

# ENVISION CONNECT GUIDE

THIS GUIDE IS TO INTERACT WITH ENSONO'S ITSM TOOL  
[03/28/2023] VERSION 1.2

## List of updates

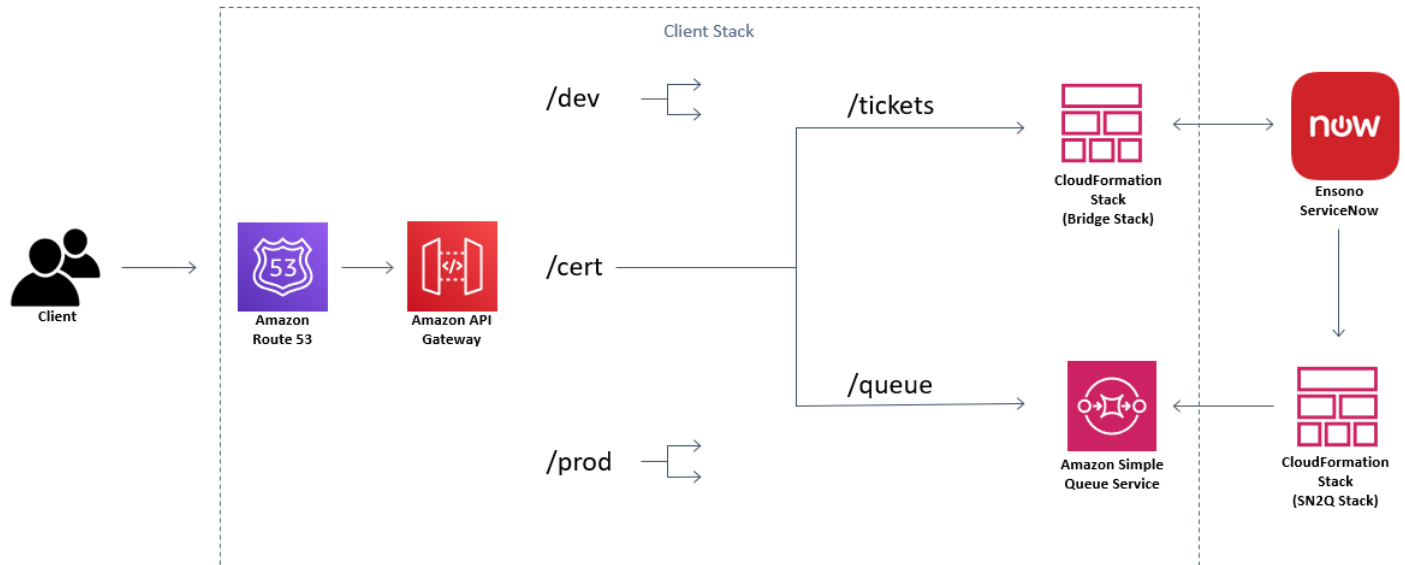
1. Removed “/close” endpoint and all associated documentation. The update endpoint should be used for all status/state updates.
2. Added functionality to close tickets after a ticket has been resolved. A status update of “closed” will now move the ticket to the “closed” state. The ticket must be in the “resolved” state for it to be closed.

ARCHITECTURE .....	4
Client Stack .....	4
Bridge Stack .....	4
Authentication .....	5
Environments .....	5
INCIDENT AND REQUEST INTEGRATION .....	6
The Ensono Process .....	6
Client to Ensono API Methods (Synchronous) .....	7
Client to Queue API Methods (Asynchronous) .....	7
Client Responsibilities .....	8
Bridge Data .....	11
Fields for Opening a Ticket .....	12
Fields for Updating a Ticket .....	13
Comments Array Format .....	14
Using the API -Inbound to Ensono .....	15
Create Ticket .....	15
Update Ticket and Add Comments .....	16
Get Ticket .....	17
Return Client Ticket Number to Ensono .....	25
Add Comments .....	26
Update to Resolved .....	27
Transferring Files To Ensono .....	28
Reading from the Queue .....	30
Deleting from the Queue .....	32
Using the API – Outbound from Ensono .....	33
GET /queue Response .....	33
Incident/Request Message Body and Fields .....	34
Create .....	35
Comment .....	36
Update .....	37
Update to Resolved/Closed .....	38
Attachment Upload from Ensono .....	39
Return Ensono Number .....	40
CHANGE INTEGRATION .....	41
The Ensono Process: .....	41
Flow Diagram: .....	42
Client to Ensono API Methods (Synchronous) .....	44
Client to Queue API Methods (Asynchronous) .....	44
Using the API -Inbound to Ensono .....	45
Ensono Change Message Body and Fields .....	49
Change Create .....	51
Change State Update .....	52
Change Approval Request .....	53
Change Redraft .....	54

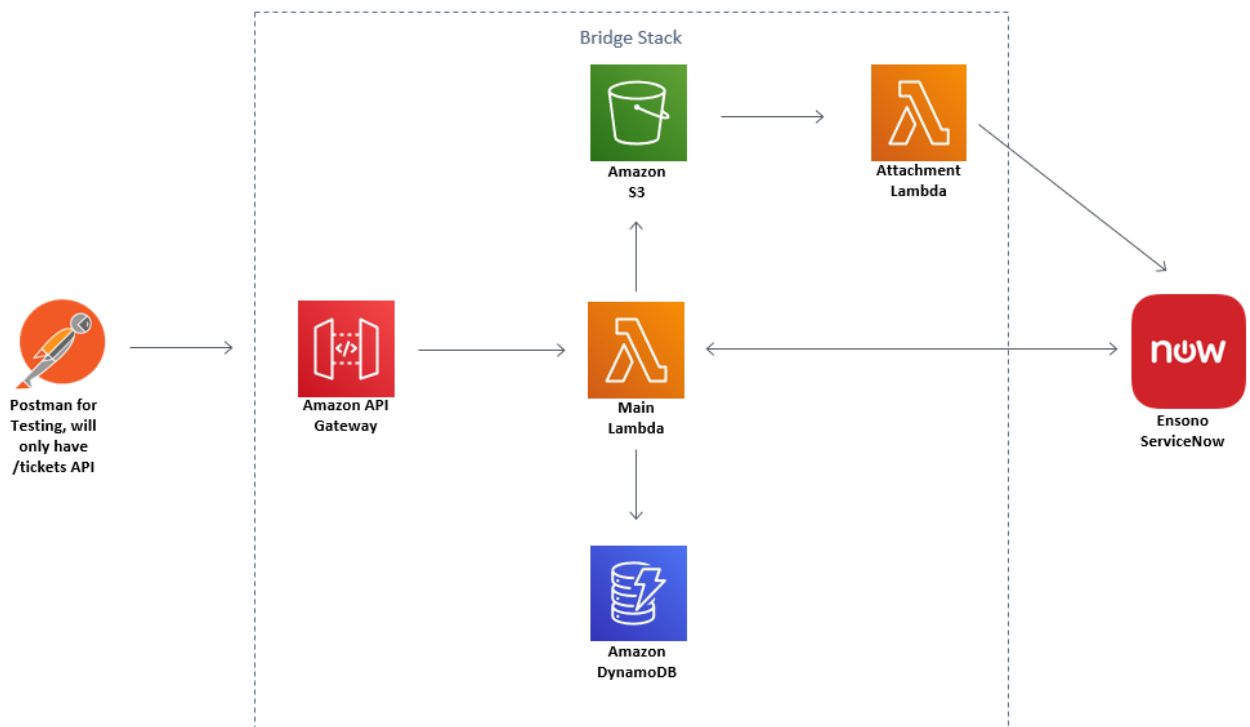
Change Attachment.....	55
Change Comment.....	56
Change Closure.....	58
Normal Change Scenarios.....	59
COMMON ERROR RESPONSES .....	60
POST PRODUCTION SUPPORT .....	62
Emergency Situation “Bridge Down” .....	62
Non-fatal Incidents, Troubleshooting Assistance, or Testing.....	62
Enhancement Requests .....	62
Group Mapping Update Requests .....	62

# Architecture

## Client Stack



## Bridge Stack



## Authentication

Authentication will be handled by an x-api-key header and a clientName parameter that must be included with each request. The x-api-key and clientName values will be provided separately once the client has been setup in the Ensono ITSM system(s).

All calls are sent via HTTPS using TLS 1.2 with a 2048-bit SSL Certificate.

IP Addresses and Ranges can be white-listed as needed.

API-Keys can be rotated manually as needed.

## Environments

Each client stack will be configured with the following environments:

### **/dev/**

Dev is used for initial setup, development, unit testing, and bug fixes.

### **/echo/**

Echo can be used in conjunction with Dev to test the queue reading and processing. Echo calls the same Ensono SNOW instance as Dev, but any transactions sent via the integration are sent back to the queue as if an Ensono user had made the action. This can be used to initiate outbound-from-Ensono transactions or to confirm the success of inbound-to-Ensono transactions. There is no /echo/queue, instead transactions will flow to the /dev/queue. Echo currently works only for incidents and requests.

### **/cert/**

Cert is used for system testing, user acceptance testing, and testing the migration of the integration code and components.

### **/prod/**

Prod is the production environment of the integration.

# Incident and Request Integration

## The Ensono Process

### ENSONO TRIGGERS

ALL Incidents and Requests have the same fields and process. ALL Case Incidents and Case Requests created for the client in the Ensono ITSM system will trigger transactions to the client's Queue. ALL updates to certain fields on those tickets will cause update transactions to be sent to the client's Queue. For attachments, if the "Is Public" flag is set to true after uploading, then Ensono will send a message to the queue with download information for the client.

### ENSONO FIELDS

The field updates that trigger update transactions from Ensono:

- description
- shortDescription
- priority (SLA at Ensono)
- group
- comments
- status
- public attachments

### RESOLUTION/CLOSURE

Ensono has both a Resolved and a Closed status. Once Resolved, the ticket can still receive comments. A Resolved ticket can also be moved back to open status if the resolution is rejected. Otherwise, a resolved ticket is set to Closed after 7 calendar days. Once a ticket has been Closed, no more updates are allowed to be made to the ticket.

Depending on the client's re-open policy, the re-open scenario can be discussed in depth.

## Client to Ensono API Methods (Synchronous)

### /tickets

**POST:** Creates a ServiceNow ticket and returns the ticket number. Adds integration record and comments asynchronously.

*Ticket Types: Incident, Request*

### /tickets/{TICKETNUMBER}

**PUT:** Updates the fields of the specified ticket with the passed values. Returns successful or a SN error. Updates integration record and comments asynchronously.

*Ticket Types: Incident, Request*

**GET:** Returns the current values of all fields on the specified ticket.

*This API is only available in dev environment.*

*Ticket Types: Incident, Request*

### /tickets/{TICKETNUMBER}/returnptn

**PUT:** Updates the fields of the specified ticket's integration record with the passed PTN.

*Ticket Types: Incident, Request*

### /tickets/{TICKETNUMBER}/notes

**PUT:** Adds 1 or more comments to the specified ticket asynchronously. Returns successful.

*Ticket Types: Incident, Request*

### /tickets/{TICKETNUMBER}/attachments/{FILENAME}

**GET:** Returns pre-signed URL to PUT file.

*Ticket Types: Incident, Request*

## Client to Queue API Methods (Asynchronous)

### /queue

**GET:** Returns 1-10 messages in the queue to be processed by the client. Once a message has been processed, the client should use the ReceiptHandle of the message to delete the message from the queue to prevent duplicate processing. Queue Messages can be grouped by their MessageGroupID (ticket number) and ordered by SentTimestamp. Since the Queues are FIFO (First In First Out), messages belonging to the same MessageGroupID must be deleted before the next messages can be retrieved.

### /queue/{RECEIPTHANDLE}

**DELETE:** Deletes the specified message from the queue.

Note: receiptHandle needs to be URL encoded.

