



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.

Real-time Reporting with Pulse

Statistic Properties

6/3/2026

Contents

- 1 Alias
- 2 Display Alias
- 3 Description
- 4 Display Format
- 5 Filter Out LoggedOut Agents
- 6 Filters
- 7 Formula
- 8 GroupBy
- 9 Insensitivity
- 10 Notification Mode
- 11 Notification Frequency
- 12 Statistic Type
- 13 Time Profile
- 14 Time Range
- 15 What do I do next?



- Supervisor

Properties of statistics that you can use in your templates and reports.

Related documentation:

-
-

When you select a statistic within the template wizard, Genesys Pulse displays the values of the statistic properties that are described below.

Tip

You can modify a statistic definition while defining a template. See Standard Report Templates.

Alias

The Alias must be a unique name that represents the technical name of the statistic. Use an ASCII letter for the first character.

Display Alias

The Display Alias is the name displayed on the report.

Description

The Description provides the functional meaning of the statistic.

Display Format

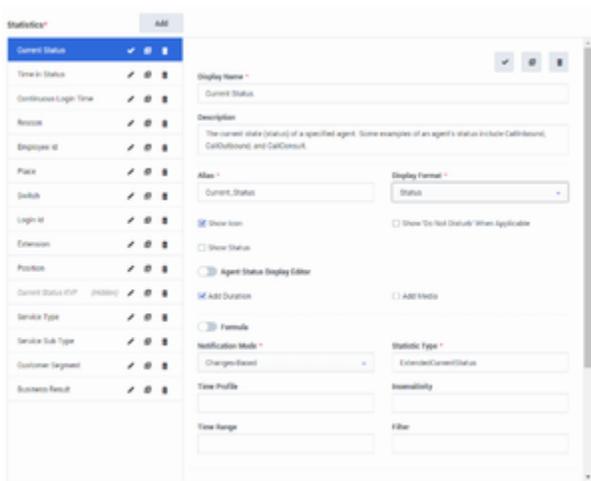
The Display Format specifies whether values are shown as time or numbers, and, if numbers, the

number of decimal places. Depending on the statistic you chose, the available formats in the drop-down list are time-based or numerical.

List of Values: Time, Integer, Number, Percent, String, Status

Starting with release 9.0.0, new **Status** display format is available for statistic with Statistic Type = ExtendedCurrentStatus. The **Status** display format allows to select the current agent status properties that you want to display:

- Show Status
- Show Icon
- Add Duration
- Add Media
- Show 'Do Not Disturb' when applicableAgent Status Display Options



Starting with release 9.0.001, the Advanced Display Options Editor is available for the Status display format. It allows to fully replace the standard representation with a user-designed representation based on the JavaScript string literals syntax.

The following variables can be used to construct the status to display:

- `${status}`-status name
- `${media}`-media name
- `${duration}`-duration of the current status in the time format (hh:mm:ss)
- `${dnd}`-duration of the "Do Not Disturb" status in the time format (hh:mm:ss). The Show "Do Not Disturb" When Applicable checkbox must be selected to display the Do Not Disturb status and duration even if the `${dnd}` variable is defined in the custom format via the Editor.

When the `${media}` or `${dnd}` value is unavailable, it is displayed as Undefined. To avoid this, you can specify conditions and use ternary operators.

Examples:

Display Format	Displayed data example
<pre> <code> $\{\{media \&\& \!dnd ? media + " \{\{!\}\}\{\{!\}\} " : ""\}\{\{status\}\}\{\{duration\}\}$</code> </pre> <p>(standard format representation when all Status properties are checked)</p>	<pre> <code> voice \{\{!\}\}\{\{!\}\} WaitForNextCall(00:23:12) or LoggedOut(01:20:15) or DoNotDisturb(00:03:10) </code> </pre>
<pre> <code> $\{\{duration + " in " + status\}\{\{media ? "(" + media + ")" : ""\}$ </code></pre>	<pre> <code> 00:23:12 in WaitForNextCall(voice) or 01:02:00 in LoggedOut </code> </pre>

