WEEK 2
ServiceNow Administration Fundamentals

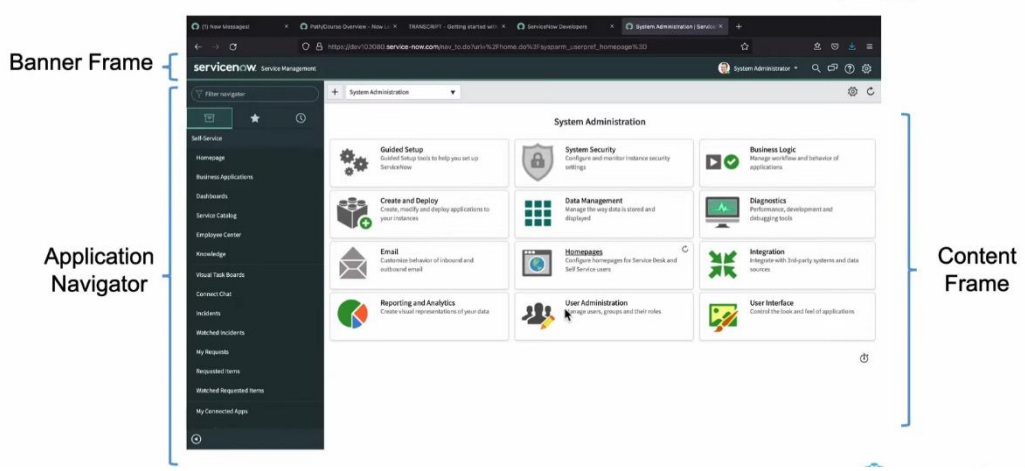
Servicenow Architecture

- The Now Platform is an Application Platform as a Service (APaaS).
- It is cloud based
- It supports and provides Infrastructure computer resources.
- It provides a platform where you can develop your own custom solutions.
- It provides a robust set of applications and workflows to support most common business processes.
- All applications (OOB and custom) for the entire enterprise are supported by a single, common, data model and database.
- 4 full backups per week and 6 days of daily differential backups are supported.
- The platform is secured via multiple technologies which have been certified by various third-party security organizations.

Servicenow Platform UI

Three primary elements that make up the Servicenow platform UI:

- Banner Frame
  Logo, User Menu, Tools, System settings
- Application Navigator
  Navigation Filter, All applications, Favorites, History
- Content Frame



User Authentication

When a user logins to an instance, Servicenow validates their identity and gives access to functions and information according to the user's assigned roles and groups.
The platform can support several methods of user authentication:

- OAuth 2.0
- Single Sign-On (SSO)
- Multi-factor authentication
- Digest Token

- LDAP
- Local database authentication

User Interfaces

Servicenow provides 3 user interfaces for interacting with the Now Platform:

The Now Platform is the primary UI. It is best suited for desktop and laptops. It can be accessed via a web browser and the instance URL.

The Servicenow Mobile Apps are best suited for mobile devices and can be installed from the app store.

- The Servicenow Agent app targets fulfilling requests.
- The Now Mobile app is built for the needs of the employees.
- The Onboarding app is built for the needs of new-hire employees.

The Service Portal is a user- friendly, self-service, widget-based portal accessed via a web browser and special URL.

Servicenow supports most of the major browsers like Chrome, Microsoft Edge, Firefox and Apple Safari.

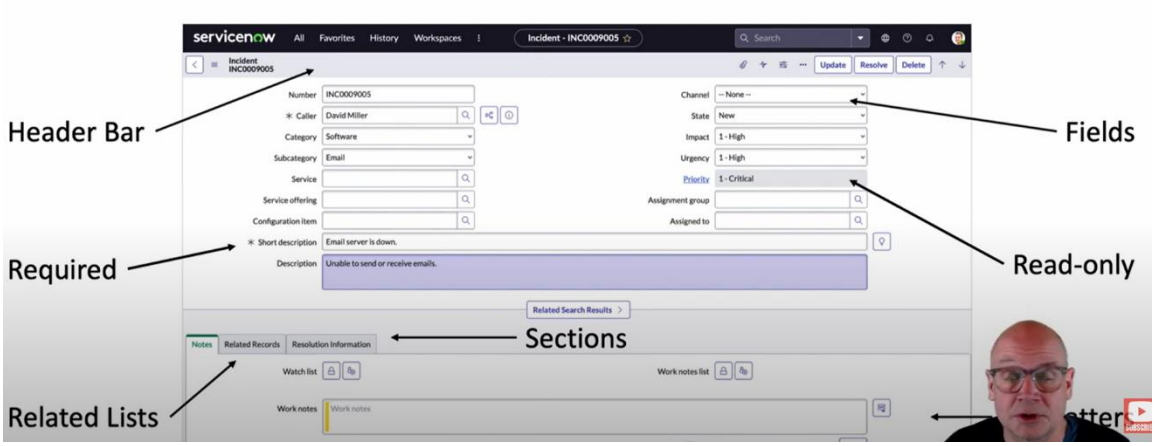
Servicenow Branding

Applying your distinct corporate identity across the Now Platform UI to create a shared identity, build trust and speed adoption.
Guided Setup:
It provides the system admin step-by-step instructions to configure applications and modules within your instance.
To access guided setup, locate the guided setup application in the application navigator and select the ITSM guided setup or ITOM guided setup module.

Service Portal and UI Builder:
Service Portal and UI Builder are two additional tools to brand the interface.
Service Portal is a widget-based tool that allows creation of intuitive, user-friendly interfaces to the Now platform.
UI Builder allows you to build out a functional page by clicking from a library of components (data and visualizations) and layouts.

Forms

A form in ServiceNow is a common set of tools and user-interface elements used to view and update a single record from the database.

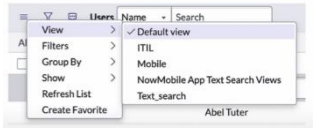


Form field types:

- String field
- Boolean(true/false) field
- List field
- Choice field
- Reference field
- Journal field

Form Related Lists

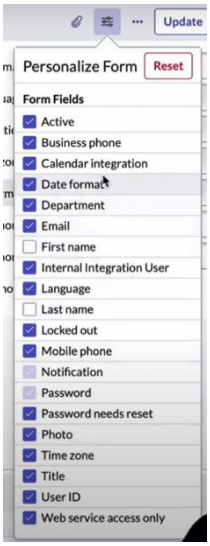
A related list is a special form element that displays a list of other records from another table related to the current record. (one to many relationship)



Form Formatters

A formatter is a special form element that displays information that is not a field in the record.

Form Views



Not every user wants to see record data in the same way. Form view provide the ability to display and organize fields, related lists and formatters in different ways to meet the needs of specific users.

Form Personalization

Form personalization allows users to select which fields are displayed on a particular form view. These changes don't impact the form view of other users.

Tasks

A task is some item of work that needs to get done. In servicenow, each task is represented by a record in a database table named Task [task].

Most commonly used types of Tasks in servicenow:
Change request. Incident, Problem.
They are extensions of the task table. They inherit the common properties of the task table and then add their own attributes as needed.

Task records in ServiceNow are only created when entries are added to those extended tables so you never actually create a task record in the task table directly.

Task Management

- Defining and managing tasks in ServiceNow allow you to take common work that needs to be done and build repeatable processes to efficiently get it done.
- Assignment Rules auto assign tasks to users or groups, making sure they are handled by the most appropriate team members.
- Approvals can be created for a list of approvers (manually or automatically) according to approval rules.
- Service Level Agreements track the amount of time a task has been open to ensure they are completed within an allotted time.
- Inactivity Monitors ensure that tasks don't fall through the cracks by notifying when tasks untouched for a predefined period.
- Workflow are processes that can be created and applied to tasks that meet certain conditions

Task Assignment
Tasks can be assigned to an individual user or a group of users (or both).
Assignment Rules
An assignment rule is a record in servicenow database that tells the platform how to automatically populate the assign to an assignment group attributes when tasks are created.
AppNav>System policy>Rules>Assignment
Table: Assignment Rule [sysrule_assignment]
Assignment Lookup Rules
These can only be created against the Incident Tasks so you don’t have the option of choosing which task table you want these to apply to. Also, they only give you the ability to choose from a small predefined set of fields to build your conditions against.
AppNav>System policy>Assignment Lookup Rules

Task Collaboration
Oftentimes the process of researching and completing tasks requires input and communication with multiple stakeholders.
User Presence allows multiple stakeholders to view and update a record simultaneously.

The Active Viewers icon will display on the header bar when another user is viewing that record at the same time.
The real-time editing (Pulse) icon will display beside a field that has been changed by another user.

Additional comments are viewable by a large audience including the customer or the person that created the task.
Work notes are the place to put notes meant for your internal working team members.

Notifications

When records are created, updated or any event is generated, servicenow can send notifications to the users configured in those notifications.

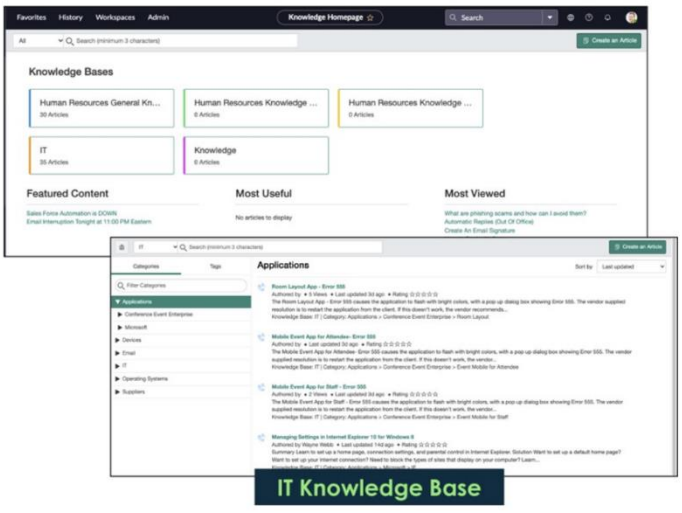
Outbound and Inbound

Servicenow sends a notification which is received by a user and this is called Outbound notification.

Users can also respond to the emails received or create new email and send it to the servicenow which can basically perform some action on the records of the table which is called inbound action.

OOB Notifications
Servicenow also has out of the box notifications configured for existing modules and applications provided by servicenow like incident notifications, change notifications, problem management notifications or request notifications.
Servicenow admins can edit existing notifications or they can create new notifications.
You can access notifications via module notifications under system notification.The table under all the notifications are stored: [sysevent_email_action].

