Integration Guide

Public – Dec 2024

Version History

Date Changed	Version	Author(s)	Changes
May 25, 2024	1.0	Sean Dulac (Appko) Paul Esch (TSANet)	First version. Includes Salesforce reference integration
June 13, 2024	1.1	Sean Dulac (Appko) Paul Esch (TSANet)	API completed to include notes and response update.
Dec 13, 2024	1.2	Sean Dulac (Appko) Paul Esch (TSANet)	Major updates to the API V 1.0.8 with new endpoints. Updated SFDC section and added MS PowerApp

Table of Contents

Contributors and References

Contributors

Paul Esch, TSANet, https://www.linkedin.com/in/pauleesch/
Sean Dulac, Appko, https://www.linkedin.com/in/seandulac/

References

The following documents explain the details behind the integration framework Technology and Methods.

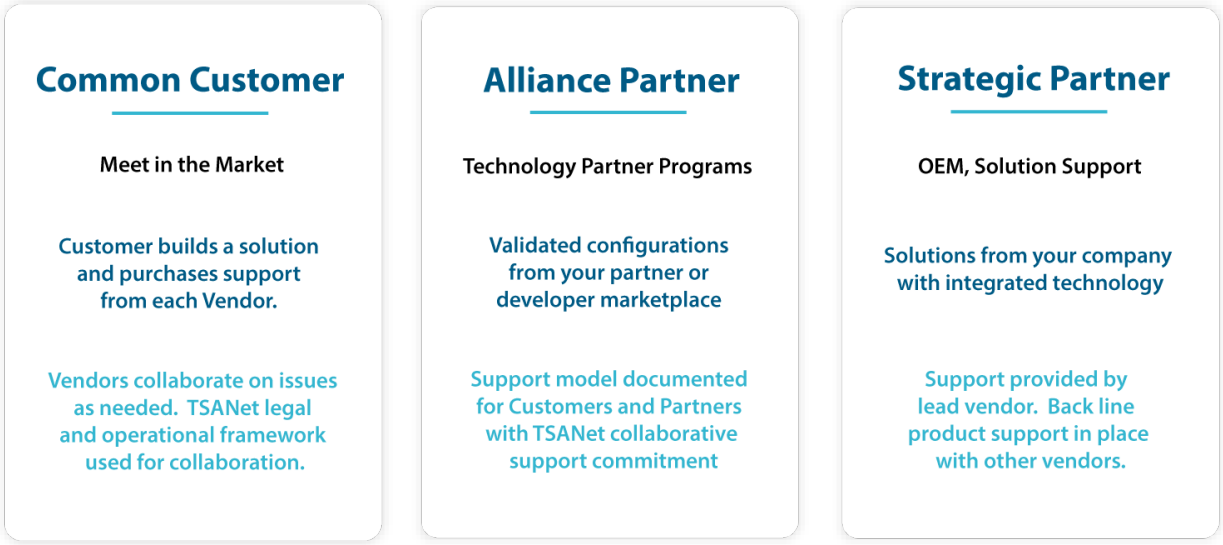
Enterprise Integration Patterns: Designing, Building, and deploying messaging solutions.
By Grregor Hohpe and Bobby Woolf -
https://www.amazon.com/o/asin/0321200683/ref=nosim/enterpriseint-20

Apache Camel: https://developers.redhat.com/products/redhat-build-of-apache-camel/overview

Executive Summary

TSANet is a member-driven, not-for-profit, global collaborative alliance of 900+ technology companies working together to improve the support experiences of their shared customers. Member companies use TSANet to enable collaboration with their Technology Partners. The TSANet Partner Framework is flexible and supports several relationship models.