Filter Out LoggedOut Agents

The screenshot shows the configuration interface for the 'Current Status' report. On the left is a list of properties including 'Time in Status', 'Login Time', 'Continuous Login Time', 'Reason', 'Employee Id', 'Place', 'Switch', 'Login Id', 'Extension', 'Position', 'Current Status KVP (Hidden)', 'Service Type', 'Service Sub Type', 'Customer Segment', and 'Business Result'. On the right, the configuration panel includes fields for 'Display Name' (Current Status), 'Description', 'Alias' (Current_Status), 'Display Format' (String), 'Show Agent State Icon' (checked), 'Formula' (unchecked), 'Notification Mode' (Changes-Based), 'Statistic Type' (ExtendedCurrentStatus), 'Time Profile', 'Time Range', 'Filter', 'Hide Statistic' (unchecked), 'Filter Out LoggedOut Agents' (checked), 'Additional Data' (Hardware Reason Codes and Software Reason Codes checked, User Data unchecked).

You can filter agents in the LoggedOut status from your reports within a report template. The Filter Out LoggedOut Agents option works in conjunction with the Current Status statistics, which are based on the ExtendedCurrentStatus stat type. Once enabled, the Filter Out LoggedOut Agents option is applied to all statistics in the template.

Filters

The screenshot shows the GAX Configuration interface. The top navigation bar includes 'GAX', 'Dashboard', 'Agents', 'Configuration', 'Routing Parameters', and 'A'. Below the navigation bar, the breadcrumb path is 'Home > Applications > Applications > Stat_Server Properties'. The left sidebar contains a list of configuration categories: 'General', 'Connections', 'Ports', 'Tenants', 'Options', 'Permissions', 'Dependencies', and 'Application Options'. The 'Application Options' category is selected and expanded. The main content area is titled 'Application Options' and contains a table with the following rows:

<input type="checkbox"/>	Key
<input type="checkbox"/>	▼ Filters
<input type="checkbox"/>	Bronze
<input type="checkbox"/>	Bronze_Call
<input type="checkbox"/>	Bronze_Chat
<input type="checkbox"/>	Bronze_Email
<input type="checkbox"/>	Bronze_fax
<input type="checkbox"/>	Bronze_sms
<input type="checkbox"/>	Bronze_webcallback

The Filters represent statistical filters that define restrictive conditions on actions used while calculating the statistic.

The list of Filters is available in the Configuration section of Genesys Administrator Extension (GAX). This view is available in Stat Server application options used by Genesys Pulse.

Your account must have privileges to access this section.

Within GAX, you can add, edit or delete a filter.

Filters Example

Suppose that you want to filter calls based on language: If the enterprise set up the key Language to identify language and the value Spanish for callers who speak Spanish, you could use the PairExists UserData function to search for calls with attached data in the Language/Spanish key-value pair.

On the Options tab of the Stat Server Properties screen, you could add a SpanishLanguage option in

the [Filters] section and specify filtering for calls with attached data containing the key "Language" and the value "Spanish".

The example would have SpanishLanguage in the Name field and PairExists("Language","Spanish") in the Value field.

Now, when an agent attaches the "Spanish/Language" key-value pair to calls from a desktop application, the calls are filtered out of statistical calculations.

Formula

The screenshot shows the 'Statistics *' configuration interface. On the left is a list of statistics, with 'Answers' selected and highlighted in blue. On the right is the configuration pane for the 'Answers' statistic. The 'Display Name' is 'Answers', and the 'Description' is 'The total number of dialing attempts initiated by a Campaign Manager with a call result of Answer (when a call is answered by a human voice). In some contact centers, the call result can also mean Right'. The 'Alias' is 'Campaign_Answers' and the 'Display Format' is 'Integer'. The 'Formula' checkbox is checked and circled in black. At the bottom right, there is a 'Save' button.

From the statistic detail pane, you can create or customize statistics by creating a formula.

The formula uses a javascript-based syntax, which lets you calculate expressions with values given by other statistic and use functions provided by Genesys for more specific calculations. For example, you can calculate the ratio of the calls abandoned to the calls offered in your queue to measure the percentage of abandoned calls in your queue.

