Types of Interaction Routing

Known types of routing of customer interactions using in contact centers

September 20, 2017

Nikolay Anisimov

In many contemporary contact centers, routing of customer interactions is central and the most important part of interaction processing. Indeed, in many cases, only customer service representatives (or agents) can solve customers' problems. Finding an agent with the right skills and capabilities to handle customer interactions is a key feature of the contact center. There are several types of routing based on different properties of customers and agents. In this article, we will shortly consider all known types of routing and closely related notions.

We consider different types of routing to create a universal model that would cover all known types of routing, as well as to enable us to invent new ones. More about this project can be found in this whitepaper.

Routing environment

Before examining types of routing, let us consider a general schema of routing, see Figure 1 below.

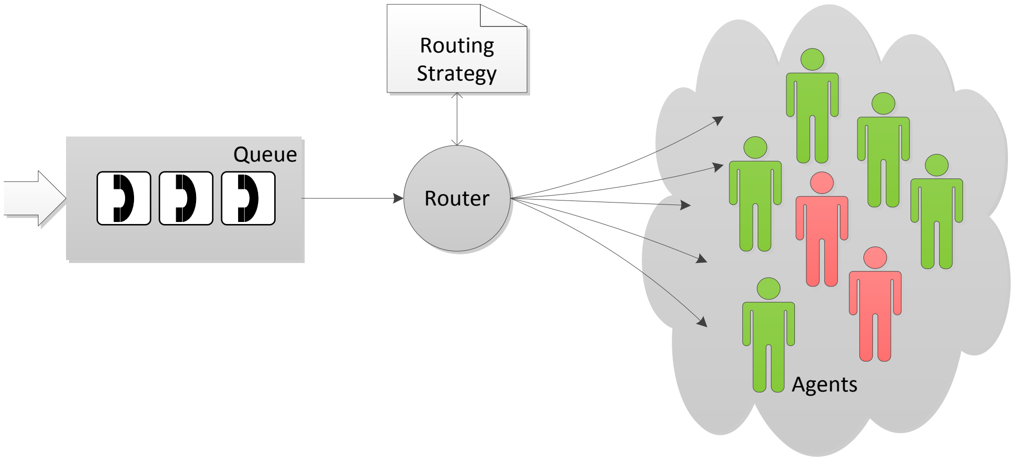

Figure 1: Simple schema of call routing

The central part is the router, a real-time engine that receives interactions from customers and distributes them to agents in accordance with a specific routing strategy. A routing strategy contains application logic that matches interactions to agents to find the most appropriate agent. Normally, a routing strategy is expressed in some formalism of scripting nature and the router interprets it in real time.

Calls are distributed only to those agents who are available (colored in green). An agent is unavailable if he/she is already engaged with a call or he/she is not in a ready state. Moreover, interactions and agents have some properties and decision based on which routing a customer interaction is made.

This routing schema is very simple and limited to simple-use case where the contact center application contains only routing interactions to agents. However, in many cases there is a need for more-sophisticated scenarios when an interaction passes through several stages. In these cases, a workflow model of resource management may help. See Figure 2 below.

