ServiceNow Administration Fundamentals

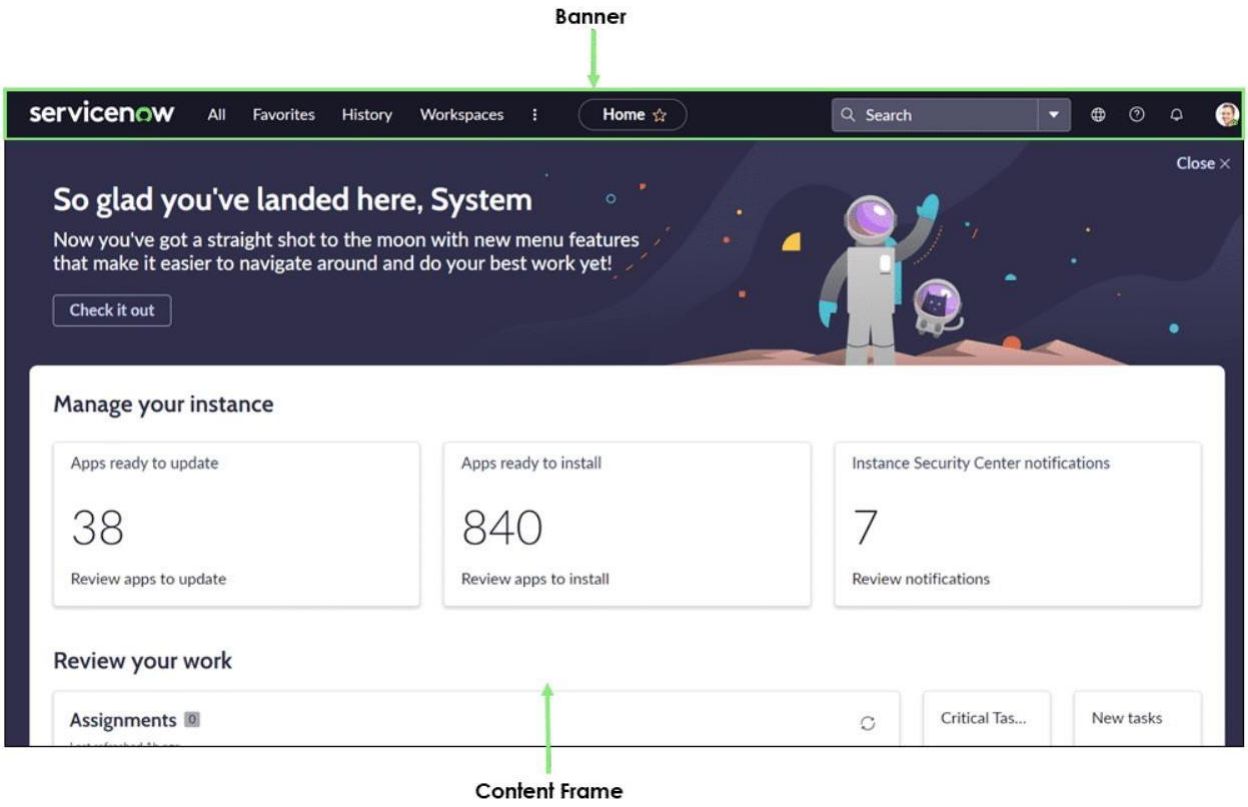
Platform Overview and Architecture

ServiceNow is an American based company established on 2003 and this company provides an Application Platform as a Service (APaaS) which is known as a Now Platform. It is a cloud-based computing model which provides the infrastructure needed to develop, run and manage applications. This platform is not limited to any business functions of any organisation and can be utilised in different areas of an organisation like HR, Finance, Security etc. Organisations can utilise this to automate manual repeatable processes and standardize their service delivery. It is a single platform to automate business processes across the enterprise by providing the infrastructure needed to develop, run and manage applications. It is a single data model which is built on a flexible table schema and delivers a common set of core capabilities and reusable components. This platform has a multi-instance architecture rather that multi-tenant architecture which is used by other cloud companies. In multi-tenant all the data get co-mingled with each other on databases making it complex that require hardware and software maintenance on a regular basis which can lead to availability issues for customers. ServiceNow cloud is built on an advanced architecture called multi-instance where the organisation's data applications and customizations reside in a unique software stack called an instance. Each instance is isolated from every other instance but they can still communicate with each other. It provides high advanced availability of data centres in any situation, like the operational failure or outage of instance traffic can be quickly rerouted to redundant servers to provide continuous availability. It provides four weekly backups of your data with six days of different daily backups. Every organisation has their own instance whose maintenance will be different from others and makes sure that they are taking the right backup of the instance. Security services can also be integrated with single sign-on services that are compliant with the standard.