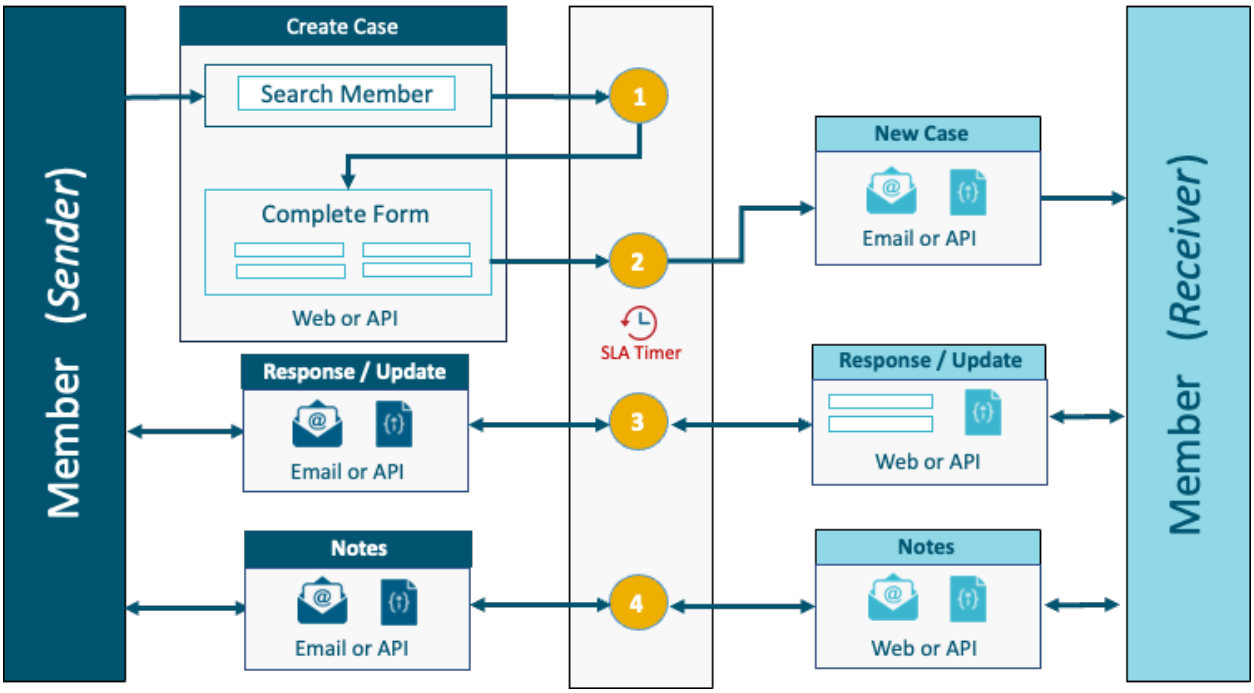
TSANet Member Benefits



TSANet Technology Partner Framework

Benefits of TSANet Connect Integration

Members use the TSANet Connect system to connect and collaborate with other members. The collaboration process is described in the image below:



Collaboration Process – Connect 2.0

Improved User Experience (Integrated B2B)

TSANet Elite Members can integrate their system into TSANet Connect to improve the user experience and process compliance. User Experience benefits are noted below.

Use Case	WebApp Collaboration	Integrated B2B
Inbound request from a member	Uses an alias or email–to-case feature of the target system.	Create a case in the system with the ability to utilize all features of the system for automation and routing
Outbound request to a member	Fragmented process. Member System, TSANet WebApp and engineer's email	End-End Process is within the Members system and workflow.
Ongoing notes and updates	user works in email/notes and manually updates the case	Updates added to the case as part of TSANet Connect notes and update API

Note: An example of user experience can be found in this document's Salesforce reference integration section.

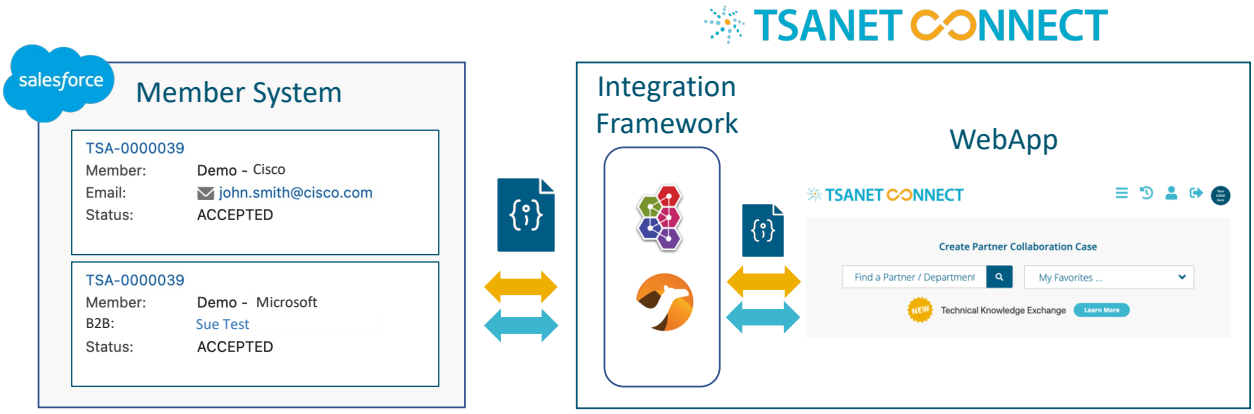
Benefits vs. Point-Point integrations

The table below highlights some of the challenges with Point-Point integrations and the benefits of TSANet Connect integration for the use cases described in this document.

