

Callback in queue (aka 'CiQ')

When a caller orders Callback in Queue (often referred to as “CiQ”), this results in 2 or more sessions (with the same call_id).

First, ordering of Callback in queue



Later, the Callback



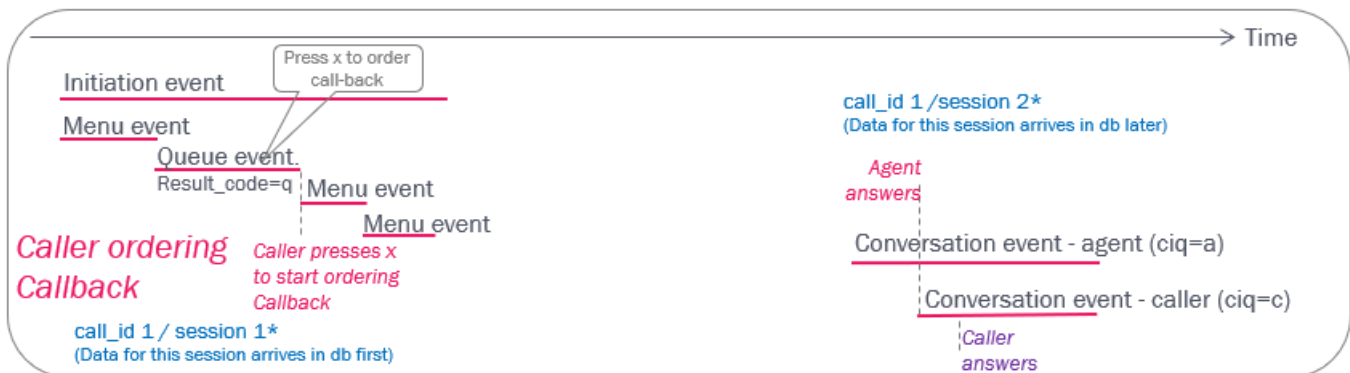
First, a session for ordering the CiQ, which lasts until the caller hangs up. Then a new session when Puzzel calls an agent and the one that ordered CiQ.

Note

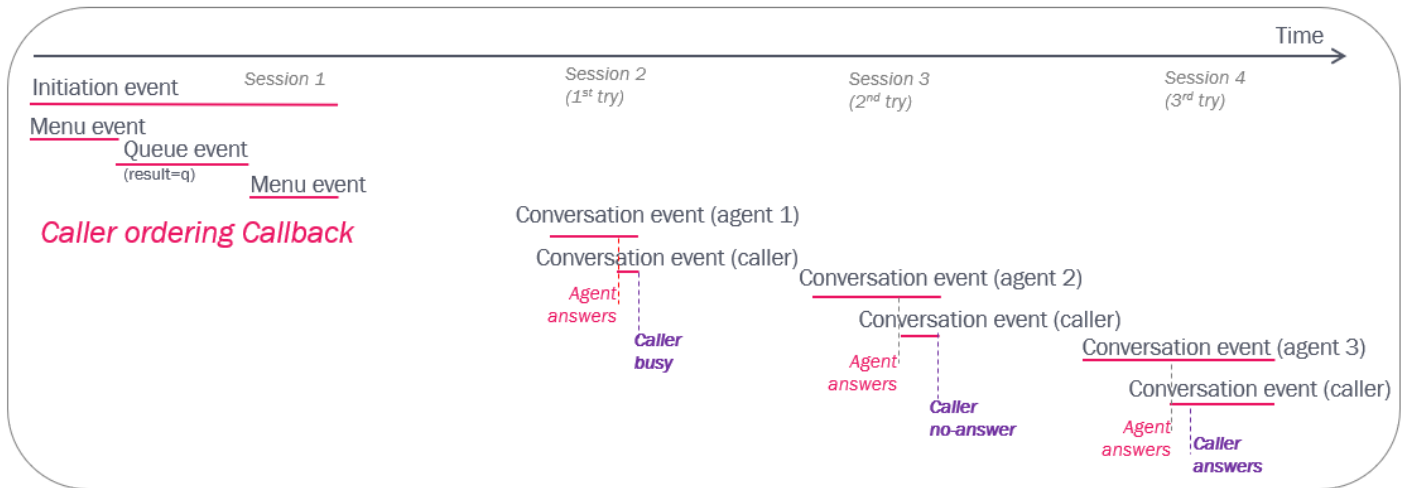
When events for a new session for an existing call_id are ready to be inserted into the database, all existing events for this call_id will be deleted, and then all old and new events for this call_id is inserted with new rec_id and new value in dte_updated.