# Client Responsibilities

## TRIGGERING

The client should determine what tickets should be sent to Ensono and when. This could be based on Assigned Group, Category, CI, Approval Status, etc. The client should distinguish between when to send a create (if the vendor/partner/Ensono ticket number field is blank) call and when to send an update, and what type of update.

## FORMATTING

**Create format:** All required JSON fields need to be included in creates JSON format

**Update format:** For updates, besides required identity fields, only fields that have changed should be included as an array.

**Note:** Including every field with every update could cause unnecessary ticket ‘hopping’ that can cause the ticket to take longer to be resolved.

## APPENDING CUSTOM OR EXTRA FIELDS

Any fields not listed in the Bridge Data section will need to be concatenated to one of the existing fields, usually description or comments. This logic is normally required when clients send Requests or Tasks to Ensono. These fields will also need to be considered when building update triggers on the client side. Important fields, such as start and end dates on a Change Task, would need to pass any updated information over the integration.

## DUPLICATE PREVENTION

**Create transaction:** The client should only send one create transaction for any given ticket, unless an error is returned.

**Update transaction:** the client should make sure only changed fields are passed

**Comment transaction:** comments/notes/logs are only sent once as note array, with or without update array as Update transaction

## MAPPING

The client may need to create mappings for certain fields. A commonly mapped field on the client side is the “status” or “state” field. Additionally, the client may map their specific Impact/Urgency/Priority to a Priority value of 1 through 4. Alternatively, the client may send the impact and urgency fields in the payload which can be mapped by Ensono. If an invalid group is passed, the error will be returned from Ensono’s ITSM. If an invalid priority is passed, the mapping will choose the default priority of 3.

## LOGGING

In case the need for troubleshooting and bug fixes arise, the client should keep logs of any processes, Requests to the bridge, and Responses from the bridge.

## ERROR HANDLING AND RETRY

Depending on the operation being performed and the error message received, the client should take different actions.

**Example 1:** If the error is related to the group provided, the client should send a ‘default’ group.

**Example 2:** If the API call times out, wait for 60 to 120 sec and then try to resend the transaction.



## READING, PROCESSING, AND DELETING FROM THE QUEUE

Creates & Updates & Public Attachments from Ensono will be added to the client's queue that they can interact with via the API. The client should check the queue at a regular interval (normally 30-300 seconds) for messages. If messages are returned, the client should process those messages as they see fit and delete the message from the queue to prevent duplicate processing.

The queues are FIFO (First In, First Out) so the first messages must be read, processed, and deleted before the later messages can be read. When reading the queue, an array of message objects will be returned (examples below). Each message Object will have multiple fields, but only a few may be needed.

The Body element will have the ticket JSON from the Ensono ticket system for the relevant transaction, the ReceiptHandle element will be needed to DELETE messages from the queue, and the last two fields are in the Attributes object. In this array are objects with important fields, including the SentTimestamp and MessageGroupID fields.

Transactions from each queue read should be grouped by MessageGroupID (Ticket number) and ordered by SentTimestamp. Different ticket numbers can be processed in any order, but the transactions for each ticket should be processed sequentially by their SentTimestamp. Once the transaction is processed, the ReceiptHandle should be used to DELETE that message from the queue so that it is not read again and the next transaction can be processed.

When messages are read from the queue, the messages are made invisible to other read calls for the next 30 seconds. If a message is not DELETE-ed from the queue, the message will be resent (and no new messages sent for the same MessageGroupID) during the next read call up to 20 times before the message is moved out of the normal queue and to a failure queue. The queue also has a message expiration of 4 days, which means if the messages isn't read 20 times or deleted within 4 days of being added to the queue, then the message will be deleted automatically.

## RETURNING CLIENT TICKET NUMBERS

If the client reads their queue and finds a ticket that needs to be created in their system, then once the ticket is created the client should call **PUT /tickets/{ticketnumber}/returnptn** to return their Partner Ticket Number and ID to Ensono's ITSM.

Any transaction from Ensono where the "clientTicketNumber" is an empty string or the attribute ClientReferenceAvailable is set to false means the client's ticket number has not been returned to the Ensono system.

## QUEUE ALERTS

There are 2 alarms setup by default to send alerts to a client defined email address. The first alarm fires when the unread messages count exceeds 10 for a 30 minute period, meaning the prod queue isn't being read. The second alarm fires when the average age of the messages is over 1 hour old, meaning messages aren't being deleted from the prod queue. The alarms will be setup to send to the client email address so a ticket can manually be created on the client side to investigate integration issue.

## INTERNAL ASSIGNMENTS

When a bridged ticket has been assigned to an internal client group in the client's system, Ensono needs a bridge transaction to let our users know to no longer work the ticket. The most common approach to this is to hard code an update that moves the ticket to our generic client group and sets the ticket to a Pending status. If the ticket is then closed internally at the client, the bridge will need to send a resolution/closure update to close the Ensono ticket.

## OUT-OF-SCOPE TRANSACTIONS

Because Ensono sends all clients' transaction over the bridge, some transactions may be out of scope, depending on each client's contract. If there are any bridge cases that are out of scope, additional logic will be needed on the client's side to DELETE any of those messages from the queue. This will prevent the client's queue from getting 'clogged' with unneeded transactions.

## TROUBLE-SHOOTING

When an issue occurs, client should check the log that was kept in client system to troubleshoot.

Example 1: If a ticket was created in client system but not bridged to Ensono system, check if client system has sent the create transaction. If the transaction was sent, what was the response. If the response was successful but still the ticket did not show in Ensono system, please reach out to Ensono team.

Example 2: If there were multiple tickets found in client system that have the same Ensono ticket number, it indicates that the messages in the queue might not be deleted successfully. Check the log if there was an error occurred when deleting messages from the queue. If no error found, reach out to Ensono team for assistance.

## RE-OPEN

A Resolved Ensono ticket can also be moved back to open status if the resolution is rejected. If the client allows reopening of that ticket, when they receive Ensono status update from resolved to previous status, reopen the ticket. If the client doesn't allow reopening of that ticket, one option is to create a new ticket and call /returnptn to send Ensono the new ticket number. The other option is for the client to receive all updates after the ticket is resolved as comments.

If the client re-opened a ticket, the client needs to send Ensono a status update transaction.

If the client closes a ticket, send Ensono a closure transaction and Ensono will close the ticket. Likewise, if the client receives a closure transaction from Ensono, the client needs to close the ticket.

## Bridge Data

The API endpoint expects a JSON string including all the information about a ticket to be received.

### SPECIAL CHARACTERS

Double-escape special characters such as newlines, tabs, double quotes and backslashes in free-text fields (description, comments, etc) to prevent parsing errors from formatted fields.

Example: {"Description": "This is a \\r\\n\\"Test\\""}

### JSON OBJECT

```
{
  "payload": {
    "clientName": "",
    "ticketType": "",
    "clientTicketNumber": "",
    "priority": "",
    "impact": "",
    "urgency": "",
    "group": "",
    "contactName": "",
    "contactPhone": "",
    "contactEmail": "",
    "shortDescription": "",
    "description": "",
    "comments": []
  }
}
```

### EXTRANEIOUS FIELDS

Clients' extra fields and any other valuable information can be appended to the description or sent as a comment. If added to the description, updates back to the client may include the expanded description. Any custom fields added to JSON will be ignored.

## Fields for Opening a Ticket

Field	On Open	Limits
clientName	Required	Value will be provided by Ensono
clientTicketNumber	Required	
ticketType	Required	Incident or Request
description	Required	4000 characters
shortDescription	Required	160 characters
group	Optional	<i>Mapping to be determined</i>
priority	Required*	1, 2, 3, or 4 (Defaults to 3 if not provided or invalid) if the client didn't use impact and urgency
impact	Required*	1, 2, or 3 (Defaults to 3 if not provided or invalid) can also be mapped
urgency	Required*	1, 2, or 3 (Defaults to 3 if not provided or invalid) can also be mapped
cis	Optional	Comma separated
contactName	Required	Free-text
contactEmail	Required	Free-text
contactPhone	Optional	Free-text
comments Array	Optional	Use if any comments were added before bridging, use the same comments format as updates

\* The client needs to choose whether to send the priority value, or the impact/urgency values. This decision needs to remain consistent for all create payloads.

## Fields for Updating a Ticket

Field	On Update	On Comments	On Resolution or Closure	Limits
clientName	Required	Required	Required	
clientTicketNumber	Required	Required	Required	
ticketType	Required	Required	Required	Incident or Request
status	If Updated	<i>Ignored</i>	Required	open, workingprogress, pending, resolved, closed, cancelled
closeNotes	If Updated	<i>Ignored</i>	Required	4000 characters
description	If Updated	<i>Ignored</i>	<i>Ignored</i>	4000 characters
shortDescription	If Updated	<i>Ignored</i>	<i>Ignored</i>	160 characters
group	If Updated	<i>Ignored</i>	<i>Ignored</i>	Mapping to be determined
priority	If Updated	<i>Ignored</i>	<i>Ignored</i>	1, 2, 3, or 4
Impact	If Updated	<i>Ignored</i>	<i>Ignored</i>	1, 2, or 3 or can be mapped
urgency	If Updated	<i>Ignored</i>	<i>Ignored</i>	1, 2, or 3 or can be mapped
comments Array	Optional	Required	<i>Ignored</i>	
comments.body	Optional	Required	<i>Ignored</i>	4000 characters
comments.commenter	Optional	Optional	<i>Ignored</i>	Free-text
comments.commentTime	Optional	Optional	<i>Ignored</i>	Free-text
comments.commenterEmail	Optional	Optional	<i>Ignored</i>	Free-text
comments.commenterPhone	Optional	Optional	<i>Ignored</i>	Free-text

## Comments Array Format

Any notes or comments should be an object in the comments array, i.e.

```
{"body":"","commenter":"","commentTime":"","commenterEmail":"","commenterPhone":""}.
```

Only the body is a required field; the commenter fields are optional, and if provided will be appended to the body of the comment. Multiple comments can be passed at once (see example below).

```
"comments": [{
  "body": "test",
  "commenter": "Jason Doe",
  "commentTime": "2017-07-03 17:55:48",
  "commenterEmail": "Jason.Doe@abc.com",
  "commenterPhone": "555-909-2395"
},
{
  "body": "moving to different group ",
  "commenter": "Jason Doe",
  "commentTime": "2017-07-03 17:55:50",
  "commenterEmail": "Jason.Doe@abc.com",
  "commenterPhone": "555-909-2395"
}
]
```

## Using the API -Inbound to Ensono

### Create Ticket

#### Request:

POST ENVIRONMENTURL/tickets

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Incident",
    "clientTicketNumber": "123456",
    "priority": "4",
    "group": "Ensono Service Desk",
    "contactName": "John Doe",
    "contactPhone": "800-555-1234",
    "contactEmail": "hello@world.com",
    "shortDescription": "test",
    "description": "test",
    "cis": "testCI1,testCI3,testCI6",
    "comments": [{
      "body": "test",
      "commenter": "Jason Doe",
      "commentTime": "2017-07-03 17:55:48",
      "commenterEmail": "Jason.Doe@abc.com",
      "commenterPhone": "555-909-2395"
    }]
  }
}
```

#### Response:

CODE: 200

#### BODY:

```
{"Response": "e-CS0204070"}
```

## Update Ticket

### Request:

**PUT** ENVIRONMENTURL/tickets/e-CS0204070

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{  
  "payload": {  
    "clientName": "YourCompanyName",  
    "ticketType": "Incident",  
    "clientTicketNumber": "123456",  
    "priority": "3",  
    "group": "Ensono Network Support",  
    "shortDescription": "test update"  
  }  
}
```