Knowledge Management



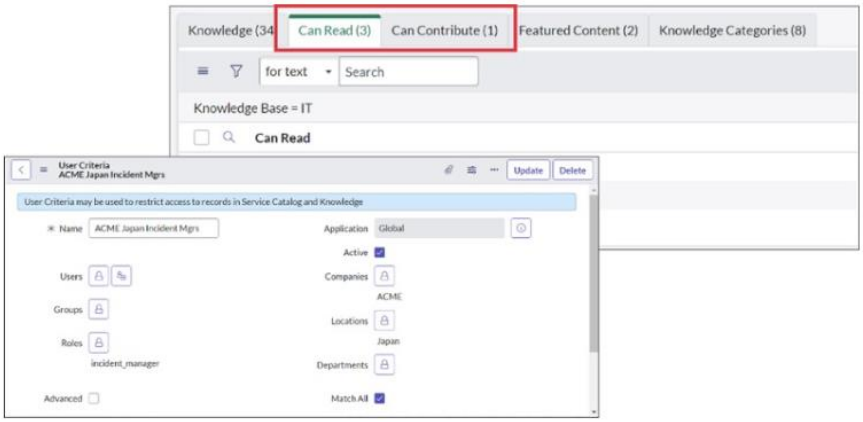
A knowledge article is a record in a knowledge base that provides information to users. A knowledge articles can be a policy, self-help tips, troubleshooting and resolution steps.
Benefits:

- One stop shop to find answers.
- Provides a centralized location for the creation, categorization, viewing of articles.
- Stores information in knowledge bases.
- Articles are referred as KB articles.

Knowledge Security and Visibility:

User Criteria defines conditions that are evaluated against users to determine which users can create, read, write and retire knowledge articles.

- You can apply several user criteria records to knowledge content.
- User criteria is applied at the knowledge base level.



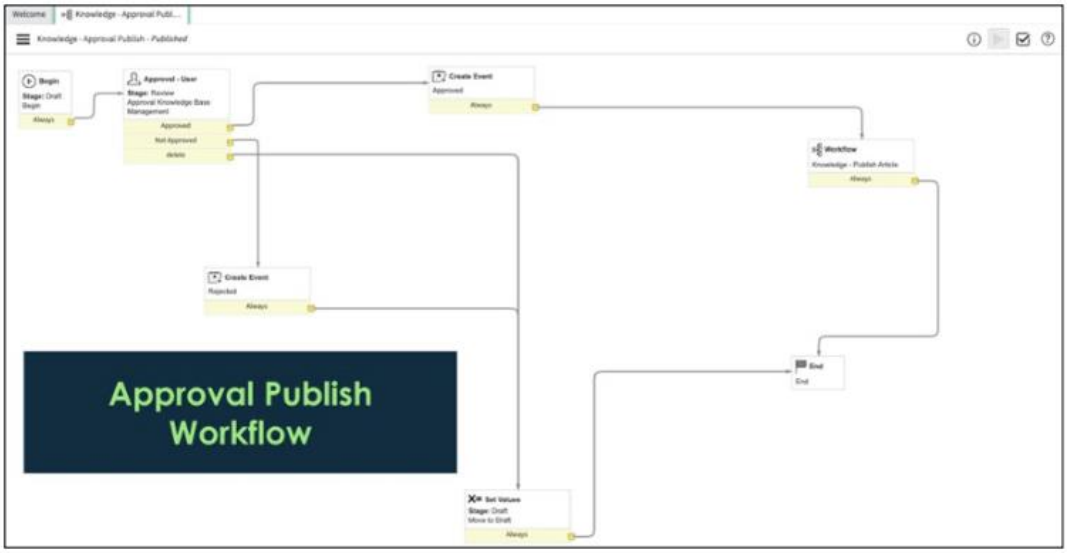
If a knowledge base has no criteria selected, articles within that knowledge base are available to all users.

User criteria outcomes include:

- canRead
- cantRead
- canContribute
- cantContribute

<u>Knowledge base: Workflows</u>

The publishing and retirement processes for a knowledge article are controlled by workflows defined for the knowledge base that article belongs to.
You can assign different workflows to each knowledge base.
You can use one of the default workflows or create your own workflow to define custom publishing and retirement processes for different types of knowledge.



The knowledge base workflows available in servicenow baseline instance include:

- Knowledge-Approval Publish: Requests approval from a manager of the knowledge base before moving the article to the published state. If the manager rejects the request, the article remains in the draft state and the workflow is cancelled.
- Knowledge-Approval Retire: Requests approval from a manager of the knowledge base before moving the article to the retired state. The workflow is cancelled, and the article remains in the published state if the manager rejects the request.
- Knowledge-Instant publish: Immediately publishes an draft article without requiring an approval.
- Knowledge-Instant Retire: Immediately retires a published article without requiring an approval.
- Knowledge-Publish Knowledge: A subflow that moves the knowledge article to the published state. You can use this subflow when defining your own workflow.
- Knowledge-Retire Knowledge: Moves a knowledge article to retired state.

To import a Word document into the knowledge base:

- Navigate to All>knowledge>articles>Import articles
- Select the knowledge base
- Select the Category*
- Add the word file
- Click Import
- Click Continue

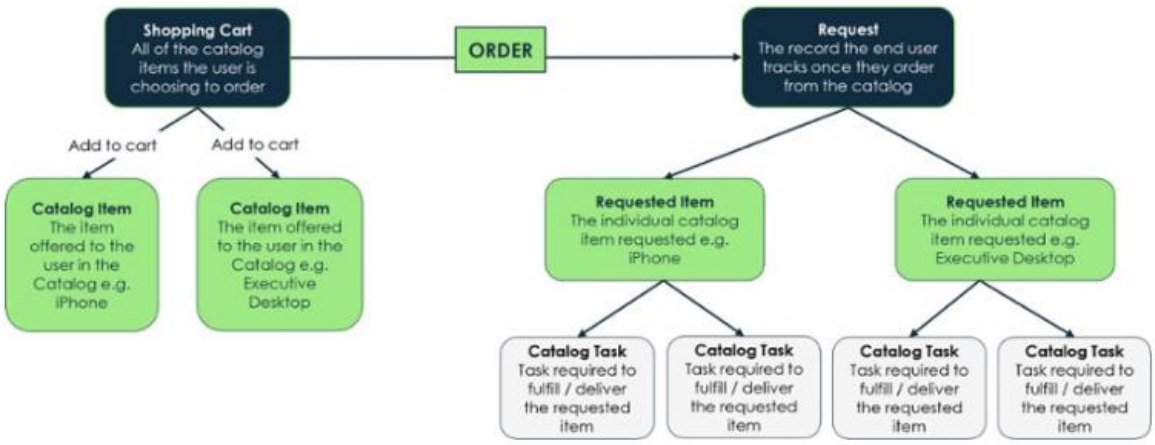
The import a word doc feature allows you to upload more than one doc at a time. When uploading multiple docs, one article is created for each uploaded item.

Service Catalog

Service catalog is a request ordering system to request services and products offered by different departments of any organization.

The service catalog lets users to see a list of things they might need(to create a request for) or would like to have- usually but not limited to, IT products and services.
Administrators and users with various catalog roles can define catalog items, including formatted descriptions, photos, and prices.
Categories define the organization for Service Catalog items. It organizes service catalog items into logical groups.
Users with admin or catalog_admin role can manage multiple Service Catalogs and provide services to different teams within the organization.

Service Catalog and Request Management:
Service Catalog: The users browse the catalog foe items they wish to order or request.
Request Management: Once the order has been placed the request needs to be tracked and fulfilled.



After placing an order, the request management process begins. For each catalog item, workflows facilitate the approval process as well as the fulfilment tasks.

Service Catalog major components

Items are the building blocks of the service catalog including hardware, software and services. Items once selected and submitted result in requests or other records like incidents. To create a new item or modify an existing item, navigate to All>Service Catalog>Catalog Definitions>Maintain Items

Record Producers appear as simplified forms, allowing users to provide information that is translated into task-related records being added or modified in the database. It is an interface used as an alternative to lists and forms. Each Record Producer focuses on a specific process or task and can be used anywhere in the servicenow platform. In the Service Catalog, Record Producers are presented in categories along with catalog items. Service Catalog can be used as a complete front-end UI.

Variables are global by default and provide options to tailor a catalog item to the customer's needs. It defines the questions to ask the end user ordering the catalog item. Question choices can define the available options and might affect the order price.

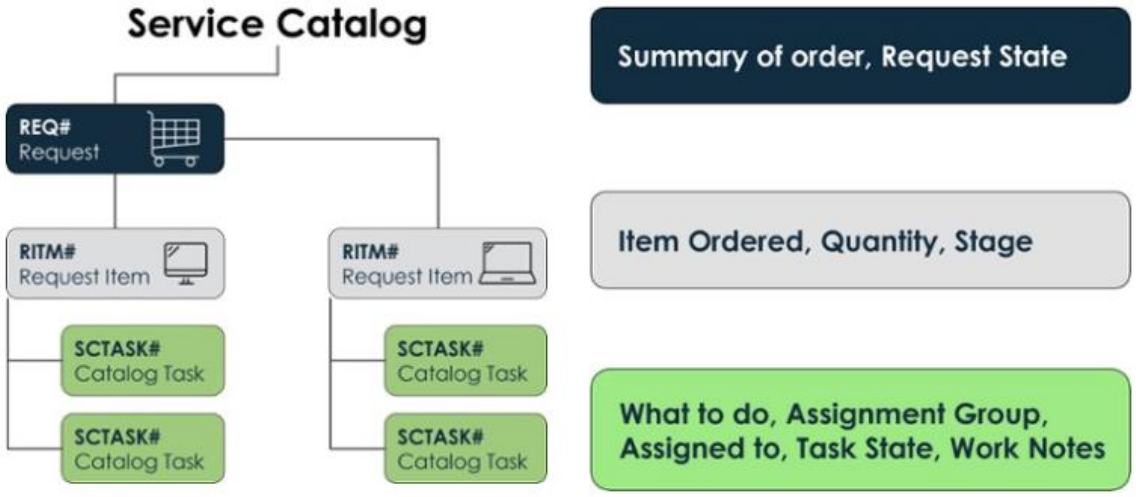
Examples: Which monitor size?, Who is the hiring manager?, What is the budget code?

The service catalog variables help define the structure of a catalog item form that is displayed to the customer.

Variable set is just a container, so it has only two fields: Name and Description. Navigate to All>Service Catalog>Catalog Variables>Variable Sets to create a new variable set.

Order Guides provide the ability to order multiple, related items as one request. Variables are represented by the Order field number.. Use an order Guide to assist users in determining what items the need.

Service Catalog Item Request Output



For catalog items, a request is created. A request can have one or more items associated with it. An item can have one or more tasks associated with it. Each output is stored in the appropriate corresponding table.

- REQ# Requested Item [sc_request] table: A request number generated to keep track of an order. Records on this table begin with REQ and behave like containers.
- RITM# Requested Item [sc_req_item] table: Records on this table begin with RITM and manage the delivery of each individual item in the request. Each discrete item ordered is given a specific “Requested Item Number” known as RITM (number).
- SCTASK# Catalog Task [sc_task] table: Records in this table begin with SCTASK and are the assigned tasks needed to complete the delivery of each request item from start to finish.