It is possible to configure that Puzzel calls the one that ordered callback first, and then call an agent. Since very few customers use this, the examples in this document are based on calling agent first. Raw data for callback 'call caller first' is shown at the end of this article.

Example with callback answered on the first attempt:



If the customer does not answer, Puzzel might do 1 or 2 more tries (configurable).



All events for all sessions related to one CiQ will have the same call_id.

The Queue event will have result=q for a caller that ordered Callback.

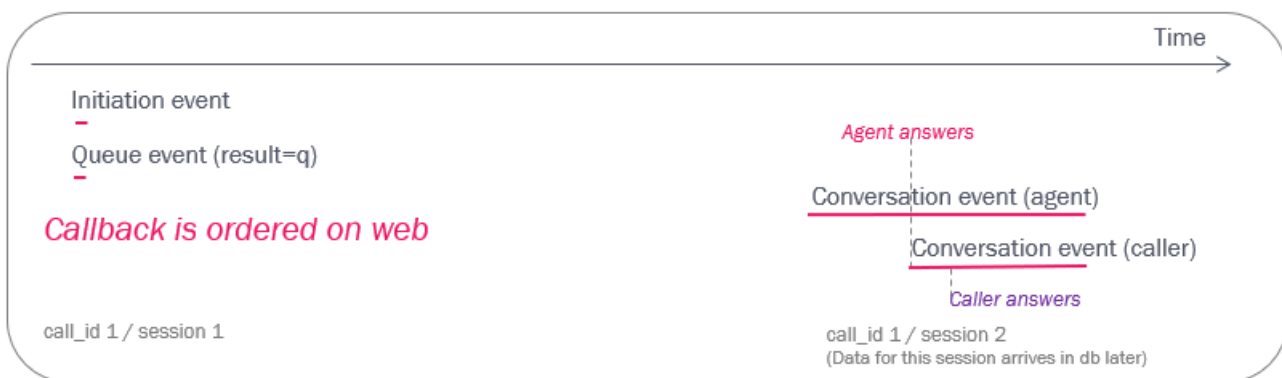
Only Conversation events in a callback (or Callout or Dialler) session will have a value in field ciq:

- ciq=a for call to the agent
- ciq=c for call to the number callback is ordered to (destination or contact)

When a call to the one that ordered callback results in busy, the answered call to the agent in this session will have a very short speaktime. When a call to the one that ordered callback is not answered, the call to the agent in this session will typically have 15-30 seconds speaktime.

