



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.

# Conversation Rules Templates Guide

Conditions

12/18/2025

# Conditions

Condition	Example Usage	Parameters	Description
<b>Time Sensitive</b>			
Today is a work day	Today is a work day	N/A	To use this condition, a business calendar must be associated with the rule. Based on the definition of the business calendar, this condition evaluates true if the current day is a "work day".
It is currently during business hours	It is currently during business hours	N/A	To use this condition, a business calendar must be associated with the rule. Based on the definition of the business calendar, this condition evaluates true if the current time is during business hours (without regard to the day)
Today is a work day and it is currently during business hours	Today is a work day and it is currently during business hours	N/A	To use this condition, a business calendar must be associated with the rule. Based on the definition of the business calendar, this condition evaluates true if the current day and time are on a business day and during business hours
<b>Media-Related</b>			
Media type is "{mediaType}"	Media type is "voice" Media type is "chat" Media type is "email"	mediaType <ul style="list-style-type: none"> <li>• fetched from Business Attributes -&gt; Media Type</li> </ul>	To use this condition, the media type needs to be passed in a separate fact field, as this value is not passed in on the standard JSON structure from a Query Customer Profile block. The media type can be extracted from the interaction.
<b>Customer-Related</b>			
<b>Usage Notes:</b> To use the following customer-related conditions, the Composer application must use the Query Customer Profile block to retrieve details about the customer, and the result of Query Customer Profile must be assigned to the CustomerProfile fact.			

Condition	Example Usage	Parameters	Description
Customer "{contactAttribute}" "{stringOperator}" "{stringValue}"	<ul style="list-style-type: none"> <li>Customer Last Name starts with Sh</li> <li>Customer Zip Code equals 27613</li> <li>Customer Country is Canada</li> </ul>	<p>contactAttribute</p> <ul style="list-style-type: none"> <li>• fetched from Business Attributes -&gt; Contact Attributes</li> </ul> <p>stringOperator</p> <ul style="list-style-type: none"> <li>• contains</li> <li>• ends with</li> <li>• equal to</li> <li>• equal to ignore case</li> <li>• starts with</li> </ul> <p>stringValue</p> <ul style="list-style-type: none"> <li>• any string</li> </ul>	<p>The rule author can choose any of the defined string fields from the drop-down list. This list is populated from Configuration Server, so will contain any new fields added for the solution. The rule author can choose any of the operators from a drop-down list, and will then type in a value to compare against.</p> <p>The rule author can negate any of the conditions by using the not operator in GRAT, and can group related conditions together using the grouping feature.</p> <p>The Composer application can pass in the result from Query Customer or Identify Customer blocks as a single variable to the rule. This condition will extract the field from the input variable and compare it to the stringValue using the operator specified in stringOperator.</p>
Customer (numeric) "{contactAttribute}" "{operator}" "{numericValue}"	<p>Customer (numeric) "age" is greater than 50</p> <p>Customer (numeric) "credit score" is less than 500 Customer (numeric) "weight" is less than or equal to 150</p>	<p>contactAttribute</p> <ul style="list-style-type: none"> <li>• fetched from Business Attributes -&gt; Contact Attributes</li> </ul> <p>operator</p> <ul style="list-style-type: none"> <li>• not equal to</li> <li>• equal to</li> <li>• greater than</li> <li>• greater than or equal to</li> <li>• less than</li> <li>• less than or equal to</li> </ul> <p>numericValue</p> <ul style="list-style-type: none"> <li>• any number</li> </ul>	<p>The rule author can choose any of the defined string fields" from the drop-down list. This list is populated from Configuration Server, so will contain any new fields added for the solution. The rule author can choose any of the operators from a drop-down list, and will then type in a value to compare against.</p> <p>The Composer application can pass in the result from Query Customer or Identify Customer blocks as a single variable to the rule. This condition will extract the field from the input variable and compare it to the intValue using the operator specified in operator.</p>
Customer segment is "{customerSegment}"	Customer segment is "Gold"	customerSegment	The rule author can choose any of the