Progress Stages for a Requested Item:

While viewing the requested item, you can expand the flow stages which provide summary-level feedback about the progress or state of an item in the delivery process.

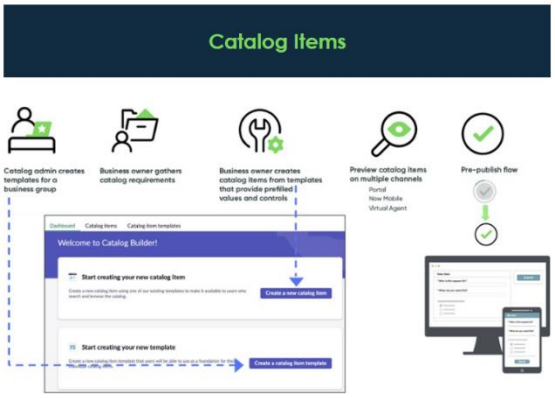
After a request has been submitted, users are able to easily track it by navigating All>Self-Service>My Requests and opening the record associated with the request.

The Flow stages attached to an item indicate the progress or state of an item in the delivery process with one of the following stages:

- Waiting for approval(In progress)
- Approved
- Pending(has not started)
- Fulfilment(In progress)
- Deployment/Delivery
- Completed

Catalog Builder

Create or edit a catalog item or record producer using a visual and guided experience along with specified instructions. The catalog builder experience enables you to delegate the creation and maintenance of the catalog.



Users can also create a template that can be used to create catalog items. While creating template you can specify values or restrictions for items create using the template.

From the homepage of the catalog builder, you can do the following:

- Create a catalog item
- Create a catalog template
- View the available catalog items
- View the available catalog item templates
- View the catalog items that are recently updated
- View the configured content that describes the catalog building process in your organization.

Flow Designer

Flow Designer is a non-technical interface for building and enabling process automation capabilities, known as flows.

Navigate to All>Process Automation>Flow Designer

Flows automate business logic for a particular application or process such as approvals, tasks, notifications, and record operations.

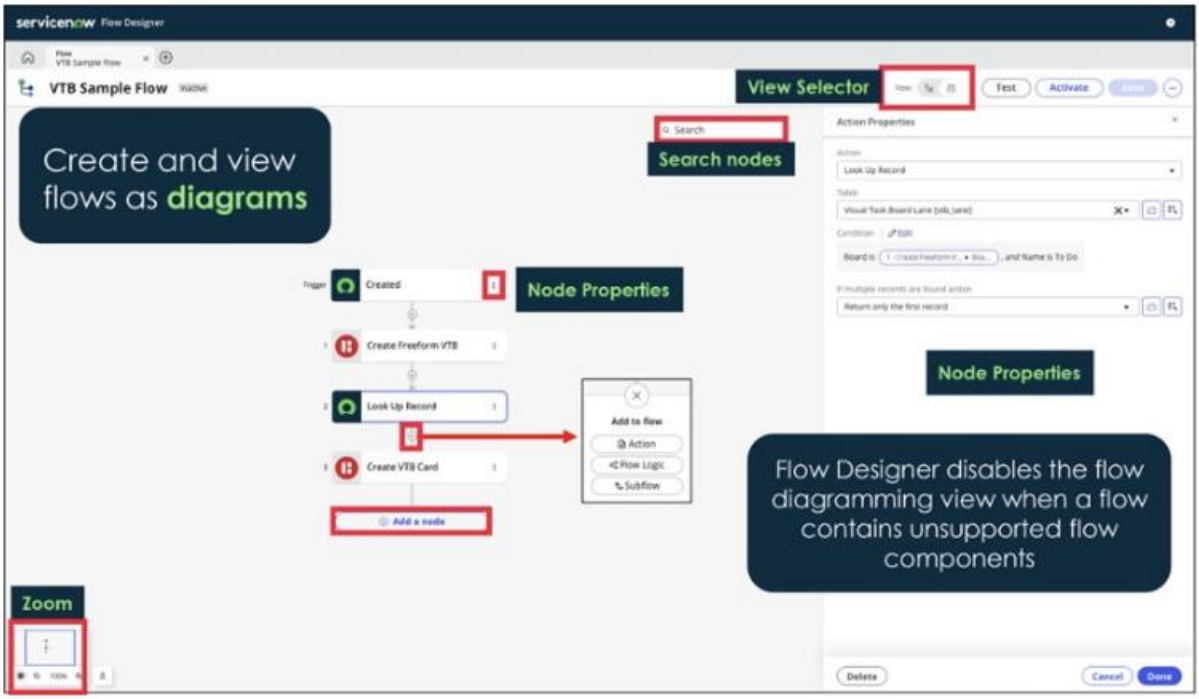
The following roles might be required to work with the Flow Designer in one capacity or another: flow_designer, flow_operator and action_designer.

Benefits

- Single environment to build and visualize business processes
- Configuration and runtime information available to create, operate, and troubleshoot flows from a single interface
- Provides natural-language-descriptions of flow logic
- Promotes process automation by enabling subject matter experts to develop and share reusable actions
- Allows extending Flow Designer content by subscribing to Integration Hub or installing spokes
- Create a flow with an SLA Task Trigger

In situations like service catalog, where you may have hundreds of items with many associated flows, moving forward, Flow Designer flows are a preferred approach.

Flow Diagramming View



The diagramming view offers features for working with your flow in a visual diagram.

Supported flow components

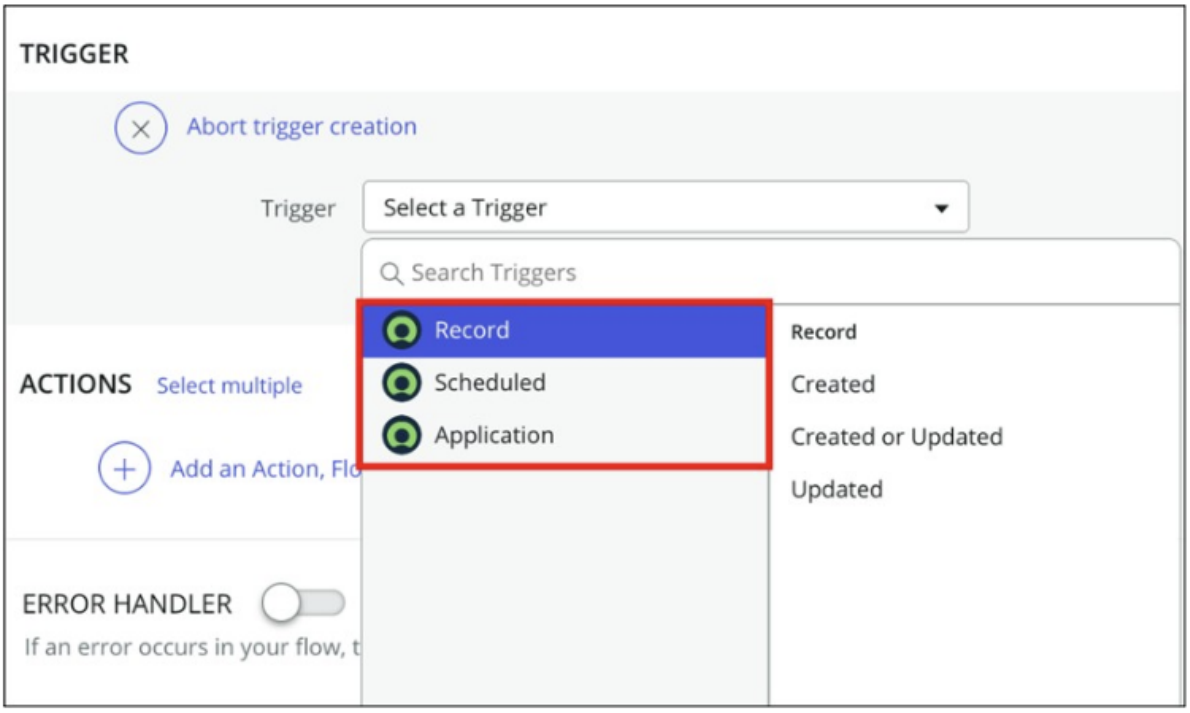
The flow diagramming view only displays flows with these trigger types.

- Record triggers
- Date triggers
- Inbound email
- Service Catalog
- SLA Task

The flow diagramming view only displays flows containing these flow logic types.

- Call a workflow
- Do the following in parallel
- Dynamic Flows
- Else If
- End Flow
- For Each
- Get Flow Outputs
- If
- Set Flow Variables
- Wait for a duration of time

Flow Components: Triggers

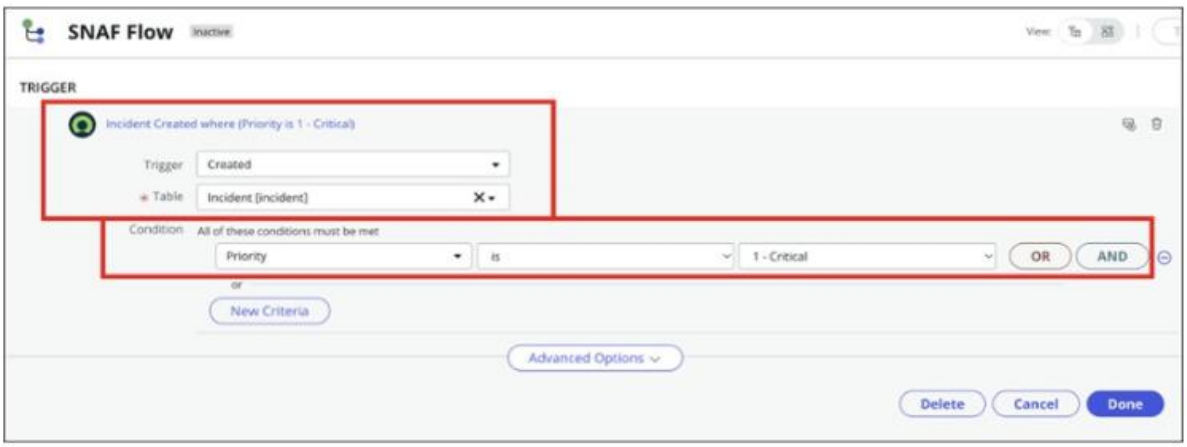


Triggers can be record-based, date-based, or application-based.

Triggers instantiate the flow and can be record-based, date-based, or application-based.

