#

# From oslo.messaging

#

# Name for the AMQP container. must be globally unique. Defaults to a

# generated UUID (string value)

#container\_name = <None>

# Timeout for inactive connections (in seconds) (integer value)

#idle\_timeout = 0

# Debug: dump AMQP frames to stdout (boolean value)

#trace = false

# Attempt to connect via SSL. If no other ssl-related parameters are

# given, it will use the system's CA-bundle to verify the server's

# certificate. (boolean value)

#ssl = false

# CA certificate PEM file used to verify the server's certificate

# (string value)

#ssl\_ca\_file =

# Self-identifying certificate PEM file for client authentication

# (string value)

#ssl\_cert\_file =

# Private key PEM file used to sign ssl\_cert\_file certificate

# (optional) (string value)

#ssl\_key\_file =

# Password for decrypting ssl\_key\_file (if encrypted) (string value)

#ssl\_key\_password = <None>

# By default SSL checks that the name in the server's certificate

# matches the hostname in the transport\_url. In some configurations it

# may be preferable to use the virtual hostname instead, for example

# if the server uses the Server Name Indication TLS extension

# (rfc6066) to provide a certificate per virtual host. Set

# ssl\_verify\_vhost to True if the server's SSL certificate uses the

# virtual host name instead of the DNS name. (boolean value)

#ssl\_verify\_vhost = false

# DEPRECATED: Accept clients using either SSL or plain TCP (boolean

# value)

# This option is deprecated for removal.

# Its value may be silently ignored in the future.

# Reason: Not applicable - not a SSL server

#allow\_insecure\_clients = false

# Space separated list of acceptable SASL mechanisms (string value)

#sasl\_mechanisms =

# Path to directory that contains the SASL configuration (string

# value)

#sasl\_config\_dir =

# Name of configuration file (without .conf suffix) (string value)

#sasl\_config\_name =

# SASL realm to use if no realm present in username (string value)

#sasl\_default\_realm =

# DEPRECATED: User name for message broker authentication (string

# value)

# This option is deprecated for removal.

# Its value may be silently ignored in the future.

# Reason: Should use configuration option transport\_url to provide the

# username.

#username =

# DEPRECATED: Password for message broker authentication (string

# value)

# This option is deprecated for removal.

# Its value may be silently ignored in the future.

# Reason: Should use configuration option transport\_url to provide the

# password.

#password =

# Seconds to pause before attempting to re-connect. (integer value)

# Minimum value: 1

#connection\_retry\_interval = 1

# Increase the connection\_retry\_interval by this many seconds after

# each unsuccessful failover attempt. (integer value)

# Minimum value: 0

#connection\_retry\_backoff = 2

# Maximum limit for connection\_retry\_interval +

# connection\_retry\_backoff (integer value)

# Minimum value: 1

#connection\_retry\_interval\_max = 30

# Time to pause between re-connecting an AMQP 1.0 link that failed due

# to a recoverable error. (integer value)

# Minimum value: 1

#link\_retry\_delay = 10

# The maximum number of attempts to re-send a reply message which

# failed due to a recoverable error. (integer value)

# Minimum value: -1

#default\_reply\_retry = 0

# The deadline for an rpc reply message delivery. (integer value)

# Minimum value: 5

#default\_reply\_timeout = 30

# The deadline for an rpc cast or call message delivery. Only used

# when caller does not provide a timeout expiry. (integer value)

# Minimum value: 5

#default\_send\_timeout = 30

# The deadline for a sent notification message delivery. Only used

# when caller does not provide a timeout expiry. (integer value)

# Minimum value: 5

#default\_notify\_timeout = 30

# The duration to schedule a purge of idle sender links. Detach link

# after expiry. (integer value)

# Minimum value: 1

#default\_sender\_link\_timeout = 600

# Indicates the addressing mode used by the driver.

# Permitted values:

# 'legacy' - use legacy non-routable addressing

# 'routable' - use routable addresses

# 'dynamic' - use legacy addresses if the message bus does not

# support routing otherwise use routable addressing (string value)

#addressing\_mode = dynamic

# Enable virtual host support for those message buses that do not

# natively support virtual hosting (such as qpidd). When set to true

# the virtual host name will be added to all message bus addresses,

# effectively creating a private 'subnet' per virtual host. Set to

# False if the message bus supports virtual hosting using the

# 'hostname' field in the AMQP 1.0 Open performative as the name of

# the virtual host. (boolean value)

#pseudo\_vhost = true

# address prefix used when sending to a specific server (string value)

#server\_request\_prefix = exclusive

# address prefix used when broadcasting to all servers (string value)

#broadcast\_prefix = broadcast

# address prefix when sending to any server in group (string value)

#group\_request\_prefix = unicast

# Address prefix for all generated RPC addresses (string value)

#rpc\_address\_prefix = openstack.org/om/rpc

# Address prefix for all generated Notification addresses (string

# value)

#notify\_address\_prefix = openstack.org/om/notify

# Appended to the address prefix when sending a fanout message. Used

# by the message bus to identify fanout messages. (string value)

#multicast\_address = multicast

# Appended to the address prefix when sending to a particular

# RPC/Notification server. Used by the message bus to identify

# messages sent to a single destination. (string value)

#unicast\_address = unicast

# Appended to the address prefix when sending to a group of consumers.

# Used by the message bus to identify messages that should be

# delivered in a round-robin fashion across consumers. (string value)

#anycast\_address = anycast

# Exchange name used in notification addresses.

# Exchange name resolution precedence:

# Target.exchange if set

# else default\_notification\_exchange if set

# else control\_exchange if set

# else 'notify' (string value)

#default\_notification\_exchange = <None>

{%- if \_data.default\_notification\_exchange is defined %}

default\_notification\_exchange = {{ \_data.default\_notification\_exchange }}

{%- endif %}

# Exchange name used in RPC addresses.

# Exchange name resolution precedence:

# Target.exchange if set

# else default\_rpc\_exchange if set

# else control\_exchange if set

# else 'rpc' (string value)

#default\_rpc\_exchange = <None>

# Window size for incoming RPC Reply messages. (integer value)

# Minimum value: 1

#reply\_link\_credit = 200

# Window size for incoming RPC Request messages (integer value)

# Minimum value: 1

#rpc\_server\_credit = 100

# Window size for incoming Notification messages (integer value)

# Minimum value: 1

#notify\_server\_credit = 100

# Send messages of this type pre-settled.

# Pre-settled messages will not receive acknowledgement

# from the peer. Note well: pre-settled messages may be

# silently discarded if the delivery fails.

# Permitted values:

# 'rpc-call' - send RPC Calls pre-settled

# 'rpc-reply'- send RPC Replies pre-settled

# 'rpc-cast' - Send RPC Casts pre-settled

# 'notify' - Send Notifications pre-settled

# (multi valued)

#pre\_settled = rpc-cast

#pre\_settled = rpc-reply