Genesys Pulse assumes the offered calls are defined by a statistic alias Offered and the abandoned calls are defined by a statistic alias Abandoned.

The formula must return a Result value to be valid and can access any statistics of the template with the following syntax: `Data..Value`

All formulas must contain an assignment for the Result variable (for example, `Result=`). The Result of the formula calculation is the final value of this variable.

For example, here is a formula using the function `G.GetAgentNonVoiceStatus()`:

```
Result = G.GetAgentNonVoiceStatus(Data.Current_Status.Value, email);
```

GroupBy

The top screenshot shows a table titled "CallInternal (group by language and segment)". The table has four columns: Name, CallInternal, Segment, and Language. The data is grouped by Name. The first group is "Green, Anna" with 9 rows. The second group is "Qwerty, Mary" with 6 rows.

Name	CallInternal	Segment	Language
Green, Anna	9	N/A	N/A
Green, Anna	8	N/A	N/A
Green, Anna	1	Silver	N/A
Green, Anna	3	N/A	English
Green, Anna	1	Bronze	English
Green, Anna	2	Gold	English
Green, Anna	1	Silver	English
Green, Anna	1	N/A	Russian
Qwerty, Mary	6	N/A	N/A

The bottom screenshot shows a zoomed-in view of the table, focusing on the "Qwerty, Mary" group. The table has four columns: Name, CallInternal, Segment, and Language. The data is grouped by Name. The first group is "Green, Anna" with 6 rows. The second group is "Qwerty, Mary" with 3 rows.

Name	CallInternal	Segment	Language
Green, Anna	3	N/A	English
Green, Anna	1	Bronze	English
Green, Anna	2	Gold	English
Green, Anna	1	Silver	English
Green, Anna	1	N/A	Russian
Qwerty, Mary	6	N/A	N/A
Qwerty, Mary	5	N/A	N/A
Qwerty, Mary	1	Silver	N/A

The GroupByColumns option facilitates provisioning of Pulse widgets and enable multi-dimensional data presentation (GroupBy capability) in Grid widgets and in the Data view on an expanded widget tab.

GroupBy columns should contain valid unique aliases, separated by comma.

StatType, used with GroupBy capability, must define the following additional attributes:

- **GroupBy**—contains grouping expressions separated by comma
- **GroupByColumns**—contains aliases for grouping expressions separated by comma

Example:

To monitor the number of internal calls grouped by Language and Segment, instead of explicitly defined and applied filters to a metric, define the GroupBy and GroupByColumns for the StatType in the Stat Server:

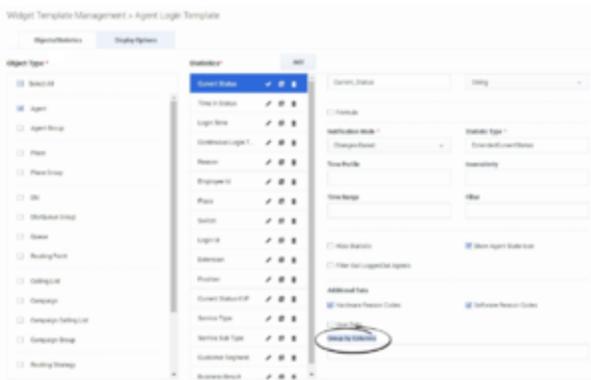
```
[Total_Calls_Grouped]
Category=TotalNumber
GroupBy=GetString(UserData,"Language"), GetString(UserData,"Segment")
GroupByColumns=Language, Segment
MainMask=CallInternal
Objects=Agent
Subject=DNAAction
```

Include a metric based on this StatType in your widget template and make sure **Group By Columns** field is filled correctly with "Language, Segment" string.

Important

- All metrics in the StatType GroupByColumns attribute aliases must be included in the **Group by Columns** field or all GroupBy data is ignored.
- GroupBy support relies on StatServer functionality that was introduced in release 8.5.103.
- Pulse supports snapshots when GroupBy is applied to the same expression for either all or no statistics.
- When you use a GroupBy expression that involves user data which is changed during call, that call will be counted in the group with the old value (or no value) AND in the group with new value. Therefore, for the TotalNumber statistics, if you add up all values for all groups, you have more than the statistic's total value.

