

Table of Contents

Overview of Project Requirements.....	2
Method 1: Basic ticket management.....	2
Method 2: eBonding	2
a. Ticket created or updated by Toyota	2
b. Ticket updated by Denovo	2
c. Ticket resolved (but not closed) by Denovo	2
Ticket Workflow Diagram.....	3
Configuration Requirements for API eBonding.....	4
Design of REST API Integration	4
Denovo's Scripted REST API	4
Denovo's Create Ticket Endpoint	5
Example of Request Body	5
Example cURL command to create new ticket	5
Example of Response.....	6
Denovo's Update Ticket Endpoint.....	6
Example cURL command to update existing ticket (belonging to authorized user company)	7
Example of Response When Status Is Resolved or Closed (and Closed Notes Included).....	7
Toyota's Update Ticket Endpoint – Example.....	8
Ticket Type Field Mapping	9
eBonding Response.....	9
Denovo's Outbound REST Web Service.....	9
Toyota's Outbound REST Web Service	9
Denovo's Business Rules	11
Toyota's Business Rules (Suggestion).....	11
Deliverables Responsibility Matrix.....	11

Overview of Project Requirements

Toyota has elected to integrate their ticket creation and management process with Denovo's. Toyota requires the ability to create and manage tickets using two methods:

- **basic ticket management** via a service call or by entering a ticket in Denovo on Demand
- **eBonding** where Toyota creates a ticket for Denovo using Toyota's ServiceNow instance

Method 1: Basic ticket management

In basic ticket management, Toyota requests that Denovo create the ticket, and Denovo maintains the ticket data until it is resolved. Toyota can request that a ticket be created by calling Denovo, emailing Denovo's support team, or by entering ticket details in Denovo on Demand. In this scenario, there is only one domain where the ticket is tracked: Denovo's ServiceNow instance. Toyota is notified when the ticket is created, when customer notes are added, and when the ticket is resolved.

Method 2: eBonding

eBonding is a scenario where Toyota's ServiceNow instance integrates directly with Denovo's ServiceNow instance via REST API. This scenario assumes Toyota has integrated their ServiceNow ticketing system with Denovo's ServiceNow ticketing system. When Toyota creates or updates a ticket for Denovo in Toyota's system, it is synchronized with a ticket that is created or updated by Denovo's system. Only Toyota can create a ticket in this scenario. However, once the ticket is created, Toyota can call Denovo's published eBonding REST API endpoints to send and receive information on the ticket residing in Denovo's system.

In order for Toyota's ServiceNow instance to receive updates from Denovo, Toyota must provide Denovo with a published REST API that interacts with their ServiceNow instance. This allows changes that Denovo makes to the ticket to be reflected in Toyota's instance.

There are three types of workflows within the eBonding scenario, described below:

a. Ticket created or updated by Toyota

Any time a ticket is created or updated by Toyota, Toyota triggers a call to Denovo's REST API to create or update the ticket in Denovo's ServiceNow instance.