User Interface and Branding

The ServiceNow user interface has two components: -

- Banner
- Content frame



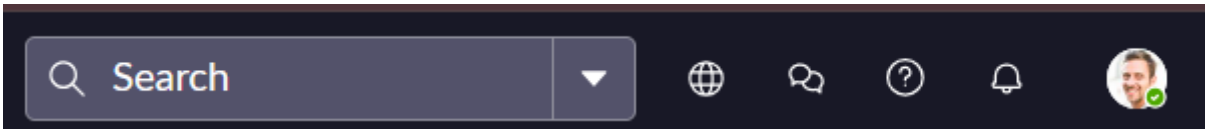
The banner spans the top of every page. It contains a logo, navigation menus, the title of the page in the content frame, global search, help, notifications, and a user menu.

Global Search: Global Search is used to search results that includes all the items the search term appears in. By default, global search results can include: -

- Cases
- Customers or Users
- ServiceNow Community Questions
- Incidents
- Change Requests
- Problems

- Knowledge Articles

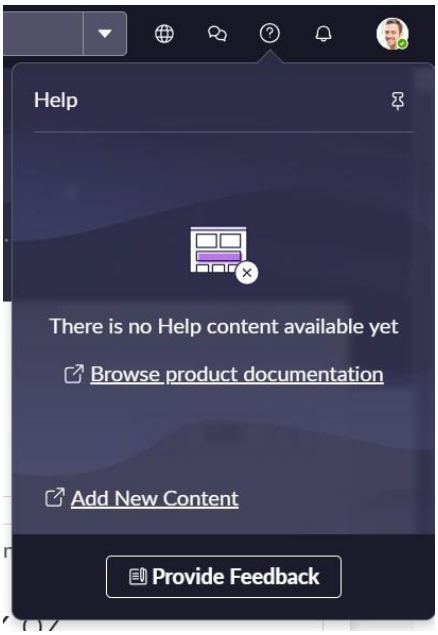
To enter search terms, we need to click in the search icon and enter the search term in the search field.



Connect Chat: Connect chat is a real-time messaging tool that enables users to chat with individuals and groups, quickly share files and collaborate on any record by connecting with right people instantly. (IMPORTANT: Connect Chat is not available in Next Experience. Starting with Washington DC, certain Connect Chat functions are available in Next Experience byusing Sidebar.) Connect Chat animates communication around records, Visual Task Boards, topics of interest, or groups of people. Features include:

- Direct conversations between two users.
- Group conversations between three or more users.
- Conversations linked to records. Comments and work notes appear in conversations in real time and users can update the record directly from the conversation.
- Drag-and-drop sharing of links, files, and records.

Contextual Help: The ServiceNow system provides several preconfigured help contexts that are connected to a link that display the exact information needed for the current list, form, or record. Help topics can be defined by own which then configures all to appear when the help icon is clicked in a ServiceNow record. Users click the help icon to open the default help pages provided in the base system. For any page that does not have context-sensitive help defined, the instance displays the help system welcome page. Users can use the search feature or the index to find the correct help topic. The location of the help icon depends on the user interface version.



Application Navigator:

Application navigator appears at the left of the interface and provides access to all available applications and modules, favorites, and recently viewed items. It is used to quickly find information and services. It consists of a navigation filter and the following tabs.

- All applications- Displays all application menus and modules. Each application appears as a section in the application navigator denoted by an application label. The tab lists modules by name under each application label.
- Favorites- Displays items that have been added as favorites.
- History- Displays items that are recently accessed.