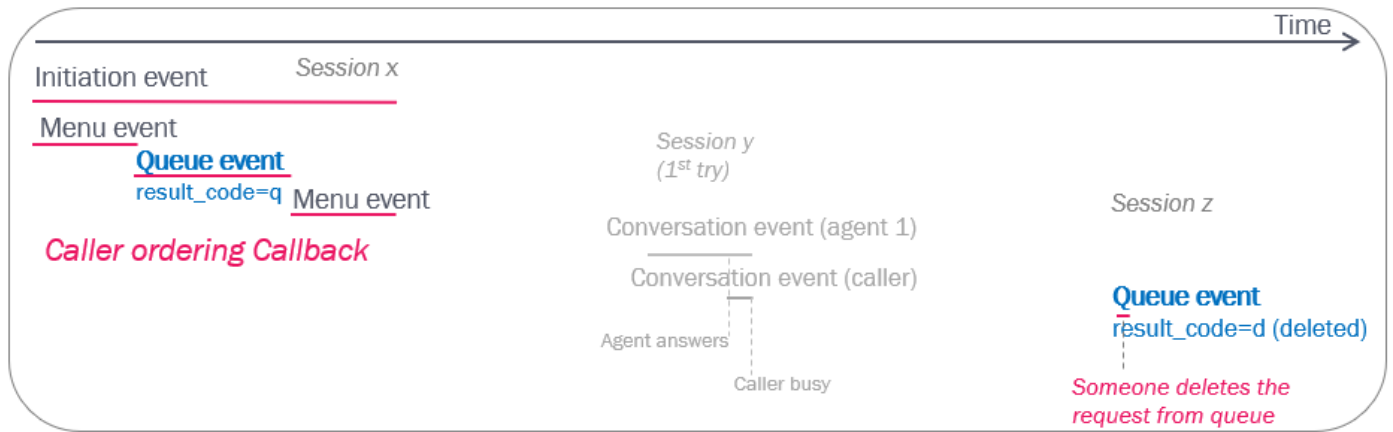
Callback ordered on web

If Callback is ordered by filling in a form on a web-page, this results in an Initiation- and a Queue event, both with duration 0 (or 1 sec). As for callback ordered by phone, the following Conversation events will arrive in the db later, that is, after the calls are done:



Delete Callback request from Queue

If a Puzzel administrator (with special access) deletes a Callback request from queue, this will result in a new Queue event with result_code=d (deleted) in addition to the already generated queue event with result_code=q. If you want to report total number of requests to a queue, you should count queue events with result_code unlike d.

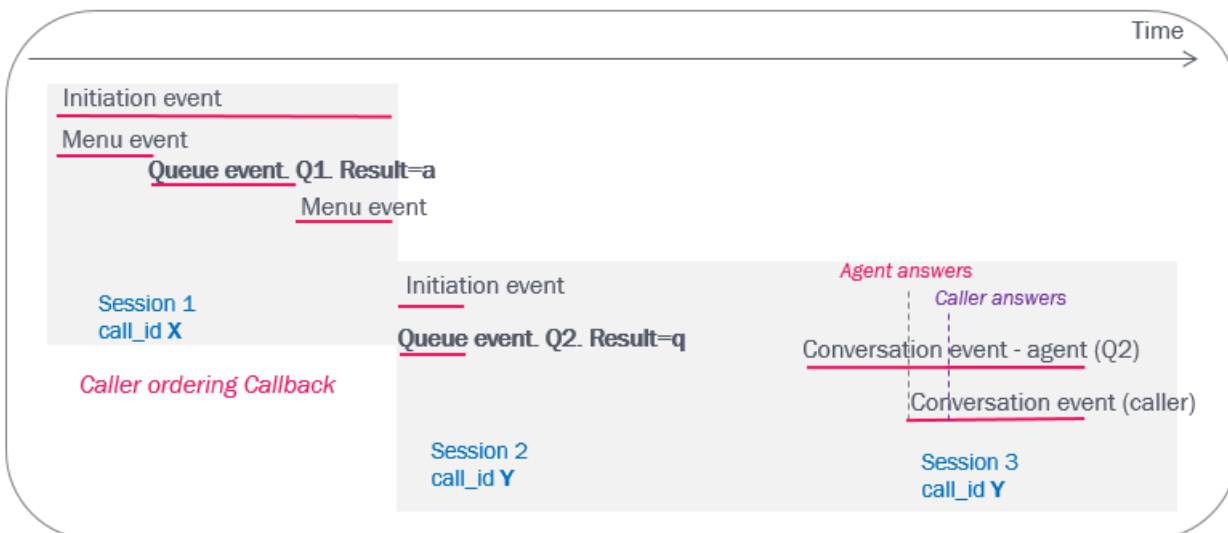


Note
 result_code=d means deleted for Queue events and default for Menu events. See also [Delete email from queue](#).