### Response:

**CODE:** 202

#### BODY:

```
{"Response": "Update Successful"}
```



## Get Ticket

### Request:

GET ENVIRONMENTURL/tickets/e-CS0879714

### HEADERS:

x-api-key: provided separately

### BODY:

none

### Response:

CODE: 200

### BODY:

```
{
  "Response": {
    "task_ci": {},
    "attachments": {
      "4266611f1becedd026a2da49b04bcbcb": {
        "average_image_color": "",
        "chunk_size_bytes": "700,000",
        "compressed": "true",
        "content_type": "application/pdf",
        "encryption_context": "",
        "file_name": "Envision Connect Client Guide v4.2.2.pdf",
        "hash": "f6371a3c6da7c90194183178edecadd483875b2f09d6fef70088ee08023ccf04",
        "image_height": "",
        "image_width": "",
        "size_bytes": "890348",
        "size_compressed": "748741",
        "state": "Available",
        "sys_created_by": "Ensono.User@Ensono.com",
        "sys_created_on": "2023-01-27 18:24:27",
        "sys_domain": "b360a33bdb32c5109786176a48961991",
        "sys_domain_path": "!!!/!!&/!!)/",
        "sys_id": "4266611f1becedd026a2da49b04bcbcb",
        "sys_mod_count": "2",
        "sys_tags": "",
        "sys_updated_by": "Ensono.User@Ensono.com",
        "sys_updated_on": "2023-01-27 18:24:41",
        "table_name": "sn_customerservice_case",
        "table_sys_id": "0816e9db1becedd026a2da49b04bcb33",
        "u_public": "false",
        "x_aito_data_sync_public": "true"
      }
    },
    "comments": {
```

```

"3f86e91f1becedd026a2da49b04bcb45": {
  "element": "comments",
  "element_id": "0816e9db1becedd026a2da49b04bcb33",
  "name": "sn_customerservice_case",
  "sys_created_by": "Ensono.User@Ensono.com",
  "sys_created_on": "2023-01-27 18:25:07",
  "sys_id": "3f86e91f1becedd026a2da49b04bcb45",
  "sys_tags": "",
  "value": "test comment"
},
"incident": {
  "account": "Ensono",
  "action_status": "",
  "active": "true",
  "active_account_escalation": "",
  "active_escalation": "",
  "activity_due": "",
  "additional_assignee_list": "",
  "agile_story": "",
  "approval": "Not Yet Requested",
  "approval_history": "",
  "approval_set": "",
  "asset": "",
  "assigned_on": "",
  "assigned_to": "",
  "assignment_group": "CS - 1st Line",
  "auto_close": "false",
  "auto_created_case": "false",
  "business_duration": "",
  "business_impact": "",
  "business_service": "UNKNOWN",
  "calendar_duration": "",
  "case": "Test ticketCS0879714",
  "case_action_summary": "",
  "case_report": "CSR0859822",
  "category": "Mainframe / Mid-range",
  "cause": "",
  "caused_by": "",
  "change": "",
  "child_case_creation_progress": "false",
  "closed_at": "",
  "closed_by": "",
  "cmdb_ci": "",
  "company": "Ensono",
  "consumer": "",
  "contact": "Ensono User",

```

```
"contact_local_time": "12:28:06",
"contact_time_zone": "US/Central",
"contact_type": "Web",
"contract": "",
"contributor_groups": "",
"contributor_users": "",
"correlation_display": "",
"correlation_id": "",
"delivery_plan": "",
"delivery_task": "",
"description": "Test ticket",
"due_date": "",
"entitlement": "",
"escalation": "Normal",
"expected_start": "",
"first_response_time": "",
"follow_the_sun": "false",
"follow_up": "",
"group_list": "",
"impact": "",
"incident": "",
"initiated_as_request": "false",
"internal_contact": "",
"internal_user": "",
"knowledge": "false",
"location": "",
"made_sla": "true",
"major_case_state": null,
"needs_attention": "false",
"notes_to_comments": "false",
"notify": "Do Not Notify",
"number": "CS0879714",
"opened_at": "2023-01-27 18:22:57",
"opened_by": "Ensono User",
"order": "",
"parent": "",
"partner": "",
"partner_contact": "",
"post_case_review": "",
"priority": "4",
"proactive": "false",
"probable_cause": "",
"problem": "",
"product": "",
"reassignment_count": "0",
"recipient_list": "",
"rejection_goto": "",
```

```

"related_party_consumers": "",
"related_party_users": "",
"resolution_code": null,
"resolved_at": "",
"resolved_by": "",
"route_reason": "",
"service_offering": "",
"short_description": "Test ticket",
"skills": "",
"sla_due": "",
"sn_app_cs_social_social_profile": "",
"sold_product": "",
"stage": null,
"state": "open",
"subcategory": "APM",
"support_manager": "",
"sync_driver": "false",
"sys_class_name": "sn_customerservice_case",
"sys_created_by": "Ensono.User@Ensono.com",
"sys_created_on": "2023-01-27 18:23:37",
"sys_domain": "b360a33bdb32c5109786176a48961991",
"sys_domain_path": "!!!/!!&/!!/",
"sys_id": "0816e9db1becedd026a2da49b04bcb33",
"sys_mod_count": "6",
"sys_tags": "",
"sys_updated_by": "Ensono.User@Ensono.com",
"sys_updated_on": "2023-01-27 18:25:06",
"task_effective_number": "CS0879714",
"task_for": "",
"time_worked": "",
"u_affected_cis": "",
"u_attachment_changes": "27-Jan-23 12:24:28 - Ensono User (Attachment
Changes)\nAttachment added: Envision Connect Client Guide v4.2.2.pdf by
Ensono.User@Ensono.com\n\n",
"u_case_type": "Incident",
"u_classified": "2023-01-27 18:23:37",
"u_custom_fields": "{\"manual_intervention\":\"false\"}",
"u_default_project": "",
"u_everbridge_delivery_details": "<a href=\"https://ipaas-
extui.everbridge.net/?key&#61;311440963534895:41de1e69-f115-4b83-ab27-8fdda79881dc\"
target=\"_blank\" rel=\"noopener noreferrer nofollow\">Everbridge Delivery Details</a>",
"u_everbridge_delivery_details_internal": "https://ipaas-
extui.everbridge.net/?key=311440963534895:41de1e69-f115-4b83-ab27-8fdda79881dc",
"u_hold_reason": "",
"u_integration_log": "27-Jan-23 12:24:33 - Envision Connect Integration (Integration
Log)\nUpdated using Update Case (V1)-SAPI: \n\n",
"u_made_fix_sla": "true",

```

```

    "u_off_hold_date": "",
    "u_project_selector": "All Projects",
    "u_reason_problem_not_related": null,
    "u_reopen_count": "0",
    "u_request": "",
    "u_requested_completion_date": "",
    "u_resolution_percentage_complete": "0",
    "u_responded": "",
    "u_response_percentage_complete": "0",
    "u_stop_notification": "false",
    "u_third_party_reference": "",
    "u_vendor_ticket_reference": "",
    "universal_request": "",
    "upon_approval": "Proceed to Next Task",
    "upon_reject": "Cancel all future Tasks",
    "urgency": "3",
    "user_input": "",
    "variables": "",
    "watch_list": "",
    "wf_activity": "",
    "work_end": "",
    "work_start": "",
    "x_aito_data_sync_client_number": "INC0010299",
    "x_aito_data_sync_reopened_at": "",
    "x_aito_everbridge_delivery_details": "",
    "x_aito_everbridge_delivery_details_internal": "",
    "x_aito_everbridge_stop_notification": "false",
    "x_aito_idle_mon_breach_count": "0",
    "x_aito_idle_mon_reminder_count": "0",
    "closeNotes": ""
  },
  "case": {
    "account": "Ensono",
    "action_status": "",
    "active": "true",
    "active_account_escalation": "",
    "active_escalation": "",
    "activity_due": "",
    "additional_assignee_list": "",
    "agile_story": "",
    "approval": "Not Yet Requested",
    "approval_history": "",
    "approval_set": "",
    "asset": "",
    "assigned_on": "",
    "assigned_to": "",
    "assignment_group": "CS - 1st Line",

```

```
"auto_close": "false",
"auto_created_case": "false",
"business_duration": "",
"business_impact": "",
"business_service": "UNKNOWN",
"calendar_duration": "",
"case": "Test ticketCS0879714",
"case_action_summary": "",
"case_report": "CSR0859822",
"category": "Mainframe / Mid-range",
"cause": "",
"caused_by": "",
"change": "",
"child_case_creation_progress": "false",
"closed_at": "",
"closed_by": "",
"cmdb_ci": "",
"company": "Ensono",
"consumer": "",
"contact": "Ensono User",
"contact_local_time": "12:28:06",
"contact_time_zone": "US/Central",
"contact_type": "Web",
"contract": "",
"contributor_groups": "",
"contributor_users": "",
"correlation_display": "",
"correlation_id": "",
"delivery_plan": "",
"delivery_task": "",
"description": "Test ticket",
"due_date": "",
"entitlement": "",
"escalation": "Normal",
"expected_start": "",
"first_response_time": "",
"follow_the_sun": "false",
"follow_up": "",
"group_list": "",
"impact": "",
"incident": "",
"initiated_as_request": "false",
"internal_contact": "",
"internal_user": "",
"knowledge": "false",
"location": "",
"made_sla": "true",
```

```

"major_case_state": null,
"needs_attention": "false",
"notes_to_comments": "false",
"notify": "Do Not Notify",
"number": "CS0879714",
"opened_at": "2023-01-27 18:22:57",
"opened_by": "Ensono User",
"order": "",
"parent": "",
"partner": "",
"partner_contact": "",
"post_case_review": "",
"priority": "4",
"proactive": "false",
"probable_cause": "",
"problem": "",
"product": "",
"reassignment_count": "0",
"recipient_list": "",
"rejection_goto": "",
"related_party_consumers": "",
"related_party_users": "",
"resolution_code": null,
"resolved_at": "",
"resolved_by": "",
"route_reason": "",
"service_offering": "",
"short_description": "Test ticket",
"skills": "",
"sla_due": "",
"sn_app_cs_social_social_profile": "",
"sold_product": "",
"stage": null,
"state": "open",
"subcategory": "APM",
"support_manager": "",
"sync_driver": "false",
"sys_class_name": "sn_customerservice_case",
"sys_created_by": "Ensono.User@Ensono.com",
"sys_created_on": "2023-01-27 18:23:37",
"sys_domain": "b360a33bdb32c5109786176a48961991",
"sys_domain_path": "!!!/!!&!!)/",
"sys_id": "0816e9db1becedd026a2da49b04bcb33",
"sys_mod_count": "6",
"sys_tags": "",
"sys_updated_by": "Ensono.User@Ensono.com",
"sys_updated_on": "2023-01-27 18:25:06",