- Record-based triggers run a flow after a record has been created, updated, or deleted. When using a record-based trigger, the triggering record can be used later in the flow as input for actions.
- Date-based triggers run a flow at the specified date and time: daily, weekly, monthly, etc. The execution time can be used as an input for actions in the flow.
- Application-based triggers are added when the associated application spoke(contains triggers and actions dedicated to a particular application eg. ITSM Spoke contains actions for managing Task records such as the Create Task action. Spokes are activated when their parent application is activated.) is activated. In some instances, a plug-in might need to be activated as well (All>System Definition>Plugins)

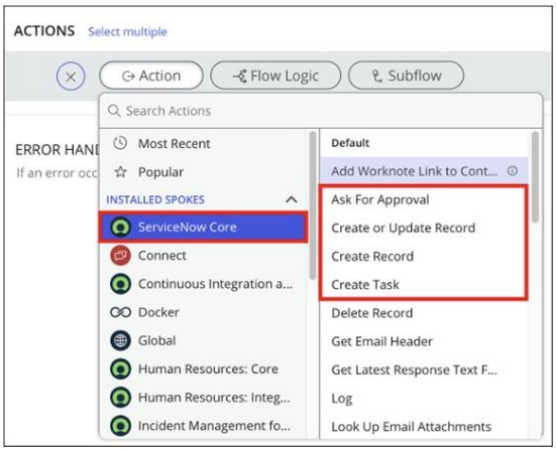
<u>Flow components: Triggers and conditions</u>



A flow can include triggers and conditions.

Trigger: An activity that initiates the flow, such as record created in a specified table or a scheduled job.

Conditions: Statements that determine when or how an action runs. For example, run an action only if a field is over a certain value.



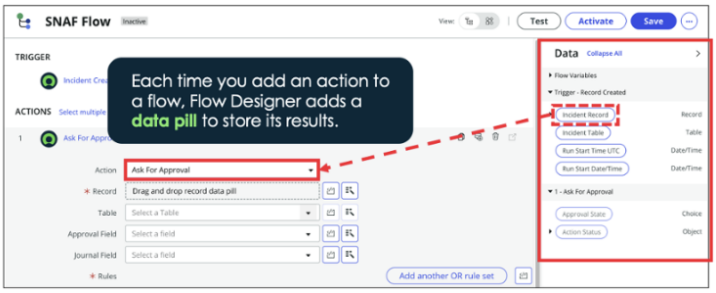
Flow Components: Actions

Actions are operations executed by the system, such as looking up a record, updating a field value, requesting an approval, or logging a value.

In flow designer, under actions you can view the most recent and most popular actions that have been used in the last 7 days for all Flow Designer users.

Some servicenow core actions include:

- Ask for approval
- Create Record
- Delete Record
- Look up record
- Wait for condition



Flow Components: Data

The **Data** section of the flow designer contains the data pills that can be used in subsequent actions. To reference the data stored in the data pill, drag and drop the data pill from the Data section to the appropriate field in the flow or click on the **Data Pill Picker** icon.

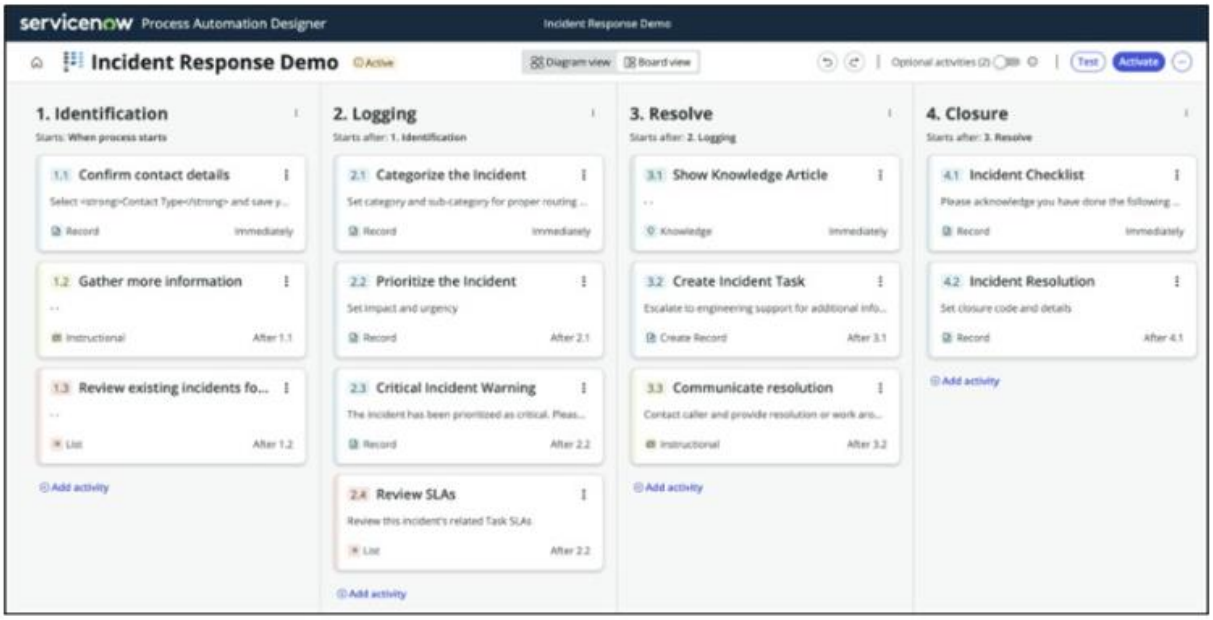
Integration Hub

Integration Hub offers several pre-built sets of integration actions to interact with common third-party applications.
These sets of integration actions are referred to as spokes.
Automate integration tasks using servicenow compoments for Flow Designer or develop custom integrations.

Process Automation Designer

Process Automation Designer enables owners to author cross-enterprise workflows and create a single unified process.

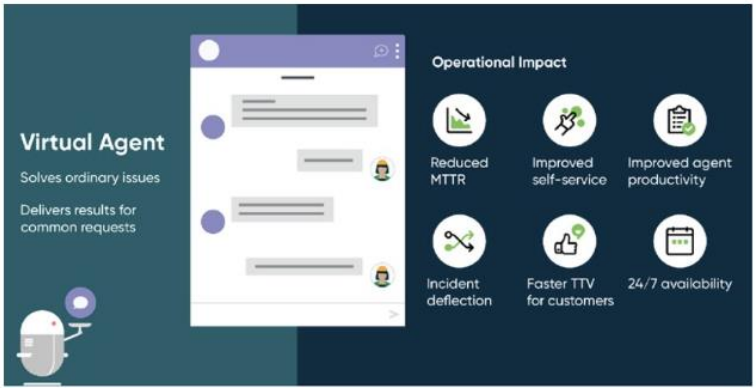
It can also be used to provide end users with a simplified, task-oriented view of your process.



Using PAD, citizen developers and process owners can automate existing servicenow platform capabilities (send notifications, create new records, submit approvals, etc.) PAD is based on the technology of Flow Designer and can be used to create flows, actions and manage flow executions.

<u>Virtual Agent</u>

Virtual Agent is a conversational platform that helps users obtain information, make decisions, and perform common work tasks within a messaging interface.



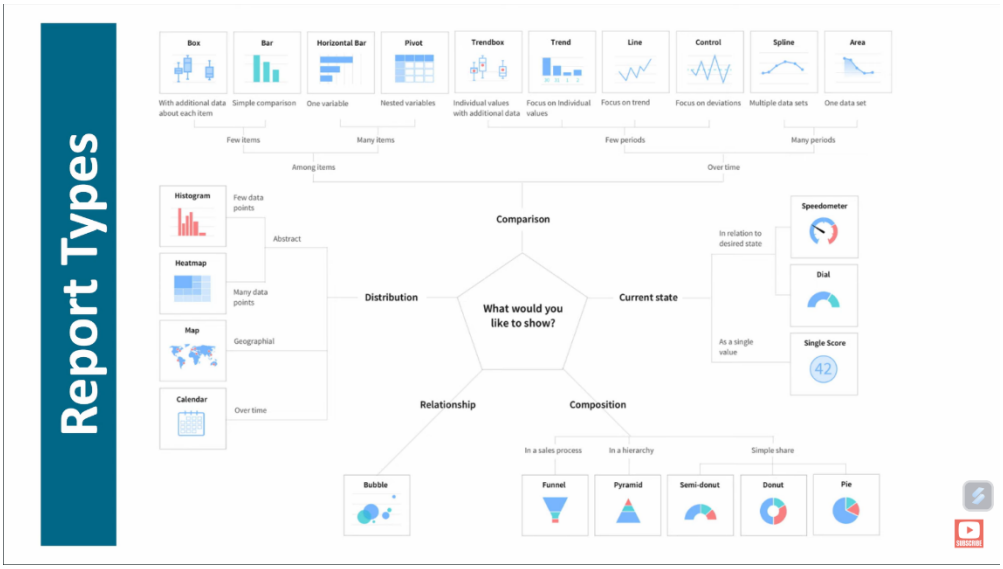
Through live agent support, users have the option to switch to a human agent for assistance at any time, ensuring they receive the help they need.
It's an intelligent conversational experience for your users that gives them answers to common questions and helps them resolve routine issues faster. It uses natural-language understanding-artificial intelligence- to figure out what people are saying.

Reporting

Reports, prepared on an ad-hoc basis, show results by allowing users to view and analyze servicenow data.
Run predefined reports or create new custom reports with the report designer.
Reports can be visually represented in many ways, including bar graphs, pie charts, dials, lists, pivot tables, donuts and more.

Report types

The servicenow base instance comes with over 25 standard report types



The All>Reports>View/Run module contains a library of reports which you can run and use to create your own custom reports. Many of these reports came with the platform and others were created by your reporting administrators specifically for your company.

Report Designer

Using the built-in Report Designer, anyone can easily create reports by following guided flows to configure, preview, edit and share reports.

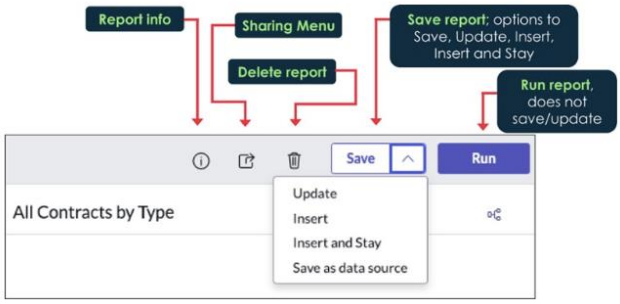
You can also:

- **Schedule** reports, so they are run and distributed on a regular basis.
- **Integrate** reports into intuitive dashboards with interactive filters and widgets that let you visualize data in the most effective way.

Each section of the report designer provides different configuration options:

- **Data:** Provide a name for the report, as well as select the source from where your data comes from. You can choose a data source, which is a predefined data set used for creating reports
- **Type:** Select the visualization of your report by choosing a report type. There are 28 different types to choose from
- **Style:** Adjust the look of your report, from coloring to titles, as well as making adjustments to the report legend.

Report Actions and Options



Report actions become available once the report has been saved and they depend on your role.