b. Ticket updated by Denovo

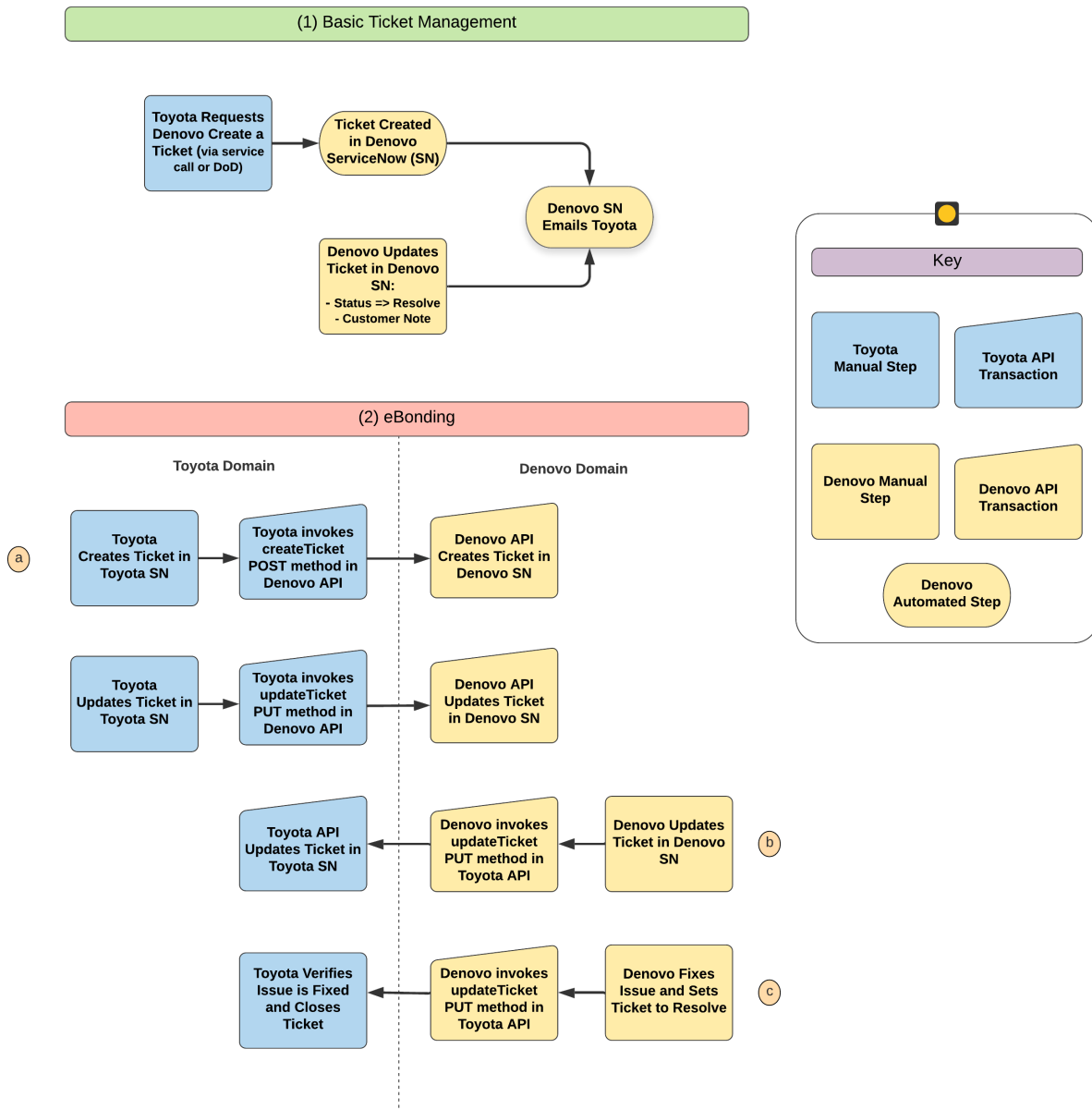
When a ticket is updated by Denovo staff (including the adding of customer notes), Denovo calls Toyota's REST API to update the ticket in Toyota's instance.

c. Ticket resolved (but not closed) by Denovo

When Denovo fixes the issue and changes the status of a ticket created by Toyota to RESOLVED, Denovo calls Toyota's REST API to change the status of the ticket. After Toyota verifies the ticket has been fixed, they manually change the status to CLOSED.

Ticket Workflow Diagram

The following diagram details the workflows for the two scenarios of basic ticket management and eBonding:



Configuration Requirements for API eBonding

1	Authentication	Toyota's API client must send valid user credentials to Denovo's API using basic authentication over SSL . This means a valid user and password must be configured for a service account used as the API User for the client requesting access to the published API resources. Note that the API User is only a service account and not a standard user or employee of the company.
2	Blank or null values	Any parameter submitted in the body of the API request that does not have a value (or has a blank value) will be ignored.
3	Unsupported parameters	Unsupported parameters submitted in the body of the API request will be ignored.
4	Field mapping	Optionally, Toyota's ticket types should be mapped to Denovo's ticket types - this includes the categories and assignment groups for the ticket. See <i>Ticket Type Field Mapping</i> .
5	Coordinated testing	Denovo engineering and the operations team will engage Toyota for testing before enabling API eBonding in production.

