



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

AI Core Services

9.0.011.00

12/22/2025

9.0.011.00

AI Core Services Release Notes

9.x AI Core Services is part of 9.x starting in **9.0.006.05**.

Release Date	Release Type	Restrictions	AIX	Linux	Solaris	Windows
07/13/18	General	Under Shipping Control		X		

What's New

This is a general release for this component. For availability of this release, contact your Genesys representative. This release includes the following new features and enhancements:

- You can now generate and purge predictor data using the Predictive Routing API. For details, see the [Predictive Routing API Reference](#). (This file requires a password to open it. Contact your Genesys representative if you need access.)
- You can now configure parameters to control password-related behavior such as how often users must change them, blocking users after a specified number of login attempts, and adding a custom message when users are blocked. For a full description of how to configure password-related behavior, see [Password policy configuration](#). This functionality requires you to run two upgrade scripts, **upgrade_40a_users.py** and **upgrade_41a_accounts.py**, as documented in the [Upgrade Notes](#) section in this Release Note.
- The audit trail functionality has been improved, to record additional actions and provide the ability to specify how long audit trail records are kept. All actions related to logins, object modification/creation/deletion, and so on, whether performed using the GPR application or the API, are logged. The records include the ID for the user who performed that action and the date and time. For details, see [Audit trails](#).
- You can now create a new predictor by copying an existing one. To do so, send a POST request to the new **copy_predictor** endpoint. This functionality is available using the Predictive Routing API only. For details, see the [Predictive Routing API Reference](#).

Helpful Links

Releases Info

- [List of 9.0.x Releases](#)
- [9.0.x Known Issues](#)

Product Documentation

- [Genesys Predictive Routing](#)

Genesys Products

- [List of Release Notes](#)

- This release supports Mongo DB 3.6. The IP includes the scripts required to upgrade your MongoDB database, as well as the Predictive Routing application. For details, see the [Upgrade Notes](#) section in this Release Note.
 - You can now use GET commands to retrieve dataset and predictor details using the Predictive Routing API. For details, see the **dataset** and **predictor** sections of the [Predictive Routing API Reference](#).
 - Predictive Routing now correctly recognizes columns with any combination of the following Boolean values: y/n, Y/N, Yes/No. Previously, only columns with true/false and 0/1 values were discovered as Booleans. The identification is case insensitive.
 - The way Predictive Routing recomputes cardinalities when you append data to Agent or Customer Profiles via the Predictive Routing API has been changed.
 - Cardinalities are no longer recomputed automatically across the whole collection each time you append data. Full automatic computation happens only once, when an Agent or Customer Profile is uploaded the first time for schema discovery. When you append data to an Agent or Customer Profile via the API, cardinalities are computed only for the appended data portion and only when the number of agents or customers set in the **ADD_CARDINALITIES_EVERY_N_RECORDS** parameter is reached. The results of computation are added to the already-stored cardinality values. This new behavior significantly improves speed when loading new data by avoiding simultaneous recomputations on the full data collection when there are multiple frequent appends done in small batches.
 - The **ADD_CARDINALITIES_EVERY_N_RECORDS** parameter has been added to the **tango.env** file with the default value of 1000. Each time the counter for appended agents/customers reaches this number, computation for the last appended 1000 records takes place. The default value can be changed in the **tango.env** file, which is located in the **IP_<version>/conf** directory. When you change the value, restart the application to have the new value take effect.
 - You can force recomputation of cardinalities on the full Agent or Customer Profiles collection using the new POST **compute_cardinalities** API endpoint. For details, see the [Predictive Routing API Reference](#).
- NOTE:** This functionality is available only when you use the Predictive Routing API. If you append using the Predictive Routing application interface, all cardinalities are recalculated, which is the same behavior as in previous releases.
- You can now upload data (agent, customer, and dataset) using zip-archived .csv files. Only one .csv file per archive is supported. This applies to uploads made via either the Predictive Routing application or the Predictive Routing API.
 - The performance of the Predictors administration when you use the Predictive Routing application has been significantly improved.
 - You can now retrieve information on the currently deployed platform using the new **version** endpoint, which has been added to the Predictive Routing API. For details, see the [Predictive Routing API Reference](#).
 - You can configure the Predictive Routing application to display custom messages on the login screen. To add a message, configure the LOGIN_MESSAGES environment variable, as explained in [Set Values for Environment Variables](#).
- **Notes:**
 - The text of the login message is not centered.
 - Non-ASCII characters must be escaped, using the equivalent HTML symbol code.
 - You must reconfigure the login message every time you install a new version of the AI Core Services component.
-

Resolved Issues

This release contains the following resolved issues:

Genesys Predictive Routing (GPR) now automatically sets the value of the **OMP_NUM_THREADS** environment variable to 1, which enables the operating system to properly distribute CPU threads among the various running processes. Previously, this variable had to be set manually. (PRR-2919)

The display for **Agent%** in predictor models has been corrected.