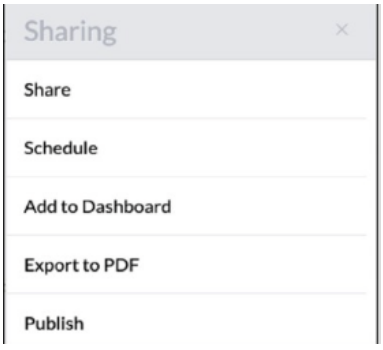
- **Update**: Overwrite report, return to the report list
- **Insert**: Save a duplicate copy of the report, return to the report list
- **Insert and Stay**: Save a duplicate copy of the report, remaining on the report
- **Save as report source**: Allows you to create a prefiltered data set that can be used for creating reports (role reqd: report_admin)

Report Distribution

To make a report visible to a particular group or user, use the **Share** option to select Groups and/or Users.

Steps to publish a report:

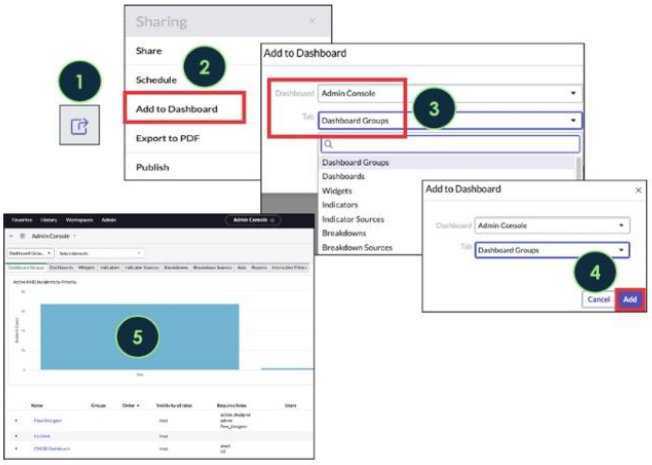
1. With desired report displayed click the **Sharing** menu icon, then click **Publish**.
2. Click the **Copy report link** icon from the report header to copy the URL to your clipboard.
3. **Open URL** in browser.



When distributing a report, sharing makes the report visible to authenticated users within servicenow.
From the **Sharing** menu, the following options are available:

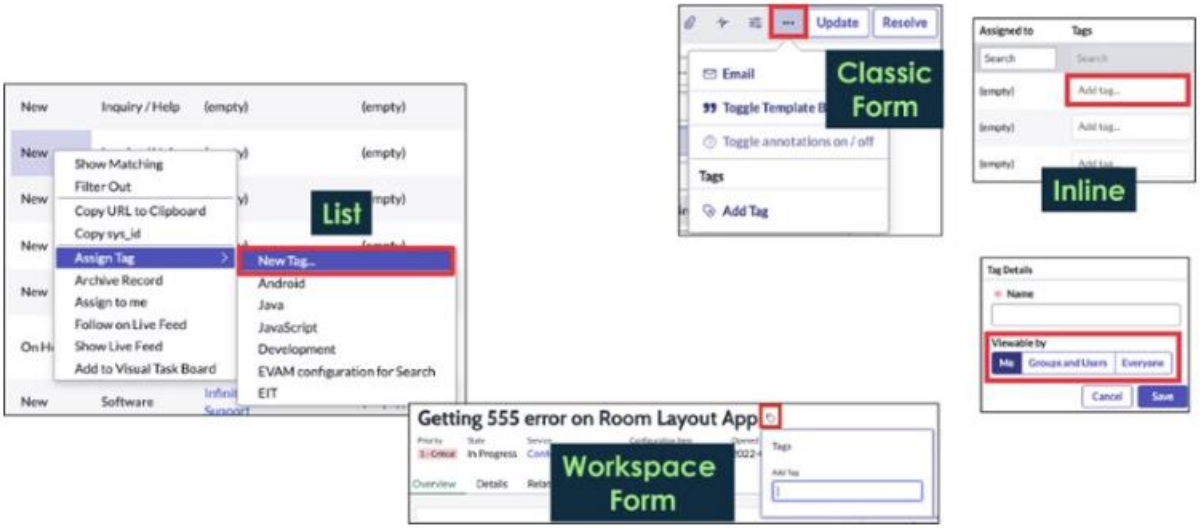
- **Share**: Specifies who can see the report. Options include:
    - Me (the user who created the report)
    - Everyone (all servicenow users)
    - Groups and Users (specific groups and/or specific users)
- **Schedule**: Creates a scheduled email of the report as an attachment
- **Add to Dashboard**: Adds directly to a Dashboard you choose. If the dashboard has multiple tabs, you can select a specific tab.
- **Export to PDF**: Converts the report to a PDF which can be generated immediately or sent as an email.
- **Publish**: Creates a URL for the report which can be used by internal and external audiences. However, users may need to log into servicenow to view all of the data. Published reports can be unpublished and no longer accessible.

<u>Add Report to a Dashboard</u>



<u>Tags – Personalize or Configure</u>

Tags provide an easy way to categorize, flag, and locate records. Tags can be created against any record from a list or form view.



Use the **viewable by** field to control how it is shared.

New tags can be made visible to:

- The current logged in user (Me)
- Groups and Users
- Everyone

In the workspace view, the tag icon appears next to the primary value in the form header. An unfilled tag icon means that no tags have been assigned to the record. When tags are assigned and they are visible to you, the tag icon is filled.

<u>Reporting Compared to Performance Analytics</u>

When you **report** on a table (for example, Incident or Problem), information about the **current state of platform data** displays.

Performance Analytics provides information about performance **iteratively, over time.**

Performance Analytics (PA) allows users to create dashboards with widgets to visualize data over time in order to identify areas of improvement.

The performance analytics architecture:

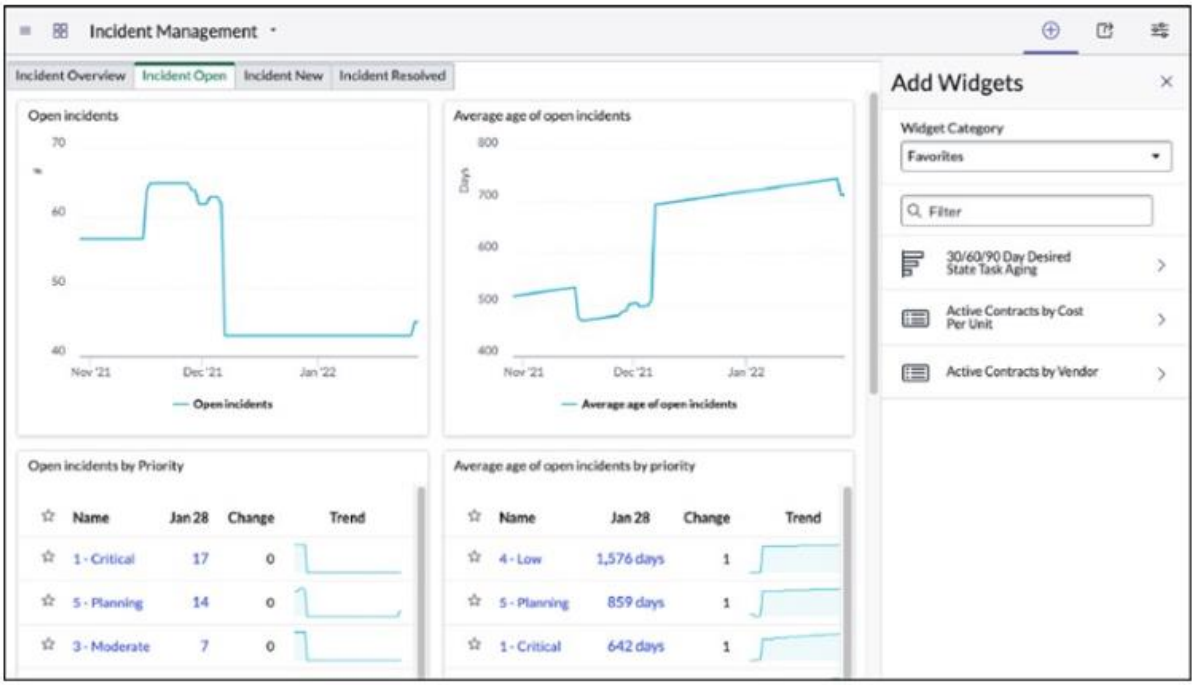
- Widget-Saved view of indicator or breakdown
- Tables: Indicator Source-calculates scores
- Data Collector-recurring jobs taking data snapshots
- Dashboard-Custom arrangement of widgets

Reporting answers the question of “Where are we today?” while Performance Analytics answers what is happening over time.

<u>Dashboards</u>

Dashboards enable you to display multiple Performance Analytics, reports, and other widgets on a single screen.

Use dashboards to **create a story with data** that can be shared.



Dashboards may be responsive or nonresponsive. Responsive dashboard functionality is enabled by default. Non-responsive dashboards have limitations including who can create, view, and edit them.
If your instance requires dashboards by navigating to All>System Properties>Dashboard Properties and clearing the Enable responsive dashboard.
With dashboards you can:

- Share Performance Analytics and Reporting visualizations on both Workspaces and classic dashboards.
- Create and edit Performance Analytics reports and other widgets directly from the dashboard.
- Use the Add Widgets pane to quickly find and preview widgets, then add them to the dashboard.
- Easily share dashboards with other users from the integrated sharing pane.
- Use quick layouts to snap widgets into a predefined layout, then adjust the layout as desired.
- Set dashboards as your homepage so you can quickly access information that you use frequently.