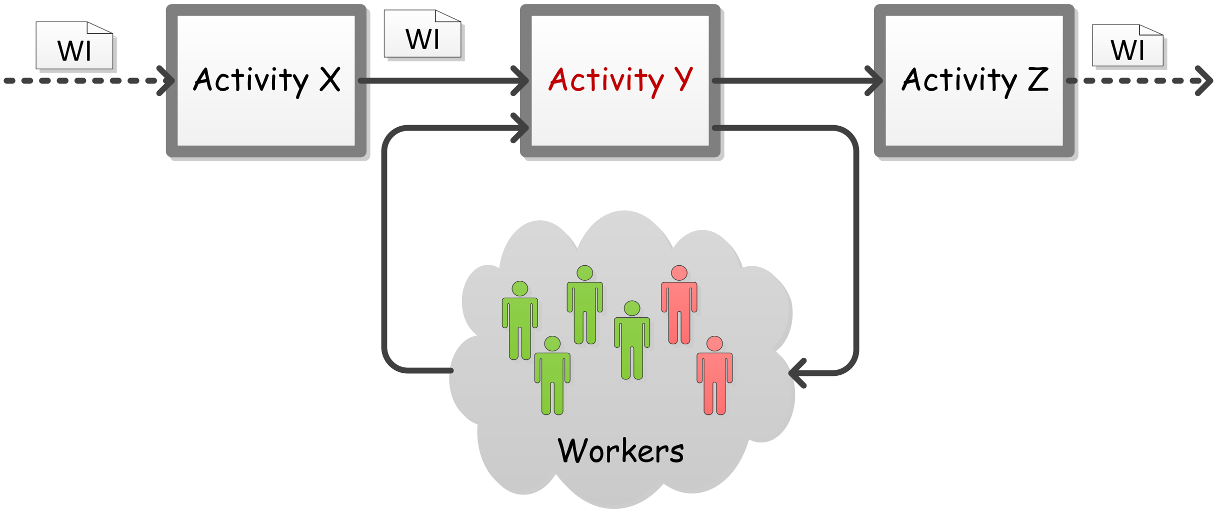

Figure 2: Workflow and resources (WI – work-item)

According to the workflow model, the whole process of work item processing is represented as a graph where nodes are activities. Each activity is a portion of work on the work item. Some activities may require involvement of people - workers. When a work item reaches such an activity point (activity Y in Figure), it enters the activity only when an appropriate worker is found and allocated to this work item. When the work item is processed by the activity, the allocated worker is not available for other work items. When the processing of the work item is finished by the activity, the work item proceeds to another activity and the worker is freed up and returns to the worker pool.

For both routing schemes, the routing procedure comprises two operations – matching and physical assignment of interaction and agent. The matching operation involves an interaction and determines the most appropriate agent in accordance with the routing strategy. Conversely when an agent becomes available, the matching procedure determines an appropriate interaction.

A physical assignment operation executes mechanisms that physically allocate agents to interactions. This operation may be complicated when agents are distributed and may require an agent reservation procedure.

It should be stressed that each type of routing takes a group of agents and reduce it to some subset. The subset may contain only one agent who is exactly the one needed for the routed interaction. If the subset is zero, that means that no agents are able/free to handle an interaction. If the subset contains more than one agent, we need to apply one more routing procedure to reduce the subset to one agent.

Known types of interaction processing

ACD and statistical-based routing

When interactions are routed to agents, in many cases agents' performances are taken into consideration. For example, in ACD an interaction is usually routed to the longest available agent (LAA), trying to even out agents' idle time[1]. To arrange such a type of routing, each agent is associated with a set of metrics. During contact center operation, these metrics are calculated, resulting in a set of statistics. One of the most popular agent metrics is agent occupancy, a percentage of time the agent is in a busy state, calculated on some chosen time interval.

Schematically, ACD-like routing is illustrated in Figure 3 below.

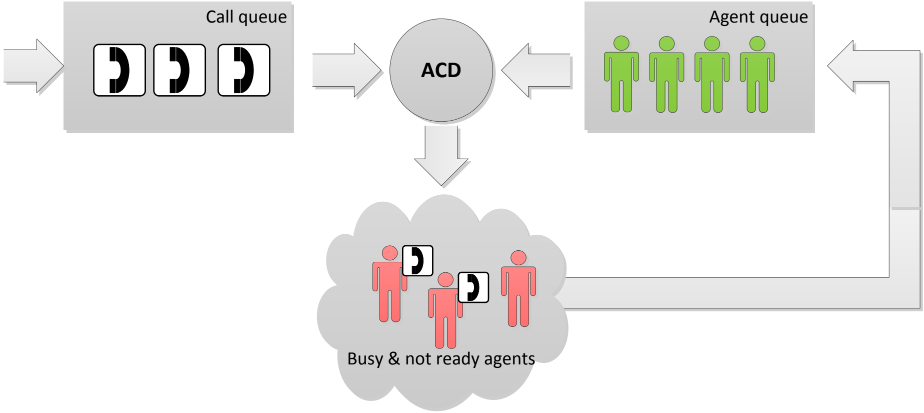

Figure 3: ACD-like routing

The schema contains two FIFO queues for calls and agents. Calls are arranged in the queue in accordance with entering time. Agents are arranged in accordance with the time agents have spent in a ready state (aka idle time). When there is at least one call in the call queue, and at least one agent in the agent queue, the first call and the first agent are assigned to each other and both leave the queues. We will say the agent is handling the call (and is busy) and the call is assigned to the agent. When the agent completes handling the call, he/she moves to a not-ready state. Usually in this state the agent is doing some call-related work (aka wrapping up) and this status is referred to as after-call-work (ACW). When the agent completes after call work and presses the ready button, he/she is placed back in the agent queue.

