

# VNGAGE INTEGRATION INTERFACE

---

Technical documentation

# 1 Document history

Version	Date	Author	Note
<b>1.0</b>	2016-05-19	Bartul Bonacic	Created
<b>1.1</b>	2016-12-15	Anders Hellström	Suggested changes
<b>1.2</b>	2017-01-27	Stefan Möhl	First draft new version
<b>2.0</b>	2017-01-31	Bartul Bonacic	New 2.0 version
<b>2.0.1</b>	2017-02-01	Stefan Möhl	Re-ordered 7.3.4-7.3.9
<b>2.0.2</b>	2017-02-03	Stellan	Formating text
<b>2.0.3</b>	2017-02-03	Stefan Möhl	

## 2 Table of content

<b>1 Document history</b>	<b>2</b>
<b>2 Table of content</b>	<b>3</b>
<b>3 Introduction</b>	<b>4</b>
3.1 Intention	4
3.2 Abbreviations	4
<b>4 Business Use Cases</b>	<b>5</b>
4.1 Terms used	5
4.2 Visitors to the attached web site seeking contact via VEP web site behavior tracking plugin	5
4.3 Transfer ongoing chat conversation to another agent / group	5
4.4 Visitors to the site request call back (Web callback)	5
4.5 Visitors need to co-browse the attached web site during a phone call	6
4.6 Visitors in an ongoing chat conversation requests changing to voice phone conversation	6
<b>5 Technical Overview</b>	<b>7</b>
<b>6 Message Exchange Infrastructure</b>	<b>8</b>
6.1 Message Serialization	8
6.2 Message Routing	8
6.3 Message Time-to-live	8
6.4 Security	8
6.5 Extensibility	8
<b>7 Integration Model</b>	<b>10</b>
7.1 Conversation	10
7.1.1 <i>conversation.created</i>	10
7.1.2 <i>conversation.ended</i>	10
7.2 Case	11
7.2.1 <i>case.opened</i>	11
7.3 External visitor routing	11
7.3.1 <i>externalVisitorRouting.routingRequested</i>	11
7.3.2 <i>externalVisitorRouting.routingCancelled</i>	12
7.3.3 <i>externalVisitorRouting.transferredToGroup</i>	13
7.3.4 <i>externalVisitorRouting.routedToAgentById</i>	13
7.3.5 <i>externalVisitorRouting.routedToAgentByEmail</i>	14
7.3.6 <i>externalVisitorRouting.routedToGroup</i>	14
7.3.7 <i>externalVisitorRouting.routedToAgentByIdCompleted</i>	15
7.3.8 <i>externalVisitorRouting.routedToAgentByEmailCompleted</i>	15
7.3.9 <i>externalVisitorRouting.routedToGroupCompleted</i>	16
7.4 Visitor handover	17
7.4.1 <i>visitorHandover.callBackRequested</i>	17
7.4.2 <i>visitorHandover.phoneToWebRequested</i>	17
7.5 Agent and group mapping	18

## 3 Introduction

### 3.1 Intention

This document aims at describing the integration interface provided by Vergic for associated partners and customers who wish to integrate their systems with VEP (Vergic Engage Platform). The intended audience is technical staff external to Vergic. It describes the system-level communication workflows and technical contracts.

### 3.2 Abbreviations

Abbreviation	Explanation
<b>VEP</b>	Vergic Engage Platform
<b>AMQP</b>	Advanced Message Queuing Protocol is an open standard application layer protocol for message-oriented middleware
<b>SSL</b>	Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), both of which are frequently referred to as 'SSL', are cryptographic protocols that provide communications security over a computer network.
<b>GUID</b>	A globally unique identifier (GUID) is a unique reference number used as an identifier in computer software.

## 4 Business Use Cases

The purpose of this integration is to accomplish the business use cases described in this chapter.

### 4.1 Terms used

Primary users of the overall system functionality are the contact center agents, hereinafter referred to as *agent*. A competence group is a logical set of agents that accept visitors from the same queue, usually because they share the same business competences. Such a competence group is hereinafter referred to as *group*. The other party to the agent is seeking contact and is hereinafter referred to as *visitor*. The dialogue of chat messages between agents and visitors is hereinafter referred to as *conversation*.

### 4.2 Visitors to the attached web site seeking contact via VEP web site behavior tracking plugin

A visitor to an organization's website is able to request contact with an agent via the VEP web site contact banner functionality. When a visitor clicks a join chat button, the contact request is transferred to the Partner System through the integration and queued in the Partner System for prioritization and routing to an agent. Once the visitor is routed, the agent performs the conversation with the visitor through Vergic's product functionality.