Notifications

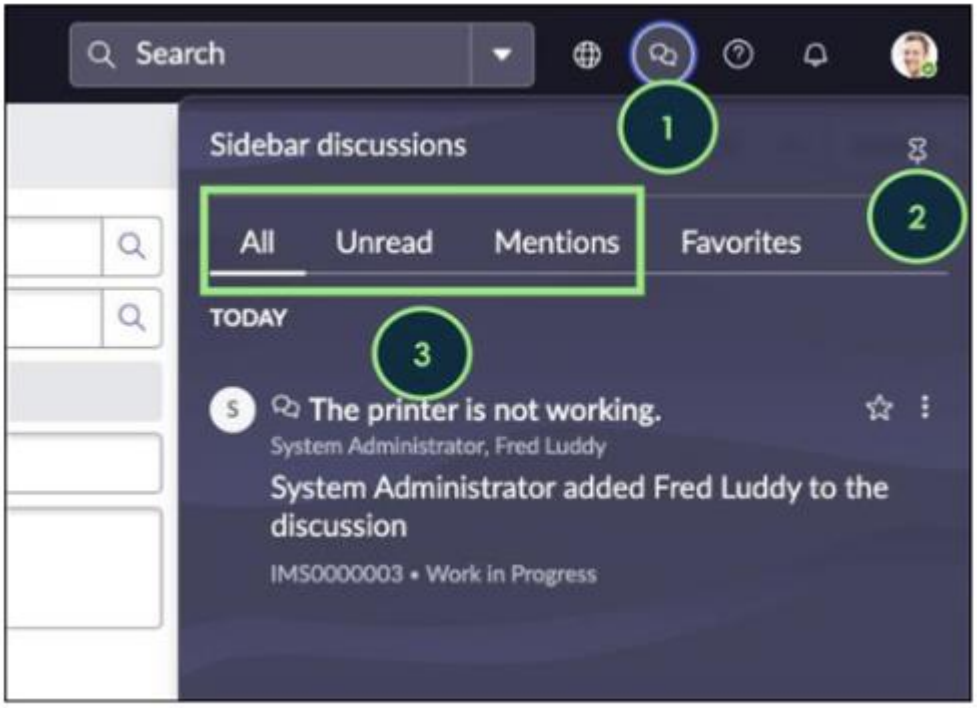
Notifications can be triggered by events in the platform and require no scripting knowledge. Use notifications to notify users about activities in servicenow (i.e. updates to incidents or change requests).A notification is a tool for alerting users when events that concern them have occurred through the following methods:

- Email
- SMS
- Meeting Invitation

Email notifications are used to send selected users email or SMS notifications about specific activities in the system, such as updates to incidents or change requests. To access a new notification record navigate to All>System Notification>Email>Notifications.

Using Sidebar in Supported Workspaces

Using **Sidebar**, agents can have real-time collaboration with other based around a Workspace task-based or interaction-based record.



1. Sidebar discussions: Display Sidebar discussions dialog
2. Pin: You can permanently pin the menu to the side of the screen
3. All, Unread, Favorites: Select whether you want to view all discussions, unread discussions, or only ones marked as favorites.

Sidebar is supported in these workspaces:

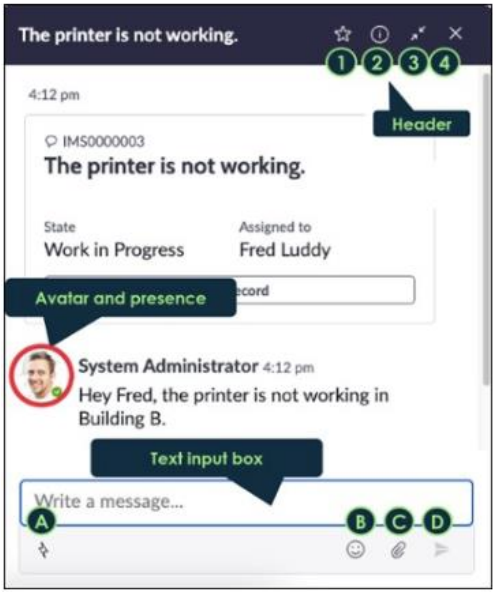
- CSM Configurable Workspace
- CSM Manager Workspace
- HR Agent Workspace
- ITSM Manager Workspace
- Vendor Management Workspace

Docked Chat Windows (Fulfillers)

Fulfillers can access multiple Sidebar discussions at the same time using the **docked windows** feature.

When a sidebar discussion is created, the record card is automatically added at the start of the chat window.

Docked Chat Windows include these features:



1. **Favorites**: Marks the sidebar discussion as a favorite. Discussions that have been marked as favorites display under the Favorites tab
2. **Information**: Opens the discussion info panel that includes the About submenu, Participants submenu and the leave discussion button.
3. **Collapse**: Collapses the discussion so it displays only as a tab at the bottom of the screen.
4. **Close**: Closes the window.

User Avatar and Presence: Displays the user's avatar and presence (online status). The dot indicates the user's presence depending on whether they are logged in to the platform

- Green dot- user is logged in
- No dot- user is not logged in
- Orange dot- user recently logged out

Activity Stream in Sidebar

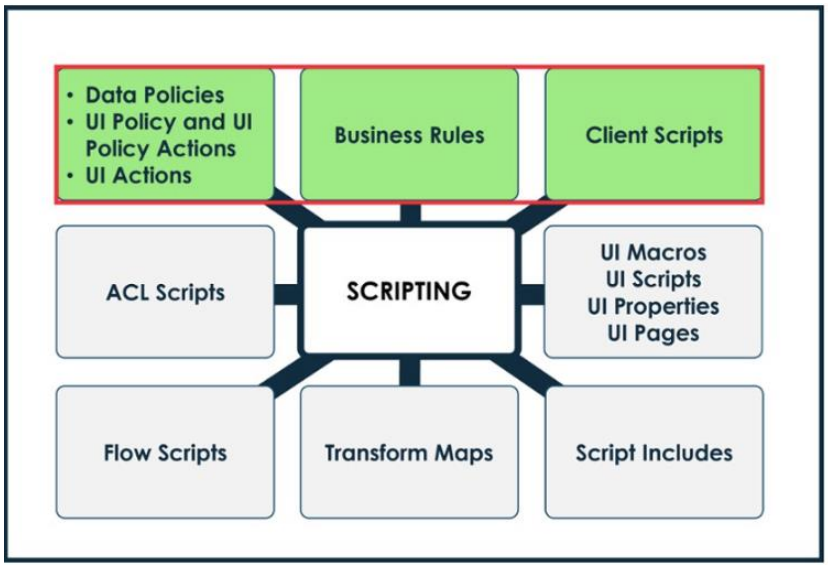
Sidebar is integrated with **activity stream**. When you start a Sidebar discussion, a tile corresponding to the discussion is automatically added to the activity stream. Sidebar and discussion tiles are only available with Next Experience.

Customers and non-fulfiller roles do not have access to the record nor the conversation tile, similar to work notes.

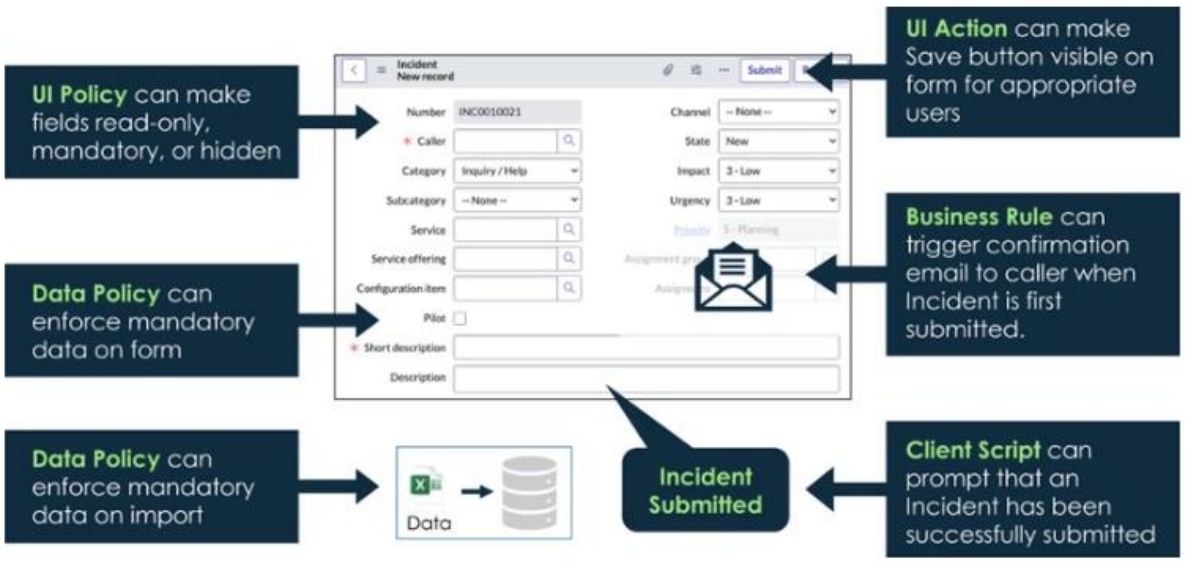
UI Policies and Business Rules

Scripting Areas in ServiceNow

ServiceNow has over 30 places where code can be inserted to change the behavior of the platform.



Sample Policies, Actions, Roles, and Scripts



Business Rules: Can do much more than trigger notifications. For example, when changes occur on the record or a field, other actions may be triggered. A baseline example might be a business rule responsible for updating child incidents if something happens on the parent incident.
Client Scripts: Can do more than show-up messages. Many Client Scripts change values in one field in response to another field being updated. For example: If an assignment group value is updated, the client script can clear the "Assigned to" field if the Assigned to is not a member of the new assignment group.
Data Policy: Can not only enforce mandatory fields but also read only fields. Therefore, they can ensure data is not touched on an import or by a script/user. Plus, they work on lists.

What is Scripting?
Scripting in ServiceNow or Platform Scripting is the customization of an instance and/or applications using Javascript.
Javascript may execute on the **client side**(web browser) or the **server side**(ServiceNow database) and can fundamentally alter the baseline instance and user experience.

Client refers to an application or system that accesses a remote service or another computer system.
A **server** is the computer program running one or more service; a physical computer dedicated to running one or more services, or a system running a database.

Client-to-server round-trips take time and make the end-user wait for the round-trip to complete. **Request + Response = Round Trip**

UI Policy and UI Policy Actions

A **User Interface (UI) Policy** is a rule that is applied to a form to dynamically change information or the form itself. **UI Policies execute on the client side.**

Use a UI Policy to set fields on a form to:

- Mandatory or Optional
- Hidden or Visible
- Read-only or Editable

To apply a UI Policy to all views, set the **Global** setting to **true.**

Once a UI Policy is saved, **UI Policy Actions** determine what happens on the form, including:

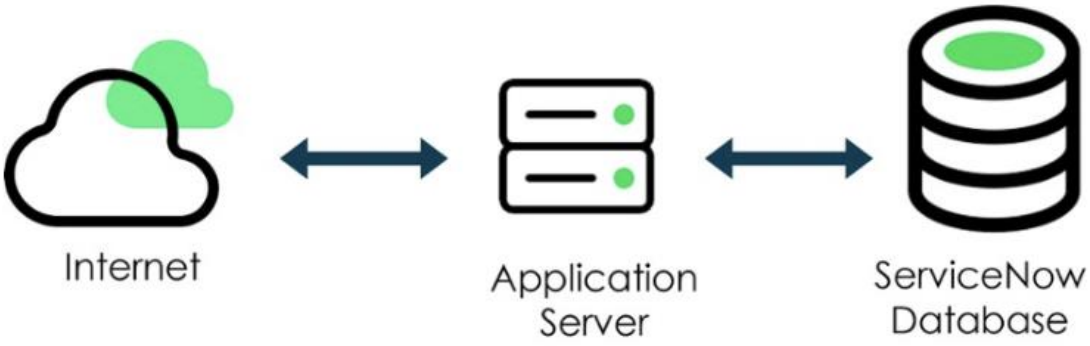
- Setting a field as **mandatory** – requiring a value in order to save the record
- Setting a field as **hidden** – no longer displaying a field on the form
- Setting a field as **read-only** – preventing a user from updating its value

UI Policies are not about security, they are about managing the **user experience** and **data integrity**.