```

```

"task_effective_number": "CS0879714",
"task_for": "",
"time_worked": "",
"u_affected_cis": "",
"u_attachment_changes": "27-Jan-23 12:24:28 - Ensono User (Attachment
Changes)\nAttachment added: Envision Connect Client Guide v4.2.2.pdf by
Ensono.User@Ensono.com\n\n",
"u_case_type": "Incident",
"u_classified": "2023-01-27 18:23:37",
"u_custom_fields": "{\"manual_intervention\":\"false\"}",
"u_default_project": "",
"u_everbridge_delivery_details": "<a href=\"https://ipaas-
extui.everbridge.net/?key=&#61;311440963534895:41de1e69-f115-4b83-ab27-8fdda79881dc\"
target=\"_blank\" rel=\"noopener noreferrer nofollow\">Everbridge Delivery Details</a>",
"u_everbridge_delivery_details_internal": "https://ipaas-
extui.everbridge.net/?key=311440963534895:41de1e69-f115-4b83-ab27-8fdda79881dc",
"u_hold_reason": "",
"u_integration_log": "27-Jan-23 12:24:33 - Envision Connect Integration (Integration
Log)\nUpdated using Update Case (V1)-SAPI: \n\n",
"u_made_fix_sla": "true",
"u_off_hold_date": "",
"u_project_selector": "All Projects",
"u_reason_problem_not_related": null,
"u_reopen_count": "0",
"u_request": "",
"u_requested_completion_date": "",
"u_resolution_percentage_complete": "0",
"u_responded": "",
"u_response_percentage_complete": "0",
"u_stop_notification": "false",
"u_third_party_reference": "",
"u_vendor_ticket_reference": "",
"universal_request": "",
"upon_approval": "Proceed to Next Task",
"upon_reject": "Cancel all future Tasks",
"urgency": "3",
"user_input": "",
"variables": "",
"watch_list": "",
"wf_activity": "",
"work_end": "",
"work_start": "",
"x_aito_data_sync_client_number": "INC0010299",
"x_aito_data_sync_reopened_at": "",
"x_aito_everbridge_delivery_details": "",
"x_aito_everbridge_delivery_details_internal": "",
"x_aito_everbridge_stop_notification": "false",

```



```

        "x_aito_idle_mon_breach_count": "0",
        "x_aito_idle_mon_reminder_count": "0",
        "closeNotes": ""
    },
    "u_task_integ": {}
}
}

```

## Return Client Ticket Number to Ensono

### Request:

**PUT** ENVIRONMENTURL/tickets/e-CS0204070/returnptn

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```

{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Incident",
    "clientTicketNumber": "1234567"
  }
}

```

### Response:

**CODE:** 200

#### BODY:

```

{"Response": "PTN added"}

```

## Add Comments

### Request:

**PUT** ENVIRONMENTURL/tickets/e-CS0204070/notes

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Incident",
    "clientTicketNumber": "123456",
    "comments": [{
      "body": "moving to different group",
      "commenter": "Jason Doe",
      "commentTime": "2017-07-03 17:55:48",
      "commenterEmail": "Jason.Doe@abc.com",
      "commenterPhone": "555-909-2395"
    }]
  }
}
```

### Response:

**CODE:** 200

#### BODY:

```
{"Response": "Comments added"}
```

## Update to Resolved

### Request:

PUT ENVIRONMENTURL/tickets/e-CS0204070

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Incident",
    "clientTicketNumber": "123456",
    "status": "resolved",
    "closeNotes": "no longer needed"
  }
}
```

### Response:

CODE: 200

#### BODY:

```
{"Response": "Ticket Closed Successfully"}
```

## Transferring Files To Ensono

Files can only be added to an existing ticket and must be added via a separate REST call. To upload a file, make a GET request to `ENVURL/tickets/{e-CS0000000}/attachments/{example1-file-name.txt}` where `{INC0000000}` is the Ensono ticket number to which the attachment should be added and `{example1-file-name.txt}` is the name of file including file type. This will return a one-time use URL.

To upload attachments, PUT to the URL returned from GET `/tickets/{ticketnumber}/attachments/{filename}` with the Body being the binary file data and an appropriate Content-Type header for the file.

The x-api-key header is not required for the one-time URL PUT.

### LIMITATIONS

1. The file name should contain only alphanumeric characters, hyphens, and dots. And must begin with a letter or number.
2. The only file type not allowed to be uploaded is exe, all other valid file types may be passed (docx, pdf, jpg, txt, xlsx, log, eml, zip, pptx, etc).

**Note:** ServiceNow has file type validation, so if **TestFile.xlsx** is renamed to **TestFile.csv** it will not be added to the ticket

### GETTING AN UPLOAD URL

#### Request:

GET `ENVIRONMENTURL/tickets/e-CS0204070/attachments/test1.pdf`

#### HEADERS:

x-api-key: provided separately

#### BODY:

*none*

#### Response:

CODE: 200

#### BODY:

```
{
  "Response": {
    "code": 200,
    "body": "https://bpm-ensono-sn-bridge-test-files.s3.us-east-2.amazonaws.com/7794fb9edb19a340c5123ebd7c9619d4.test1.pdf?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ASIAYCQITFFPEKI7NDLC%2F20181025%2Fus-east-2%2Ffs3%2Faws4\_request&X-Amz-Date=20181025T192345Z&X-Amz-Expires=120&X-Amz-Security-Token=FQoGZXIvYXdzEBsaDIOPGherAwyfavuJyLtAW0gVfO1%2BGtJ9xkY%2B4wDYwImHBY0e6de1QeUMRn%2BYIcqT%2FATZjYEcaiEtTPMty2K94n5FRY9VTX8qK9gFlvpLx2e9eOsTokyo8aM1VhhK5KtOvsDRXlvt951fkmKZLMXoB8TNP6C9zmJB%2FGXu28UDxEgD1gKSJehCj"
  }
}
```

```
tAjbHcrEU19fPmU9GMPmYwO9n0XfEwugCp2b2%2B42BoA3GLRNHcHtCDbnTdOM5JGAt
wT13yGLHMy6R2QpgmVHoJvWwQobeR9rbfDbQLmipc5I58phyqCq3fQZOe6XnF8EMjUQf
Q9e2M3ZBsm0r5ICIFL4bpjim%2BMfeBQ%3D%3D&X-Amz-
Signature=a121d8c8ddb97f3858ea736e462c4d9d4f156218afec773332793e22114a0b0f
&X-Amz-SignedHeaders=host"
}
}
```

## UPLOADING THE FILE

### Request:

**PUT** (URL FROM RESPONSE ABOVE)

### HEADERS:

Content-Type: application/pdf

- *Content-Type MUST match file type (ex: text/plain, image/jpeg, text/html)*

### BODY:

*File Binary Data Only, no JSON wrapper needed*

### Response:

**CODE:** 200

**BODY:**

## USING BASE64 ENCODED DATA INSTEAD OF BINARY DATA TO UPLOAD FILES

To upload the file data as a Base64 string instead of binary, a few adjustments are needed.

1. When making the GET call for the URL, add '.base64' to the end of the file name.  
(test.docx → test.docx.base64, ticketInfo.png → ticketInfo.png.base64)
2. When making the PUT call, pass the Base64 string as the body, where the binary data would normally be sent. Also, set the Content-Type header to 'text/plain'.

## UPLOADING A BASE64 FILE

### Request:

**PUT** (URL FROM RESPONSE ABOVE)

### HEADERS:

Content-Type: text/plain

### BODY:

VGhpcyBpcyBhbiBhdHRhY2htZW50IHVwbG9hZCB0ZXN0

### Response:

**CODE:** 200

**BODY:**

## Reading from the Queue

### Request:

GET ENVIRONMENTURL/queue

### HEADERS:

x-api-key: provided separately

### BODY:

*none*

### Response with Messages:

CODE: 200

### BODY:

```
{
  "ReceiveMessageResponse": {
    "ResponseMetadata": {
      "RequestId": "14b203db-99d8-5b2f-aac0-5f8a6fd2272e"
    },
    "ReceiveMessageResult": {
      "messages": [{
        "MessageId": "0eb49bcf-53a7-4769-b37a-2342b3658a00",
        "ReceiptHandle":
"AQEBD4eNzuQjJoBJ+TRMwOo5Ub/VvF/AmGYEVooeBKrkEwUh+o9fOpYRpTJL0QhcOhOii+HM9xT41o5P0
20GA4Ugn0RPJyP2pSY7gfrs0VI4QTB6i9GbedsW1T6zwAtRqNPj62qoyudi0c//YtU9jWozN5kPwJ7Oe5OuWg
z/4cCC1EwOFXVzhjrRLaFSmn+IEmpYn6npnuc0bkphh1kzb8fdDPUMnoSSdDOq3bfpi7z6fAD9cnPWzzB5VvH
iudqdkORIJPdEKL60pa7IM3CO5QuNpifiykUeEJb4Lho2Vn3d48=",
        "MD5OfBody": "385b136e9129e7f0b282ed0657772ff5",
        "Body": {
          "clientName": "testClient",
          "ticketNumber": "e-CS0879727",
          "clientTicketNumber": "",
          "ticketType": "Incident",
          "status": "open",
          "description": "test",
          "shortDescription": "test",
          "group": "Ensono Support",
          "ensonoGroup": "CS - 1st Line",
          "priority": "4",
          "impact": "3",
          "urgency": "3",
          "openedTime": "2023-01-27T22:24:53Z",
          "category": "Private Cloud",
          "origin": "Web",
          "cis": ""
        },
        "Attributes": {
          "SenderId":
"AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
```

```

        "ApproximateFirstReceiveTimestamp": "1674858487529",
        "ApproximateReceiveCount": "3",
        "SentTimestamp": "1674858334956",
        "SequenceNumber": "18875507807458288128",
        "MessageDeduplicationId":
"fb1cf0cacb6943d49cf62593e60d7ecbb4b6baff0edaba7984d8d3ecaf41801c",
        "MessageGroupId": "e-CS0879727",
        "ClientReferenceAvailable": false,
        "TransactionType": "Create"
    }
}
}
}
}

```

#### Response Without Messages:

**CODE:** 200

**BODY:**

```

{
  "ReceiveMessageResponse": {
    "ReceiveMessageResult": {
      "ResponseMetadata": {
        "RequestId": "a4ddf399-6d19-5670-a8bc-a6e26670337a"
      },
      "ReceiveMessageResult": {
        "messages": null
      }
    }
  }
}

```

## Deleting from the Queue

ReceiptHandle from the message will be used to delete the message. Before adding it to the URL, it needs to be URL encoded.

### Request:

**DELETE** ENVIRONMENTURL/queue/URLENCODE ({ReceiptHandle highlighted above})

#### **HEADERS:**

x-api-key: provided separately

#### **BODY:**

*none*

### Response:

**CODE:** 200

#### **BODY:**

```
{
  "DeleteMessageResponse": {
    "ResponseMetadata": {
      "RequestId": "8d9a7dc2-42c5-5e95-89c4-287741bf1a2f"
    }
  }
}
```



## Using the API – Outbound from Ensono

### GET /queue Response

Each response will look like the below JSON with any ticket transactions contained in the highlighted array. Each ticket transaction will be a JSON object with its own body and set of attributes, examples below.