Topic	Point – to - Point	TSANet Connect
Implementation	Multi-company IT project. Increases approvals, complexity, and cost. Avg $100k to put a point – point connection in place. Multiply this by number of Partners	Each Member has a single integration project with TSANet Connect. This then enables that Member to collaborate with all other 900+ members
Ongoing support / Change management	Change management must be coordinated between the vendors. This increases the cost of ongoing support	TSANet manages changes (Example adding a field that they require for a new case). Members see this change the next time they make a collaboration request
Flexibility to support all Partners	Can only work with Partners that have similar IT capabilities. Most small and medium-sized companies do not have the ability to integrate systems.	TSANet members can use the TSANet Connect WebApp, Salesforce Package, MS Power App or APIs to fully integrate their systems.

System Overview

The TSANet Connect system includes a Web App and an Integration Framework based on Apache Camel on Spring Boot. The Integration Framework provides a complete set of REST APIs for creating integrations. This document will describe the TSANet Connect REST APIs and how they were used to develop the Salesforce package and Microsoft Power App.



TSANet Connect – Integration Framework

The TSANet Connect Integration Framework is based on Apache Camel on Spring Boot. Apache Camel is an open-source integration framework facilitating seamless communication between applications, systems, and services. It simplifies connecting disparate components by providing a flexible and extensible platform. In addition to the provided TSANet Connect REST APIs, the system can support custom integrations as needed.

Key features of Apache Camel:

- Routing Engine: Camel allows you to define routes that connect various endpoints (such as databases, messaging systems, APIs, and files). These routes guide the flow of data between components.
- Component-Based: Camel supports many components (e.g., HTTP, JMS, FTP, REST) out of the box. You can also create custom components tailored to your specific needs.
- Domain-Specific Language (DSL): Camel offers a concise DSL for defining integration routes. This DSL abstracts away the complexities of low-level APIs.
- Enterprise Integration Patterns (EIP): Camel implements EIPs, which provide standardized solutions for common integration challenges.

- Extensibility: You can easily extend Camel by adding custom processors, components, and data transformations.
- Testing Support: Camel provides robust testing capabilities to ensure the correctness of your integration flows.

TSANet Connect – Database Structure

TSANet Connect is designed to provide a simple and flexible method for Members to collaborate. The following objects are found in the database to support the collaboration process:

Groups: Groups are used to create relationships between Members. A Member could be part of several groups. One-Many and Many-Many groups are supported. TSANet-hosted groups and Member-hosted groups are used. Groups define the rules for collaboration (For example SLA response times)

Accounts: Accounts provide high-level information about a Member.

Departments: An account can have one or more departments. This allows you to assign separate process forms for departments (e.g., *Division A and Division B*).

Users: Users are associated with an account. A Members Identify management system can manage user access through SAML2.0.

Process Forms: An account can have one or more forms. These forms are linked to a department or group to support inbound workflow. They define what data the member receives for inbound requests.

TSANet Cases: Metadata for a collaboration request (Sender and Receiver metadata).

- **Case Notes**: Notes related to a collaboration

A facade pattern has been implemented to facilitate the translation of API consumer language to database persistence language.

TSANet Connect 2.0 REST API

TSANet provides a complete set of APIs to create apps and custom integrations. The endpoints are summarized below, and detailed documentation is found in the Swagger documentation below.

https://app.swaggerhub.com/apis/TSA_Net/Connect-Prod
https://app.swaggerhub.com/apis/TSA_Net/Connect-Beta

Postman Collection is also available at:
https://www.tsanet.org/wp-content/uploads/2024/12/Connect-2.0.postman_collection.json_.zip

Contact TSANet at membership@tsanet.org to request access to the Beta site. The Salesforce reference integration section provides an example of how these APIs are used.

TSANet Case Status and Priority.

The TSANet case status and Priority are separate from the member's case. Members can map these to their case or keep them separate.

Priority: P1 (High), P2 (Medium), P3 (Low). The system defines and monitors each SLA's initial response. If the SLA is not met, an escalation workflow triggers, alerting both the sender and receiver management contacts. Default SLAs are set in TSANet Hosted groups (P1=2hr, P2=4hr, P3=24Hr). Member can modify these defaults in their Hosted groups.

Status: The following status definitions are used.

- **Open** (New waiting initial response)
- **Information** (Sent if information is needed to accept or reject)
- **Accepted** (Collaboration Accepted)
- **Rejected** (Collaboration rejected. For example, end customer does not have support). This status is not used for solution support.
- **Closed** (set by submitter or auto-close after 30 days of inactivity)

The following list of available APIs is in OSA3.0 format:

Endpoint (POST): login
Use: Login to TSANet and get a token for further calls.

Endpoint (GET): me
Use: Returns the TSANet user profile.