Callback to another queue (non-standard)

The standard callback solution is that the caller orders Callback in queue to the queue he/she was waiting in. However, some customers require that callbacks are placed in a separate queue. When this is configured, this results in different events than for the standard solution:

- Firstly, the initial session will have a queue event with result a instead of q, since callback is not ordered to this queue.
- Secondly, the following sessions will have a new call_id since this is a "new" ordered (web) callback to queue 2. Example:



In Puzzel statistics this will be reported like this:

Total overview:

2 Incoming calls and 1 (or 0) answered

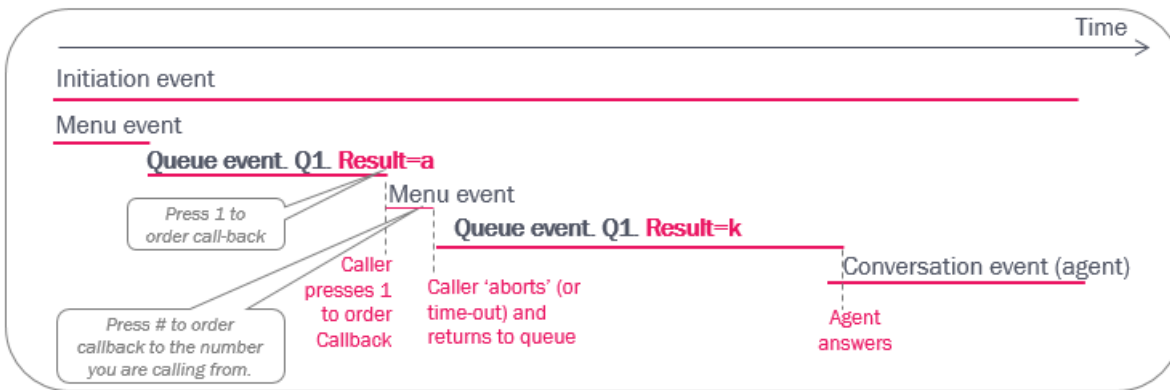
Details per queue:

Queue	Incoming calls	Total calls	Callbacks ordered	Existing queue	Answered (excl callbacks)	Answered callbacks
Queue 1	1	1	0	1	0	0
Queue 2	1	1	1	0	0	1 (or 0)

Callback order not completed

If a caller (temporarily) leaves the queue to start to order Callback, but does not finish the order callback process, the call is usually sent back to queue.

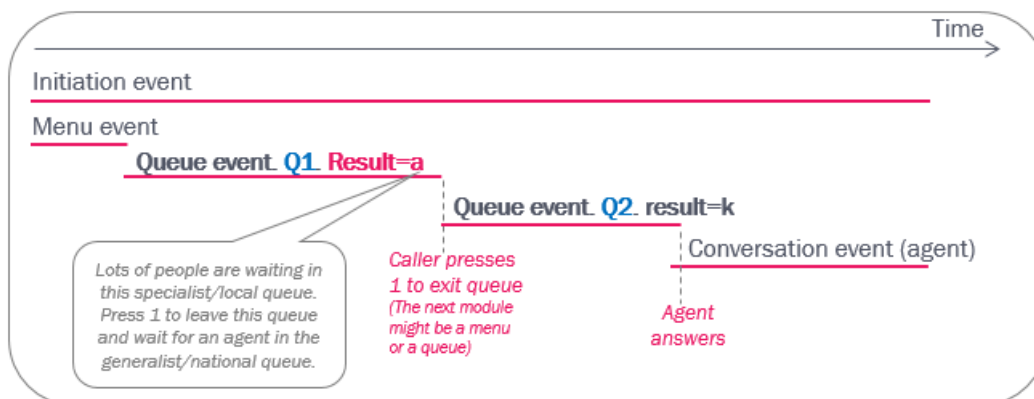
This results in a queue event with result=a, one or more menu events and a **new queue event**, which will have result=k if the call is later answered (or h if caller hangs up in queue).