When the chat conversation enters into post-processing/closing mode in VEP, VEP will notify the Partner System. Post-processing mode/closing mode is identified by; when an agent in VEP products has clicked the "Hang up" button or when the agent is notified by VEP desktop that the visitor has left the conversation, and the conversation switches to the status of "Hung Up".

### 4.3 Transfer ongoing chat conversation to another agent / group

An agent is able to transfer an ongoing chat conversation to another agent / group.

Within VEP an ongoing conversation can be transferred to another group via the "Release-to-group" button.

### 4.4 Visitors to the site request call back (Web callback)

A visitor on the website may request a telephone call from an agent. VEP may forward the call back request to the Partner System. A call back request includes name, phone number, banner headline (corresponding to the title on the banner that currently appears in the VEP queue). The telephone number is mandatory, other information may be left empty.

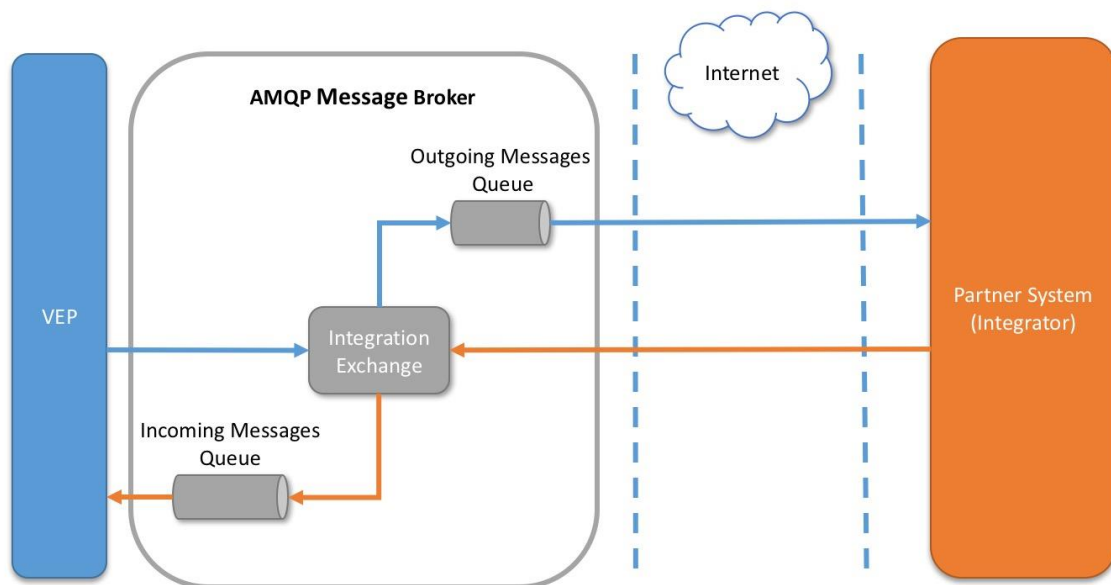
## **4.5 Visitors need to co-browse the attached web site during a phone call**

An agent is able to assist visitors already in a phone conversation by automatically synchronized navigation of the web site together with the visitor. This is the VEP phone2web co-browsing feature. VEP is able to create a connection between the visitor's web site session and the agent's VEP desktop client. VEP informs the Partner System when an agent generates a pin code to initiate phone2web / co-browse with a visitor during a phone call, by sending a message.

## **4.6 Visitors in an ongoing chat conversation requests changing to voice phone conversation**

This event takes place when visitor expresses a wish to switch to telephone call in current chat text. The agent then may make an outgoing voice phone call to the visitor by the Partner System.

## 5 Technical Overview



The integration is accomplished via message passing between systems. Vergic provides a Message Broker to facilitate necessary communication between systems. This message broker communicates via the well-established AMQP protocol. Events inside VEP that are of interest to the Partner System are published as messages and available for consumption by the Partner System in a dedicated queue marked as *Outgoing Messages Queue*, following the blue message flow in the diagram above.

Events inside the Partner System that are of interest to VEP are published as messages to the provided exchange (queue entry point) marked as *Integration Exchange*, subsequently available for consumption by VEP in the dedicated queue marked as *Incoming Messages Queue*, following the orange message flow in the diagram above.

## 6 Message Exchange Infrastructure

To realize AMQP broker functionality, Vergic currently uses a RabbitMQ server instance, accessible on a publicly available endpoint.

### 6.1 Message Serialization

All messages are serialized into JSON formatted strings, more specifically - dates are serialized in ISO format and should always be in the UTC time zone; numbers as simple number values (decimal style); Booleans as 'true' and 'false' and all other simple types are created using standard JavaScript literals for the type. In order to successfully exchange messages, the Partner System must serialize messages in same fashion.