Condition	Example Usage	Parameters	Description
		<ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; CustomerSegment</li> </ul>	defined segments from the drop-down list. This list is populated from Configuration Server, so will contain any new fields added for the solution.
<b>Service-Related</b>			
<b>Usage Notes To use the following Service-Related conditions, the Composer application must:</b> <b>1) Use the “Query Services” block for all services associated with the customer: a)Identifier = customer ID. b)Service Status = all c) Service Type (unset). 2) Populate the result of “Query Services” into “Services” fact</b>  <b>Note: Certain conditions may not require querying all service status and/or all service types. However, querying all services provide the most flexibility in rules authoring should the business decision changes.</b>			
Customer has at least one active service	Customer has at least one active service		If the customer has at least one service that is active, this condition evaluates true.
Customer has at least one active service of type “{serviceType}”	Customer has at least one active service of type "Reservation"  Customer has at least one active service of type "Merchandise Return"	serviceType  <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementService</li> </ul>	If the customer has at least one active service of the type specified, the condition evaluates true.
Customer has at least one service of type “{serviceType}” that has completed.	Customer has at least one service of type "Reservation" that has completed.  Customer has at least one service of type "Merchandise Return" that has completed.	serviceType  <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementService</li> </ul>	If the customer has at least one service of the type specified that is in completed state, the condition evaluates true.
Customer has at least {numberOfServices} services currently active and started within {time} “{timeUnit}”	Customer has at least 3 services currently active and started within 1 week  Customer has at least 2 services currently active and started within 24 hours	numberOfServices - integer value > 0  time - integer > 0 timeUnit <ul style="list-style-type: none"> <li>• hours</li> <li>• days</li> <li>• weeks</li> <li>• months</li> </ul>	If the customer has at least the specified number of services currently active, that were all started within the time specified, the condition evaluates true.
Customer has at least {numberOfServices} services that completed within the last {time} “{timeUnit}”	Customer has at least 3 services that completed within the last 2 weeks  Customer has at least 2	numberOfServices - integer value > 0  time - integer > 0	If the customer has at least the specified number of services that all completed within the time specified, the

Condition	Example Usage	Parameters	Description
	services that completed within the last 5 days	timeUnit <ul style="list-style-type: none"> <li>• hours</li> <li>• days</li> <li>• weeks</li> <li>• months</li> </ul>	condition evaluates true.
Customer has at least {numberOfServices} services of type "{serviceType}" currently active and started within {time} "{timeUnit}"	Customer has at least 3 services of type "Reservation" currently active and started within 1 month  Customer has at least 2 services of type "Product Defect" currently active and started within 3 days. Customer has at least 2 services of type "Complaints" currently active and started within 90 days	numberOfServices - integer value > 0  serviceType <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementService</li> </ul> time - integer > 0  timeUnit <ul style="list-style-type: none"> <li>• hours</li> <li>• days</li> <li>• weeks</li> <li>• months</li> </ul>	If the customer has the specified number of services, of the given type, currently active and all started within the time specified, the condition evaluates true.
Customer has at least {numberOfServices} services of type "{serviceType}" completed within the last {time} "{timeUnit}"	Customer has at least 2 services of type "Airline Reservation" completed within the last 2 months  Customer has at least 5 services of type "Complaint" completed within the last 180 days	numberOfServices - integer value > 0  serviceType <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementService</li> </ul> time - integer > 0  timeUnit <ul style="list-style-type: none"> <li>• hours</li> <li>• days</li> <li>• weeks</li> <li>• months</li> </ul>	If the customer has the specified number of services, of the given type, that all completed within the time specified, the condition evaluates true.
Customer had last completed "{serviceType}" service occur within {time} "{timeUnit}"	Customer had last completed "Complaint" service occur within 7 days	serviceType <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt;</li> </ul>	If the customer's last completed service occurred on or before the time specified, the condition evaluates true.