```
{
  "ReceiveMessageResponse": {
    "ReceiveMessageResult": {
      "ResponseMetadata": {
        "RequestId": "7742e7f3-3cdd-509d-9c97-b1e9be73ed33"
      },
      "ReceiveMessageResult": {
        "messages": [...]
      }
    }
  }
}
```

Each individual message will have some common fields to look for:

1. ReceiptHandle – used to delete the message once it has been processed
2. Body – contains the fields and values of the ticket transaction
3. Attributes.MessageGroupId – is the Ensono ticket number
4. Attributes.TransactionType – shows type of transactions, options:
  - a. Create
  - b. Update
  - c. Notes
  - d. Closure
  - e. Attachment
  - f. Redraft - When a change goes back to “New” state from some other state
  - g. ReturnEnsonoNum – Provides Ensono ticket number when the POST request to create a new ticket timed out and did not return an Ensono number in the response.
5. Attributes.SentTimestamp – the epoch time for when the transaction was added to the queue
6. Attributes.ClientReferenceAvailable – shows if Ensono received client ticket number or not

## Incident/Request Message Body and Fields

The Body of each message contains a payload similar to the inbound format above, with some differences. Client-populated fields and UUIDs are included from the Ensono System and can be ignored.

Below are all possible fields that could be in the Body of a message.

Field	On Create	On Update	In changed: {}	On Notes	On Close	Can be Ignored
clientName	Yes	Yes		Yes	Yes	
clientTicketNumber	Yes	Yes		Yes	Yes	
ticketType	Yes (Incident or Request)	Yes		Yes	Yes	
ticketNumber	Yes (Ensono number)	Yes		Yes	Yes	
urgency	Yes (1, 2, or 3) or mapped value	Yes		Yes	Yes	Yes
impact	Yes (1, 2, or 3) or mapped value	Yes		Yes	Yes	Yes
priority	Yes (1, 2, 3, or 4)	Yes	Yes	Yes	Yes	
group	Yes (mapped client value)	Yes	Yes	Yes	Yes	
cis	Yes (comma separated)	Yes		Yes	Yes	
category	Yes (Always Empty on create)	Yes		Yes	Yes	Yes
origin	Yes	Yes		Yes	Yes	Yes
description	Yes (Free text)	Yes	Yes	Yes	Yes	
openedTime	Yes (Free text)	Yes		Yes	Yes	
status	Yes (open)	Yes	Yes	Yes	Yes	
statusNextAction	(See below)	Maybe	Maybe		Yes	
shortDescription	Yes (Free text)	Yes	Yes	Yes	Yes	
ensonoGroup	(Ensono group value)	Yes		Yes	Yes	Yes
updatedAt	(UTC)	Yes		Yes	Yes	Yes
closureCategory		Yes		Yes	Yes	
changed: {}		Yes			Yes	
notes: {}	(Ensono additional comment)			Yes		
closeCode	(See below)				Yes	
closeNotes	(Free text)		Yes		Yes	

Available values for “statusNextAction” when ticket state changed to pending are: "pending-client", "pending-vendor", "pending-change".

Available values for “closeCode” when ticket state changed to resolved are: "resolved", "unresolved", "cancelled", "resolved\*", "unresolved\*".

Every field is included from Ensono in each transaction. On updates, the client system should only update fields that are found in the “changed”: {} object. Similarly, all fields are included on notes updates, but only the contents of “notes”: {} should be applied.

## Create

```
{
  "MessageId": "0eb49bcf-53a7-4769-b37a-2342b3658a00",
  "ReceiptHandle":
"AQEBD4eNzuQjJoBJ+TRMwOo5Ub/VvF/AmGYEVooeBKrkEwUh+o9fOpYRpTJL0QhcOhOii+HM9xT41o5P0
20GA4UgnORPjyP2pSY7gfrs0VI4QTB6i9GbedsW1T6zwAtRqNPj62qoyudi0c//YtU9jWozN5kPwJ7Oe5OuWg
z/4cCC1EwOFXVzhjrRLaFSmn+lEmPYn6npnuc0bkphh1kzb8fdDPUMnoSSdDOq3bfpi7z6fAD9cnPWzzB5VvH
iudqdkORIJPdEKLu60pa7IM3CO5QuNpifiykUeEJb4Lho2Vn3d48=",
  "MD5OfBody": "385b136e9129e7f0b282ed0657772ff5",
  "Body": {
    "clientName": "testClient",
    "ticketNumber": "e-CS0879727",
    "clientTicketNumber": "",
    "ticketType": "Incident",
    "status": "open",
    "description": "test",
    "shortDescription": "test",
    "group": "Ensono Support",
    "ensonoGroup": "CS - 1st Line",
    "priority": "4",
    "impact": "3",
    "urgency": "3",
    "openedTime": "2023-01-27T22:24:53Z",
    "category": "Private Cloud",
    "origin": "Web",
    "cis": ""
  },
  "Attributes": {
    "SenderId":
"AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1674858487529",
    "ApproximateReceiveCount": "3",
    "SentTimestamp": "1674858334956",
    "SequenceNumber": "18875507807458288128",
    "MessageDeduplicationId":
"fb1cf0cacb6943d49cf62593e60d7ecbb4b6baff0edaba7984d8d3ecaf41801c",
    "MessageGroupId": "e-CS0879727",
    "ClientReferenceAvailable": false,
    "TransactionType": "Create"
  }
}
```

## Comment

```
{
  "MessageId": "02faa024-f0a7-4b9f-94a7-f4b295c922a2",
  "ReceiptHandle":
    "AQEBLpUrllRyJgillFh1Te7ZD67DIA9KeW8SAa4tDrxCZsnVszyMQNPj10eQ+aFcZHX617HKd+KRwVz04+m+uo0GAHQru
    7cIVMI5zvxlBbUAQHcE7z+SHK8rTpZg48EeD1SA0neJPF+FOzjqEXSZrPG08Ry3GGjxFGzh5PAwl1gTGYh3oOdWfTCYiEe5
    pMEnJO7mykaqWdD/Ykj+0h1QfBgFvnODjkGbqTbmKPiACPUhXUMaPVFwSHygFa1z8tKkkXbEGV22SkziceNS1DQ0oN
    W57D14Uk2/bl1T3/8pH90wX8=",
  "MD5OfBody": "73e445e737b523378c5cfafbd8c15a14",
  "Body": {
    "clientName": "testClient",
    "ticketNumber": "e-CS0879727",
    "clientTicketNumber": "",
    "ticketType": "Incident",
    "status": "pending",
    "description": "test",
    "shortDescription": "test",
    "group": "Ensono Support",
    "ensonoGroup": "CS - 1st Line",
    "priority": "4",
    "impact": "3",
    "urgency": "3",
    "openedTime": "2023-01-27T22:24:53Z",
    "category": "Private Cloud",
    "origin": "Web",
    "cis": "",
    "offHoldDate": "2023-01-28T22:26:23Z",
    "statusNextAction": "pending-caller",
    "updatedOn": "2023-01-27T22:26:29Z",
    "notes": {
      "type": "comment",
      "id": "33cd921f1b64211426a2da49b04bcb1e",
      "body": "adding comment",
      "commenter": "Ensono User",
      "commenterEmail": "Ensono.User@Ensono.com"
    }
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1674858487529",
    "ApproximateReceiveCount": "3",
    "SentTimestamp": "1674858426315",
    "SequenceNumber": "18875507830846191616",
    "MessageDeduplicationId":
      "6c3bbb44e891b70217922b8d9bbe35312f424cfb87242dfb8edeb8d05177f5e",
    "MessageGroupId": "e-CS0879727",
    "ClientReferenceAvailable": false,
    "TransactionType": "Notes"
  }
}
```

## Update

```
{
  "MessageId": "fa4055b6-043e-4057-a362-4b56898aca5d",
  "ReceiptHandle":
    "AQEBn9veQicg7VA26Coh5qoIXRsgvYTVofd9SDFwqFkFeH2ll/nRofGIEl2mBmj/ZU/vtGXWcoCvG+/OA1BTIKMDj4YXOg
    cdnUnbuyTsN5HDvWsA9Wwoa5u/jyKNDtFxFuyIS24lY9P7AB2H0rMLOg2YDDKCbmLc689+3MggEURJGHld9yNLpy3Jp
    eJ3hr8RbfHOOBprFDj/BrqfkcDWZtXmI2mM2jUqcJ3u+cmuk0cXzomjTV67gsZPDQXp3nVHpMlNk/W1SACI1GI/lg/krWr
    deAWDEyuh7WWlceGLho4uio=",
  "MD5OfBody": "5433a1890594fb7cc88ede76c8219fc0",
  "Body": {
    "clientName": "testClient",
    "ticketNumber": "e-CS0879727",
    "clientTicketNumber": "",
    "ticketType": "Incident",
    "status": "pending",
    "description": "test",
    "shortDescription": "test",
    "group": "Ensono Support",
    "ensonoGroup": "CS - 1st Line",
    "priority": "4",
    "impact": "3",
    "urgency": "3",
    "openedTime": "2023-01-27T22:24:53Z",
    "category": "Private Cloud",
    "origin": "Web",
    "cis": "",
    "offHoldDate": "2023-01-28T22:26:23Z",
    "statusNextAction": "pending-caller",
    "changed": {
      "updatedOn": "2023-01-27T22:26:29Z",
      "status": "pending",
      "statusNextAction": "pending-caller",
      "offHoldDate": "2023-01-28T22:26:23Z"
    },
    "updatedOn": "2023-01-27T22:26:29Z"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1674858487529",
    "ApproximateReceiveCount": "3",
    "SentTimestamp": "1674858426053",
    "SequenceNumber": "18875507830779119616",
    "MessageDeduplicationId":
      "e71ce70ea6492c8621b64524094b45d55a0533c3e073eb846c876c501a886f8e",
    "MessageGroupId": "e-CS0879727",
    "ClientReferenceAvailable": false,
    "TransactionType": "Update"
  }
}
```

## Update to Resolved/Closed

```
{
  "MessageId": "2dc52062-aa59-4543-8381-af8724ffd002",
  "ReceiptHandle":
    "AQEBow/CxSerKljYGwdKqHKSXwxcuiCiTH/mM5IKCz0waiX/sMNRU2V8M9GNKOfmj1h06N4ZgVg/gUqduBqC4qEQHX
    w9Lejep6UUEzmnPeWAdQac6uKrrZ/gNZuwJYnwgVtZJwXknXHqucZMl63vFrex8ff5DsOgzaiM5Na9oWDNV1JGctx0SFs
    +0RI6QDRlpPF33UNPjmvWmO7MniqUrk2dDXY2lOBGeEA8znAOp2Um33Oks8GMq+jY6FCvMxcALT++wX2cFUt8g9+V
    ZPsTxEAMv3ajuvmsGe3egwegsocsdgw=",
  "MD5OfBody": "0f8caf1e63c8b0340a73cad566e4582",
  "Body": {
    "clientName": "testClient",
    "ticketNumber": "e-CS0879727",
    "clientTicketNumber": "",
    "ticketType": "Incident",
    "status": "resolved",
    "description": "test",
    "shortDescription": "test",
    "group": "Ensono Support",
    "priority": "4",
    "impact": "3",
    "urgency": "3",
    "openedTime": "2023-01-27T22:24:53Z",
    "category": "Private Cloud",
    "origin": "Web",
    "cis": "",
    "closeNotes": "test close notes",
    "closeCode": "resolved*",
    "changed": {
      "updatedOn": "2023-01-27T22:29:36Z",
      "status": "resolved",
      "closeNotes": "test close notes",
      "closeCode": "resolved*"
    },
    "updatedOn": "2023-01-27T22:29:36Z"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1674858633136",
    "ApproximateReceiveCount": "2",
    "SentTimestamp": "1674858604868",
    "SequenceNumber": "18875507876555760384",
    "MessageDeduplicationId":
      "a24e18a216babf1e8ef462dd3b9f0b3dc06eeb06d1a70fb09c5f4858a1693c4f",
    "MessageGroupId": "e-CS0879727",
    "ClientReferenceAvailable": false,
    "TransactionType": "Closure"
  }
}
```

[illegible]

**Note:** To download the attachment, make a GET call to the URL stored in `Body.attachment` and the attachment will be returned in Binary as the body of the response.

## Return Ensono Number