Design of REST API Integration

Denovo will provide a published ticketing system API to give Toyota the ability to create, view and update new incidents or service requests in Denovo's ticketing system. Denovo will also trigger calls to Toyota's published API in order to notify Toyota about any tickets Denovo updates.

There are 3 basic parts used to build the API integration: the scripted REST API, the ticket field mapping, and the eBonding response, described below.

Denovo's Scripted REST API

The **scripted REST API** accepts authorized requests on the base **/api/deno2/dod/ticket** endpoint. The available endpoints for creating a ticket and updating a ticket are described as follows:

Denovo's Create Ticket Endpoint

To create a new incident or service request in Denovo's instance, POST the desired JSON body to the resource path below:

Path	<host>/api/deno2/v1/dod/ticket	
Method	POST	
Supported Content Type	application/json	
Supported JSON Body Parameters		
id	Mandatory	ticket number provided by Toyota’s ticketing system - this is Toyota’s cross reference for any ticket communications. In Denovo’s instance, it will be saved in the u_correlation_id field.
subject	Mandatory	short description of the ticket
user	Optional	email address of an authorized user who is creating the ticket. Note: This user is not necessarily the API User whose service account is sending the transaction. If a user is not specified in the ServiceNow caller_id field when the ticket id is submitted, the default API user is supplied.
description	Optional	more detailed description of the ticket - if this is not provided, the subject is copied as the description
category	Optional	top-level category provided by the customer ticketing system
sub_category	Optional	sub-category provided by the customer ticketing system
notes	Optional	any additional notes or comments describing the ticket

Example of Request Body

```
{
  id: "22333",
  subject: "JDE Response Slow",
  category: "JDE",
  sub_category: "Performance",
  user: "jim.smith@motionmanufacturing.com",
  notes: "Login times have been slow over the last couple of hours"
}
```

Example cURL command to create new ticket

```
curl https://denovodev.service-now.com/api/deno2/v1/dod/ticket
--request POST
--header "Accept:application/json"
--header "Content-Type:application/json"
--user "ticketadmin@motionmanufacturing.com:secure_password"
--data "{ user: \"jim.smith@motionmanufacturing.com\", subject: \"JDE
Response Slow\", id: \"22333\" }"
```

Example of Response

```
{
  "result": {
    "ticket_number": "INC0730756",
    "customer_ticket_ref": "22333",
    "status": "New",
    "priority": "3",
    "subject": "JDE Response Slow",
    "description": "New eBonding Ticket sent via DoD API:\n\nSubmitted
By: jim.smith@motionmanufacturing.com\nSubject: JDE Response Slow\n\n",
    "contact_name": "Jim Smith",
    "assignment_group": "Ops Center",
    "assigned_to": "",
    "category_1": "",
    "category_2": "",
    "category_3": "",
    "category_4": "",
    "created_on_utc": "2019-09-20 20:14:23",
    "last_updated_on_utc": "2019-09-20 20:14:23",
    "created_on_denver": "2019-09-20 14:14:23",
    "last_updated_on_denver": "2019-09-20 14:14:23"
  }
}
```

Denovo's Update Ticket Endpoint

This endpoint updates an existing ticket owned by the same company as the authenticated user (the API User). To update an incident or service request in Denovo's instance, use a PUT method to add the desired JSON body to the resource path below:

Path		<host>/api/deno2/v1/dod/ticket/{number}
Path Parameters		{number} - (string – do not include the curly brackets when passing data) the Denovo ticket number to update. In Toyota’s ServiceNow instance, this value appears in the correlation_id field.
Method		PUT
Supported Content Type		application/json
Supported JSON Body Parameters		
status	Optional	must be one of the supported status values: in_progress restored resolved closed canceled
subject	Optional	an update to subject (short description) of the ticket
description	Optional	an update to the longer description of the ticket

notes	Optional	any additional notes or comments describing the ticket
user	Optional	the email address of an authorized user who is updating the ticket. Note: This user is the individual requesting the update, not the API user sending the transaction.
close_notes	Optional	notes to be added to closure details when a ticket status is set to resolved or closed (you must submit status as resolved or closed in order to include close_notes)