Condition	Example Usage	Parameters	Description
	Customer had last completed "Reservation" service occur within 1 month	<p>ContextManagementService</p> <p>time - integer &gt; 0</p> <p>timeUnit</p> <ul style="list-style-type: none"> <li>hours</li> <li>days</li> <li>weeks</li> <li>months</li> </ul>	
The number of active services associated with this customer is "{operator}"{numberOfServices}	<p>The number of active services associated with this customer is greater than 5</p> <p>The number of active services associated with this customer is less than 3</p>	<p>operator</p> <ul style="list-style-type: none"> <li>not equal to</li> <li>equal to</li> <li>greater than</li> <li>greater than or equal to</li> <li>less than</li> <li>less than or equal to</li> </ul> <p>numberOfServices integer &gt;= 0</p>	If the number of active services associated with this customer matches the condition specified (eg, "greater than 5", "equal to 10"), the condition evaluates true.
The number of completed services associated with this customer is "{operator}"{numberOfServices}.	<p>The number of completed services associated with this customer is greater than 5.</p> <p>The number of completed services associated with this customer is less than 3.</p>	<p>operator</p> <ul style="list-style-type: none"> <li>not equal to</li> <li>equal to</li> <li>greater than</li> <li>greater than or equal to</li> <li>less than</li> <li>less than or equal to</li> </ul> <p>numberOfServices integer &gt;= 0</p>	If the number of completed services associated with this customer matches the condition specified (eg, "greater than 5", "equal to 10"), the condition evaluates true.
The total number of services associated with this customer is "{operator}"{numberOfServices}	The total number of services associated with this customer is less than 3	<p>operator</p> <ul style="list-style-type: none"> <li>not equal to</li> <li>to</li> <li>greater than</li> <li>greater than or equal to</li> </ul>	If the number services (active or completed) associated with this customer matches the condition specified (eg, "greater than 5", "equal to 10"), the condition evaluates true.

Condition	Example Usage	Parameters	Description
		<ul style="list-style-type: none"> <li>less than</li> <li>less than or equal to</li> </ul> numberOfServices integer >= 0	
<b>State-Related</b>			
<b>Usage Notes: To use the following State-Related conditions, the Composer application must:</b> <b>1) Use the “Query States” block for all services associated with the service: a) Service ID = service in question; b) State Status = all; c) State Type (unset); 2)</b> <b>Populate the result of “Query States” into “States” fact of Rule Block.</b> <b>Note: Certain conditions do not require querying all state statuses and/or all state types. However, querying all states provides the most flexibility in rules authoring should the business decision changes.</b>			
Service is currently in "{state}" state.	Service is currently in "Offering Callback" state  Service is currently in "Collection" state	stateType <ul style="list-style-type: none"> <li>               Fetched from Business Attributes                -&gt; ContextManagementState             </li> </ul>	This condition will examine the active state related to the Service object and compare it to the selected value. If it matches, the condition will evaluate true.  Example: Service "Travel Reservation" could have states Query Airfares (completed) Reserve flights (completed) Make payment (active)  The following condition would evaluate true: Service is currently in "Make payment" state.
Service has completed state "{state}" within {time} "{timeUnit}"	Service has completed state "Delivering Callback" within 5 days.  Service has completed state "Payment" within 24 hours	stateType <ul style="list-style-type: none"> <li>               Fetched from Business Attributes                -&gt; ContextManagementState             </li> </ul> time - integer > 0 timeUnit <ul style="list-style-type: none"> <li>hours</li> <li>days</li> <li>weeks</li> <li>months</li> </ul>	This condition will examine the list of "states" that are provided in the Service object. If there is at least one state of the specified "type" that has completed within the specified time range, it will evaluate true.
Service has been in "{state}" state for at	Service has been in "Pending Payment"	stateType	This condition will examine the list of

