



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.

# Designer User's Guide

## Routing Algorithms

10/19/2024

---

## Contents

- 1 Agent Loading
- 2 Agent Occupancy
  - 2.1 How it works
- 3 Expected Waiting Time
- 4 Load Balance
- 5 Most Skilled Agent
  - 5.1 How it works
- 6 Randomly
- 7 Round Robin
- 8 Time in Ready State



- Administrator

Learn about the different types of routing algorithms that you can select in the **Route Call** and **Route (Digital)** blocks.

### Related documentation:

- 

The Route Call and Route blocks allow you to select certain statistic types that can be used as routing algorithms. (For more information about statistic types, see [Statistic Types](#) on the [Statistic block](#) page.)

## Agent Loading

Selects agents within an agent group and calculates a vector based on three values:

- The number of busy DNs.
- The agents' time in Ready state (the same value as **Time In Ready State**)
- A random number.

## Agent Occupancy

Enables the routing engine to route interactions to the least occupied agent (the agent with the lowest occupancy rate). The occupancy rate is determined by the ratio between the time the agent has been busy since last login compared to the agent's total login time.

**Agent Occupancy** enables the routing engine to evaluate multiple available agents and select the least occupied agent, which balances the workload among available agents.

This statistic is defined and calculated when the agent logs in and can be used only in statistics that are applied for an agent. This statistic cannot be used with agent groups.

### How it works

Consider the following scenario:

- After login, Agent 1 was on a call for five minutes and was in a Ready state for five minutes. His occupancy is 50 per cent ( $5 / (5+5)$  ).

- 
- After login, Agent 2 was on a call for one minute and was in a Ready state for two minutes. His occupancy is 33 per cent ( $1 / (1+2)$ ).

When using **Agent Occupancy**, the routing engine distributes an incoming interaction to Agent 2, as he is the least occupied agent.

## Expected Waiting Time

Provides wait-in-queue estimates for the last interaction that entered a virtual queue. This statistic has been designed for the multimedia model, but assumes agents cannot handle more than one simultaneous non-voice interaction at a time.

This statistic applies to all types of media and is calculated and refreshed internally for the last 600 seconds.

## Load Balance

Load balancing between routing targets helps to ensure there is an equal distribution of interactions among queues, DN's, agent groups and queue groups. This statistic considers the number of calls distributed between objects, the percentage of busy agents, the expected waiting time, or other routing features.

When this statistic is defined, the routing engine automatically counts calls that are routed to different DN's and queues and adjusts the logic so there is an equal distribution of interactions across the specific DN's or queues.

## Most Skilled Agent

This routing algorithm finds the most skilled agent among all available agents who match the skills selected in the Route Call (or Route) block.

### How it works

Consider the following scenario:

- A **Route Call** block has three skills selected.
- Agent A has skill levels of 7/7/7 for those skills.
- Agent B has skill levels of 10/5/5, respectively.

The algorithm finds the agent with the highest *combined* skill among the selected skills. Therefore, if both agents are available, the algorithm selects Agent A as the most skilled agent.

---

## Important

This routing algorithm only works with *skill*-based routing; *skill expression*-based routing is not supported.

## Randomly

Controls the mechanism for selecting a default target. If the target list has more than one available target, this option functions as follows:

- If this option is set to **random**, the routing engine picks a target according to its internal rules.
- If this option is set to **FIFO**, the routing engine picks the first available target in the list.

## Round Robin

Quickly selects a target from an agent group by applying a round-robin algorithm. The routing engine searches through a list of agents in the target object and returns the first available agent (Agent 1). For the next call, the routing engine bypasses the previously selected agent and picks the next available agent.

When the routing engine reaches the end of the list of agents, it returns to the beginning of the list.

## Time in Ready State

Provides the total current time of an agent in Ready state. It is calculated based on availability - that is, there is no activity currently in progress on a particular DN and the agent has placed the DN in Ready state. **Time in Ready State** aggregates the total time for the state and this information can be used to build the target list.

You can use **Time in Ready State** to route interactions to the agent that has been waiting in Ready state for the longest time.