To immediately implement updates and changes to forms and lists, you can use UI Policies which allow you to add sophisticated controls without having to write scripts and define custom process flows or tasks.

Data Policies

A **Data Policy** is a rule that enforces data consistency by setting fields as mandatory and/or read-only.



Data Policies are similar to UI Policies, but **UI Policies only apply to data entered on a form through a standard browser**. Since UI Policies can also manage the visibility of fields on a form, you want to augment UI Policies with Data Policies.

Data Policy controls are similar to UI Policies, but UI Policies are only enforced on data entered into a form (passing through the UI).

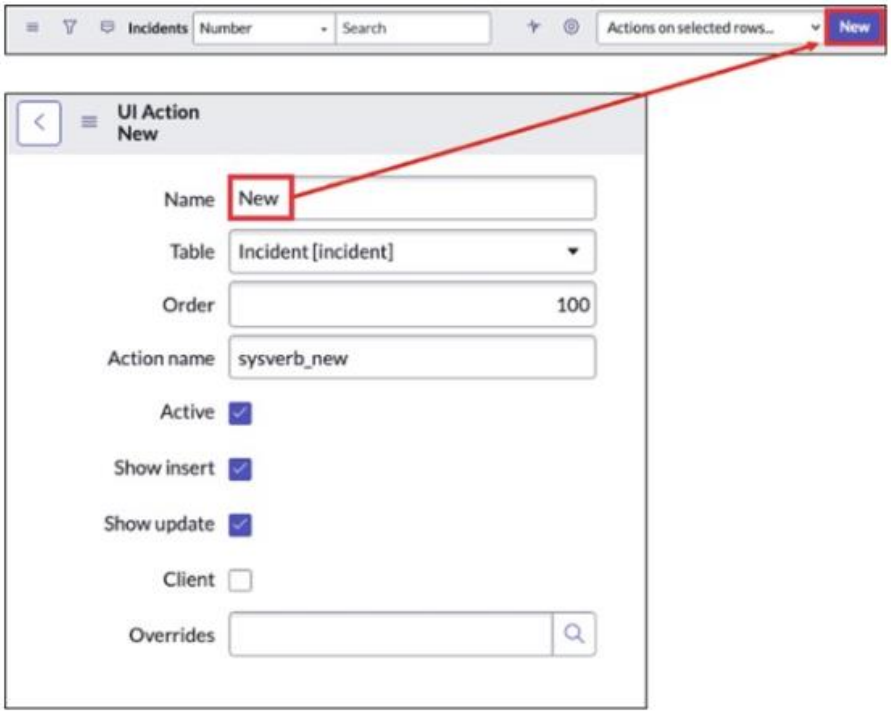
Data Policies are applied to all data entered into the platform: form(UI), Import Sets, or Web Services.

A Data Policy executes on the **server** side but can also run as a UI Policy on the client side.

Data Policies can be used on lists to make a field read-only; the field will appear to be editable, but the update will fail.

UI Actions

User Interface (UI) Actions add buttons, links, and context menu items on forms and lists, making the UI more interactive, customizable, and specific to user activities.



The **New** button on the existing list of All Incidents is an example of a UI Action.
UI Actions can contain scripts that define custom functionality. UI Actions can be **server or client side** depending on the 'client' check box selection. This setting determines when a UI Action can appear.
UI Actions include:

- Form buttons
- Form context menu items (right click header)
- Form Links (Related Links in a form)
- List buttons
- List context menus (right-click header)
- List choices (at the bottom of a list)
- List links (Related Links at the bottom of a list)

When Order 100 is specified, UI Actions with Order numbers greater than 100 will display after this UI Action, while UI Actions less than 100 will display before this UI Action im the user interface.
When the UI Actions **Active** box is checked, the UI Action is visible and can be interacted with by a user.
To turn on the Save button/UI Action, navigate to **All>System Properties>UI Properties** and select **Show "Save", "Insert", and "Insert and Stay" buttons on forms.**

Client Script

Client Scripts make “real-time” changes to the appearance of the user interface, especially forms. Client Scripts execute on the **client side.**

Client Scripts can be created to do the following:

- Automatically update the location field to reflect the value (user) entered into the caller field
- Disable the attachment link of a closed record when the form is loaded so a user is unable to add or modify attachments.
- Display a notice at the top of the page to confirm a catalog request was submitted.

Several types of scripts are supported:

- **onCellEdit()**: runs when a cell on a list changes value through use of the list editor
- **onChange()**: runs when a particular field changes value
- **onLoad()**: runs when a form is loaded
- **onSubmit()**: runs when a form is submitted

Catalog Client Script Example

Client Scripts: Control the behavior of catalog items when presented to users

Tips from the Field: If no-code alternatives can achieve the same goal, use them instead!

When to Run

onLoad	onChange	onSubmit
• Modify choice list options available	• On change of variable A, clear value of variable B • On change of variable A, populate values of variables B and C	• Validate field values – start date < end date – Justification > X charactrs long

Business Rule

A **Business Rule** is configured to run when a record is displayed, inserted, updated, deleted, or when a table is required.

Business Rules can be set to run **before** or **after** the database action has occurred.

The **When** setting determines when the business rule executes and has the following choices:

- **Before** a record is saved to the database
- **After** a record is saved to the database

- **Async** (queued); client and server work independently so the client is not waiting for the server
- **Display** before the record is displayed

Business Rules execute on the **server side.**
Unlike UI Policies, Business Rules are NOT real-time:

- They do not monitor fields on a form
- They monitor records as they are inserted or updated

Migration and Integration

- Application scoping protects applications by identifying and restricting access to available artifacts and data.
- Applications developed prior to application scoping are in the global scope.
- All custom applications have a private scope that uniquely identifies them and their associated artifacts.

  Administrators can specify what parts of an application are accessible to other applications from the custom application record and each application table record.
  By default, the application can access and change its own tables and business logic but not other applications unless the receiving application allows access.

System Update Sets

An **Update Set** is a group of configuration changes that can be moved from one instance to another. Update Set allow administrators to group a series of changes into a named set and then move them as a unit.
Every instance of SN has a **default** update set, however, admins should use **named** update sets for moving customizations between instances.
An update set is an XML file that contains:

- A set of record details that uniquely identify the update set
- A list of configuration changes
- A state that determines whether another instance can retrieve and apply configuration changes

Basically, an Update set record is a “point in time” XML snapshot of process records. An Update Set works by writing changes from tracked tables to the **Customer Update [sys_update_xml] table.**

An Update Set is a container for configuration records. By navigating to **All > System Update Sets > Local Update Sets**, you can create a new Update Set or set an existing one as your current Update Set.

Batch update sets enable you to group update sets together so you can preview and commit them in bulk.

What is Captured in an Update Set?

- Business Rules
- Client Scripts
- UI Policies
- Fields
- Forms and Form Sections
- Report Definitions
- Tables
- Views
- List Configuration
- Roles
- Flows/Published Workflows

What is not captured:

- New Data Records
- Modified Data Records
- Tasks
- Modified Cis
- New Users and Groups
- Schedules
- Scheduled Job
- Dashboards

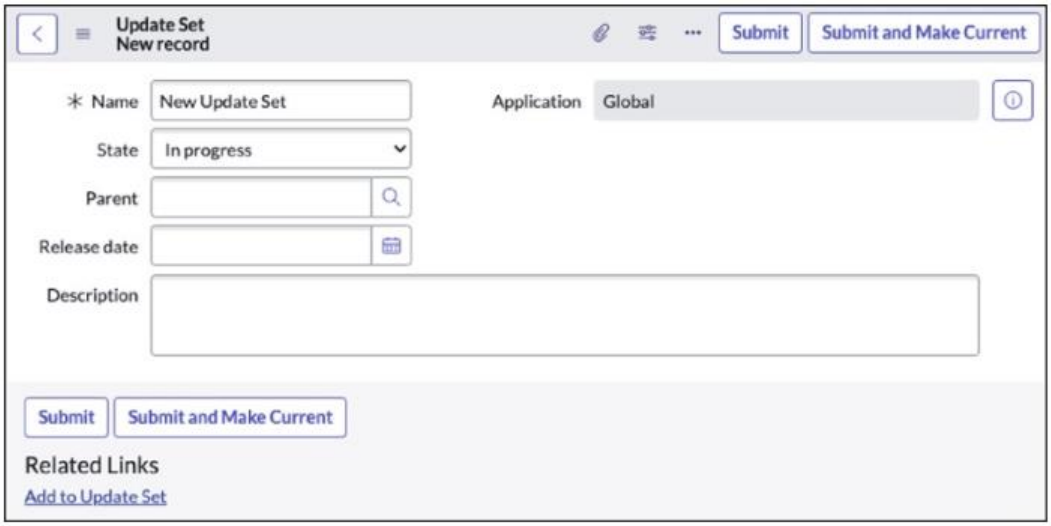
What is captured in an Update Set is typically a customization or a configuration change and does not include changes to data records.

When completing work, you may want to move data records with your updates. These records can be useful for testing or training. Data can be moved using the **Export XML** function.

The **Unload Dashboard** function unloads the entire dashboard with all related tabs, including portal pages.

Create and Select an Update Set

Create an update set to store customization changes and select it as the current set.

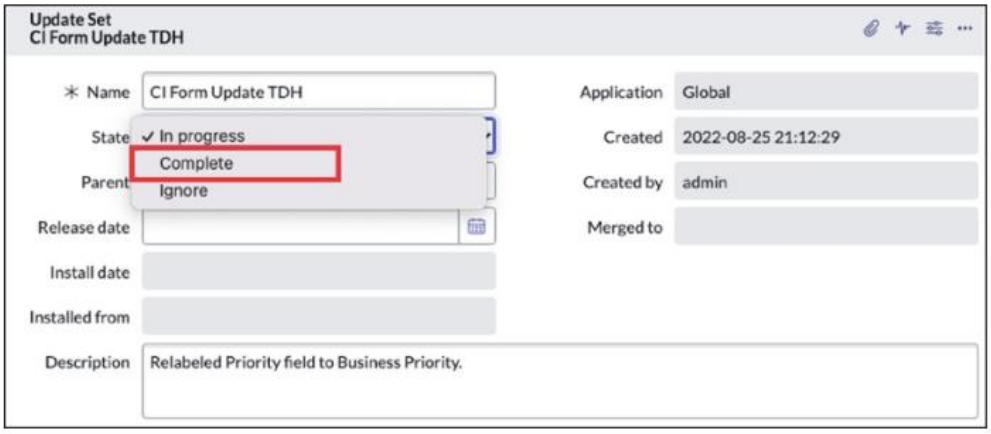


- Navigate to **All>System Update Sets>Local Update Sets** and click **New**.
- Complete the form from the fields in the table
- Click **Submit** to create the update set

If the picker is enabled and the update set is in the **In progress** table, click **Submit and make current** to select the new update set as the target for configuration changes.