You can use **Group by Columns** option when you create a template:



InSensitivity

InSensitivity describes a condition for Stat Server to send updates of statistical values to its clients. An increase in the value of this parameter usually decreases network traffic, but it also reduces reporting accuracy, because values are not updated as frequently. This setting is not visible in Stat Server configuration, but rather, clients pass its value to Stat Server along with each statistic request.

InSensitivity plays no role for reset-based statistics. For time-based or change-based notification mode, Stat Server only reports the recalculated value if the absolute value of the difference between the previous value and the recalculated value or its percentage ratio to the recalculated value is at least equal to the number specified by InSensitivity.

For example, if the result has a long integer data type—as is the case for statistics measuring time—Stat Server uses the absolute difference in values for comparison. Given an InSensitivity setting of 5 in this case, Stat Server sends the recalculated result to its client when the absolute value of the difference between the new and old result is at least 5 (seconds, usually).

Notification Mode

The Notification Mode determines when Stat Server sends updated statistical values. These are the valid options:

- **Time-Based**—Select this Notification Mode to instruct Stat Server to recalculate the statistic by the frequency displayed in Notification Frequency property. Stat Server sends a new value to Genesys Pulse only when the absolute difference from the last reported value exceeds the InSensitivity property.
- **Change-Based**—Select this Notification Mode to instruct Stat Server to notify Genesys Pulse about changes immediately.
- **No Notification**—Select this option to instruct Stat Server to not report updates. Updates are turned off in this case.
- **Reset-Based**—Select this Notification Mode to instruct Stat Server to report Genesys Pulse value right before setting it to zero (0). CurrentState statistics cannot be requested with Reset-Based notification mode.

Notification Frequency

Use Notification Frequency to set how often, in seconds, Stat Server recalculates the statistic and notifies Genesys Pulse if the statistic changes by more than the valued displayed in the Insensitivity field. This field is only used when a Time-Based Notification Mode is selected for the statistic.

Statistic Type

The screenshot shows the GAX Configuration interface. The top navigation bar includes GAX, Pulse, Agents, Configuration (highlighted), Routing Parameters, and Administration. The breadcrumb trail is Home > Applications > Applications > Stat_Server Properties. On the left, a sidebar menu lists General, Connections, Ports, Tenants, and Application Options (selected). The main content area is titled 'Application Options' and contains a table with 11 rows. Each row has a checkbox, a key name, and a dropdown arrow.

<input type="checkbox"/>	Key	▼
<input type="checkbox"/>	AbandCallsPercentage	▼
<input type="checkbox"/>	AbandonedFromRinging	▼
<input type="checkbox"/>	AbandTime	▼
<input type="checkbox"/>	ACW_Time_Inbound	▼
<input type="checkbox"/>	ACW_Time_Other	▼
<input type="checkbox"/>	ACW_Time_Outbound	▼
<input type="checkbox"/>	AgentLogInTime	▼
<input type="checkbox"/>	AgentReadyTime	▼
<input type="checkbox"/>	Agents_CurrentNumber	▼

The mandatory Statistic Type displays the parameters that define the statistic type within Stat Server.

The list of Statistic Types available in the environment should be accessible through GAX within the Configuration section. You can view them in the Application Options of the Stat Server application used by Genesys Pulse.

Your account must have privileges to access the Configuration section.

Within GAX, you can add, edit, or delete a statistic type.

This list should be the same as the list of statistic types detailed in the Genesys Pulse templates spreadsheet.