```
{
  "MessageId": "c83bcb88-58c9-4c3c-b34a-37bd0de3279f",
  "ReceiptHandle": "AQEB2+lKqBz2njMBWps6VpZ7Wuyx/cB/YG9NKSMnsW66yljMB42+s0WueAMcEdUgPKR2CnsJLOAfV70fuT9xDDQXIeyflpsvUvjaAbAOm0xil8alor/A81aUHdbXiTFyFhFJzncLTwERL5hiEFrb2c439aLY4+qgbhiJFGXz1GhSArRWXcY+NYCScHHjrhAfdyQhNBZ3pJBCQAFefQlh1rCK+yzTMlogJWh5Kd+TwKDzEezlneKRiNzC45X6UBwbf1LyTqsJVcoZ6/cdH8xX+nFBjb3FEA9Mfg5LK/K8g1hkc+s=",
  "MD5OfBody": "4938c247db4e9170b258b6138ad2c2fd",
  "Body": {
    "clientName": "testClient ",
    "ticketNumber": "e-CS0881847",
    "clientTicketNumber": "INC1235678",
    "ticketType": "Incident"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPCNWDH6BTL:envxdevbridge1MainBridge",
    "ApproximateFirstReceiveTimestamp": "1675883908676",
    "ApproximateReceiveCount": "1",
    "SentTimestamp": "1675883883744",
    "SequenceNumber": "18875770347948015872",
    "MessageDeduplicationId": "c10e950df35014765f8199ea7011608ef047e0ad62e52f7f1966123378625687",
    "MessageGroupId": "e-CS0881847",
    "ClientReferenceAvailable": true,
    "TransactionType": "ReturnEnsonoNum"
  }
}
```



# Change Integration

The change integration standard process is from Ensono to Client. When the client needs to request a change to Ensono, the client is required to open a regular request to Ensono.

## The Ensono Process:

### ENSONO TRIGGERS

Ensono Change triggers are a bit different than Incidents or Requests. Changes are only initiated over the integration from Ensono when the CHG moves out of Draft and to the Assess state. For Normal Changes with external approvers, along with the 'Status Update' transaction, there will be a 'Change Approval Request' transaction sent for each external approver listed on the Change, **as well as transactions for each public attachment** that has been added to the Change.

After the initial transaction has been sent to the client, any comments made, public attachments added, or state updates that occur on the Ensono Change will be sent to the queue.

### EXTERNAL APPROVALS

For Normal Change, Ensono requires External Approval from client. An approval group which includes all the potential approvers needs to be added to Ensono system as external approver. It must have a valid email address.

After the corresponding change is created in client system, it should go through client approval process. Once the change is approved or rejected by client CAB team, client should send Ensono approve or reject response along with the name of the FINAL approver and the reason for rejecting the change in a comment.

Standard and Emergency change do not require client approval through the integration.

### CLIENT ACTIONS

When a corresponding change is created in client system, client needs to return client change number back to Ensono.

Once external approvals have been sent to the client system, client must either approve or reject the Change. Only one approval response is needed to approve or reject the Change, and any approval or rejection actions made after the first will not be considered.

Throughout the life of the Change, comments can also be added from the client side, but no other fields can be directly updated via the integration.

### ENSONO FIELDS

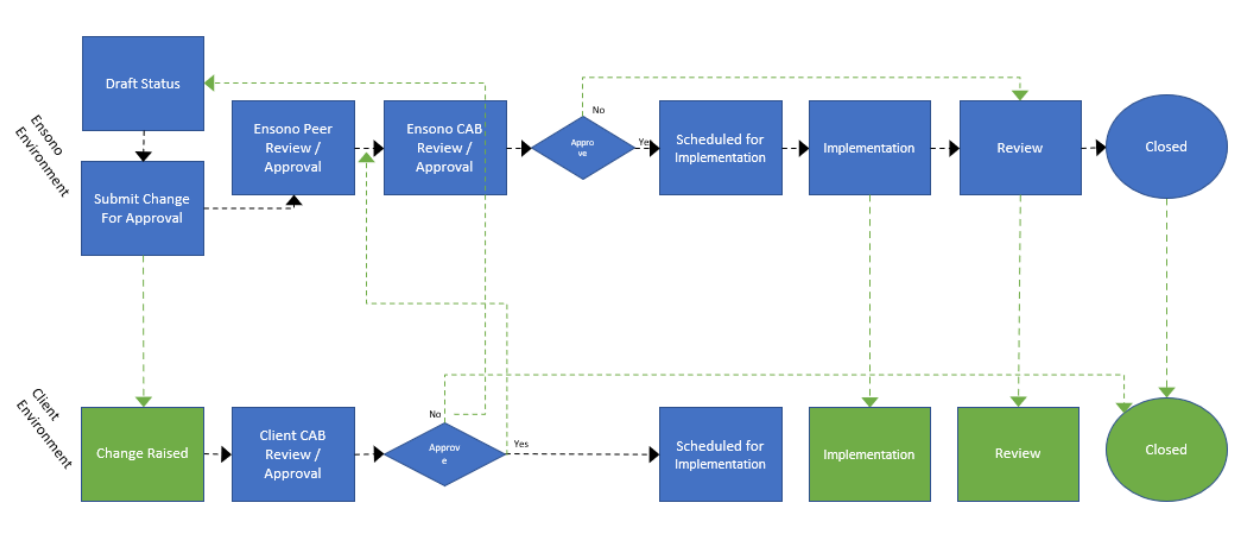
The fields updated on Changes that trigger update transactions from Ensono are:

- state
- comments
- public attachments

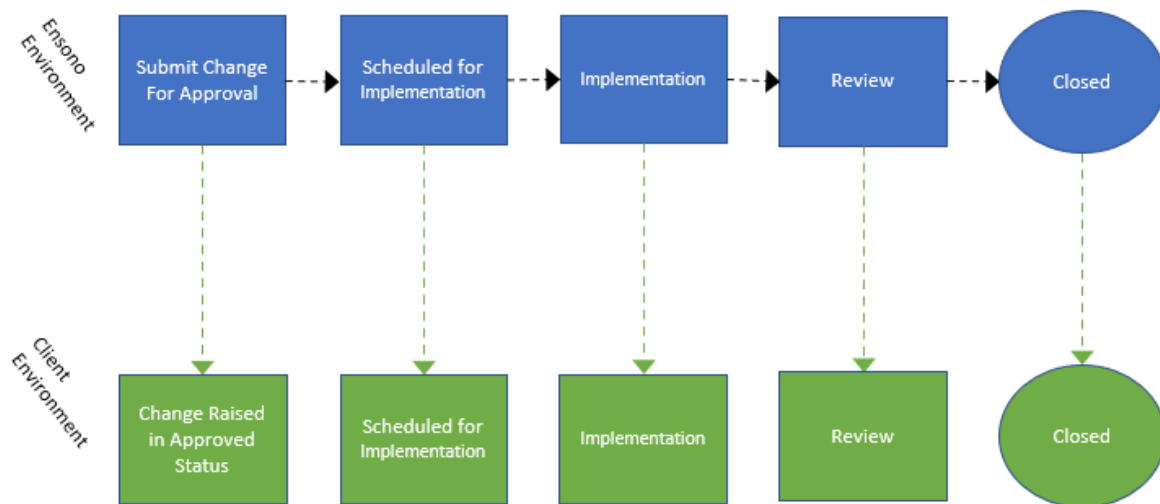
The CHG States are: Draft, Assess, Authorize, Scheduled, Implement, Review, Closed, Cancelled

## Flow Diagram:

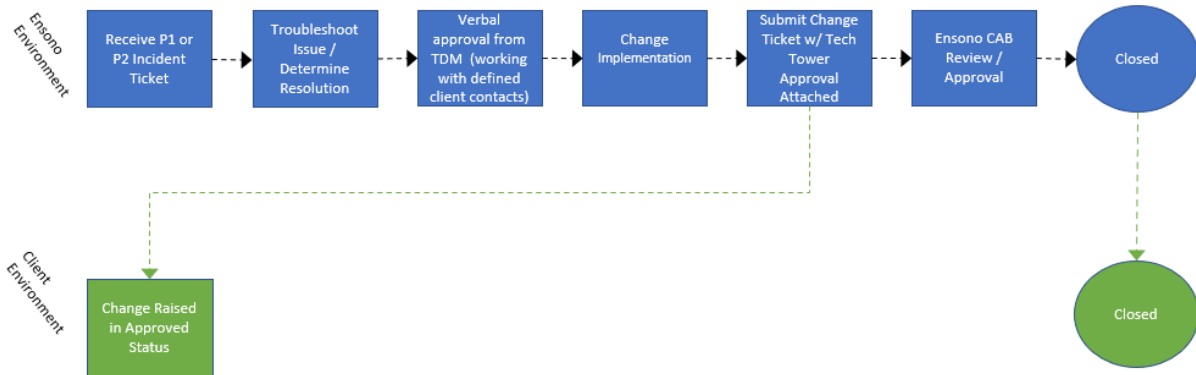
### NORMAL CHANGE



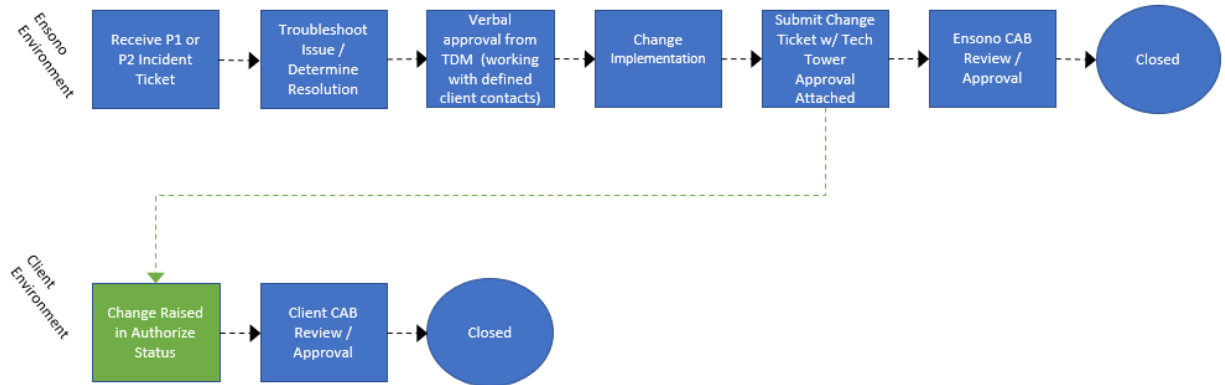
### STANDARD / PRE-APPROVED CHANGES



## EMERGENCY CHANGE OPTION 1



## EMERGENCY CHANGE OPTION 2



## DIFFERENCE BETWEEN EACH TYPE OF CHANGE

Change Type	First Transaction	Payload Data Sent to Client	Require Client Approval
Normal	Move from New to Assess	"state": "Assess"	Yes
Standard	Move from New to Scheduled	"state": "Scheduled"	No
Emergency	Move from New to Authorize	"state": "Authorize"	No

## Client to Ensono API Methods (Synchronous)

`/tickets/{TICKETNUMBER}/returnptn`

**PUT:** Updates the fields of the specified ticket's integration record with the passed PTN.

`/tickets/{TICKETNUMBER}/notes`

**PUT:** Adds 1 or more comments to the specified ticket asynchronously. Returns successful.

`/tickets/{TICKETNUMBER}/approve`

**PUT:** Approves an Approval Task from the client

`/tickets/{TICKETNUMBER}/reject`

**PUT:** Rejects an Approval Task from the client

## Client to Queue API Methods (Asynchronous)

`/queue`

**GET:** Returns 1-10 messages in the queue to be processed by the client. Once a message has been processed, the client should use the ReceiptHandle of the message to delete the message from the queue to prevent duplicate processing. Queue Messages can be grouped by their MessageGroupID (ticket number) and ordered by SentTimestamp. Since the Queues are FIFO (First In First Out), messages belonging to the same MessageGroupID must be deleted before the next messages can be retrieved.

`/queue/{RECEIPTHANDLE}`

**DELETE:** Deletes the specified message from the queue.

Note: receiptHandle needs to be URL encoded.

## Using the API -Inbound to Ensono

### RETURN CLIENT TICKET NUMBER TO ENSONO

#### Request:

**PUT** ENVIRONMENTURL/tickets/e-CHG0121253/returnptn

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Change",
    "clientTicketNumber": "1234567",
  }
}
```

#### Response:

**CODE:** 200

#### BODY:

```
{"Response": "PTN added"}
```

## ADD COMMENTS

### Request:

**PUT** ENVIRONMENTURL/tickets/e-CHG0121253/notes

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Incident",
    "clientTicketNumber": "123456",
    "comments": [{
      "body": "moving to different group",
      "commenter": "Jason Doe",
      "commentTime": "2017-07-03 17:55:48",
      "commenterEmail": "Jason.Doe@abc.com",
      "commenterPhone": "555-909-2395"
    }]
  }
}
```

### Response:

**CODE:** 200

#### BODY:

```
{"Response": "Comments added"}
```

## APPROVING A CHANGE

### Request:

**PUT** ENVIRONMENTURL/tickets/e-CHG0000642/approve

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Change",
    "clientTicketNumber": "123456",
    "approver": "approver@clientname.com",
    "approvalId": {value read from queue transaction},
    "comments": [{
      "body": "Jason Doe, Email: Jason.Doe@abc.com, Phone: 000-000-000 approved
Change Request CHG00029641 via Email at 2021-02-19 06:41:54 CST."
    }]
  }
}
```

### Response:

**CODE:** 200

#### BODY:

```
{"Response": "Approval Received Successfully"}
```

Notes: the comments field in this payload is different from the one in the comment transactions. It only contains 'body' field. Please concatenated the user info in the body field.

