RedisGears 1.0 release notes

Here's what's changed recently in RedisGears.
To learn more, see the linked pull requests.

Requirements

RedisGears v1.0.9 requires:

- Minimum Redis compatibility version (database): 6.0.0
- Minimum Redis Enterprise Software version (cluster): 6.0.0

v1.0.9 (January 2022)

This is a maintenance release for RedisGears 1.0

Update urgency: LOW : No need to upgrade unless there are new features you want to use.

Details:

- Improvements:
  - #652 Support for keys space notifications coming from other modules.

v1.0.8 (November 2021)

This is a maintenance release for RedisGears 1.0.

Update urgency: MODERATE - Program an upgrade of the server, but it's not urgent.

Details:

- Minor features:
  - #624, #626 Update Python interpreter version to 3.7.12

- Bug fixes:
  - #610 Fix symbol collision with RediSearch and RedisGraph that causes deadlock
  - #609, #611 Crash on stream reader where stream is deleted during read
  - #612 Return error when GearsBuilder is used after registered or run
  - #613 Rare issue where stream reader might trigger executions on replica
  - #629 Rare deadlock on stream reader

v1.0.7 (August 2021)

This is a maintenance release for version 1.0.
Update urgency: LOW - No need to upgrade unless there are new features you want to use.

Details:

- Minor Features:
  - #594 Added RedisGears info section to Redis info command
  - #587 inorder option for the command reader
  - #592, #586 Background executions will now appear on Redis slowlog, available in Redis 6.2 and above.

**v1.0.6 (March 2021)**

This is a maintenance release for version 1.0.

- Update urgency: Medium

- Highlights:
  This release improves overall stability and provides fixes for issues found after the previous release.

- Bug fixes:
  - #496 Allow parallel execution on Command Reader.
  - #505 Crash on a client that sends cluster set and disconnects.

**v1.0.5 (18 January 2021)**

This is a maintenance release for version 1.0.

- Update urgency: High

- Highlights:
  This release improves overall stability and provides fixes for issues found after the previous release.

- Bug fixes:
  - #480 Crash when map/accumulate/accumulateby raises an error which causes the same pyobject to be freed twice.
  - #480 'Cluster set' happened before resending hello request, which caused a crash.
  - #480 Memory leaks on 'cluster set'.

**v1.0.4 (17 January 2021)**

This is a maintenance release for version 1.0.

- Highlights:
  This release improves overall stability and provides fixes for issues found after the previous release.

- Bug fixes:
  - #458 Crash on StreamReader when the stream is dropped during processing of the function.
  - #477 Rare crash on accumulate step.
v1.0.3 (November 2020)

This is a maintenance release for version 1.0.

- **Highlights:**
  
  This release improves overall stability and provides fixes for issues found after the previous release.

- **Bug fixes:**
  
  - #427 Rare issue where messages might get lost and cause executions timeouts.
  
  - #434 Triggering an execution on uninitialized cluster state should raise a cluster uninitialized error.

v1.0.2 (October 2020)

This is a maintenance release for version 1.0.

- **Highlights:**
  
  This release improves overall stability and provides fixes for issues found after the previous release.

- **Details:**
  
  - **Minor Features:**
    
    - #368 Support for buffer input on createTensorFromBlob when integrating with RedisAI API.
    
    - #373 Registered execution plans will not re-send their metadata when they are triggered (this feature reduces network overhead and improve performance of distributed registrations).
    
    - #371 Full details on import requirement failure.
  
  - **Bug fixes:**
    
    - #374 Fix RedisAI toFlatList function to handle long long encoding.
    
    - #375 Call OnRegister on RDBLoad.
    
    - #388 Fix crash on hello request failure (internal protocol between shards), a retry will be triggered after 1 second.
    
    - #395 Fix rare wrong results on aggregate and aggregateby. Use deep copy on the zero value to avoid those incorrect results.

v1.0.1 (July 2020)

This is a maintenance release for version 1.0.

- **Highlights:**
  
  - *Import and Export requirements* - With this capability, you can export the python requirements present in RedisGears and import them into another instance using gears-cli.

    This enables, amongst others, to import requirements to environments without internet access. Notice that the requirement should be exported from and imported into machines with the same OS and version. This is the recommended tool to use for productionising RedisGears functions (for example to be used in your CI).

  - *Optimised requirements installation mechanism* - requirements will now be distributed once to each shard and not each time an execution is triggered.
- **RedisAI API fixes and additions** - fixes/additions related to the integration with RedisAI.

- **Details:**
  - **Features:**
    - #330 **SendMsgRetries** configuration parameter indicating how many times RedisGears should try sending messages between shards.
    - #330 **PythonInstallReqMaxIdleTime** configuration parameter for the maximum amount of time to wait for requirements to be installed.
    - #330 **RG.PYDUMPREQ** command that output all the currently available python requirements.
    - Added the following API functionality to the RedisAI integration:
      - #346 Script run will now return a list of tensors.
      - #347 Allow tensor creation from bytes.
      - #350 Added mget and mset to get/set multiple tensors from/to the keyspace.
      - #345 Free python GIL before calling redisAI model/script run.
  - **Bug fixes:**
    - #325 Circular reference that caused a logical memory leak.
    - #330 Crash on wrong reply status.
    - #345 Crash on python session (private data on rg.dumpregistration) ‘tostr’ function when there is no requirements.
    - #334 Type check to RedisAI API to prevent potential crashes.

**v1.0.0 (May 2020)**

This is the first general availability (GA) release of RedisGears (v1.0.0).

- **Highlights:**
  - Program everything you want in Redis - With a built-in C-API and Python interpreter, RedisGears lets you use full-fledged Python scripts and libraries to process data inside Redis.
  - Write once, deploy anywhere - Write your functions against a standalone Redis database and deploy them to production clusters—with no changes required.
  - Run your serverless engine where your data lives - RedisGears lets you process events and streams faster by running in memory next to your data in Redis.

To get started please check out [redisgears.io](https://redisgears.io). We plan to release a blogpost soon and will cross link it here.

**Bug fixes (compared to RC1):**

- #288 dependencies with version will not crash the server.
- #309, #313 Prevent crashes on RedisAI integration and update the low level C api of RedisAI.
- #311 Fix out-of-order reply in certain situations.
- #321 Return error when unknown argument is given.
Updated: January 11, 2022

**Note:**
- The version inside Redis will be 10000 or 1.0.0 in semantic versioning.
- This version requires the Redis version to be 6.0 and above.