If a caller starts ordering Callback but hangs up in the menu before the order process is finished, the queue event gets result=a (abort) and the menu event gets result=h (hang-up). If the caller does not press the needed digit(s) to confirm callback, the call might also be terminated.

Interrupt from queue

If the Puzzel solution offers callers in queue to leave queue (not order Callback) by pressing a digit, and the caller presses the digit, the call leaves the queue the queue event will have result=a.



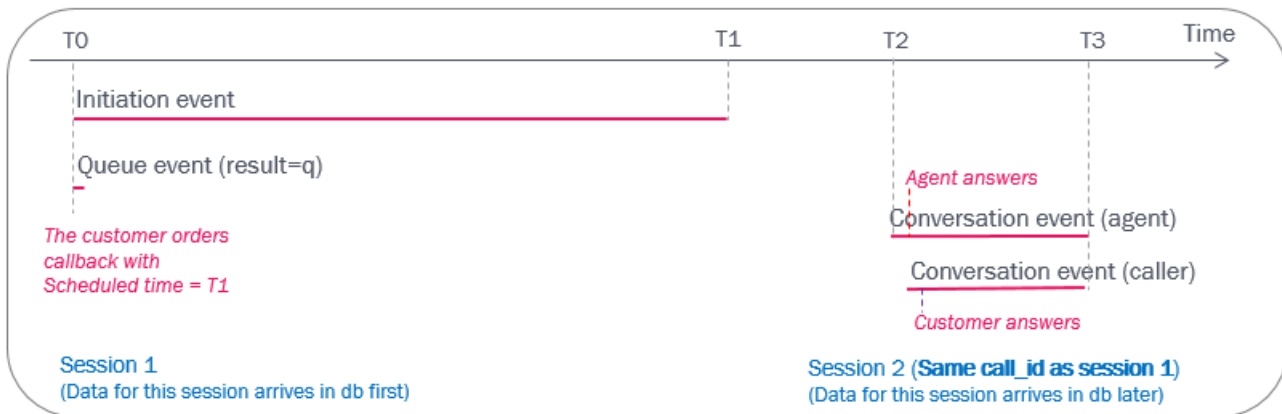
The call is routed to the defined module (e.g. another queue) following the interrupt exit.

Scheduled Callback ordered on web

If the customer that orders callback on a web page can specify a desired time to be called (the “scheduled” time), the Initiation event will have duration until the scheduled time!

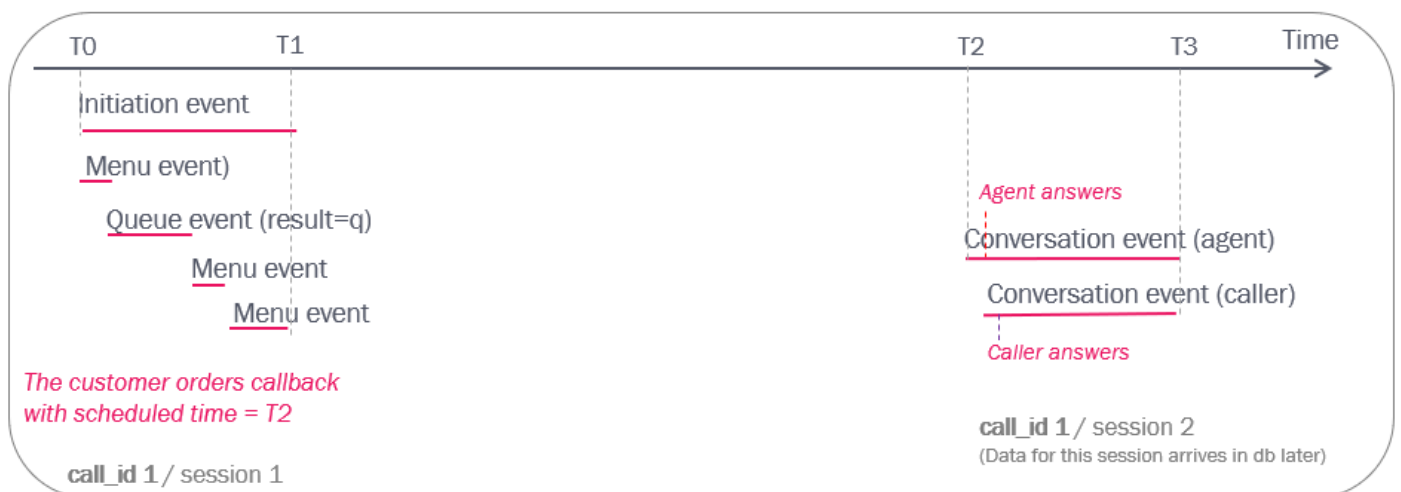
Example (see illustration after the bullet list):

