator</code> object, which is used by a peer to verify the credentials of the connecting peer.	<i>not set</i>
security-peer-verifymember-timeout	Used with authentication. Timeout in milliseconds used by a peer to verify membership of an unknown authenticated peer requesting a secure connection.	1000

Setting	Definition	Default
security-post-processor	Specifies the implementation of the <code>PostProcessor</code> interface that implements user-defined callbacks that can change the returned results of region get operations.	<i>not set</i>
server-bind-address	<p>Relevant only for multi-homed hosts - machines with multiple network interface cards. Network adapter card a Geode server binds to for client/server communication. You can use this to separate the server's client/server communication from its peer-to-peer communication, spreading the traffic load.</p> <p>This is a machine-wide attribute used for communication with clients in client/server and multi-site installations. This setting has no effect on locator configuration.</p> <p>Specify the IP address, not the hostname, because each network card may not have a unique hostname.</p> <p>An empty string causes the servers to listen on the same card used for peer-to-peer communication. This is either the <code>bind-address</code> or, if that is not set, the machine's default card.</p>	<i>not set</i>
socket-buffer-size	Receive buffer sizes in bytes of the TCP/IP connections used for data transmission. To minimize the buffer size allocation needed for distributing large, serializable messages, the messages are sent in chunks. This setting determines the size of the chunks. Larger buffers can handle large messages more quickly, but take up more memory.	32768
socket-lease-time	<p>Time, in milliseconds, a thread can have exclusive access to a socket it is not actively using. A value of zero causes socket leases to never expire. This property is ignored if <code>conserve-sockets</code> is true.</p> <p>Valid values are in the range 0..600000.</p>	60000
ssl-enabled-components	Components for which to enable SSL. Comma-separated list of one or more of (cluster, gateway, http, jmx, locator, server) or "all".	all
ssl-require-authentication	Boolean. Require two-way authentication for SSL-enabled components. Applies to all components except http.	true
ssl-http-require-authentication	Boolean. Require two-way authentication for http component.	false

Setting	Definition	Default
ssl-default-alias	String. Default certificate name. If empty, use first certificate in key store.	
ssl- <i>component</i> -alias	String. Certificate name for specified <i>component</i> , which is one of: cluster, gateway, http, jmx, locator, or server.	
ssl-ciphers	Comma-separated list of SSL ciphers or “any”	any
ssl-protocols	Comma-separated list of SSL protocols or “any”	any
ssl-keystore, ssl-keystore-password	Strings. Path to key store, key store password.	
ssl-truststore, ssl-truststore-password	Strings. Path to trust store, trust store password.	
start-dev-rest-api	If set to true, then the developer REST API service will be started when cache is created. REST service can be configured using <code>http-service-port</code> and <code>http-service-bind-address</code> properties.	false
start-locator	<p>If set, automatically starts a locator in the current process when the member connects to the distributed system and stops the locator when the member disconnects.</p> <p>To use, specify the locator with an optional address or host specification and a required port number, in one of these formats:</p> <pre>start-locator=address[port1] start-locator=port1</pre> <p>If you only specify the port, the address assigned to the member is used for the locator.</p> <p>If not already there, this locator is automatically added to the list of locators in this set of <code>gemfire properties</code>.</p>	<i>not set</i>
statistic-archive-file	The file to which the running system member writes statistic samples. For example: “StatisticsArchiveFile.gfs”. An empty string disables archiving. Adding .gz suffix to the file name causes it to be compressed.	<i>not set</i>
statistic-sample-rate	<p>How often to sample statistics, in milliseconds.</p> <p>Valid values are in the range 100..60000.</p>	1000

Setting	Definition	Default
statistic-sampling-enabled	<p>Whether to collect and archive statistics on the member.</p> <p>Statistics sampling provides valuable information for ongoing system tuning and troubleshooting purposes. Sampling statistics at the default sample rate does not impact system performance. We recommend enabling statistics sampling in production environments.</p> <p>Note:</p> <p>This setting does not apply to partitioned regions, where statistics are always enabled.</p>	false
tcp-port	<p>The TCP port to listen on for cache communications. If set to zero, the operating system selects an available port. Each process on a machine must have its own TCP port. Note that some operating systems restrict the range of ports usable by non-privileged users, and using restricted port numbers can cause runtime errors in Geode startup.</p> <p>Valid values are in the range 0..65535.</p>	0
tombstone-gc-threshold	<p>The number of tombstones that can accumulate before the Geode member triggers garbage collection for tombstones. See How Destroy and Clear Operations Are Resolved.</p>	100000
udp-fragment-size	<p>Maximum fragment size, in bytes, for transmission over UDP unicast or multicast sockets. Smaller messages are combined, if possible, for transmission up to the fragment size setting.</p> <p>Valid values are in the range 1000..60000.</p>	60000
udp-recv-buffer-size	<p>The size of the socket buffer used for incoming UDP point-to-point transmissions. If disable-tcp is false, a reduced buffer size of 65535 is used by default.</p> <p>The default setting of 1048576 is higher than the default OS maximum buffer size on Unix, which should be increased to at least 1 megabyte to provide high-volume messaging on Unix systems.</p>	1048576