Example cURL command to update existing ticket (belonging to authorized user company)

```
curl https://<host>/api/deno2/dod/ticket/INC0730756
--request PUT
--header "Accept:application/json"
--header "Content-Type:application/json"
--user "jim.smith@motionmanufacturing.com"
--data "{ status: \"cancelled\" }"
```

*Example of Response When Status Is **Resolved** or **Closed** (and Closed Notes Included)*

```
{
  "result": {
    "ticket_number": "INC0730756",
    "customer_ticket_ref": "22333",
    "status": "Closed",
    "priority": "2",
    "subject": "JDE Response Slow",
    "description": "New eBonding Ticket sent via DoD API:\n\nSubmitted
By: jim.smith@motionmanufacturing.com\nSubject: JDE Response Slow\n\n",
    "contact_name": "Jim Smith",
    "assignment_group": "Unix Admins",
    "assigned_to": "Brian Jones",
    "category_1": "Software - Enterprise",
    "category_2": "JD Edwards EnterpriseOne",
    "category_3": "Platform",
    "category_4": "Performance",
    "notes": [
      {
        "created_on_utc": "2019-09-20 22:59:38",
        "created_by": "Jim Smith",
        "note": "Please let us know before rebooting the system. We
need to schedule with operations team."
      },
      {
        "created_on_utc": "2019-09-20 20:12:28",
        "created_by": "Jim Smith",
        "note": "I am in the process of validating the reboot. Stand by for
more."
      }
    ],
    "created_on_utc": "2019-09-20 21:14:23",
    "last_updated_on_utc": "2019-09-20 21:14:23",
    "created_on_denver": "2019-09-20 14:24:50",
    "last_updated_on_denver": "2019-09-20 14:24:50",
    "close_code": "Closed/Resolved by Caller",
    "closed_on_utc": "2019-09-21 09:29:07",
    "close_notes": "The issue has been resolved, many thanks!"
  }
}
```

Notes On Closing Tickets

- If a status of **resolved** is submitted, **close_notes** are required.
- If a status of **closed** is submitted, **close_notes** are optional.
- If close_notes are provided (and not empty), these notes are applied as the final close notes of the ticket.
- If close_notes are not provided (or field is empty string), a default message is created that the ticket was closed by customer IF close_notes had not been previously applied when ticket was set to resolved (see above).

Toyota's Update Ticket Endpoint – Example

The following table shows a sample configuration for Toyota's update ticket endpoint which Denovo will use to update a ticket. Toyota will not be required to publish a create ticket endpoint because Denovo will not be creating tickets in Toyota's ServiceNow instance. Note that the values below are only examples; all final configuration specs will need to be determined by Toyota.

Sample Test Endpoint (TBD by Toyota)	https://<toyota_dev>.service-now.com/api/now/import/dnv_integration_inc/{ticket_number}
Sample Production Endpoint (TBD by Toyota)	https://<toyota_prd>.service-now.com/api/now/import/dnv_integration_inc/{ticket_number}
Path Parameters	{ticketNumber} - (string) the Toyota ticket number that will be updated.
Http Method	PUT
Supported JSON Body Parameters	u_short_description u_description u_comments u_correlation_id: (mandatory) Denovo ticket number. (In Toyota's ServiceNow instance, the Denovo ticket number is displayed in the u_correlation_id field. u_state u_close_notes
Authentication type	Basic Auth
Username	dnv.api.user (example)
Password	***** TBD
Supported Content Type	application/json
Sample Request Body	{ "u_short_description": "Test dnv 01", "u_description": "Test desc dnv 01", "u_comments": "Test comment", "u_correlation_id": "INC0001", }

Ticket Type Field Mapping

Each ticket type that Toyota sends to Denovo's ticketing system may be optionally defined in a mapping table (see example below). Ideally, each type of incident or service request should be mapped to Denovo's set of categories and assignment groups in order for the eBonding integration to be more efficient. Otherwise, Denovo's support desk will have to evaluate or "triage" each ticket to categorize and assign it.

