Genesys Predictive Routing

default-predictor and <predictor_name> Section

12/14/2025

default-predictor and <predictor_name> Section

> ### Important
> All options in this section can be configured either in the **[default-predictor]** section or the **[<predictor_name>]** section, where *<predictor_name>* is the user-configured name of a specific Predictor.

Some functionality has multiple options controlling the desired behavior:

- Agent Occupancy Options
- Agent Holdout Options
- Dynamic Interaction Priority Options

Agent Occupancy Options

Agent occupancy is the percentage of time that an agent is working while logged in, a service objective that can be specified when building a staffing forecast. Agent occupancy data is taken from Stat Server by URS using the SData function. Stat Server collects agent occupancy data using the StatAgentOccupancy statistic. The routing strategy filters agents by occupancy in the ScoreIdealAgent callback subroutine. The agent occupancy results are used to sort the agents in the target agent group; over-occupied agents drop down lower in the sorted list.

max-agent-occupancy-factor
max-agent-occupancy-threshold
min-agent-occupancy-factor
min-agent-occupancy-threshold
use-agent-occupancy

Agent Holdout Options

Agent hold-out enables you to have an interaction wait a specified time, even when an agent has become available, if the available agent is has a low score for the interaction and there is a chance a better-matched agent might become available within the configured time window.

initial-threshold-timeout
score-base-threshold
threshold-relaxation-step
threshold-relaxation-timeout

Dynamic Interaction Priority Options

If an interaction has a low score for all targeted agents, it can stay in a queue for a long time. To avoid such situations, you can configure a schedule for incremental priority increases. The schedule is set once for each interaction processed by GPR. The following options control interaction priority increments.

> ### Important
>
> If you already use priority increments for the strategy into which you are inserting the GPR subroutines, you doo not need to configure these options. If you are using priority increments only for predictive routing, use the following options to configure it.

priority-increment
priority-init-interval
priority-interval
set-dynamic-priority

- ab-test-gpr-off-period
- ab-test-gpr-on-period
- ab-test-time-slice
- context-id-key
- default-agent-score
- initial-threshold-timeout
- login-status-expression
- max-agent-occupancy-factor
- max-agent-occupancy-threshold
- max-score

- min-agent-occupancy-factor
- min-agent-occupancy-threshold
- platform-scoring-url
- priority-increment
- priority-init-interval
- priority-interval
- prr-mode
- score-base-threshold
- set-dynamic-priority
- setreadycondition-timeout

- threshold-relaxation-step
- threshold-relaxation-timeout
- use-action-filters
- use-agent-occupancy
- use-crm-query
- use-keepqueue
- use-login-status
- use-setreadycondition

ab-test-gpr-off-period

Default Value: 0
Valid Values: 0, positive integers
Changes Take Effect: On the next interaction
Dependencies: prr-mode, ab-test-gpr-on-period

Introduced: 9.0.018.01

Specifies the period of time, in seconds, that skills-based routing should route interactions when using time-sliced A/B test mode. This option alternates with the **ab-test-gpr-on-period** option, with each routing method used for the length of time you specify in the corresponding option. The split percentage is calculated based on the ratio of the two option values.

Important: These options take effect ONLY when the value set for the **prr-mode** option is `ab-test-time-sliced`.

For each interaction, the **gpmUse** KVP records whether GPR (gpmUse = 1) or skills-based routing (gpmUse = 0) routed the interaction.

If either the **ab-test-gpr-on-period** or **ab-test-gpr-off-period** option is set to 0, or has a negative or invalid value, GPR defaults to a 50/50 time split with the duration of each routing mode defined by the value of the **ab-test-time-slice** option.

ab-test-gpr-on-period

Default Value: 0
Valid Values: 0, positive integers
Changes Take Effect: On the next interaction
Dependencies: prr-mode, ab-test-gpr-off-period
Introduced: 9.0.018.01

Specifies the period of time, in seconds, that GPR should route interactions when using time-sliced A/B test mode. This option alternates with the **ab-test-gpr-off-period** option, with each routing method used for the length of time you specify in the corresponding option. The split percentage is calculated based on the ratio of the two option values.

Important: These options take effect ONLY when the value set for the **prr-mode** option is `ab-test-time-sliced`.

