Triggers and Actions available for Make and N8N

The following triggers and actions are offered by the OpenEMM App for Make (https://www.make.com/en/hq/app-invitation/5cffc5c0eba6ddca073a63853c7baeb4) and the OpenEMM Nodes for N8N (Settings → Community nodes → Install `@openemm/n8n-nodes-openemm`).

You can download demo scenarios for Make and demo workflows for N8N here: https://www.agnitas.de/en/downloads-related-to-integration-platforms/

If you want to go into the details of the OpenEMM APIs that the connectors for Make and N8N are using:

- **Triggers** use the Callback API for **Webhooks**: agnitas_emm_webhooks_1.1.4.pdf
- **Actions** use the **RESTful** API: https://emm.agnitas.de/manual/en/pdf/EMM_Restful_Documentation.html (open link in new tab)

You can find the source code of the OpenEMM nodes (npm package) at GitHub: https://github.com/agnitas-org/n8n-nodes-openemm (open link in new tab)

Triggers (Make and N8N)

ID	Functionality	Name of Make Module	Name of N8N Node	Output Parameters
TM0	Webhook has been triggered	Watch events	-	event_type, event_count, events: event_id, event_timestamp, event_data
TM1	mailing dispatch has been completed	Watch completed mailing deliveries	On mailing sent	event_type, event_count, events: event_id, event_timestamp, event_data: mailing_id
TR1	mail has been delivered to recipient	Watch individual mail deliveries	On mail delivered	event_type, event_count, events: event_id, event_timestamp, event_data: mailing_id, recipient_id, recipient_data: profile_field(s)
TR6	test mail has been delivered to recipient	Watch individual test mail deliveries	On test mail delivered	event_type, event_count, events: event_id, event_timestamp, event_data: mailing_id, recipient_id, recipient_data: profile_field(s)
TR2	mail has been bounced	Watch mail bounces	On mail bounced	event_type, event_count, events: event_id, event_timestamp, event_data: mailing_id, recipient_id, recipient_data: profile_field(s)
TR3	mail has beend opened by recipient	Watch mail openings	On mail opened	event_type, event_count, events: event_id, event_timestamp, event_data: mailing_id, recipient_id, recipient_data: profile_field(s)

ID	Functionality	Name of Make Module	Name of N8N Node	Output Parameters
TR4	link in mail has been clicked by recipient	Watch link clicks	On link clicked	event_type, event_count, events: event_id, event_timestamp, event_data: mailing_id, link_id, recipient_id, recipient_data: profile_field(s)
TR5	mailing has been unsubscribed by recipient	Watch mailing unsubscriptions	On mailing unsubscribed	event_type, event_count, events: event_id, event_timestamp, event_data: mailinglist_id, mediatype, recipient_id, status, opt_out_by, recipient_data: profile_field(s)
TR7	profile field of recipient has been changed	Watch changed profile field	On profile field changed	event_type, event_count, events: event_id, event_timestamp, event_data: recipient_id, recipient_data: profile_field(s)

Actions for Make

ID	Functionality	Name of Make Module	Input Parameters	Output Parameters
AM1	Send out mailing	Send a mailing	mailing_id, optional: send_type, send_date, customer_id, user_status	-
AM2	Get data of a mailing	Get a mailing	mailing_id or mailing_name	mailing_id, name, description, type
AM3	Create a new mailing list	Create a mailing list	name, description	mailinglist_id, name, description, creation_date, change_date
AM4	Copy an existing mailing	Copy a mailing	mailing_id	mailing_id
AM5	Update a mailing	Update a mailing	mailing_id, shortname, description, mailinglist_id or mailinglist_name, mailingtype (action, date, followup, interval, normal), mailing_content_type (advertising, transaction), sender_address, reply_address, subject, target_expression, open_action_id, click_action_id	-
AM6	Delete a mailing	Delete a mailing	mailing_id	-
AM7	Create a new mailing based on a given template ID	Create a mailing based on template	template_id	mailing_id