Mark an Update Set Complete

When you have completed the configurations and compared local update sets to resolve conflicts, mark the update set as **Complete.**



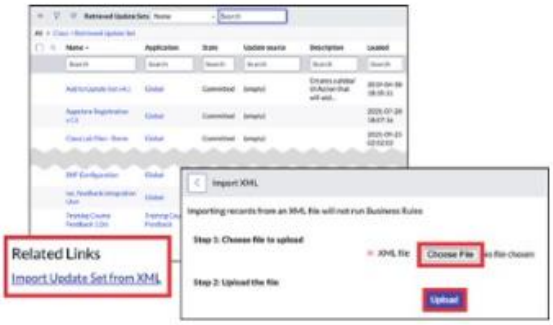
- Open the update set record
- Change the **State** of the update set to **Complete**

The update set is available for other instances to retrieve.

Mark an update set as **Complete** only when it is ready to migrate. Once an update set is complete, do not change back to **In progress.** Instead, create another update set for the rest of the changes, and be sure to commit them together in the order that they were created.

Applying an Update Set

1. Retrieve
   a. Navigate to **All>System Update Sets>Retrieved Update Sets**
   b. Click **Import Update Set from XML**
   c. Choose a file to upload
   d. Upload the file



2. Preview

3. Commit

Previewing compares an update set retrieved from a remote instance to updates on the local instance to detect potential problems.
Determine the changes to make in a single Update set since SN recommends limiting Update Sets to a max of 100 records to reduce the number of potential conflicts and make it easier to identify and review changes.

Retrieval from a Remote Instance
In the Production instance:

1. Navigate to **All>System Update Sets>Update Sources**
2. Select the development instance with the completed update set you want to commit

Additional Administrator Resources

Perform Basic Monitoring and Upgrade Functions
There are several tools that may aid you in monitoring performance and upgrading your instance:

- Performance Dashboard
- Instance Scan
- HealthScan
- Release Notes
- Upgrade Center

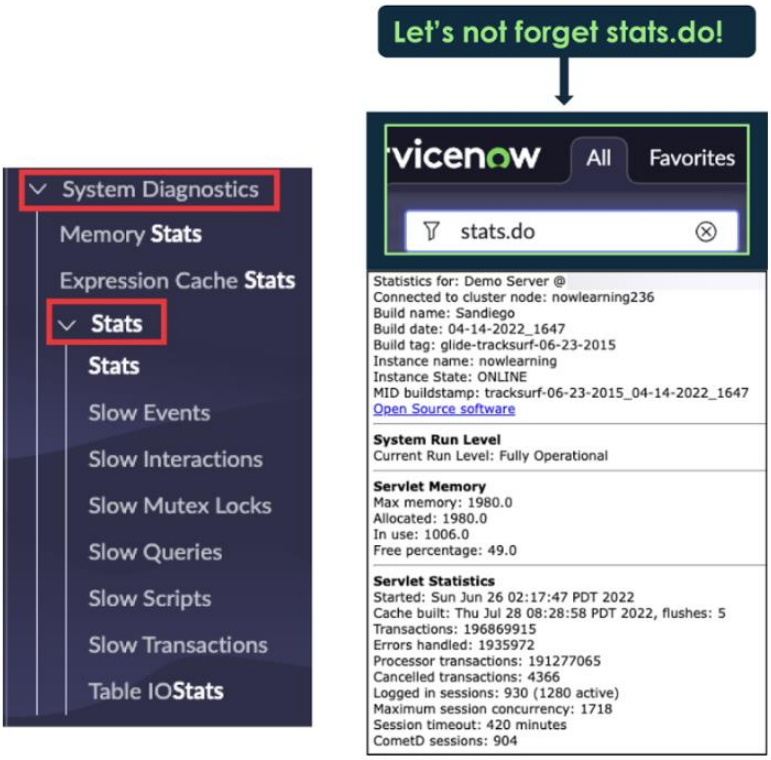
Performance Dashboard
The SN Performance dashboard feature provides central access to different graph sets for monitoring the performance of your instance.
The graph sets provide performance data for the following areas:

- Asynchronous Message Bus (AMB)
- Database
- Instance View
- MYSQL Global Status
- ServiceNow Servlet
- Slow Pattern

You can also filter the data in each graph by using different measurements, such as:

- Maximum and minimum values
- Means
- Medians

System Diagnostics Stats Tools
To aid in performance evaluation, **Stats Tools** records statistics for system activities that affect performance such as the execution of queries, scripts, and transactions.
The Stats tools plugin is activated by default. It requires the admin role to activate or upgrade and it requires the **com.snc.jrobin**.



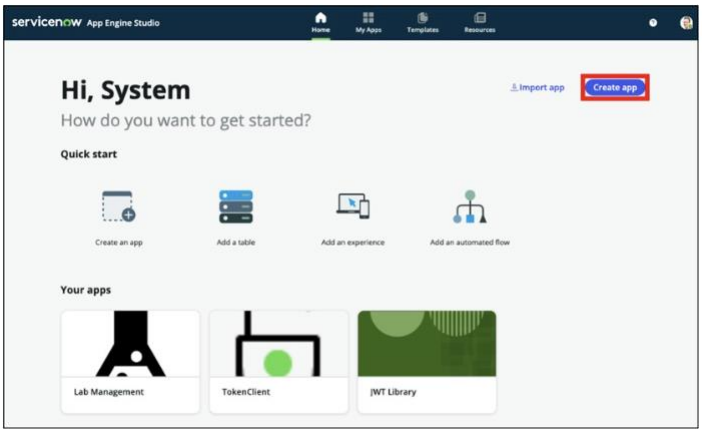
Stats Tools adds modules under **System Diagnostics > Stats**, including **Slow Queries, Slow Scripts**, and **Slow Transactions**.
Each module accesses a table of activity patterns [sys_query_pattern], [sys_script_pattern], [sys_transaction_pattern].

<u>Automated Test Framework</u>
Consider the ATF to create and run automated tests on your SN instance after modifying it. These tests provide enough flexibility to help confirm the instance still works as designed. By default, the system property to run automated tests is disabled to prevent you from accidentally running them on a production system.
Run tests only on development, test, and other non-production instances to avoid data corruption and outage.

<u>App Engine Studio (AES)</u>

- View and search created applications
- Explore tutorials and helpful resources to get started
- Add data, experience, logic and security to our app

With AES, you can leverage development tools and build apps quickly using templates for pre-built solutions. Engage with real-time data and empower employees in your organization with this intuitive app development env.

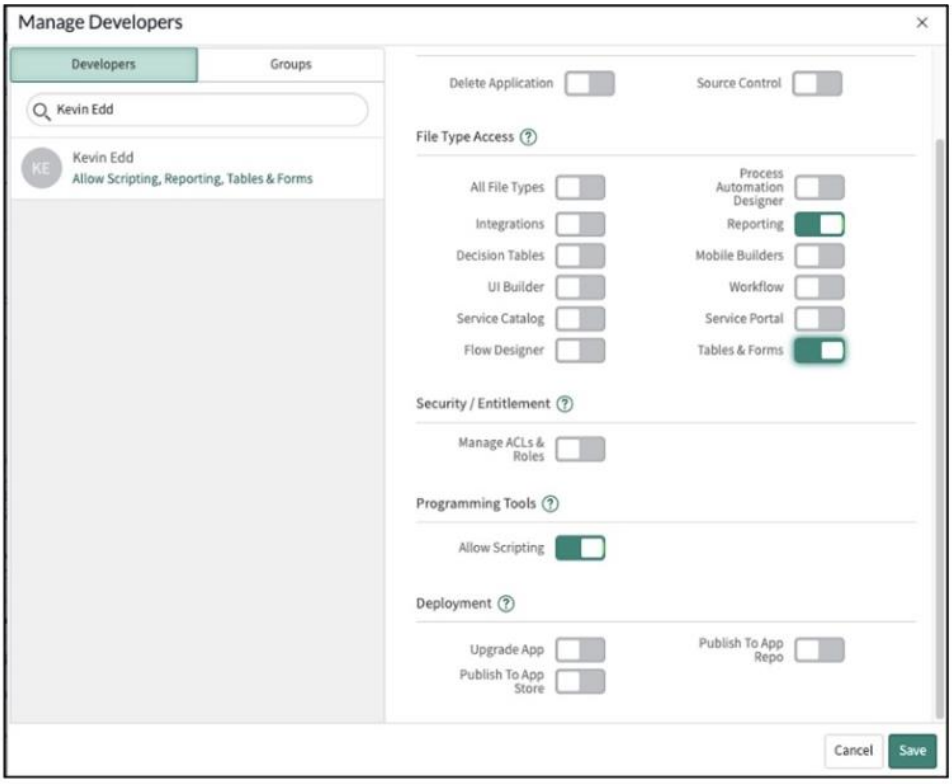


Using AES, you can also:

- Create applications from scratch
- Access a list of recent applications you have built
- Access links to add objects to apps, browse app templates, and learn more about the available tools
- Browse through a list of templates you can use to create an app with preconfigured data, experience, automation, and security

Delegated Developers

Delegated Developers are non-administrator users and groups which are assigned one or more permissions to develop applications.
Each permission grants one or more delegated-development-specific roles to retain control over the system without having the admin role.



Upgrades and Releases

Upgrading your instance involves planning, testing, and validation.
Release nots guide you through completing the phases and tasks for a successful upgrade.

System administrators can check which release is running on an instance with **stats.do.**

The supported release version is one of the two most current releases. Supported versions provide technical help 24 hours a day, 7 days a week with access to patches and fixes.
To check the running release:

- Locate the Build name, Build date, and Build tag.
- Match the build name to the release name in the Release Notes.
- Match the build date and tag to the build date and tag in the Release Notes.

Confirm upgrade and Patch:
The event system.upgraded can trigger the System Upgraded notification. If the event has a State of error, the notification is not sent; however, the upgrade may have completed successfully.
To verify the upgrade is complete:

- **Navigate to All>System Diagnostics>Upgrade Log**
- Locate the message “Notifying HI that upgrade has been completed.”

To confirm that a patch has been applied, find the system.patched event. You can create a notification for this event; it is not available by default.

Feature

- Introduces new features
- Includes all available firxes to existing functionality
- Is production-oriented: quality and stability are of the highest priority throughout the lifecycle

Patch

- Supports existing functionality with a collection of problem fixes
- Generally does not include new features

Hot fix

- Supports existing functionality with a specific problem fix fro==or a feature release
- May or may not include any previos fixes for a given release
- Does not include new features