## REJECTING A CHANGE

### Request:

**PUT** ENVIRONMENTURL/tickets/e-CHG0000642/reject

#### HEADERS:

Content-Type: application/json

x-api-key: provided separately

#### BODY:

```
{
  "payload": {
    "clientName": "YourCompanyName",
    "ticketType": "Change",
    "clientTicketNumber": "123456",
    "approver": "approver@clientname.com",
    "approvalId": {value read from queue transaction},
    "comments": [{
      "body": "Jason Doe, Email: Jason.Doe@abc.com, Phone: 000-000-000 rejected
this Change Request CHG00029641 due to time conflict via Email at 2021-02-19
06:41:54 CST."
    }]
  }
}
```

### Response:

**CODE:** 200

#### BODY:

```
{"Response": " Rejection Received Successfully"}
```



## Ensono Change Message Body and Fields

Below are all possible fields that could be in the Body of a message for an Ensono Change.

### CHANGE STATE UPDATE FIELDS

Field	Comment
clientName	
clientTicketNumber	
ticketType	Change
ticketNumber	Ensono ticket number
ensonoGroup	Ensono group name
group	Mapped client group name
approval	Approved, Rejected, Not Yet Requested, Requested
openedAt	UTC
updateOn	UTC
requestedByName	
shortDescription	Free-text. 160 characters
type	Normal, Emergency, Standard
category	Hardware, Software, Service, System Software, Applications Software, Network, Telecom, Documentation, Patching, Removal
startDate	UTC. Schedule start date. Need to be converted to local time
endDate	UTC. Schedule end date. Need to be converted to local time
risk	Low, Moderate, High. Calculated based on risk assessment.
Impact	Low, Medium, High
description	Free-text. Description of the change. 4000 characters
justification	Free-text. Reason details. 4000 characters
preChangeChecks	Free-text. 4000 characters
implementationPlan	Free-text. 4000 characters
testPlan	Free-text. 4000 characters
backoutPlan	Free-text. 4000 characters
state	New, Assess, Authorize, Scheduled, Implement, Review, Closed, Cancelled
cis	List of Cis separated by “,”
notes:{}	<b>Only included when transactionType is “Notes”</b>
closeNotes	Free-text. 4000 characters. <b>Only included when state is “Closed”</b>
closeCode	Successful, Successful with issues, Successful Outside Change Window, Unsuccessful <b>Only included when state is “Closed”</b>

## CHANGE APPROVAL REQUEST FIELDS

Field	Comment
clientName	
ticketType	"Change"
ticketNumber	Ensono ticket number
clientTicketNumber	Client ticket number
description	Free-text
state	"Assess"
approvalState	"Requested"
approvalId	Needed to send Ensono approve/reject call
approverName	User's name
approverEmail	Needed to send Ensono approve/reject call
group	Mapped client group name
ensonoGroup	Unmapped Ensono group

## Change Create

```
{
  "MessageId": "6c11b270-b88f-465e-b8ba-71f712ab7f61",
  "ReceiptHandle": "AQEBG39tqeND8aOlk08/m6qmpwi2rHFyCdqT0y4pKE..."
  "MD5OfBody": "ae4c0ae9f1f7cedff3416e685ff64599",
  "Body": {
    "clientName": "testClient",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032103",
    "group": "CLIENT - Ensono Other",
    "ensonoGroup": "Platform BPMDevandSupport",
    "openedAt": "2023-02-08T16:25:44Z",
    "updatedOn": "2023-02-08T16:25:48Z ",
    "requestedByName": "Ensono User",
    "shortDescription": "test",
    "type": "Normal",
    "category": "Other",
    "approval": "requested",
    "startDate": "2023-02-08T16:25:48Z",
    "endDate": "2023-02-08T16:25:52Z",
    "risk": "High",
    "impact": "Low",
    "description": "test",
    "justification": "test",
    "preChangeChecks": "test",
    "implementationPlan": "test",
    "testPlan": "test",
    "backoutPlan": "test",
    "state": "Assess",
    "priority": "4",
    "cis": "2000"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675873891617",
    "ApproximateReceiveCount": "4",
    "SentTimestamp": "1675873625881",
    "SequenceNumber": "18875767721935087872",
    "MessageDeduplicationId": "55ae5bacba7835629ec24a58f4517791d92d5",
    "MessageGroupId": "e-CHG0032103",
    "ClientReferenceAvailable": false,
    "TransactionType": "Create"
  }
}
```

## Change State Update

```
{
  "MessageId": "ea7e604b-b697-45e9-89c6-e2e8dc40058c",
  "ReceiptHandle": "AQEBhSxISlxyg6nY1ioOOTH7busJAawkUbvewo..."
  "MD5OfBody": "bf92c2b8eae7f1951ab41eecfe2abe70",
  "Body": {
    "clientName": "testClient",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032103",
    "group": "CLIENT - Ensono Other",
    "ensonoGroup": "Platform BPMDevandSupport",
    "openedAt": "2023-02-08T16:25:44Z",
    "updatedOn": "",
    "requestedByName": "Ensono User",
    "shortDescription": "test",
    "type": "Normal",
    "category": "Other",
    "approval": "requested",
    "startDate": "2023-02-08T16:25:48Z",
    "endDate": "2023-02-08T16:25:52Z",
    "risk": "High",
    "impact": "Low",
    "description": "test",
    "justification": "test",
    "preChangeChecks": "test",
    "implementationPlan": "test",
    "testPlan": "test",
    "backoutPlan": "test",
    "state": "Authorize",
    "priority": "4",
    "cis": "2000"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675876440401",
    "ApproximateReceiveCount": "1",
    "SentTimestamp": "1675875874559",
    "SequenceNumber": "18875768297596655872",
    "MessageDeduplicationId": "9ee98fbf2c5f7bad6e244cfa2cdf",
    "MessageGroupId": "e-CHG0032103",
    "ClientReferenceAvailable": false,
    "TransactionType": "Update"
  }
}
```

## Change Approval Request

```
{
  "MessageId": "3234041c-eefe-4dd7-9eb8-cefd0d4d9da7",
  "ReceiptHandle": "AQEBIMHq3/XhyFnvujJNfKLwKALN9JWa/...",
  "MD5OfBody": "9ecd8092a59461b272e2cb8508a4e6cd",
  "Body": {
    "clientName": "testClient",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032103",
    "group": "CLIENT - Ensono Other",
    "ensonoGroup": "Platform BPMDevandSupport",
    "description": "test",
    "state": "Assess",
    "approvalState": "Requested",
    "approvalId": "157efa0f1b702d1026a2da49b04bcbf4",
    "approverName": "Default User",
    "approverEmail": "noreply@client.com"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675873891617",
    "ApproximateReceiveCount": "4",
    "SentTimestamp": "1675873626479",
    "SequenceNumber": "18875767722088175872",
    "MessageDeduplicationId": "d89d7b98156d8a50a68",
    "MessageGroupId": "e-CHG0032103",
    "ClientReferenceAvailable": false,
    "TransactionType": "Approval Request"
  }
}
```

## Change Redraft

```
{
  "MessageId": "44266d28-6cf8-4555-8cd4-295105926804",
  "ReceiptHandle": "AQEBmbX5ClgMk9clzKdP9YCF1TpP8oS08kTLI7k3RWGU+n+..."
  "MD5OfBody": "2fc0b6b363f98de45b45bfbbce4ab87f",
  "Body": {
    "clientName": "testClient",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032075",
    "group": "CLIENT - Ensono Other",
    "ensonoGroup": "Platform BPMDevandSupport",
    "openedAt": "2023-02-03T15:37:16Z",
    "updatedOn": "",
    "requestedByName": "Ensono User",
    "shortDescription": "test",
    "type": "Normal",
    "category": "Other",
    "approval": "not requested",
    "startDate": "2023-02-10T15:37:18Z",
    "endDate": "2023-02-10T15:37:20Z",
    "risk": "Low",
    "impact": "Low",
    "description": "test",
    "justification": "test",
    "preChangeChecks": "test",
    "implementationPlan": "test",
    "testPlan": "test",
    "backoutPlan": "test",
    "state": "New",
    "priority": "4",
    "cis": "2000"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675877682032",
    "ApproximateReceiveCount": "1",
    "SentTimestamp": "1675877674804",
    "SequenceNumber": "18875768758459375872",
    "MessageDeduplicationId": "85597299277ed71d97ce4cf09597319a54091c7112b7f0581408269c48348175",
    "MessageGroupId": "e-CHG0032075",
    "ClientReferenceAvailable": false,
    "TransactionType": "Redraft"
  }
}
```

```
{
  "MessageId": "e9b249e4-7916-484b-a803-f4f9e3a2a6b7",
  "ReceiptHandle": "AQEBi9c4yTNeaXEqbtR5pPejA5QvSzEt4VZK4OWWuCxU9V4hwEUfgv...",
  "Body": {
    "clientName": "testClient",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032103",
    "group": "CLIENT - Ensono Other",
    "ensonogroup": "Platform BPMDevandSupport",
    "attachmentName": "rocket.jpg",
    "attachmentInfo": "https://envisiontest.service-now.com/api/now/attachment/22aac7831bb02d1026a2da49b04bcb36",
    "attachmentSize": "138092",
    "attachmentType": "image/jpeg",
    "attachment": "https://envxs2qdev1outbound-bucket.s3.us-west-1.amazonaws.com/rocket.jpg?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ASIAYCQITFFPAUMR3XNB%2F20230208%2Fus-west-1%2Fs3%2Faws4_request&X-Amz-Date=20230208T172243Z&X-Amz-Expires=7200&X-Amz-Security-Token=IQoJb3JpZ2luX2VjEPL%2F%2F%2F%2F%2F%2F%2F%2F%2FwEaCXVzLXdlc3QtMSJHMEUCIHT5YgEViFSh0WzINW3C5AwGU%2BIpaEAZxgVPCDblAhAAIEA8RD7Q2ce8qz2VoRI6TIkv4l5xtbc8BFWo%2F%2BmZxj7jcEqhAMlexADGgw1NTUxNDI1NTU5OTgiDIYHIL5xm1iX%2FuWNDCrhAgBL7B6LkmwdJp%2BMXE8s7R111GsliJq043WLYRhataHXt%2FlaQTOKztT8Tp0v4oT23m8%2FXSLM74BU2c1v1wFFB14YSkhx%2FbvrrOrZspbgHVgWj%2FDgIVrKONQY9b%2FBK8iOncwp2aWhdlYRGylF%2FCbr6ox8FcF4i25WUswjwLOlaRinEoUBF7ax%2Bx8xseLiKhf%2FglkMOFlukbwYMHL0KaqhSEL%2B4xd7JZ5QA0Dsrg%2Ftpl1vnMn%2B9n8yCJEznM2nTnM9dWNUagTZmQvpls3PJWf1F%2FtQRcqMabSFCaj7X1m0Xy55u5sbxRAHzcx%2B8yTV%2BCcoE%2FTto6S1Zj7YgmUZgPIRckRcCVBC00oiCRmaci505GhdJMpzMpERNRoIJHeFr%2FFTkpJ1ABm6pPdIFkte5zisqs%2BA0x6lr4gwAf%2Bih2uyfloL2ivex5ZF83n6qtTUMG16lAs13Xu0gl30Uc1GLmbhMOGOj58GOp4BSNQX%2Fa2EkeQ5KBHTjLi1eFHxFRldB2x6F51dPo% "
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxs2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675876963051",
    "ApproximateReceiveCount": "1",
    "SentTimestamp": "1675876866318",
    "SequenceNumber": "18875768551486960128",
    "MessageDeduplicationId":
      "9ae6b5f3eb5a5a0ad5dde8b7472ddd8592cfd9bd10c3fa900653843d095cf43b",
    "MessageGroupId": "e-CHG0032103",
    "ClientReferenceAvailable": false,
    "TransactionType": "Attachment"
  }
}
```