ACLs: Each row represents an individual access control list (ACL). The sequence (#) in the results the order in which ACLs are evaluated. At a table level, ACLs are evaluated only for roles and security attributes, conditions and scripts aren't evaluated. Rules for access control lists (ACLs) restrict access to data by requiring users to pass a set of requirements before they can interact with it.

UI Policies: UI policies dynamically change the behaviour of information on a form and control custom process flows for tasks. For example, UI policies can be used to make the number field on a form read-only, make the short description field mandatory, and hide other fields. Basic UI policies do not require any scripting, however for more advanced actions, use of the **Run scripts** option is required. Client scripts can be used to perform all of these actions, but for faster load times UI policies are used when possible.

Client scripts for UI policies

Any scripts can be created for UI policies to run on the client side.

We can use different options in the UI Policy form to control when and how the UI policy is applied. These options include client scripts, OnLoad execution, and view-specific UI policies. Administrators can use the UI Policy form to create client scripts that run onChange when the UI policy conditions are met (**Execute if true**) or not met (**Execute if false**). To display these scripting fields in the UI Policy form, in the Script section, select the **Run scripts** check box.

Business Rules: A business rule is a server-side script that runs when a record is displayed, inserted, updated, or deleted, or when a table is queried. Business rules are scripts that run when certain server-side conditions are met. Business rule conditions include when to run a business rule in relation to a database operation, and what record operations the business rule applies to. There are other scripting options available on the platform for client-side conditions, such as client scripts and UI actions.

Client Scripting: Client scripts allow the system to run JavaScript on the client (web browser) when client-based events occur, such as when a form loads, after form submission, or when a field changes value.

Branding: Branding in ServiceNow enables us to customize our configurable workspace to our Company Branding. It is done by applying our distinct corporate identity across the Now Platform UI to create a shared identity, build trust and speed adoption (for e.g. Company Logo, font etc.).

Guided Setup: it provides system admin a step-by-step instruction to configure various Applications and Modules within the instance to suite the need of the users. It assists us with planning the roll-out of the product and performing the basic configuration to go live. To access Guided Setup; locate application in the Application Navigator and select ITSM Guided Setup or ITOM Guided Setup.

ITSM Guided Setup includes the following categories: **Company**, Connectivity, Foundation Data, CMDB, Incident Management, Major Incident Management, Problem Management, Change Management, Service Catalog, Knowledge Management, Continual Improvement Management, Project Communication, Go Live

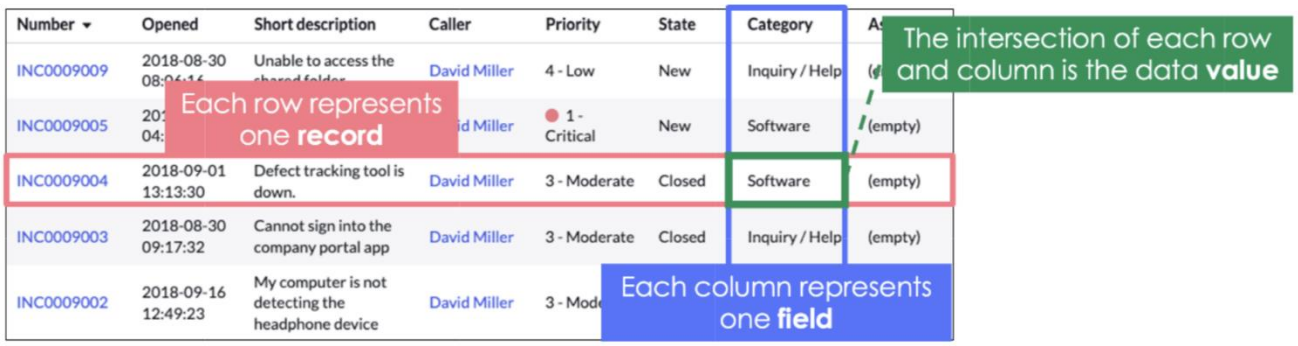
ITOM Guided Setup includes the following categories: MID Server, Discovery, Event Management, Operational Intelligence, Cloud Provisioning and Governance

ServiceNow Portal: It is a widget-based tool that allows creation of intuitive, user-friendly interfaces to the Now Platform.