For each interaction, the **gpmUse** KVP records whether GPR (gpmUse = 1) or skills-based routing (gpmUse = 0) routed the interaction.

If either the **ab-test-gpr-on-period** or **ab-test-gpr-off-period** option is set to 0, or has a negative or invalid value, GPR defaults to a 50/50 time split with the duration of each routing mode defined by the value of the **ab-test-time-slice** option.

ab-test-time-slice

Default Value: 0
Valid Values: 0, positive integers
Changes Take Effect: Immediately
Dependencies: prr-mode
Modified: 9.0.017.00

To enable 50/50 A/B testing, which when Predictive Routing and skills-based routing are turned on alternately for equal periods, set the **prr-mode** configuration option in the same section to `ab-test-time-sliced` and configure a value for this `ab-test-time-slice` option. The time slice you set is

how long each routing method is enabled before switching to the other.

Note: To configure non-50/50 A/B test periods, set the `ab-test-gpr-on-period` and `ab-test-gpr-off-period` options in such a way as to create the on-off split you want to use. For instructions, see the "Configure a non-50/50 comparison test time split" section in the "Routing scenarios using GPR" topic of the *Genesys Predictive Routing Deployment and Operations Guide*.

When enabled and set to one of the following values, time-sliced A/B testing works as follows:

- 0 (the default) - The SetIdealAndReadyCondition subroutine uses a formula based on the time of day and the date to determine whether particular interaction belongs to Group A or Group B. The change from A to B and then back to A, and so on, occurs hourly. Which period starts at midnight depends on the whether the day of the year is even or odd. If the date is an odd number, A starts at midnight; if it is an even number, B starts at midnight.

- A nonzero value (in seconds) - Genesys recommends that you do not set the value of this option to less than 3600 seconds in a production environment. The SetIdealAndReadyCondition subroutine uses a formula based on the configured time slice to determine whether a particular interaction belongs to Group A or Group B.

In release 9.0.017.00, the default value for this option was changed from 1741 to 0.

context-id-key

Default Value: No default value
Valid Values: A valid customer ID or the ANI
Changes Take Effect: Immediately
Modified: 9.0.017.01

Specifies the name of the user data key containing an ID for the current interaction. The Predictive Routing scoring engine uses this key to retrieve a record from an internal database of customer profiles (such as a CRM database) and uses features from the record to compute agents scores for the interaction.

To incorporate customer profile data into models for matching the agents, you must upload this data to the GPR Core Platform before you train a model. The URS ActivatePredictiveRouting subroutine attaches a `context_id` key to the scoring request body and provides the value of the user data key defined by this option as the `context_id` value.

- If you do not set a value for this option, the score response and the score log have an empty value for the context id.

- If you set the option value to ANI, the score response and the score log have the ANI as the value for the context id.

- If you set the option value to a valid user data key name, the score response and the score log have the value for this user data key for the context id.

- If you configure this option with an invalid user data key name, the score response and the score log have the value INVALID for the context id.

default-agent-score

Default Value: 0
Valid Values: max, median, global, min, 0
Changes Take Effect: On the next interaction

The option specifies the value the ScoreIdealAgent and isAgentScoreGood subroutines should use as the agent score for an interaction for those agents who belong to the target Agent Group but that GPR did not score. For example, an agent might be logged out, or in another status configured as unavailable, until after the scoring request it sent. If such an agent then becomes available before the interaction is routed, GPR assigns that agent the default score.

If an agent is assigned a score of 0, the agent is unlikely to receive an interaction from the queue.

Important: This option functions differently depending on the release of URS Strategy Subroutines you have deployed:

- In release 9.0.015.00 and higher, gpmAgentScore records the default score assigned to agents GPR did not score. The ScoreIdealAgent subroutine uses this value to sort the scores and the isAgentScoreGood subroutine compares it against any threshold you have configured to determine whether the agent is acceptable.

- In release 9.0.014.04 and lower, the gpmAgentScore user data KVP always contains the value 0 for unscored agents. The score specified in this option is used only when URS is sorting the agents in the target group according to their scores.

Valid values:

- **max** - Use the maximum score calculated for an agent in the target agent group.

- **median** - Use the median score calculated for the target agent group.

- **global** - Use the average global score for the agents in the target group.

- **min** - Use the minimum score calculated for an agent in the target agent group.