- -1 is displayed if the agent profile is not synced.
- 0 is displayed if the agent profile is synced but no agent in the profile data has more than ten interactions.

The **Agent%** is displayed only for disjoint and hybrid models. (PRR-2761)

This release corrects the issue that sometimes caused a Document Too Large error when running a Feature Analysis using a continuous target metric. (PRR-2746)

When a scoring request is made, only the content of the request itself is now printed to the logs. Previously, a large quantity of unnecessary DEBUG-level messages were logged. (PRR-2638)

When you add agents to an existing Agent Profile by means of an API POST request, Predictive Routing now checks whether the agent IDs are already present in the database. If so, the existing record is updated. Previously, Predictive Routing created a duplicate record for the agent. (PRR-2567)

When you upload data, GPR now skips any record containing unsupported characters and continues processing the upload starting from the next record. Previously, the data upload failed if GPR encountered any unsupported characters. (PRR-2514)

Passing a numeric Agent or Customer ID in a request now successfully returns the corresponding record from the database. (PRR-2513)

When you train a predictor with a large number of features, it is normal for a scoring request to contain only a subset of the predictor features. To reduce scoring response time, the response message no longer includes a warning about the missing features. In addition, the warning Missing keys set is no longer printed in the tango logs. If you need to see which were the omitted features, add the following parameter to the scoring request: "warnings":true. (PRR-2214)

Upgrade Notes

Use the following special procedure to upgrade to release 9.0.011.00.

Important

These instructions assume that your current Mongo DB version is 3.2.

Single-Server Deployment

Important

It is possible to run 9.0.011.00 without upgrading the Mongo DB database. However, Genesys strongly recommends to upgrade to version 3.6. After you upgrade Mongo DB to version 3.6, you cannot install any version of Predictive Routing earlier than the 9.0.011.00 release.

1. Download the IP for release 9.0.011.00. For detailed instructions, see [Deploying AICS on a Single Host](#)
 1. In the new IP folder, run the following command: `bash scripts/install.sh`
 2. Then, also in the new IP folder, run the following command: `bash/scripts/upgrade_gpr_services.sh`

At this point, release 9.0.011.00 is installed and your Mongo DB version remains at 3.2.
2. Run the following commands to update your data to the correct format for Mongo DB 3.6:

```
docker exec -ti tango /bin/bash cd src/gpr  
MODE=prod python common/scripts/versioning/upgrade_40a_users.py  
MODE=prod python common/scripts/versioning/upgrade_41a_accounts.py
```

The upgrade to version 9.0.011.00 is now completed.
3. To upgrade the database itself, now run the following scripts:
 1. In the new IP folder, run the following command **two** times (the first time it upgrades to version 3.4 and then to version 3.6): `bash scripts/upgrade_to_mongo36.sh`
 2. Then in the new IP folder, run the following script: `bash scripts/restart.sh`

HA Deployment

Important

It is possible to run 9.0.011.00 without upgrading the Mongo DB database. However, Genesys strongly recommends to upgrade to version 3.6. After you upgrade Mongo DB to version 3.6, you cannot install any version of Predictive Routing earlier than the 9.0.011.00 release.

1. Download the IP for release 9.0.011.00. For detailed instructions, see [Deploying: High Availability](#)
 1. On *each* node, in the new IP folder on that node, run the following command: `bash scripts/install.sh`
 2. Then, on the *primary* node, also in the new IP folder, run the following command: `bash/scripts/upgrade_gpr_services.sh`

At this point, release 9.0.011.00 is installed and your Mongo DB version remains at 3.2.
2. Run the following commands to update your data to the correct format for Mongo DB 3.6:
[root@node-1 ha-scripts]# `docker exec -ti <tango_container_id> /bin/bash` (insert the correct value for your tango container id)

`cd src/gpr`

`MODE=prod python common/scripts/versioning/upgrade_40a_users.py`

`MODE=prod python common/scripts/versioning/upgrade_41a_accounts.py`

The upgrade to version 9.0.011.00 is now completed.
3. To upgrade the database itself, now run the following scripts:
 1. In the new IP folder on the *primary* server, run the following command **two** times (the first time it upgrades the Mongo DB cluster to version 3.4 and then to version 3.6): `bash scripts/upgrade_to_mongo36.sh`
 2. Then in the new IP folder, run the following script: `bash scripts/restart.sh`