### 6.2 Message Routing

All messages to VEP must be published to the exchange referred as *Integration Exchange*; the exact name will be provided by Vergic once the interface is configured. Each message must be tagged with the routing key of the data model name for the message sent. The Partner System is responsible for queue creation and queue binding.

### 6.3 Message Time-to-live

All messages have TTL of 5 minutes, which means that the message must be consumed by the Partner System in that timeframe or else it is lost. VEP will never retry message sending and will not keep track of non-routed messages.

### 6.4 Security

The Partner System must use specific authentication credentials and virtual host name to successfully connect to the message broker, all provided by Vergic once the interface is configured. Isolation of each integrator is accomplished by separating each on a dedicated virtual host. As the message broker is configured to use the SSL protocol exclusively when communicating over the Internet, the Partner System must use SSL as well.

### 6.5 Extensibility

The messages and their parameter fields described below reflect the current version of the interface. Other messages may at any time be added to the interface, and other fields may at any time be added to the messages, including temporary additions for debugging



purposes. Because of this, any messages or fields not mentioned here must be expected and ignored.

## 7 Integration Model

### 7.1 Conversation

#### 7.1.1 conversation.created

When a conversation in VEP is initiated messages will soon begin to flow. Note that a conversation may be initiated with zero, one or more participants in the conversation. For example, if a visitor enters a queue for a dialogue, there will be a single participant in the conversation while the visitor is waiting to be routed to an agent. If a pre-booked web meeting has been arranged via the booked meetings functionality in VEP, the conversation will begin at the designated time even if all participants are late to the meeting, giving a conversation with initially zero participants. VEP will notify the Partner System by publishing the following message with routing key – *conversation.created*.

##### 7.1.1.1 Message properties

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>conversationSubject</b>	String	Text describing the conversation, usually the invoked VEP solution, the visitor's question or other text relevant to the agent.

#### 7.1.2 conversation.ended

When a conversation in VEP ends, no further chat messages will occur in this conversation. VEP will notify the Partner System by publishing the following message with routing key – *conversation.ended*.

##### 7.1.2.1 Message properties

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP

## 7.2 Case

### 7.2.1 case.opened

When a conversation in VEP is initiated, an associated case is opened. A new case may also be opened mid-conversation by agent initiative, if the agent feels that the conversation shifts to a new case. VEP will notify the Partner System when a case is opened by publishing following message, with routing key – *case.opened*.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>caseId</b>	GUID	Associated VEP case

When a conversation in VEP completes the associated case may be closed. A case may also be closed mid-conversation by agent initiative, if the agent feels that the conversation shifts to a new case. VEP will notify the Partner System by publishing following message, with routing key – *case.closed*.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>caseId</b>	GUID	Associated VEP case

## 7.3 External visitor routing

Message exchange between systems when external visitor routing is enabled.

### 7.3.1 externalVisitorRouting.routingRequested

When a visitor enters a queue in VEP, the visit needs to be associated with an agent in order for communication to proceed. The process is called routing and if group routing inside VEP is set to *external* then the Partner System is expected to handle routing for that particular group, otherwise routing is handled inside VEP and no associated message is published. VEP will notify the Partner System of a visitor in need of routing via a routing request by publishing following message, with routing key – *externalVisitorRouting.routingRequested*.

Property name	Type	Description
<b>routingRequestId</b>	GUID	A unique identification that the response must use to correlate with this request
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>groupId</b>	GUID	Associated VEP group where routing is needed

### 7.3.2 externalVisitorRouting.routingCancelled

When a visitor leaves the queue for some reason and is not assigned to an agent, VEP will notify the Partner System by publishing following message, with routing key – *externalVisitorRouting.routingCancelled*.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>reason</b>	String	The reason for cancelling routing, for logging/debugging

### 7.3.3 externalVisitorRouting.transferredToGroup

When a visitor successfully makes a group transfer in VEP, VEP will notify the Partner System by publishing following message, with routing key – *externalVisitorRouting.transferredToGroup*.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>previousGroupId</b>	GUID	Previously associated VEP group
<b>previousGroupRoutingTypeId</b>	Integer	Routing type of previously associated VEP group, where possible values are: 0 for manual; 1 for automatic and 2 for external
<b>groupId</b>	GUID	Currently associated VEP group
<b>groupRoutingTypeId</b>	Integer	Routing type of currently associated VEP group, where possible values are: 0 for manual; 1 for automatic and 2 for external

### 7.3.4 externalVisitorRouting.routedToAgentById

When the Partner System assigns an agent identified by an agentId (see Agent and group mapping) to a visitor, the Partner System will notify VEP by publishing following message, with routing key – *externalVisitorRouting.routedToAgentById*.