UI Builder: It allows to build out a functional page by choosing from a library of components and layouts.

List and Filters

A list displays a set of records from a table within the content frame. This example shows a list in the classic environment. Each row in a list represents one record and each column in the list represents one fields.



ServiceNow Lists interface or list view is a user interface pagedisplayed in the content frame that is designed specifically for lists of records in database table. It includes tools that make it easy to sort, search, filter and analyse list data quickly. Italso provides the ability to select single item from the list to display in more detail via a formview. There are many ways to access lists in ServiceNow. One is the Application navigator for
e.g. includes links to lots of different lists by navigating to the incident all item in the application navigator that opens the list interface for the incident table. Another method is by using the dot list command in the application navigator filter by entering the name of any

desired table followed by dot list to open the list interface for that table. For e.g.

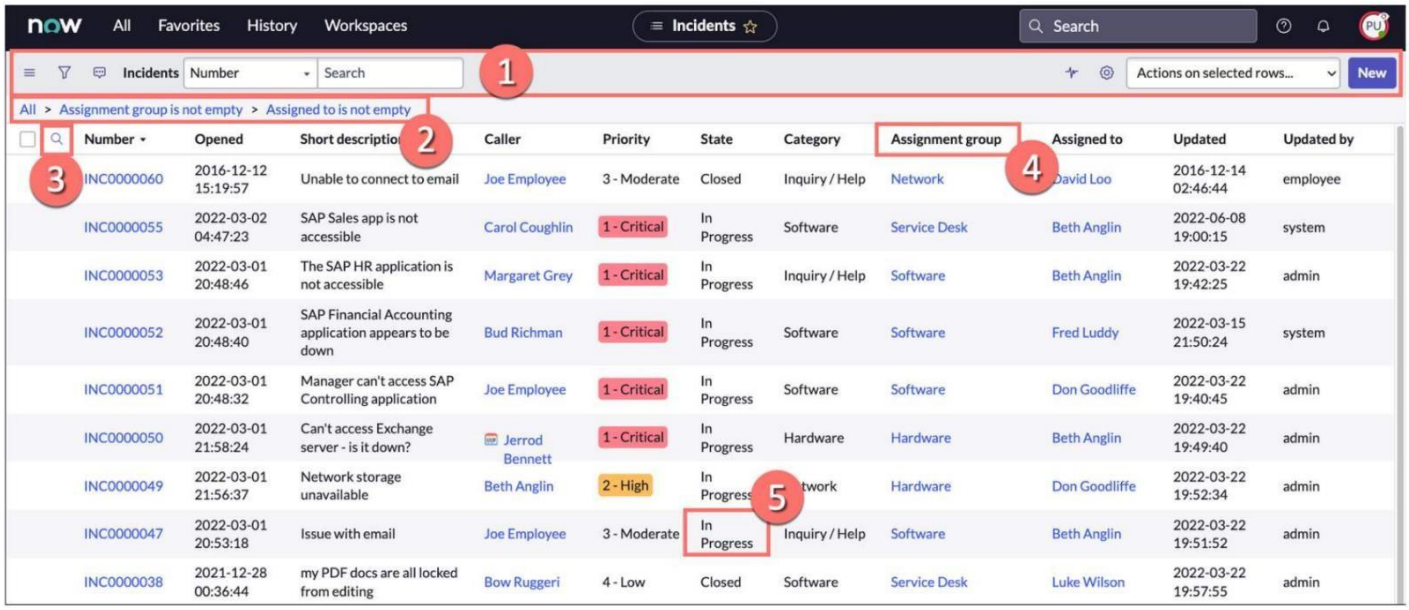
table.list → Task Table

incident.list → Incident Table

sys_user.list → User Table

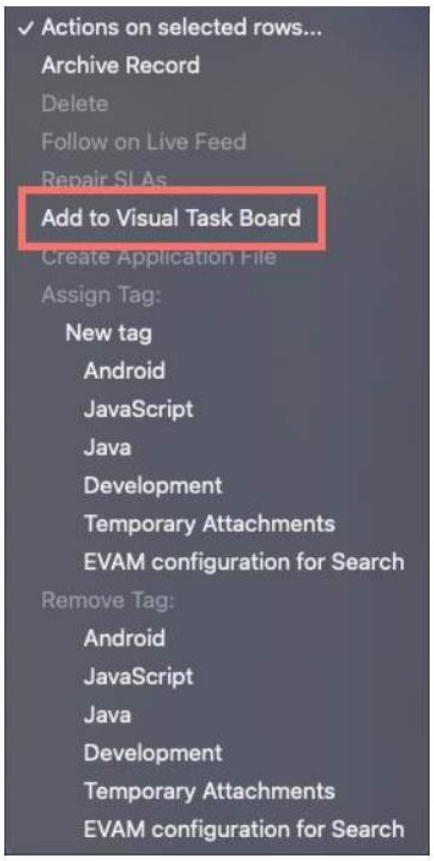