Time Profile

The screenshot shows the GAX Configuration interface. The navigation bar includes GAX, Dashboard, Agents, Configuration (selected), Routing Parameters, and Administration. The breadcrumb trail is Home > Applications > Applications > Stat_Server Properties. The left sidebar contains a menu with options: General, Connections, Ports, Tenants, Options, Permissions, Dependencies, and Application Options (selected). The main content area is titled 'Application Options' and contains a table with the following data:

<input type="checkbox"/>	Key
<input checked="" type="checkbox"/>	OneDay,Growing
<input checked="" type="checkbox"/>	OneHour
<input checked="" type="checkbox"/>	OneHour,Growing
<input checked="" type="checkbox"/>	OneHourSlide,Sliding
<input checked="" type="checkbox"/>	OneMinute
<input checked="" type="checkbox"/>	OneMinute,Growing
<input checked="" type="checkbox"/>	SinceLogin
<input checked="" type="checkbox"/>	SinceLogin,SinceLogin

Use the Time Profile to define the Time Profile for the statistic and specify the interval over which historical aggregate values are calculated. All time profiles are defined as configuration options in the Time Profiles of the Stat Server Application object in Genesys Configuration.

The list of Time Profiles available in the environment should be accessible in the GAX Configuration section. This view is available in Stat Server application options used by Genesys Pulse.

Your account must have privileges to access this section.

Within GAX, you can add, edit, or delete a Time Profile.

The Time Profile contains four main types:

- Growing
- Sliding
- Selection
- SinceLogin

Time Profiles Examples

- **Default, Growing**—The Default time profile uses a Growing interval type and resets statistics to zero (0) every night at midnight. The default value is set to 00:00.
- **LastHour, Sliding**—The LastHour time profile uses a Sliding interval type and tracks the last hour of activity with a sampling taken every 15 seconds. The default value is set to 3600:15.
- **SinceLogin, SinceLogin**—SinceLogin resets statistics to zero (**0**) at the moment of agent login. Statistics continue to accumulate as long as the agent is logged into (any) DN. The SinceLogin interval type aggregates statistical data only for agent-object statistics.
- **Shifts, Growing**—A time profile named Shifts resets statistics to zero when shifts change at 3:00 AM, 7:00 AM, 11:00 AM, 1:00 PM, 7:00 PM, and 1:00 AM. The default value is set to 3:00 +4:00, 13:00 +6:00.

Time Range

The screenshot shows the GAX Configuration page for Stat_Server Properties. The navigation bar includes GAX, Dashboard, Agents, Configuration (highlighted), and Routing Parameters. The breadcrumb trail is Home > Applications > Applications > Stat_Server Properties. On the left is a sidebar menu with options: General, Connections, Ports, Tenants, Options, Permissions, Dependencies, and Application Options (selected). The main content area is titled 'Application Options' and contains a table with the following rows:

<input type="checkbox"/>	Key
<input type="checkbox"/>	▼ TimeRanges
<input type="checkbox"/>	EWT_Announce_TR
<input type="checkbox"/>	Less_3sec
<input type="checkbox"/>	Range0-10
<input type="checkbox"/>	Range0-120
<input type="checkbox"/>	Range0-15
<input type="checkbox"/>	Range0-20
<input type="checkbox"/>	Range0-30

The Time Range specifies when to collect data for a limited set of statistics.

The list of Time Ranges is available in the Configuration section of GAX. This view is available in the options of the Stat Server application used by the Genesys Pulse solution.

Your account needs to have privileges to access this section.

Within GAX, you can add, edit, or delete a time range.

Time Ranges apply to statistics in following categories:

-
- TotalNumberInTimeRange
 - TotalNumberInTimeRangePercentage
 - CurrentNumberInTimeRange
 - CurrentNumberInTimeRangePercentage
 - ServiceFactor1
 - TotalTimeInTimeRange

Time Range Example

Suppose that you want to calculate the total number of calls answered within 30 seconds. To do so, enter Range0-30 in the Name field, and 0-30 in the Value field.

In this example, a Pulse statistic that calculates the total number of calls is based on the time range "Range0-30". If one call is answered after being in a queue for 25 seconds, a second call after 40 seconds, and a third call after 10 seconds, Stat Server counts only the first and third calls.

What do I do next?

You might want to learn more about:

- [Widget templates](#)
- [Report formulas](#)
- [Template function library](#)