Property name	Type	Description
<b>routingRequestId</b>	GUID	A unique identification given by externalVisitorRouting.routingRequested that the response must use to correlate with the request
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>agentId</b>	GUID	Associated VEP agent

### 7.3.5 externalVisitorRouting.routedToAgentByEmail

When the Partner System assigns an agent identified by an agent's email (see Agent and group mapping) to a visitor, the Partner System will notify VEP by publishing following message, with routing key – *externalVisitorRouting.routedToAgentByEmail*.

Property name	Type	Description
<b>routingRequestId</b>	GUID	A unique identification given by externalVisitorRouting.routingRequested that the response must use to correlate with the request
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>agentEmail</b>	String	Associated email address of the agent to assign visit to

### 7.3.6 externalVisitorRouting.routedToGroup

When the Partner System requests a transfer of an agent to a group queue, the Partner System will notify VEP by publishing following message, with routing key – *externalVisitorRouting.routedToGroup*.

Property name	Type	Description
<b>routingRequestId</b>	GUID	A unique identification given by externalVisitorRouting.routingRequested that the response must use to correlate with the request
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>groupId</b>	GUID	Associated VEP group

### 7.3.7 externalVisitorRouting.routedToAgentByIdCompleted

When a visitor has been routed by the Partner System via the externalVisitorRouting.routedToAgentById message, a result of the transfer will occur. VEP will notify the Partner System by publishing following message, with routing key – externalVisitorRouting.routedToAgentByIdCompleted.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>routingRequestId</b>	GUID	A unique identification correlating to the associated request
<b>statusCode</b>	Integer	0 if the routing was successful, a positive integer otherwise
<b>statusMessage</b>	String	“OK” if the routing was successful, an error message describing the problem otherwise

### 7.3.8 externalVisitorRouting.routedToAgentByEmailCompleted

When a visitor has been routed by the Partner System via the externalVisitorRouting.routedToAgentByEmail message, a result of the transfer will occur. VEP will notify the Partner System by publishing following message, with routing key – externalVisitorRouting.routedToAgentByEmailCompleted.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>routingRequestId</b>	GUID	A unique identification correlating to the associated request
<b>statusCode</b>	Integer	0 if the routing was successful, a positive integer otherwise
<b>statusMessage</b>	String	“OK” if the routing was successful, an error message describing the problem otherwise

### 7.3.9 externalVisitorRouting.routedToGroupCompleted

When a visitor has been routed to a group by the Partner System via the externalVisitorRouting.routedToGroup message, a result of the transfer will occur. If successful VEP will asynchronously (possibly before this result message) send an externalVisitorRouting.transferredToGroup message. VEP will notify the Partner System by publishing following message, with routing key – *externalVisitorRouting.routedToGroupCompleted*.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>routingRequestId</b>	GUID	A unique identification correlating to the associated request
<b>statusCode</b>	Integer	0 if the routing was successful, a positive integer otherwise
<b>statusMessage</b>	String	“OK” if the routing was successful, an error message describing the problem otherwise



## 7.4 Visitor handover

### 7.4.1 `visitorHandover.callBackRequested`

When visitor asks for an agent to do a callback via the telephone, VEP will notify the Partner System by publishing following message, with routing key – *visitorHandover.callBackRequested*.

Property name	Type	Description
<b>Name</b>	String	Name of person to do callback to
<b>phoneNumber</b>	String	Phone number to do callback to
<b>message</b>	String	Message entered when initiated callback request
<b>metadata</b>	Key value array	Any related information not defined at present
<b>agentId</b>	GUID	Associated VEP agent
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP

### 7.4.2 `visitorHandover.phoneToWebRequested`

When an agent generates a phone to web code to enable a visitor to chat and co-browse with the agent alongside the phone call, VEP will notify the Partner System by publishing the following message, with routing key – *visitorHandover.phoneToWebRequested*.

Property name	Type	Description
<b>createdAt</b>	Date and time	The time when this event was created
<b>conversationId</b>	GUID	A unique identifier of the conversation in VEP
<b>phoneToWebCode</b>	String	The code generated by the agent and sent to visitor

## 7.5 Agent and group mapping

Agent, agent email and group are VEP internal entities and thus have associated identity values by which they are uniquely identified inside VEP, referred as - agentId, agentEmail and groupId.

In most cases the Partner System will have similar concepts of its own, with different internal unique identity values, thus a mapping table is required to translate the same business concepts from VEP to the Partner System. Such a mapping table is maintained and owned by the Partner System as it needs one to properly send messages to VEP.

In order to accomplish that, a detailed list of all agent and groups accompanied by respective ids will be provided by Vergic once the interface is configured.