Condition	Example Usage	Parameters	Description
least {time} "{timeUnit}"	state for at least 1 week.  Service has been in "Confirm Reservation" state for at least 24 hours	<ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementState</li> </ul> time - integer > 0  timeUnit <ul style="list-style-type: none"> <li>• hours</li> <li>• days</li> <li>• weeks</li> <li>• months</li> </ul>	"states" that are provided in the Service object. If the specified state type has been active for at least the specified time range, it will evaluate true.
<b>Task-Related</b>			
<b>"Usage Notes: To use the following Task-Related conditions, the Composer application must:</b> <b>1) Use the Query Tasks block for all tasks associated with the service/state - this means</b> <b>a)Service ID = service in question; b)State ID = state in question, if tasks are associated with state; c) Task Status = all; d) Task Type (unset); 2)Populate the result of Query Tasks into "Tasks" fact of the Rule Block.</b>			
<b>Note: Certain conditions do not require querying all task statuses and/or all task types. However, querying all tasks provides the most flexibility in rules authoring should the business decision change.</b>			
Task "{task}" is active	Task "Make Payment" is active  Task "Pay Taxes" is active	taskType <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementTask</li> </ul>	This condition will examine the list of active "tasks" that are provided in the State object. If there is an active task of the specified "type", then the condition will evaluate true.
Task "{task}" has been completed	Task "Make Payment" has been completed  Task "Pay Taxes" has been completed	taskType <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementTask</li> </ul>	This condition will examine the list of completed "tasks" that are provided in the State object. If there is a completed task of the specified "type", then the condition will evaluate true.
Task "{task}" has been completed within {time} "{timeUnit}"	Task "Call Customer" has been completed within 1 day  Task "Process Payment" has been completed within 1 week	taskType <ul style="list-style-type: none"> <li>• Fetched from Business Attributes -&gt; ContextManagementTask</li> </ul> time - integer > 0  timeUnit	This condition will examine the list of completed "tasks" that are provided in the State object. If there is a completed task of the specified "type" that completed within the specified time range, it will evaluate true.



Condition	Example Usage	Parameters	Description
		<ul style="list-style-type: none"><li>• hours</li><li>• days</li><li>• weeks</li><li>• months</li></ul>	
Task "{task}" has been active for at least {time} "{timeUnit}"	<p>Task "Call Customer" has been active for at least 8 hours</p> <p>Task "Mail check" has been active for at least 6 months</p>	<p>taskType</p> <ul style="list-style-type: none"><li>• Fetched from Business Attributes -&gt; ContextManagementTask</li></ul> <p>time - integer &gt; 0</p> <p>timeUnit</p> <ul style="list-style-type: none"><li>• hours</li><li>• days</li><li>• weeks</li><li>• months</li></ul>	<p>This condition will examine the list of tasks that are provided in the State object. If there is at least one task of the specified "type" that has been active for at least the specified time range, it will evaluate true.</p>
Miscellaneous			
<p><b>Usage Notes: The concept of a "Contract" is abstract and will vary for different customers using the CM Templates. If the user wants to test for contract expiration, they can retrieve the actual contract via the Orchestration application (for example, database fetch, web services, and so on) and then pass in the end date in the "Contract" fact. This allows the rule author to test the end date and integrate this condition in with others (Customer, Service, State, Task related).</b></p>			
Contract will expire within {time} "{timeUnit}"	<p>Contract will expire within 7 days</p> <p>Contract will expire within 3 months</p>	<p>time - integer &gt; 0</p> <p>timeUnit</p> <ul style="list-style-type: none"><li>• hours</li><li>• days</li><li>• weeks</li><li>• months</li></ul>	<p>"Contract" will have to be defined as a separate fact. The customer will have to map their actual contract object end date (obtained from their back-end databases using Orchestration or other techniques) to the Contract fact and pass it in. We can then examine the end date passed in and determine if it is within the range specified.</p>

<disqus> </disqus>