- **0** - Use the value 0 as the score.

initial-threshold-timeout

Default Value: 0
Valid Values: (integer) 0 - <max>
Changes Take Effect: On the next interaction processed

Defines a timeout, in seconds, during which the `isAgentScoreGood` URS callback function uses an initial minimum agent score, defined by the **[default-predictor].score-base-threshold** option, to match agents to an interaction. After this timeout expires, the minimum score required to allow an agent to handle the interaction is gradually decreased.

login-status-expression

Default Value: no default value
Valid Values: &((loginStatus>0&loginStatus<23)|loginStatus>23), &(loginStatus=4|loginStatus=9)
Changes Take Effect: On the next interaction

If you set the value of the **[default-predictor].use-login-status** option to **true**, the value of the **login-status-expression** option is added to the action_filters expression in the ActivatePredictiveRouting_v3 subroutine when the scoring request is created.

- &((loginStatus>0&loginStatus<23)|loginStatus>23) - Instructs the scoring engine to evaluate scores for those agents identified as part of the target group by a skill expression or an Agent Group name who are logged into the voice channel.

- &(loginStatus=4|loginStatus=9) - Instructs the scoring engine to evaluate scores for those agents identified as part of the target group by a skill expression or an Agent Group name who are ready to accept an interaction, or have status AfterCallWork on the voice channel.

The following numerical values correspond to the following agent login states:

- 4 - WaitForNextCall (Ready)
- 5 - OffHook
- 6 - CallDialing
- 7 - CallRinging
- 8 - NotReadyForNextCall
- 9 - AfterCallWork
- 13 - CallOnHold
- 16 - ASM_Engaged
- 17 - ASM_Outbound
- 18 - CallUnknown
- 19 - CallConsult
- 20 - CallInternal
- 21 - CallOutbound
- 22 - CallInbound
- 23 - LoggedOut

max-agent-occupancy-factor

Default Value: 0.5
Valid Values: Float numbers between 0.0 and 1.0
Changes Take Effect: On the next interaction

If you set the value of the **use-agent-occupancy** option to **true**, and the value of the agent occupancy statistic is higher than the threshold specified in the **max-agent-occupancy-threshold** option, the ScoreIdealAgent subroutine multiplies the score received for each agent for the current interaction by a coefficient defined by this option.

max-agent-occupancy-threshold

Default Value: 100.0
Valid Values: Float numbers between 0.0 and 100.0
Changes Take Effect: On the next interaction

If you set the value of the **use-agent-occupancy** option to **true**, the ScoreIdealAgent subroutine compares the value of the occupancy statistic with the value you set in this option. If the occupancy value is higher than the specified threshold, the subroutine multiplies the score received for an agent for the current interaction by a coefficient defined in the **max-agent-occupancy-factor** option.

max-score

Default Value: 100
Valid Values: (integer) 1 - <max>
Changes Take Effect: On the next interaction processed

Defines the maximum score that an agent can be assigned for an interaction. The value of this option is used by the `ScoreIdealAgent` callback function to re-scale the agent score as the distance from an ideally matched agent for the interaction (assumed by URS to be 0).

The value you set should correspond to the largest possible value returned by this Predictor from the scoring engine. To function properly, this value must be consistent with the value configured for the Predictor **Score expression** field. Because the GPR scoring engine and URS have different scales, you might need to adjust returned scoring values using the **Score expression** field in the Predictor configuration. See the instructions for how to configure this field in the "Creating and Updating Predictors" topic in the *Genesys Predictive Routing Help* for more information.

To take advantage of the most precise values, set **max-score** to 10000 and the value for Score expression in the Predictor configuration to 10000 * p_score. (*p_score* is a term used in the GPR documentation to indicate the raw score returned from the scoring engine. It is not in any way derived from or related to the statistical term *P value*.) For example, if scores range from -4 to 10, use the following **p_score** - ((p_score + 5) / 16) * 100.

min-agent-occupancy-factor

Default Value: 2.0
Valid Values: Float numbers higher than 1.0
Changes Take Effect: On the next interaction