Endpoint (GET): partners (search)
Use: Search for a partner or department. This searches member and department names to return a companyId and departmentId if the member has multiple departments. Tags are also supported as part of the search

Endpoint (GET): form/company
Use: Get the collaboration form for a member company without departments in TSANet. Custom fields, including tier select, are supported

Endpoint (GET): form/department
Use: Get the collaboration form for a specific department in a member's company. Custom fields, including tier select, are supported

Endpoint (POST): cases/collaboration-request
Use: Create TSANet collaboration request case by submitting the form

Endpoint (GET): cases
Use: Get cases. Allows filters (type, status,updated date, create date)

Endpoint (GET): cases/{internalCaseNumber}
Use: Get a case by the members internal case number reference. Allows filters

Endpoint (POST): cases/{caseId}/approve
Use: Approve a collaboration request

Endpoint (POST): cases/{caseId}/update/approve
Use: Update an approved request (Example change of assigned engineer)

Endpoint (POST): cases/{caseId}/request-information
Use: request additional information needed to approve a collaboration request

Endpoint (POST): cases/{caseId}/ information-response
Use: provide a response to request additional information message

Endpoint (POST): cases/{caseId}/reject
Use: Reject a collaboration request – example customer unknown

Endpoint (POST): cases/{caseId}/close
Use: Close a collaboration request

Endpoint (POST): cases/{caseId}/notes/create
Use: Post a case note for a collaboration request

Endpoint (GET): cases/{caseId}/notes
Use: Get case notes for a collaboration request. Filters available (dates)

Integration Process

Beta Environment

Elite Members can access the Beta environment to develop and test integrations. TSANet will provide the following:

1. User account on the system (Beta-Member_Name). User/password for API
2. Connection to the Test Account or other Members also in Beta to support end-end testing.
3. Link to Swagger documents
4. Postman collection
5. Access to the Salesforce Package for Members using Salesforce.
6. Access to the Microsoft Power App for Microsoft Dynamics and Teams
7. Access to a TSANet developer resource for questions and assistance with integration.

TSANet will work with members to co-develop standard integrations for reuse in the most common systems. For an example of this approach, see the Salesforce reference implementation section.

Production Environment

After validating the Member integration, TSANet will work with the Member to transition to the Production environment.

Reference Integration – Salesforce Managed Package

TSANet provides a Salesforce Package as a reference integration to show the steps to building integration and the user experience. This is a managed package, and members have complete source code visibility in GitHub. https://github.com/tsanetgit/SFDC

View the Webinar below for an overview and demo of the Salesforce integration:

https://www.tsanet.org/introduction-to-tsanet-connect-2-0-salesforce-sep-24-2024/

Salesforce Install Guide and User Guide

The Salesforce install guide provides the information needed to install and configure the Salesforce package. The Salesforce User Guide includes information on how a user uses the integration (Members can modify this user guide as needed). The install guide and user guide can be found in the readme at https://github.com/tsanetgit/SFDC

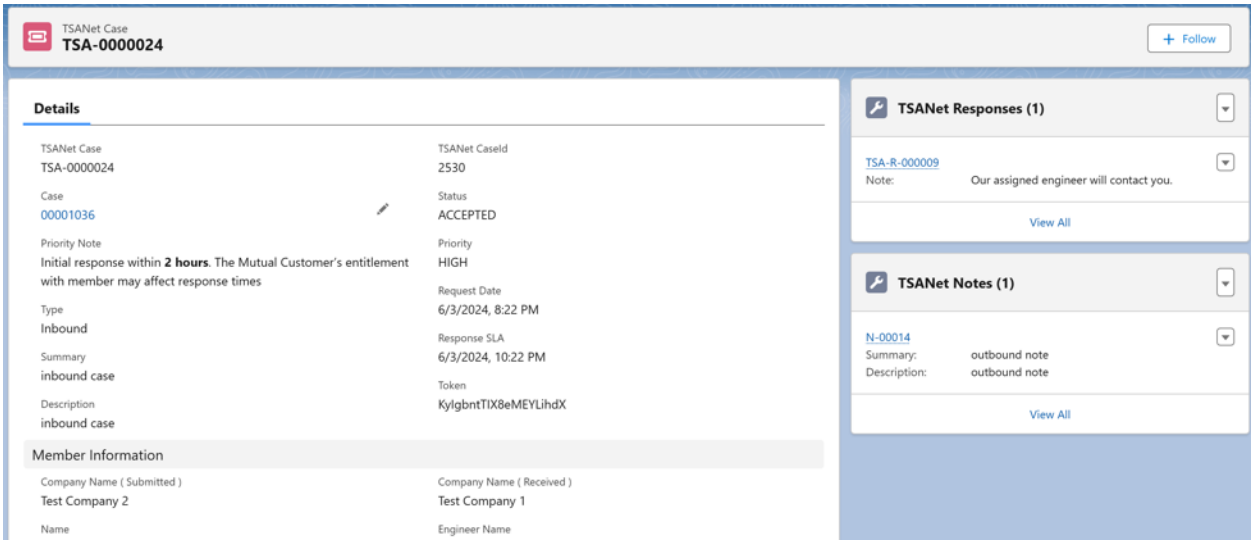
Salesforce Package Components

After installation, the TSANet Connect Package adds the following to Salesforce.