In case the user don't know the name of the table: sys_db_object.list → Database of all tables.

Standard Paradigm:

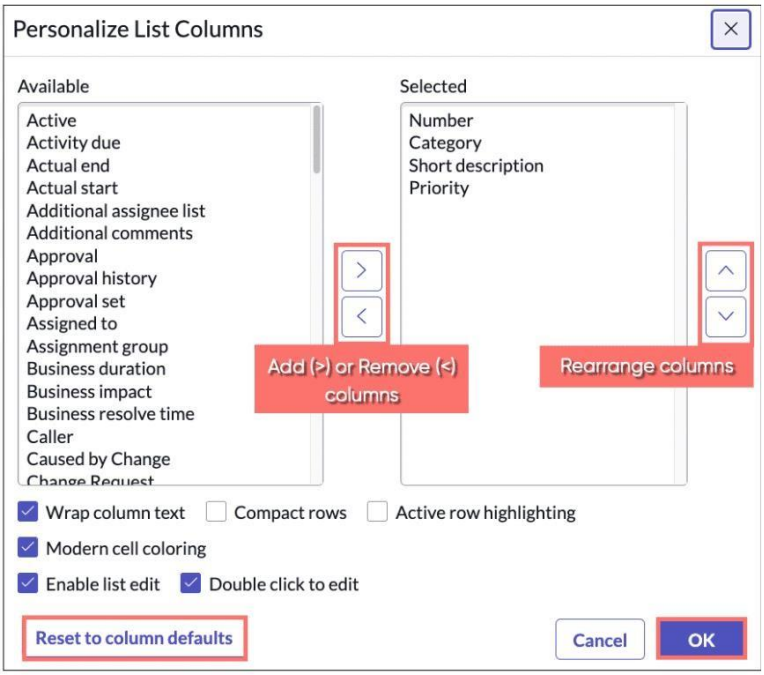


1. **Title bar**: Displays the list title and view name (when not on the default view), as well as search list values. The filter icon allows you to create custom filters on your list. The gear icon allows you to personalize the list.
   List controls: By selecting the **list controls** icon next to the filter (funnel) icon to access options related to viewing and filtering the entire list.

Actions on selected rows: By hovering over and selecting any row in a list (or using the **Select All** feature) and selecting the **Actions on select rows** drop-down menu to see this list of actions. A popular item here is to add certain records from a list to a *Visual Task Board (VTB)*.



Personalised list (gear icon): Personalizing a list to show or hide fields on a view, as well as change the list column order for the current logged-in user.



2. **Breadcrumbs**: Offers a quick form of filter navigation. Breadcrumbs are created by using the condition builder to apply filters to a list (see next lesson, "Filters and Tags").

3. **Column header search**: Provides a search within a specific column.
4. **Column headers:** Displays the title (field on a form) of the column.
5. **Field values**: Right-click on a field value to access additional actions.