[1] This does not mean evening out agents' occupancy because, in general, interactions of different types may have different average handling times. Therefore, to even out agents' occupancy, we will need to use different metrics usually referred to as agents' busy factor or agents' workload.

Routing based on agent presence status

Normally, an interaction is routed to an agent who is able to process the interaction. First, the agent should be presented in the system. That means the agent should be registered in the system and presented in the system's database. Second, the agent should be logged into the system. Third, the agent should be ready to receive this type of interactions. To accomplish this, the agent's desktop is usually equipped with ready/not ready buttons on his/her phone set or softphone, enabling the agent to manifest his/her readiness to receive some type of interaction.

Skill-based routing

Skill-based routing (SBR) is the most popular type of routing in call- and contact centers. In short, SBR is a routing when interactions are distributed to agents in accordance with agent skills. There are many flavors of SBR depending on vendors' implementation. In this document, we will consider the most sophisticated one, known as interaction routing, based on skill expressions.

According to this type of routing, each agent is associated with one or several skills, collectively called an agent's skill set. Each skill is identified by its name that is normally self-explanatory. For example, skill names could be: "English", "Help Desk", "Opening New Accounts". A skill can be simple or complex. A simple skill indicates only whether an agent can do some sort of activity or not. For example, if an agent has a simple skill named "English" this means that the agent can speak English.

Complex skills, apart from their name, may have some properties. The most popular property is a skill level that is an integer number ranged within certain boundaries. For example, a level of "English" skill may vary from 1 to 10 where "10" is the highest level, meaning that the agent is fluent in English. Level "1" is the lowest level, which may mean that the agent can speak English, but very badly. Level "0" may be interpreted as the agent speaks no English.

The skill level of some agent can be set voluntarily by his/her manager, or set more formally based on results of aptitude tests.

A complex skill may have other properties, such as a date when the skill was last used , total time of experience with the skill, etc.

Routing of interactions is usually governed by a skill expression, an expression with skills and their parameters. The skill expression is applied to a set of agents and indicates a subset of agents whose skills meet the skill expression. For example, the skill expression *"English.level > 7 AND HelpDesk.level >= 2"* means that an agent should have the skill "English" with a level more than 7 and a skill "HelpDesk" with a level equal to or higher than 2. An agent who has a level of English skill equal to 9 and a level of HelpDesk skill equal to 3 satisfies the skill expression.

An example of SBR with skill expressions, virtual queues, and virtual agent groups is illustrated in Figure 4.

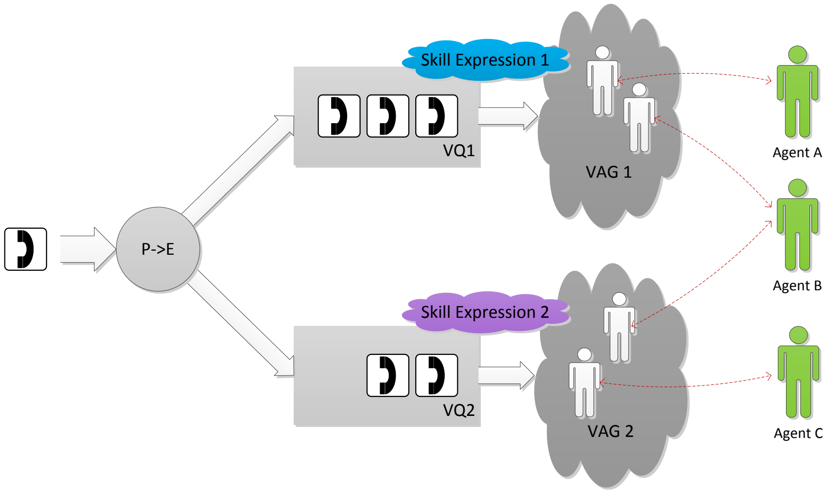

Figure 4: Skill-based routing with skill expressions

In this example, we configured two skill expressions. The configuration of each expression is implemented as virtual queue (VQ) and virtual agents group (VAG).