- TSAnet Case Object – with child records for response and notes
- LWC – for use on the case page
- Feed updates – Updates to the TSANet case show in the case feed
- Flows and actions – For members who would like to customize UI to best fit their case workflow

TSANet Case Object and Child Objects

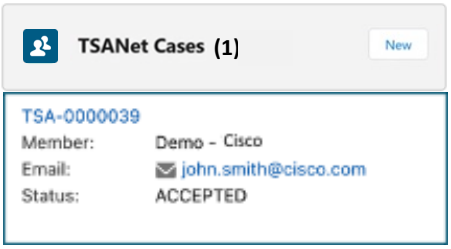
The TSANet Case Object holds data sent to and from the TSANet Connect system. Child records provide case responses and notes during the collaboration's lifecycle. Each TSANet Collaboration case is linked to a case in the member's system.



TSANet Case and Child Objects

TSANet LWC

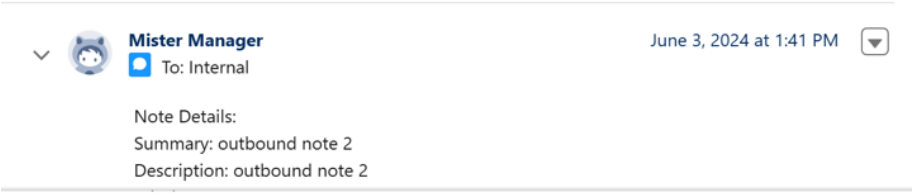
A lightning web component provides the functionality for making an outbound request. This component is added to the case or could be exposed as part of an action or flow. The user can mouse over the TSANet case to view additional details, and all updates and notes are integrated into the case feed.



TSANet Connect LWC

Case Feed Updates

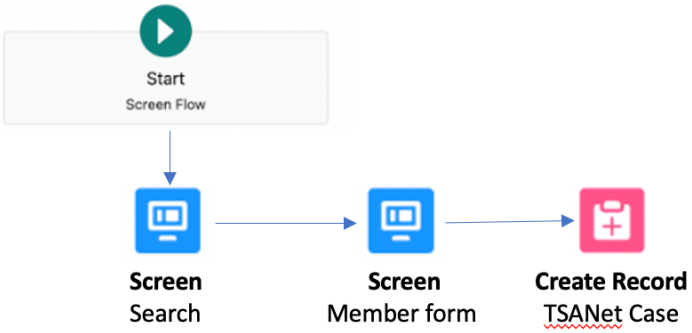
The package provides a method to update the case feed and post notes.