Setting	Definition	Default
	Valid values are in the range 2048.. OS_maximum.	
udp-send-buffer-size	The size of the socket buffer used for outgoing UDP point-to-point transmissions. Valid values are in the range 2048..OS_maximum.	65535
use-cluster-configuration	This property is only applicable for data members (non-client and non-locator). A value of “true” causes a member to request and use the configuration from cluster configuration services running on dedicated locators. Setting this property to “false” causes a member to not request the configuration from the configuration services running on the locator(s).	true
user-command-packages	A comma separated list of Java packages that contain classes implementing the <code>CommandMarker</code> interface. Matching classes will be loaded when the VM starts and will be available in the GFSH command-line utility.	<i>not set</i>

Cache.xml:

Server or peer cache:

```
<?xml version="1.0" encoding="UTF-8"?>
<cache
  xmlns="http://geode.apache.org/schema/cache"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://geode.apache.org/schema/cache
http://geode.apache.org/schema/cache/cache-1.0.xsd"
  version="1.0">
...
</cache>
```

Client cache:

```
<?xml version="1.0" encoding="UTF-8"?>
<client-cache
  xmlns="http://geode.apache.org/schema/cache"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://geode.apache.org/schema/cache
http://geode.apache.org/schema/cache/cache-1.0.xsd"
  version="1.0">
...
</client-cache>
```

Server Configuration

- [<cache> Element Reference](#)
- [<cache-server>](#)
- [<region>](#)
- [<region-attributes>](#)

Client Configuration

- [<client-cache> Element Reference](#)
- [<pool>](#)
- [<region>](#)

Tools and Modules:

- [gfsh](#)
`gfsh` (pronounced “jee-fish”) provides a single, powerful command-line interface from which you can launch, manage, and monitor Geode processes, data, and applications.
- [Gemcached](#)
Gemcached is a Geode adapter that allows Memcached clients to communicate with a Geode server cluster, as if the servers were memcached servers. Memcached is an open-source caching solution that uses a distributed, in-memory hash map to store key-value pairs of string or object data.
- [HTTP Session Management Modules](#)
The Apache Geode HTTP Session Management modules provide fast, scalable, and reliable session replication for HTTP servers without requiring application changes.
- [Geode Pulse](#)
Geode Pulse is a Web Application that provides a graphical dashboard for monitoring vital, real-time health and performance of Geode clusters, members, and regions.
- [Geode Redis Adapter](#)
The Geode Redis adapter allows Geode to function as a drop-in replacement for a Redis data store, letting Redis applications take advantage of Geode’s scaling capabilities without changing their client code. Redis clients connect to a Geode server in the same way they connect to a Redis server, using an IP address and a port number.
- [Apache Lucene® Integration](#)
The Apache Lucene® integration enables users to create Lucene indexes and execute Lucene searches on data stored in Geode.

Geode Configuration:

- **gemfire.properties.** Contains the settings required by members of a cluster. These settings include licensing, system member discovery, communication parameters, logging, and statistics. See the Geode Properties Reference.
- **gfsecurity.properties.** An optional separate file that contains security-related (security-*) settings that are otherwise defined in **gemfire.properties**. Placing these member properties into a separate file allows you to restrict user access to those specific settings. See the Geode Properties Reference.
- **cache.xml.** Declarative cache configuration file. This file contains XML declarations for cache, region, and region entry configuration. You also use it to configure disk stores, database login credentials, server and remote site location information, and socket information.

Rest API:

The Apache Geode developer REST interface runs as an embedded HTTP or HTTPS service (Jetty server) within a Geode data node.