## Change Comment

```
{
  "MessageId": "ed203d54-0a26-4112-8571-3204420ede65",
  "ReceiptHandle": "AQEBPlv7vifTAkE5CeGhC+Muqe/KEIST87ysqJcsntCKXdFfNxTKlh...",
  "MD5OfBody": "44238058805dbeb3a7976d8ed4ac3ecb",
  "Body": {
    "clientName": "testClient",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032103",
    "group": "CLIENT - Ensono Other",
    "ensonoGroup": "Platform BPMDevandSupport",
    "openedAt": "2023-02-08T16:25:44Z",
    "updatedOn": "2023-02-08T17:20:38Z",
    "requestedByName": "Ensono User",
    "shortDescription": "test",
    "type": "Normal",
    "category": "Other",
    "approval": "approved",
    "startDate": "2023-02-08T16:25:48Z",
    "endDate": "2023-02-08T16:25:52Z",
    "risk": "High",
    "impact": "Low",
    "description": "test",
    "justification": "test",
    "preChangeChecks": "test",
    "implementationPlan": "test",
    "testPlan": "test",
    "backoutPlan": "test",
    "state": "Review",
    "priority": "4",
    "cis": "2000",
    "notes": {
      "type": "comment",
      "id": "54ea87031bf02d1026a2da49b04bcb8e",
      "body": "Adding a comment",
      "commenter": "Ensono User",
      "commenterEmail": "Aubrey.Neumeier@Ensono.com"
    }
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675876963051",
    "ApproximateReceiveCount": "1",
```



```
"SentTimestamp": "1675876865373",  
"SequenceNumber": "18875768551245040896",  
"MessageDeduplicationId":  
"b57c6eaba8f89739915520f3927fb7dedbd0f119feef69846be156ad32636898",  
"MessageGroupId": "e-CHG0032103",  
"ClientReferenceAvailable": false,  
"TransactionType": "Notes"  
}  
}
```

## Change Closure

```
{
  "MessageId": "4c38091b-5c48-4e5e-b53f-7ad6eb51fc12",
  "ReceiptHandle": "AQEB1Ro19BykyfCqQBU9tXzE/eGmTHdg4R/pmt/6EP5t4Y9FhELPO6...",
  "Body": {
    "clientName": "YourCompanyName",
    "ticketType": "Change",
    "clientTicketNumber": "",
    "ticketNumber": "e-CHG0032103",
    "group": "CLIENT - Ensono Other",
    "ensonoGroup": "Platform BPMDevandSupport",
    "openedAt": "2023-02-08T16:25:44Z",
    "updatedOn": "",
    "requestedByName": "Ensono User",
    "shortDescription": "test",
    "type": "Normal",
    "category": "Other",
    "approval": "approved",
    "startDate": "2023-02-08T16:25:48Z",
    "endDate": "2023-02-08T16:25:52Z",
    "risk": "High",
    "impact": "Low",
    "description": "test",
    "justification": "test",
    "preChangeChecks": "test",
    "implementationPlan": "test",
    "testPlan": "test",
    "backoutPlan": "test",
    "state": "Closed",
    "priority": "4",
    "cis": "2000",
    "closeNotes": "close notes text",
    "closeCode": "Successful Outside Change Window"
  },
  "Attributes": {
    "SenderId": "AROAYCQITFFPPLPIRNUQG:envxsn2qdev1SN2QBridge",
    "ApproximateFirstReceiveTimestamp": "1675876963051",
    "ApproximateReceiveCount": "1",
    "SentTimestamp": "1675876955306",
    "SequenceNumber": "18875768574267887872",
    "MessageDeduplicationId": "730e17019d1b4ece7f92327929d8176d036ff9fa60cc0bf80027f6e64d5e6381",
    "MessageGroupId": "e-CHG0032103",
    "ClientReferenceAvailable": false,
    "TransactionType": "Closure"
  }
}
```

}

## Normal Change Scenarios

Client	Ensono Action	Data Sent to Client	Client Action
Approve	Implement successful	"state": "Closed" "closeCode": "Successful" "closeNotes": "Free text"	Close client change. Map close code
Approve	Implement successful with issues	"state": "Closed", "closeCode": "Successful with issues" "closeNotes": " Free text "	Close client change. Map close code
Approve	Implement Failure	"state": "Closed", "closeCode": "Unsuccessful" "closeNotes": " Free text "	Close client change. Map close code
Approve	Reject. Move to New	"state": "New"	Cancel client change
Approve	Cancelled	"state": "Cancelled"	Cancel client change.
Approve	Move to New	"state": "New"	Cancel client change.
Approve	Moved to New then set to Assess again	"state": "Assess"	Cancel previous change if not done so already. Create a new change
Reject	Cancelled	"state": "Cancelled"	Do nothing since client change should be in closed state
Reject	Move to New	"state": "New"	Cancel client change.
Reject	Moved to New then set to Assess again	"state": "Assess"	Create a new change when Ensono sends the next "Assess" Transaction

# Common Error Responses

## `{"message": "Missing Authentication Token"}`

Using the wrong combination of Method and Path.

1. May be environment typed wrong: Cert instead of cert (case sensitive)  
`https://smconnect.ensono.com/testclient57/Cert/tickets/INC0318118`  
Solution: <https://smconnect.ensono.com/testclient57/cert/tickets/INC0318118>
2. May be next URI path typed wrong (case sensitive)  
`https://smconnect.ensono.com/testclient57/cert/Tickets/INC0318118`  
Solution: <https://smconnect.ensono.com/testclient57/cert/tickets/INC0318118>

## `{"message": "Forbidden"}`

1. Incorrect ENVIRONMENTURL (case sensitive)
2. E.g. Testclient57 as part of <https://smconnect.ensono.com/Testclient57/cert/tickets/INC0318118>  
Solution: <https://smconnect.ensono.com/testclient57/cert/tickets/INC0318118>

## `{"Response": "Bad Path"}`

This is caused by a bad Method (GET, POST, PUT) and/or Path (/tickets/notes) or a misspelled path (/tickets/INC000000/notes).

## `{"Response": "APIKey not authorized for ticket company"}`

This is caused by trying to update or add to a ticket that does not belong to the company making the API call.

## `{"Response": "Unable to find mapping for ClientName"}`

This is caused by a misspelling of the clientName field, or the mapping for that company has not been set up for use via API.

## `{"Response": ["assignment_group sys_id or name not found.", "This ticket cannot be created."]}`

This is an error returned from SN if an invalid group name is provided.

```
{"Error": {"Code": "InvalidParameterValue", "Message": "Value  
AQEBvxJoo81y8ewBSlc2RH9TgJVR02/WSz7w8tUy7lo+8XpJgMEsFRgOZaMPgVufSe03V9hfT4/M/xHAK1vXv  
1Ntp9i5LOnCdb/Y5Gg1P7q4WwCQr56CXIJycM5K3T7d3hkPU9kv1tomkGaCtU85+2gMVUsSNG+Yrtd9nKbgt  
i4lygu6fN0sou7PYeKRecu45eXtewVD+z/c8vizJFCGY5BUvdoACK4GulndCmWeXV5oWc/w/23SO90xe1KW9/  
KKAerhSwTKMOw1WpbLO/T6YPHcTumL6tS3HOV5ZMeTMeCgle2jtJDyb99Jle6Tk0xfZ9ZI for parameter  
ReceiptHandle is invalid. Reason: The receipt handle has  
expired.", "Type": "Sender"}, "RequestId": "377ec271-6395-5f50-9e0b-94befbb22782"}
```

This is an error returned when trying to delete an expired ReceiptHandle. The message will need to be read again and the new ReceiptHandle can be used for the DELETE call.

```
{"Error":{"Code":"ReceiptHandleIsInvalid","Message":"The input receipt handle  
\"AQEBvxJoo81y8ewBSlc2RH9TgJVR02/WSz7pJgMEsFRgOZaMPgVufSe03V9hfT4/M/xHAK1vXv1Ntp9i5LO  
nCdb/Y5Gg1P7q4WwCQR56CXlJycM5K3T7d3hkPU9kv1tomkGaCtU85+2gMVUsSNG+Yrtd9nKbgti4lygu6fN  
0sou7PYeKRecu45eXtewVD+z/c8vizJFCGY5BUvdoACK4GulndCmWeXV5oWc/w/23SO90xe1KW9/KKAErhS  
wTKMOw1WpbLO/T6YPHcTumL6tS3HOV5ZMeTMeCgle2jtJDyb99Jle6Tk0xfZ9ZI\" is not a valid receipt  
handle.\"","Type":"Sender"},"RequestId":"0672169c-060d-5a8a-95a6-40539ee34467"}
```

This is an error returned when trying to delete a non-existent ReceiptHandle. Try URLEncoding the ReceiptHandle before adding it to the DELETE

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<Error><Code>AccessDenied</Code><Message>Request has expired</Message><X-Amz-  
Expires>120</X-Amz-Expires><Expires>2020-02-25T21:38:36Z</Expires><ServerTime>2020-02-  
25T21:43:20Z</ServerTime><RequestId>FC887DC0042A67FF</RequestId><HostId>6tHHwHVXP5Uivsbus  
xVCiOnWmaLTiV1tta2di3aw1SpE7eH3vLgXj/JScFqbmcXj4Duu64ogSN0=</HostId></Error>
```

This error is returned when the Pre-signed URL for an attachment upload has expired. The GET call will need to be made a second time and the new URL used to complete the upload.

# Post Production Support

## Emergency Situation “Bridge Down”

If the bridge is failing, not responding, and/or the client is unable to create tickets via the bridge, then please call the Ensono Help Desk (+1 833-662-7378) and open a high priority incident for the Ensono Group ‘Platform BPMDevandSupport’.

## Non-fatal Incidents, Troubleshooting Assistance, or Testing

For any non-fatal service affecting issues, please submit an appropriate priority incident to the Ensono Group ‘Platform BPMDevandSupport’ with details of what happened that should not have, or visa versa, along with as many example ticket numbers as possible. The more evidence/examples included, the faster we should be able to determine what happened.

If Ensono’s assistance is needed to troubleshoot, test, or investigate something not related to a failure, submit a request with the details of what’s needed to the Ensono Group ‘Platform BPMDevandSupport’.

## Enhancement Requests

For any enhancements to existing integrations that are not related to a failure or issue, please submit a detailed request to the Ensono Group ‘Platform BPMDevandSupport’.

## Group Mapping Update Requests

For any update to existing group mapping, please submit a detailed request to TDM team ‘US - Service Management’. If TDM approves, Ensono Group ‘Platform BPMDevandSupport’ will update the mappings. If a new group needs to be created, please submit a detailed request to Ensono Group ‘ServiceNow Admin’.