Feed is updated for all activity on a TSANet Case

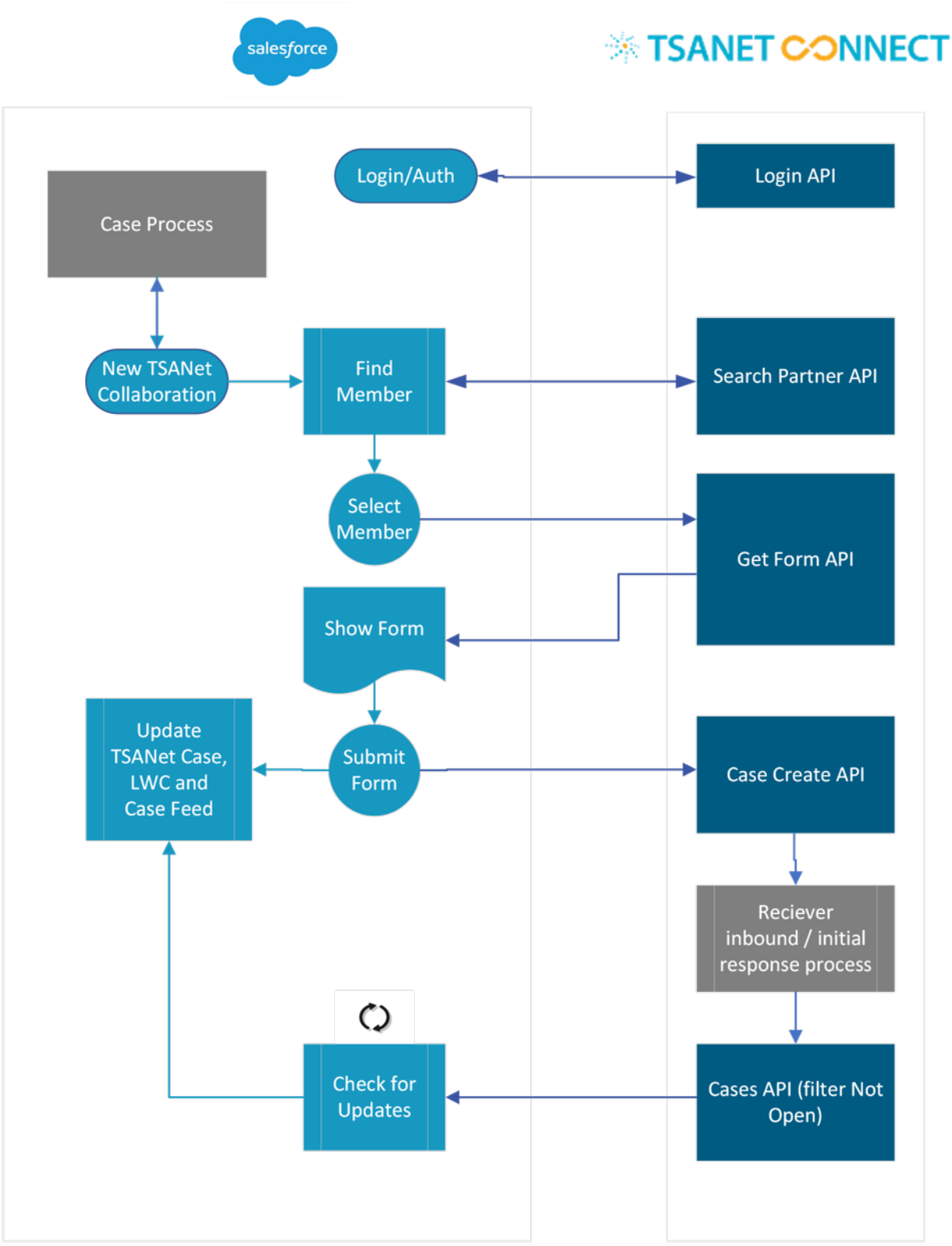
Flows

The package provides flows that members can use to customize and automate workflows.