- At T0, the end-customer orders callback («call me») with a specified time (T1)
- The request is put in the queue’s «waiting room» until the scheduled time, and then it’s moved into the queue with high priority
- The queue will then (at T2) call a ready agent. This may take some time if no agents are available at T1
- First, the events for the ordering of scheduled callback arrives in the Raw data db
 - Initiation event with duration T0 (order time) to T1 (scheduled time)
 - Queue event with duration 0
- After the calls to agent and customer has ended (T3, might be hours or days later than T0), the conversation events with the same call_id arrives in the db.



Scheduled Callback ordered by phone

If a caller already in a queue can order a scheduled callback, the Initiation event will last until the ordering process ends.



- Between T0 and T1 the caller (in queue) orders callback with a specified time (T2)
- The request is put in the queue’s «waiting room» until the scheduled time (T2), and then it’s moved into the queue with high priority
- The queue will then call a ready agent (this may take some time if no agents are available at T2)

- First, the events (type i, m and q) for the ordering of Scheduled callback arrives in the Raw data db
- After the calls to the agent and the customer is ended (T3) the conversation events with the same call_id arrives in the Raw data db

Callback - call caller first (non-standard)

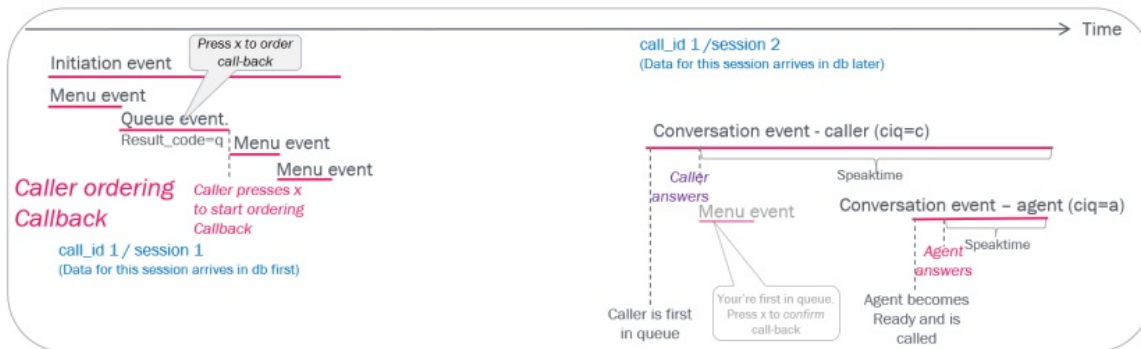
If callback with 'call caller first' is used, it might take some time from the caller has answered the callback until an agent becomes available and is called. It is possible to ask the caller to confirm, and if this is done, an extra menu record is created. This is the raw data generated:

First, ordering of Callback in queue



Later, the Callback

- Caller is called when he/she is first in queue.
- (Disconnect if no confirmation received)
- Please note it may take minutes before an agent becomes ready and is called!



The Speakttime in the caller's conversation event will be (significantly) longer than the Speakttime in the agent's conversation event.

The caller might hang up before an agent is connected!