The routing works as follows. When an interaction enters the system it contains some set of properties. Based on these properties, the system decides what skill expression is appropriate for this interaction. A set of skill expressions is configured in a system as well as corresponding virtual queues and virtual agent groups. When a target skill expression is determined, the interaction is placed in a queue corresponded to this expression. If there is an agent whose skills satisfy this skill expression, the interaction is distributed to his/her workplace.

One can think of agents with multiple skills and satisfying multiple skill expression as agents staying in different queues simultaneously. When such an agent receives an interaction from some queue, he/she leaves the other queues. For example, the agent B in Figure 4 belongs to two virtual groups simultaneously.

It should be noted that despite the fact that SBR is the most popular in the industry, there is no standard specification or standard recommendation explaining how it works. This results in a situation when implementations of SBR in different contact centers differs dramatically.

Other types of routing

Composing routing strategies

Sequential application of different types of routing, such as skill-based routing and, if there are several agents with the same skills, proceeding to statistical (e.g. LAA) distribution to determine one agent.

If you take a closer look at the example of SBR with skill expressions from the previous section, then you will find that this routing is a composition of two routing strategies – skill-based and ACD-like ones. Indeed, at the first stage we identify a set of agents with skill sets satisfying some skill expression. At the second stage we arrange the selection of a single agent with the aid of an ACD-like queue.

Moreover, you will notice that the routing also comprises a presence-based component that requires an agent to be in ready state.

Pull and push modes

In contact center routing there are two types of routing modes – pull and push. In pull mode, an interaction is placed into a common pull (queue) where it could be pulled by agents. An agent decides him/her-self when to pull the interaction.

In push mode, an interaction is immediately pushed to a selected agent forcing him/her to deal with the call. Normally, an interaction is pushed when it is a real-time interaction, such as voice call or web chat when customers cannot wait and may abandon the call if not dealt with promptly. However, in some cases, a non-real-time interaction (aka deferred) also could be pushed to an agent.

Agent reservation

When an appropriate agent is found for new interaction, we should assign him/her to the interaction and make him/her unavailable for other interactions. This is not so easy, because many interactions may try to capture the same agent simultaneously. In this case there would be a conflict and this conflict should be resolved in some way without loss of performance. Indeed, imagine that ten interactions try to reserve the same agent (e.g. because he/she is the best) and only one is successful. The other nine interactions will waste valuable time in this competition that would result in total degradation of the system. This problem is well known as an agent reservation problem.

The opposite problem of the assignation of an interaction to several agents is completely symmetrical.

Random routing

Sometimes, it is reasonable to determine a target randomly. This makes sense when we have a set of similar targets (e.g. determined during previous routing stages), but we need only one. For example, if we applied LAA-based routing (see the previous section) and obtain five LAA agents, where each waited exactly for 30 seconds. Then, it is a reasonable way to select one agent randomly. The random choice provides more or less even distribution at no expense.

Cost-based routing

In this type of routing, a work item is routed to an agent whose cost is minimal. The cost could be calculated in different ways. For example, some agents may have a price for handling one call. Other agents may have an hourly rate. Naturally, this type of routing interactions should be distributed to less expensive agents.

Value-based routing

This type of routing is similar to cost-based routing, where we take into consideration values generated by agents, not expenses. It may be revenue-based routing where calls are distributed to agents who generate more revenue.

Organization-based routing

In this type of routing, an interaction is distributed to an agent who takes a place in some organizational structure. For example, routing could be arranged to agents who belong to some department, location, positions, etc. A typical contact center is configured with a simple organization structure comprising such concepts as agents, agent groups, and tenants.

Escalation routing strategies

An escalation occurs when the system cannot find a target for some interaction. In this case (maybe after a timeout) a less discriminative routing procedure could be applied.

Using a timeout

It may be that the application of some routing strategy results in no agents. The reason for this may be that all agents are busy or not available for some reason (e.g. logged out or not ready). Then, this routing policy suspends the interaction for a timeout in the hope the applied routing strategy will return a not empty agent set. In other words, the timeout policy queues the interaction and listens to changing agent statuses.