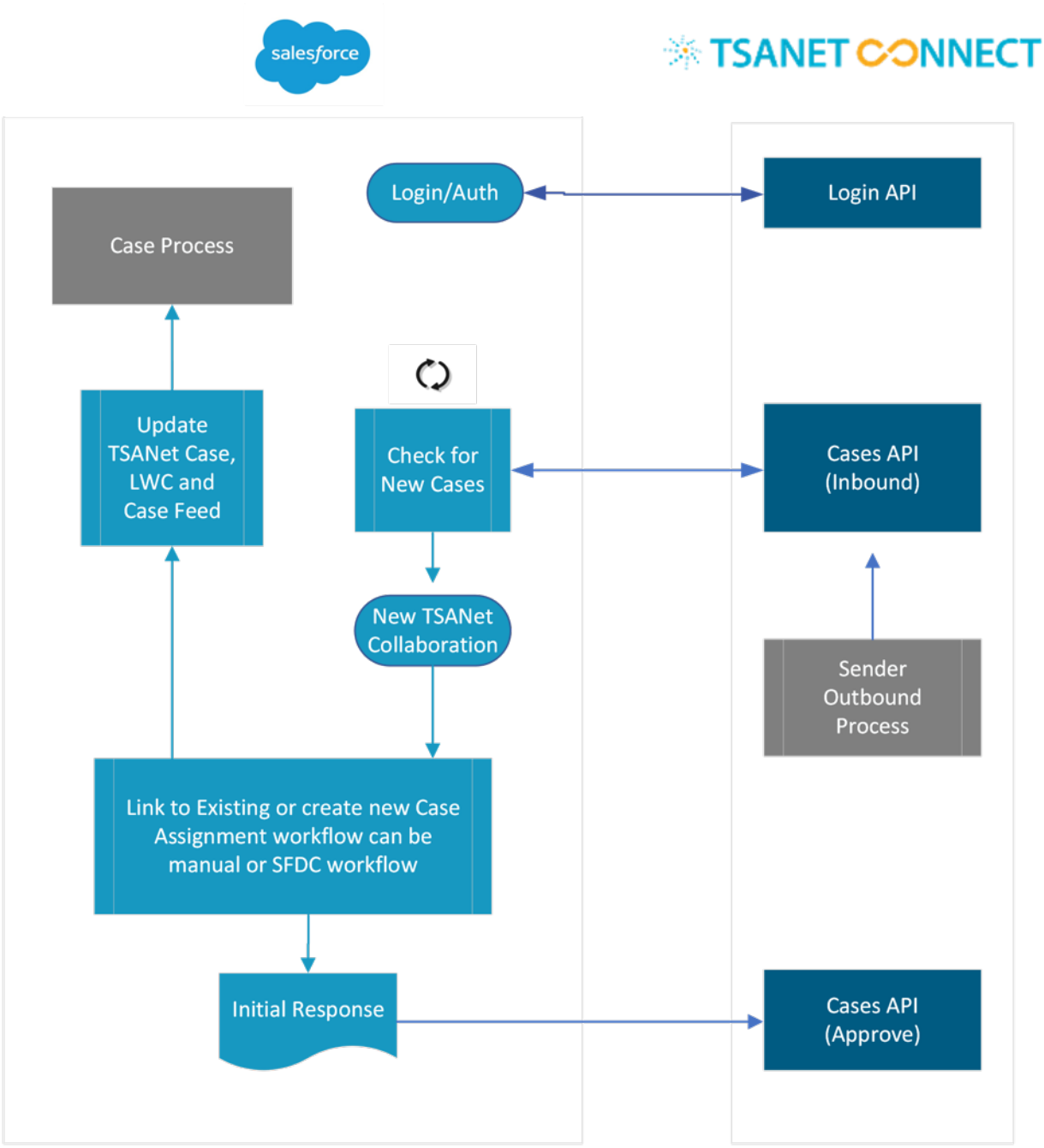


Example Flow – Create Collaboration Case

User Experience – Outbound Request (To Member)



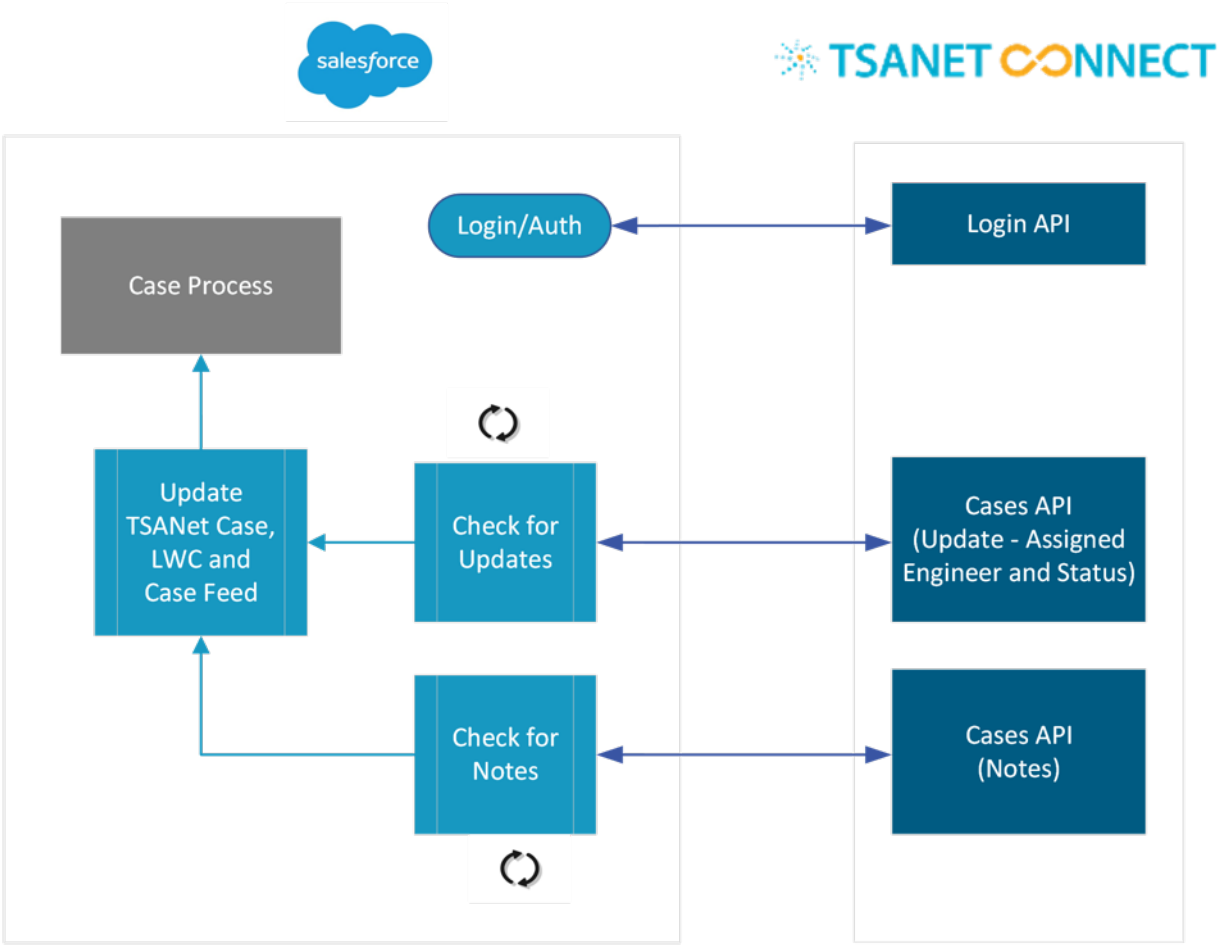
User Experience – Inbound Request (From Member)



Note: The package provides data and screens for the manual process. Members can implement workflow to automate the inbound process as needed.