All Geode REST interface classes and required JAR files are distributed as a WAR file with the Geode product distribution. You can locate the file in the following location:

\$GEMFIRE/tools/Extensions/gemfire-api.war

Note:

- ✓ To enable the developer REST API service in Apache Geode, set the **start-dev-rest-api** Geode property to **true** when starting a data node using either **gfsh** or the **ServerLauncher** API.
- ✓ To ensure that the data node is reachable on a machine with multiple NIC addresses, you can use **http-service-bind-address** to bind an address to the REST API service (as well as the other embedded web services such as Pulse.)

```
start server --name=server1 --start-rest-api=true --http-service-port=8080 --http-service-bind-address=localhost
```

Also, we can enable **rest-api** using below way.

gemfire.properties contains:

```
http-service-port=8080
http-service-bind-address=localhost
start-dev-rest-api=true
```

To Enable swagger in Apache Geode, use below command.

```
start server --name=server1 --J=-Dgemfire.start-dev-rest-api=true --J=-Dgemfire.http-service-  
bind-address=localhost --J=-Dgemfire.jmx-manager=true --J=-Dgemfire.jmx-manager-  
start=true
```

<http://localhost:8080/gemfire-api/docs/index.html>

GFSH SHELL Variables:

```
echo --string=$* //to display all predefined SHELL variables  
set variable --name aaa --value bbbb  
hint server //find list of commands of server
```

```
create region --name=region1 --type=
```

PARTITION:

A region configured with the PARTITION region shortcut is partitioned across each peer member that created the region.

PARTITION_REDUNDANT:

The region maintains an extra copy of the data in memory.

PARTITION_PERSISTENT:

A region configured with the PARTITION_PERSISTENT region shortcut has state that is partitioned across each peer member that creates the region. The region writes its state to disk and can recover that state when the member restarts.

PARTITION_REDUNDANT_PERSISTENT

PARTITION_OVERFLOW

PARTITION_REDUNDANT_OVERFLOW

PARTITION_PERSISTENT_OVERFLOW

PARTITION_REDUNDANT_PERSISTENT_OVERFLOW

PARTITION_HEAP_LRU

PARTITION_REDUNDANT_HEAP_LRU

REPLICATE:

A region configured with the REPLICATE region shortcut has local state that is kept in sync with all other peer regions configured with a REPLICATE data policy.

REPLICATE_PERSISTENT:

A region configured with the REPLICATE_PERSISTENT region shortcut has local state that is kept in sync with all other peer regions that are configured with a REPLICATE data policy. The region writes its state to disk and recovers that state when the member restarts.

REPLICATE_OVERFLOW
REPLICATE_PERSISTENT_OVERFLOW
REPLICATE_HEAP_LRU

LOCAL:

The region does not distribute data and operations to other caches.

LOCAL_PERSISTENT
LOCAL_HEAP_LRU

LOCAL_OVERFLOW:

The region does not distribute data and operations to other caches.

The region moves the values of entries to disk when it detects that the JVM is running low on memory.

LOCAL_PERSISTENT_OVERFLOW
PARTITION_PROXY
PARTITION_PROXY_REDUNDANT
REPLICATE_PROXY:

A region configured with the REPLICATE_PROXY region shortcut has no local state and forwards all

operations to a peer region configured with the REPLICATE or REPLICATE_PERSISTENT region shortcut.

XML Definition of REPLICATE_PERSISTENT

```
<region name="myRPregion">  
  <region-attributes  
    scope="distributed-ack"  
    data-policy="empty"/>  
</region>
```

For JArs:

=====

```
deploy --jars=group1_functions.jar  
list deployed  
undeploy
```

How do use an In Memory Data Grid?

Simply install your servers in a single site or across multiple sites.

Each group of servers within a site is referred to as a cluster.

Install the IMDG software on all the servers and choose the appropriate topology for the product.

For multi-site operations I always recommend a partitioned and replicated cache.

Setup your APIs, or GUI interfaces to allow replicated between the various servers.

Develop your data model and the business logic around the model.

Partitioned or Replicated Regions?

Partitioned

- Store data sets that are too large to fit in a single VM, single cache member
- 1 to Many relationships
- Suitable for 100's of GB
- Represents most typical cases so start with Partitioned regions

Replicated:

- Read heavy, small data set(reference or lookup data)
- Ideal for Many to Many relationships
- Data sets that can be entirely contained in a single member
- Represents the more rare case
- CANNOT horizontally scale

Function Execution

- ✓ A function is a body of code that resides on a server and that an application can invoke from a client or from another server without the need to send the function code itself.
- ✓ The caller can direct a data-dependent function to operate on a particular dataset, or can direct a data-independent function to operate on a particular server, member, or member group.

Steps to Write Function:

Main tasks:

1. Write the function code.
2. Register the function on all servers where you want to execute the function. The easiest way to register a function is to use the `gfs` `deploy` command to deploy the JAR file containing the function code. Deploying the JAR automatically registers the function for you. Alternatively, you can write the XML or application code to register the function.
3. Write the application code to run the function and, if the function returns results, to handle the results.
4. If your function returns results and you need special results handling, code a custom `ResultsCollector` implementation and use it in your function execution.