The value of a timeout may be finite or infinite. In the first case we will need to define an action to be executed when the timeout expires.

Multiple workers

This type of routing means assigning several workers to the processing of one interaction. Traditionally, an interaction is assigned to one agent at the moment they first handle it. This is explained by limitations of routing technologies. However, in many other areas of customer care, it is preferably to serve a customer through several workers with different specializations.

Role-based routing

In this type, an interaction is assigned to a user acting in a specific role. For example, an interaction may require a conventional agent with some skills and a manager for managing the process and to intervene into a conversation when needed.

Product-based routing

Sometimes we need to route an interaction to some person who deals with a certain product (e.g. MS Windows NT). The number of products could be astronomical (e.g. hundreds of thousands) and we should be able to determine from a customer the product and then find the right person who can consult with the customer about this product.

Social routing

According to this article, “social routing effectively channels the customer to the most appropriate service representative in the contact center. The consumer will be presented with agent options and will, in effect, select the best, most suitable customer service advisor themselves to address their query, complaint or information request”.

In other words, a customer who is connecting to a contact center is presented with a list of agents, each agent being described by his/her name, skills, biography, picture, agent rating (based on previous reviews), estimated waiting time, his/her organizational position, location, etc. The customer selects an agent and the interaction will be distributed to this agent.

In other words, in social routing, the customer explicitly participates in routing decisions.

Sales-based routing

This type of routing requires distributing a customer's interaction to an agent who handled a previous interaction with the same customer. This type of routing is needed for agents who play the role of salesmen and need to contact the same customer to continue working on the same deal (and finally to get a deserved commission). If, however, the agent is not available for some reason, the interaction should be distributed to some other member of the salesman's team.

Routing with conflicted agents

This type of routing is described in this blog where a solution is based on skill-based routing. This routing type is similar to the previous one, but with the ability to exclude 'conflicting' agents from the routing. "There are some agents who should not be offered calls from certain callers due to what the student referred to as a "conflict of interest.""

Procedure-based routing (hunting groups)

This type of routing reduces a group of agents to one agent to whom an interaction is routed. It works as follows. An interaction to be distributed is offered to several agents simultaneously. An agent who claims the interaction first receives the interaction. In telephony, such a type of distribution is called a hunting group, where an inbound call is offered to several agents by ringing their phones.

When using this type of distribution, we should envisage a situation when nobody claims the offered interaction within some predefined time period.

Routing with IVR treatment

Some routers allow inserting IVR treatments in routing strategies. This is useful when a routing strategy realizes that is has not enough information for effective determining a routing target. In this case, some IVR script is initialized that initiates a dialog with the customer to collect additional needed information. This information is fetched to the router and the routing strategy execution is resumed. Physically, such functionality is implemented as invoking IVR by the router that blocks execution of the strategy unless results from the IVR are received.

Customer segmentation

A routing procedure may depend on the customer segment (e.g. "platinum", "gold" and "silver"). For example, if a customer belongs to "platinum" segment, we immediately transfer him/her to an agent with the highest skills. If a customer belongs to "gold" we transfer him/her to IVR to collect some information and then route him/her to an agent. Finally, if a customer is "silver" we transfer him/her to IVR without any agent involvement.

Customer segment

Execution of a routing strategy may depend on a customer. Normally, all customers are partitioned into so-called customer segments depending on their importance for a business of contact center. A typical value of customer segment could be “Platinum”, “Gold”, and “Silver”. The segmentation could be done based on customer history (e.g. number of purchases in the internet shop), customer attributes and properties (e.g. balance of bank account, location, age, gender, etc.) For example, a platinum customer could be immediately routed to the most qualified agent who will resolve the customer’s problem. At the same time, a silver customer could be directed to self-service without any agent’s involvement.

Demographics-based routing

The demographic-based routing is explained with the following example.

“An example would be a customer who is a 65-year-old male from the Midwest who is calling customer service with an issue regarding the use of a product. Initially, the call will be presented to a product specialist who can assist in explaining how to use the product to the customer (e.g. using skill-based routing). Within that group of specialists, the call would then be routed to agents who have the most success working with 65-year-old males from the Midwest (demographic-based routing).”