Ongoing Updates – Collaboration with Connect 2.0 Members.

When collaborating with other TSANet Members, the system supports bi-directional notes in the case feed. Initial response updates are also supported.



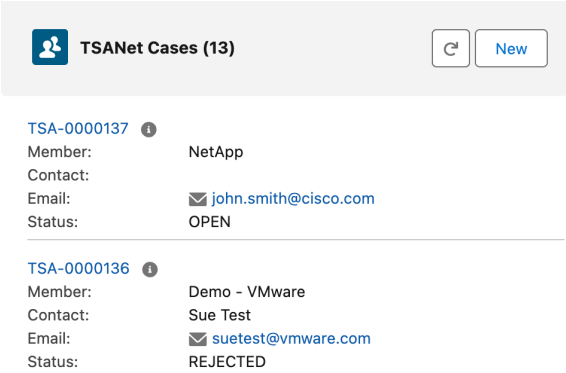
Members can control the information they send to collaborating Members. The best practice is to allow the user to define the update's visibility. TSANet suggests these options for note visibility:

- Customer + TSANet Member: updates both customer and collaborating member
- TSANet Member: Only include the TSANet member
- Internal Only: for internal notes

Response updates support updating case information, such as the assigned engineer. Salesforce Flows can also be used to message other system-generated case updates.

Multiple Collaborations on a Customer Case

The system supports collaboration with multiple TSANet members on a single customer case. Updates and notes can be directed to all members or a single member. Inbound updates from the Members are separated by Member name within the case feed.



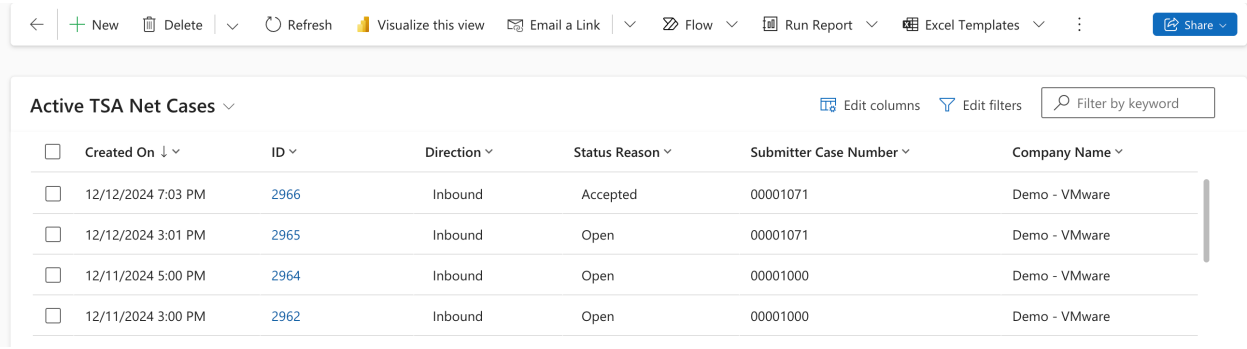
Example: VMware and NetApp in same case

Reference Integration – Microsoft Power App

TSANet has developed a Microsoft Power App for Microsoft Dynamics CRM and Microsoft Teams. This Power App has entered beta, and Members can contact TSANet at membership@tsanet.org for access.

Like the Salesforce Package, the source code can be found in GitHub.
https://github.com/tsanetgit/MS_Power_App

The Integration method and user experience are consistent with Salesforce.



Reference Integration – Custom Integration (Cisco Smart Bonding)

Custom integration can be used for Members with existing solutions or a CRM system that does not support expanding functionality with packages and apps. For example, integration with the Cisco Smart Bonding solution was accomplished by creating a custom integration. The following is a high-level view of the approach.

- Use the Cisco **create API (POST)** to create a new collaboration with Cisco (Inbound request).
- Use the TSANet **response APIs (POS**T) for Cisco to respond to a Member collaboration request.
- Use the Cisco **update API (POST)** to add notes and case updates from the other TSANet Member.
- Use the TSANet **collaboration-request API (POST)** to create a new collaboration from Cisco to Member (Outbound Request)
- Use the TSANet **API case-note (POST)** to create a new note
- Use the TSANet **API case-update (POST)** to update the collaboration (example assigned engineer change)