If you set the value of the **use-agent-occupancy** option to **true**, and the value of the agent

occupancy statistic is lower than the threshold specified in the **min-agent-occupancy-threshold** option, the ScoreIdealAgent subroutine multiplies the score received for an agent for the current interaction by a coefficient defined by this option. If the resulting adjusted score is higher than the value specified in the **max-score** option, the adjusted score is set to the value of the **max-score** option.

min-agent-occupancy-threshold

Default Value: 0.0
Valid Values: Float numbers between 0.0 and 100.0
Changes Take Effect: On the next interaction

If you set the value of the **use-agent-occupancy** option to **true**, the ScoreIdealAgent subroutine compares the value of the occupancy statistic with the value you set in this option. If the occupancy value is lower than the specified threshold, the subroutine multiplies the score received for an agent for the current interaction by a coefficient defined in the **min-agent-occupancy-factor** option.

platform-scoring-url

Default Value: none
Valid Values: A valid GPR API scoring endpoint + a valid Predictor ID
Changes Take Effect: On the next interaction processed

The ActivatePredictiveRouting strategy subroutine in URS uses the URL defined by this option as the HTTP address to send scoring requests to the GPR API. This URL should be the value for the **[default].platform-base-url** option with *<predictor_name>*/score appended.

You can locate the Predictor ID in messages returned from the GPR API or in the browser URL address when you are in the GPR web application with the page for the desired Predictor open.

priority-increment

Default Value: 1
Valid Values: (integer) any integer
Changes Take Effect: On the next interaction

Specifies the increment by which priority is increased each time.

priority-init-interval

Default Value: 300
Valid Values: (integer) any non-negative integer
Changes Take Effect: On the next interaction

Controls the time interval, in seconds, the strategy waits before starting to increment priority for a queued interaction.

priority-interval

Default Value: 10
Valid Values: (integer) any integer greater than 5
Changes Take Effect: On the next interaction

Specifies the time period, in seconds, between priority increments for a queued interaction.

prr-mode

Default Value: off
Valid Values: prod, off, ab-test-time-sliced, dry-run
Changes Take Effect: Immediately

Specifies whether an instance of Predictive Routing should run as a production instance or as a test instance.

- **prod** - All the interactions that pass through the ActivatePredictiveRouting strategy subroutine are processed using Predictive Routing.

- **off** - No interactions use Predictive Routing.

- **ab-test-time-sliced** - The periods of time when Predictive Routing and skill-based routing are alternately turned on. The duration of each period is configured in the in the **ab-test-time-slice** option located in the same section as this **prr-mode** option.

- **dry-run** - Predictive Routing scores agents for your interactions, but does not use the scores for routing.

score-base-threshold

Default Value: 0
Valid Values: (integer) 0 - <max>
Changes Take Effect: On the next interaction processed

This option defines the initial minimum agent score required for an agent to be considered a match for an interaction. After the timeout defined by the **[default-predictor].initial-threshold-timeout** option expires, the minimum score required to handle the interaction is gradually decreased. If you set the value to 0, no initial minimum score is required and agents with any score are considered for an interaction.

set-dynamic-priority

Default Value: False

Valid Values: True, False
Changes Take Effect: On the next interaction

Specifies whether dynamic priority interaction handling is enabled and handled in the GPR subroutines. When set to true interaction priority is incremented based on the settings configured for the other priority options. When set to false, dynamic priority interaction handling is not set by the Predictive Routing subroutines. If dynamic priority parameters are set elsewhere in the strategy, the option must be set to false.

setreadycondition-timeout

Default Value: 600 (seconds)
Valid Values: Any positive integer
Changes Take Effect: On the next interaction

Warning: If you need to change the value of this option, contact Customer Care to ensure that the value you specify is compatible with your environment.

Defines a timeout value that sets the maximum delay, in seconds, between the moment when URS receives an Event from T-Server and when the IsAgentScoreGood subroutine is called. If the delay is greater than the value set in this option, Predictive Routing considers that the call has waited in the IsAgentScoreGood subroutine for the configured period and routes the call to an agent in the target group without checking value configured in the **[default-predictor].score-base-threshold** option.

threshold-relaxation-step

Default Value: 1
Valid Values: (integer) 1 - <value of the **max-score** option>
Changes Take Effect: On the next interaction processed

Defines an increment by which, while an interaction remains queued, the minimum agent score required to match the interaction is decreased after each period defined by the value of the **[default-predictor].threshold-relaxation-timeout** option, following the initial period defined by the **[default-predictor].initial-threshold-timeout** option.

threshold-relaxation-timeout

Default Value: 1
Valid Values: (integer) 1 - <max>
Changes Take Effect: On the next interaction processed

This option defines a timeout, in seconds, after which the minimum agent score required for matching an interaction is decreased by the amount defined by the value of the **[default-predictor].threshold-relaxation-step** option.

use-action-filters

Default Value: false
Valid Values: true, false
Changes Take Effect: Immediately

Note: In a hybrid environment, this option should *always* be set to `false`.