Agent capacity model

The agent capacity model stemmed from introducing new media types. Applying the old rules for managing these media types resulted in the creation of some problems. The main use case that drove the invention of the agent capacity model was the inability of an agent who works with email to receive telephone voice calls. Indeed, the system considered an agent with email as busy and did not distribute a voice call to him/her. At the same time, an agent working with email (or other deferred media interactions) could easily accept a voice call, postponing email for a while.

The agent capacity model allows an agent to work with several interactions of different types, simultaneously. For example, some agents may work with two or three chat sessions at a time without sacrificing quality.

The agent-capacity model is usually defined as a set of rules, each rule being devoted to one media channel. A typical capacity rule looks like the expression: “the agent could accept a new voice call if he/she has no voice call and if he/she has one or several emails”.

Access control and complex tenancy

Multi-tenancy is a required system feature for hosted and cloud-based contact centers. A tenant is an organization that subscribes and consumes contact center functionality. Multi-tenancy means that several organizations may utilize a contact center infrastructure without impacting on each other. This means that all data and processes are strictly separated between tenants, and each tenant uses the functionality as if it is a single user.

In enterprise contact centers there are some additional requirements to tenancy. A tenant may have internal structure composed of, say, departments and one department should be logically separated from another one.

Application and system routing

Based on exploring all the above types of routing mentioned, we could distinguish two types of routing – application-based and system-based routing. Application-based routing tries to associate customers with agents and therefore to improve customer satisfaction.

System-based routing tries to pursue system-based objectives such as a fair utilization of agents, minimization of agents' cost, etc. Clearly these two types are contradictory.

Further reading

Unfortunately, the literature about interaction routing in contact centers is very poor and cannot provide any useful material that is worth reading.

The situation with resource allocation in workflow systems is better. Several useful cases of allocation of humans to workflow activities are considered in workflow resource patterns [1]. There is an OASIS recommendation on human tasks in business processes [2]. The paper [3] contains comparison of these two approaches. If you are thinking about organization-based routing, you should start with the paper on organizational management in workflow systems [4].

Interesting information about other workflow patterns (control, data, and exception patterns) can be found on the corresponded home page: http://www.workflowpatterns.com/

References

[1] Web Services – Human Task (WS-HumanTask) Specification Version 1.1. November 4, 2009 http://docs.oasis-open.org/bpel4people/ws-humantask-1.1-spec-cd-06.pdf

[2] N. Russell, A.H.M. ter Hofstede, D. Edmond, and W.M.P. van der Aalst. Workflow Resource Patterns. BETA Working Paper Series, WP 127, Eindhoven University of Technology, Eindhoven, 2004.

[3] N. Russell and W.M.P. van der Aalst. Workflow Resource Patterns as a Tool to Support OASIS BPEL4People Standardization Efforts. *BPTrends*, 6(3):1-26, March 2008.

[4] Zur Muehlen, M. Organizational Management in Workflow Applications – Issues and Perspectives. Information Technology and Management (2004) 5: 271.

Conclusion

In these notes we briefly considered the main aspects of interaction processing, including almost all types of routing strategies and related problems. Only a small part of these functionalities is implemented in contemporary contact centers because of limitations of routing technologies. The main bottleneck is a complexity of implementation of each type of strategy and a lack of tools enabling one to automate this process. For a long time, introducing a new routing strategy required hard coding rather than a simple configuring. The natural solution is developing a formal model and domain-specific language (DSL) oriented at specification and implementation of different types of strategies.

We have been working on this problem, suggesting a framework for designing interaction processing applications and corresponding execution environment in the cloud. More about this project can be found in this whitepaper.

About the Author

Nikolay Anisimov, Ph.D., is a computer scientist with a strong academic background. He is an industry veteran with twenty years' experience in contact center technologies, is the author of numerous patents, technical and research papers, whitepapers, and articles in industry journals. Nikolay has worked for Genesys, Alcatel-Lucent, FrontRange Solutions, Five9, Aspect Software, and Bright Pattern, Inc. He is a co-founder of Contact Technology Labs, Inc.

You can contact Nikolay at Email: anisimov@computer.org