ID	Functionality	Name of Make Module	Input Parameters	Output Parameters
AC1	Update mailing content by ID or name with content IDs and content texts	Update a mailing content	mailing_id, content_id or content_name, content (array of text, order and target_id)	content_id, content_name, content (array of id and text)
AC2	Delete mailing content by ID or name	Delete a mailing content	mailing_id, content_id or content_name	-
AC3	Read mailing content by ID or name	Get a mailing content	mailing_id, content_id or content_name	content_id, content_name, content (collection of id, order, text and (optional) target_id)
AR1	Create a new recipient	Create a recipient	email, any profile fields (optional), mailinglist_id (optional), binding_status (optional)	customer_id, email, firstname, lastname, gender, title, mailtype, sys_tracking_veto, sys_encrypted_sending, cleaned_date, creation_date, timestamp, lastsend_date, lastopen_date, lastclick_date, datasource_id, latest_datasource_id
AR2	Update an existing recipient	Update a recipient	email or customer_id, any profile fields (optional), mailinglist_id (optional), binding_status (optional)	customer_id, email, firstname, lastname, gender, title, mailtype, sys_tracking_veto, sys_encrypted_sending, cleaned_date, creation_date, timestamp, lastsend_date, lastopen_date, lastclick_date, datasource_id, latest_datasource_id
AR3	Delete an existing recipient	Delete a recipient	email or customer_id	-
AR4	Get data of recipient	Get a recipient	email or customer_id	customer_id, email, firstname, lastname, gender, title, mailtype, sys_tracking_veto, sys_encrypted_sending, cleaned_date, creation_date, timestamp, lastsend_date, lastopen_date, lastclick_date, datasource_id, latest_datasource_id
AR5	Update binding status of recipient on a mailing list	→ AR2		
AR6	Get binding status of recipient (on a given mailinglist)	Search recipient bindings	email or customer_id, mailing list (optional), limit (optional)	mailinglist_id, mediatype, user_type, user_status, user_remark, creation_date, change_date
-	Generic call of REST API	Make an API call	see REST API documentation	see REST API documentation

Actions for N8N

ID	Functionality	Name of N8N Node	Input Parameters	Output Parameters
AM1	Send out mailing	Send mailing	mailing_id, optional: send_type, send_date, customer_id, user_status, data (the latter for action-based mailings only)	JSON provided by RESTful endpoint `sendMailingIdPost`
AM2	Get data of a mailing	Get mailing	mailing_id or mailing_name	JSON provided by RESTful endpoint `getMailingByName` or by `getMailingById` with `view="light"` to reduce output
AM3	Create a new mailing list	Create mailing list	name, description	JSON provided by RESTful endpoint `postMailinglist`
AM4	Copy an existing mailing	Copy mailing	mailing_id	JSON provided by RESTful endpoint `copyMailingById`
AM5	Update a mailing	Update mailing	mailing_id, shortname, description, mailinglist_id or mailinglist_name, mailingtype (action, date, followup, interval, normal), mailing_content_type (advertising, transaction), sender_address, reply_address, subject, target_expression, open_action_id, click_action_id	JSON provided by RESTful endpoint `putMailing` (possible input fields: shortname, description, mailinglist_id, mailinglist_shortname, mailingtype, mailing_content_type, target_expression, open_action_id, click_action_id)
AM6	Delete a mailing	Delete mailing	mailing_id	JSON provided by RESTful endpoint `deleteMailing`
AM7	Create a new mailing based on a given template ID	Create mailing	template_id	JSON provided by RESTful endpoint `postMailing`
AC1	Update mailing content by ID or name with content IDs and content texts	Update mailing content	mailing_id, content_id or content_name, content (array of text, order and target_id or target_name)	JSON provided by RESTful endpoint `contentMailingIdContentIdPut` or `contentMailingIdContentNamePut`
AC2	Delete mailing content by ID or name	Delete mailing content	mailing_id, content_id or content_name	JSON provided by RESTful endpoint `contentMailingIdContentIdDelete` or `componentMailingIdComponentNameDelete`

ID	Functionality	Name of N8N Node	Input Parameters	Output Parameters
AC3	Read mailing content by ID or name	Get mailing content	mailing_id, content_id or content_name	JSON provided by RESTful endpoint `contentMailingIdContentIdGet` or `contentMailingIdContentNameGet`
AR1	Create a new recipient	Create recipient	email, any profile fields (optional), mailinglist_id (optional), binding_status (optional)	JSON provided by RESTful endpoint `recipientPost`
AR2	Update an existing recipient	Update recipient	email or customer_id, any profile fields (optional), mailinglist_id (optional), binding_status (optional)	JSON provided by RESTful endpoint `recipientPut`
AR3	Delete an existing recipient	Delete recipient	email or customer_id	JSON provided by RESTful endpoint `recipientEmailDelete` or `recipientCustomerIdDelete`
AR4	Get data of recipient	Get recipient	email and optionally any profile fields with required values as filter (example: gender = 0)	JSON provided by RESTful endpoint `getRecipient`
AR5	Update binding status of recipient on a mailing list	→ AR2	email, mailinglist_id, status	JSON provided by RESTful endpoint `recipientPut`
AR6	Get binding status of recipient on a mailing list	Get recipient binding	email or customer_id, mailinglist_id	JSON provided by RESTful endpoint `bindingEmailMailinglistIdGet` or `bindingCustomerIdMailinglistIdGet`
-	Generic call of REST API	HTTP Request (internal n8n node)	see REST API documentation	see REST API documentation

From:
https://wiki-iframe.agnitas.de/ - **OpenEMM Wiki**

Permanent link:
https://wiki-iframe.agnitas.de/doku.php?id=integration-apis

Last update: **2026/07/23 14:14**