- true - URS uses a skill expression or Agent Group names taken from the action_filters field in the scoring request.

- false - URS checks with the Stat Server for the target list of agents, as specified in the **login-status-expression** option, and adds the target Agent IDs to the scoring request.

Note: If **[default-predictor].login-status-expression** is set to &(loginStatus=4|loginStatus=9), indicating that the agents who are in the Ready state or ACW state (for voice calls) are the designated target agents, then the GetActionFilters subroutine uses a custom statistic called RStatGPRAgentsReadyOrACWvoice. This custom statistic is provided in the **object.kvlt** file in the URS Strategy Subroutines IP.

use-agent-occupancy

Default Value: false
Valid Values: true, false
Changes Take Effect: On the next interaction

The value you set for this option determines whether the ScoreIdealAgent subroutine checks for agent occupancy and adjusts the agent's score for an interaction to increase, or decrease, the probability that the agent receives the interaction. If you set the value of this option to **true** and the value of statistic StatAgentOccupancy is above a threshold specified by the **max-agent-occupancy-threshold** option value, the subroutine adjusts the agent score lower. The score is multiplied by a coefficient defined in the **max-agent-occupancy-factor** option. Similarly, if the agent occupancy is lower than the value of the **min-agent-occupancy-threshold** option value, it is adjusted higher. The score is multiplied by a coefficient defined in the **min-agent-occupancy-factor** option. If the adjusted score exceeds the value specified by the **max-score** option, its value is set to the maximum score specified for the Predictor.

use-crm-query

Default Value: true
Valid Values: true, false
Changes Take Effect: Immediately

Option name reserved for future use.

use-keepqueue

Default Value: true
Valid Values: true, false
Changes Take Effect: On next interaction

Specifies whether a higher-priority interaction, entering a queue, is answered before interactions that arrive after it. If the high-priority interaction has a score too low to satisfy the criteria configured for routing, setting this option to true can block routing of higher-scoring but lower-priority interactions.

- **true** - If a higher priority interaction enters a queue, it must be routed to a suitable agent before the interactions already waiting are routed.

- **false** - Interaction priority is not considered when routing waiting interactions to a suitable agent. Preserves routing behavior used before release 9.0.018.00.

use-login-status

Default Value: false
Valid Values: false, true
Changes Take Effect: On the next interaction

Set the value of this option to **true** to have the value of the **[default-predictor].login-status-expression** option added to the action_filters expression in the ActivatePredictiveRouting_v3 subroutine when the scoring request is created.

Important: Genesys recommends that you set this option to **true** and provide a valid value for **[default-predictor].login-status-expression** to reduce the number of agents for whom scores are evaluated. The value **false** should be used only for debug and troubleshooting purposes in a staging environment.

use-setreadycondition

Default Value: false
Valid Values: true, false
Changes Take Effect: On the next interaction

If option is set to **true**, the strategy executes calls to the isAgentScoreGood subroutine, which temporarily removes low-scoring agents from consideration for routing. If option is set to **false**, the strategy does not execute calls to the isAgentScoreGood subroutine and, as a result, the interaction is routed to the agent with the highest availability; that is, the one who has been waiting longest for an interaction.

- *Low-scoring agents* - Agents with scores for the current interaction lower than time-dependent minimum required score, which is defined by the following options: **[default-predictor].initial-threshold-timeout**, **[default-predictor].threshold-relaxation-timeout**, **[default-predictor].threshold-relaxation-step**.

- *Temporary* - Until the agent's score for the interaction becomes higher than the time-dependent minimum required score.

Important: This option takes effect only when the **[default-predictor].prr-mode** option is set to **prod** or to **ab-test-time-sliced** for the same predictor.