While this mapping activity is optional, in reality not performing this task would cause the integration to not function optimally. The mapping can be done after the integration is functional or incrementally, at any time.

Ticket type mapping should be performed by the Denovo Operations Team, as they are the ones that would normally triage customer tickets, and the goal of mapping is to alleviate the need for Denovo's support team to triage tickets from eBonding integrations.

The following is an example of a mapping table (additional fields should be added as needed):

Toyota Ticket Type Name	Toyota Ticket Category	Toyota Ticket Sub-Category	Denovo ServiceNow Category	Denovo ServiceNow Sub-Category	Denovo Assignment Group	Denovo Impact	Denovo Urgency
Incident	Software Update	Financial	Software – Enterprise	JDE	CST- D	2	2

eBonding Response

A push action sends an update to Toyota's ServiceNow instance when Denovo updates a ServiceNow ticket created by Toyota. Conversely, when Toyota updates a ServiceNow ticket associated with Denovo, a push action updates Denovo's ServiceNow instance. These push transactions use the REST APIs provided by both Denovo and Toyota. For Denovo, this action is active when the eBonded flag is checked in Toyota's record in Denovo's ServiceNow instance.

Denovo's Outbound REST Web Service

Denovo will add an **Outbound REST web service** named "Toyota eBonding Integration" which defines the Toyota API endpoint and user provided by Toyota (see the following figure).

Toyota's Outbound REST Web Service

Toyota will add an **Outbound REST web service** named "Denovo eBonding Integration" (or similar) which defines the Denovo API endpoint and user provided by Denovo.

REST Message
Toyota eBonding Integration

UpdateDelete

This form has annotations - click ? to toggle them - (click here to never show this again)

* NameToyota eBonding Integration

ApplicationGlobal

Accessible fromThis application scope on

DescriptionIntegration of Toyota's ServiceNow with Denovo: used for outbound notifications back to Toyota's SN instance.

* Endpointhttps://toyotadev.service-now.com/api/now/import/dnv_integration_inc

AuthenticationHTTP Request

Authentication typeBasic

Use mutual authentication

Basic auth profile

UpdateDelete

HTTP MethodsNewSearchfor textSearch

1 to 2 of 2

REST Message = Toyota eBonding Integration

	Name	HTTP method	Endpoint
	Default Update	POST	https://toyotadev.service-now.com/api/no...
	Default GET	GET	https://toyotadev.service-now.com/api/no...

Actions on selected rows...

1 to 2 of 2

Denovo's Business Rules

Denovo will add new **Business Rules** to trigger Outbound REST web services when Toyota tickets are updated or when customer notes are added in Denovo's system. The rules only apply when a ticket has been originally created by Toyota and the eBonding flag is checked in the company record.

Denovo's Business Rules are entitled: Toyota Ticket Update.

Toyota's Business Rules (Suggestion)

Likewise, Toyota will need to add new **Business Rules** (or equivalent) to trigger Outbound REST web services when Denovo tickets are created, updated or when customer notes are added in Toyota's system.

Toyota's suggested Business Rules may be entitled:

- Denovo Ticket Create
- Denovo Ticket Update

Deliverables Responsibility Matrix

The following table lists the project deliverables and responsible parties.

Deliverable	Denovo Responsibility	Toyota Responsibility
Field Mapping Definition	X	X
Denovo's Scripted REST API <ul style="list-style-type: none">• Create Ticket Endpoint• Update Ticket Endpoint• Add Toyota email(s) to whitelist	X X X	
Denovo's Outbound REST Web Service	X	
Denovo's Business Rules <ul style="list-style-type: none">• Ticket Update for Toyota• Comment Added for Toyota	X X	
Toyota's Scripted REST API <ul style="list-style-type: none">• Update Ticket Endpoint• Add Denovo email(s) to whitelist		X X
Toyota's Outbound REST Web Service		X
Toyota's Business Rules <ul style="list-style-type: none">• Ticket Update for Denovo• Comment Added for Denovo		X X
Completed Coordinated API Testing	X	X