Filter conditions:

A **filter** is a set of conditions applied to a table list to isolate a subset of data. Three components that make up a filter condition include [1] **field**, [2] **operator**, and [3] **value**. By selecting the **Show/hide filter** (funnel) icon to add, remove, or edit filter conditions and apply them.

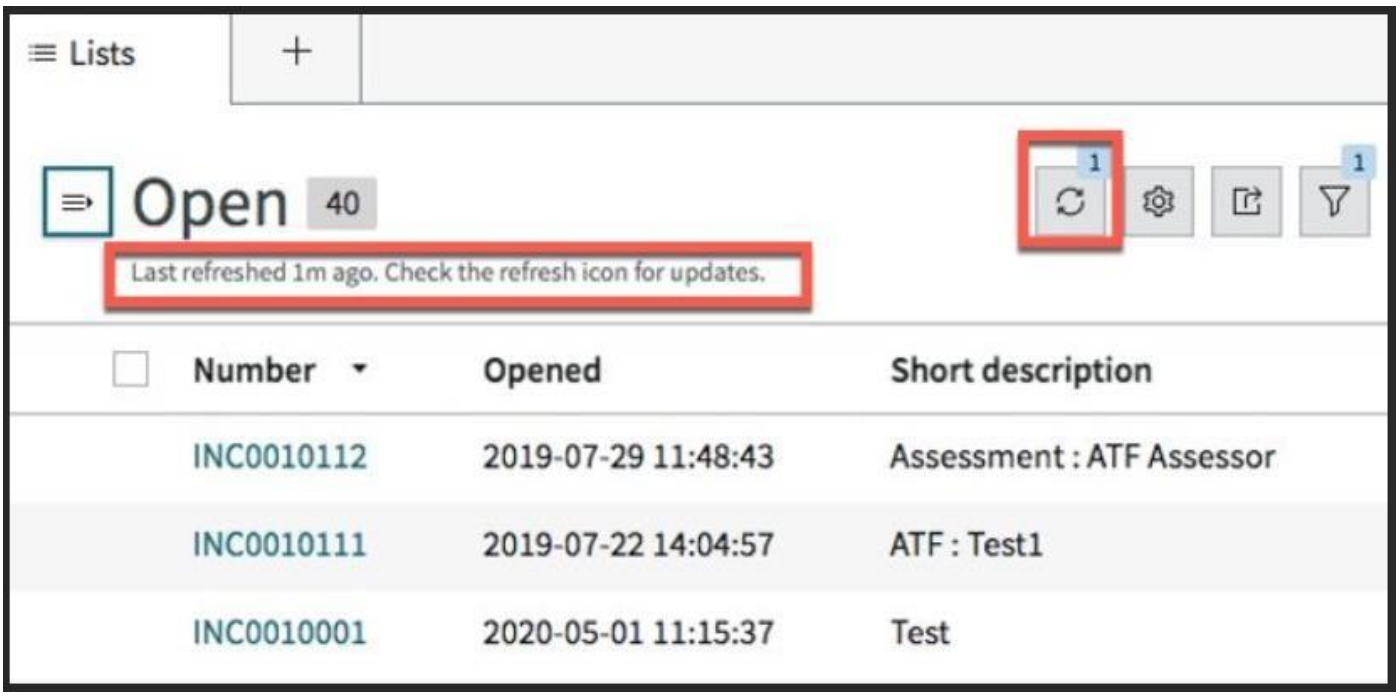
1. **Field**: A choice list based on the table and user access rights. The choice list includes fields on related tables by **dot-walking**.
2. **Operator**: A choice list based on the field type. For example, in the incident table, the greater than operator does not apply to the **Active** field but it does apply to the **Priority** field.
3. **Value**: A text entry field or a choice list, depending on the field type. For example, in the Incident table, the Active field offers a choice list with values true, false, and empty, while the Short description field offers a text entry field.

Refresh List:

To enable the list refresh, we need to prompt the system property with the Admin as the required role.

a. Entering sys_properties.list in the Workspace filter navigator.
b. Searching for glide.lists.live_list_enabled in the System Properties list search bar.
c. Clicking the value column field and change the value to true.
d. Selecting the checkmark icon to save the selection.

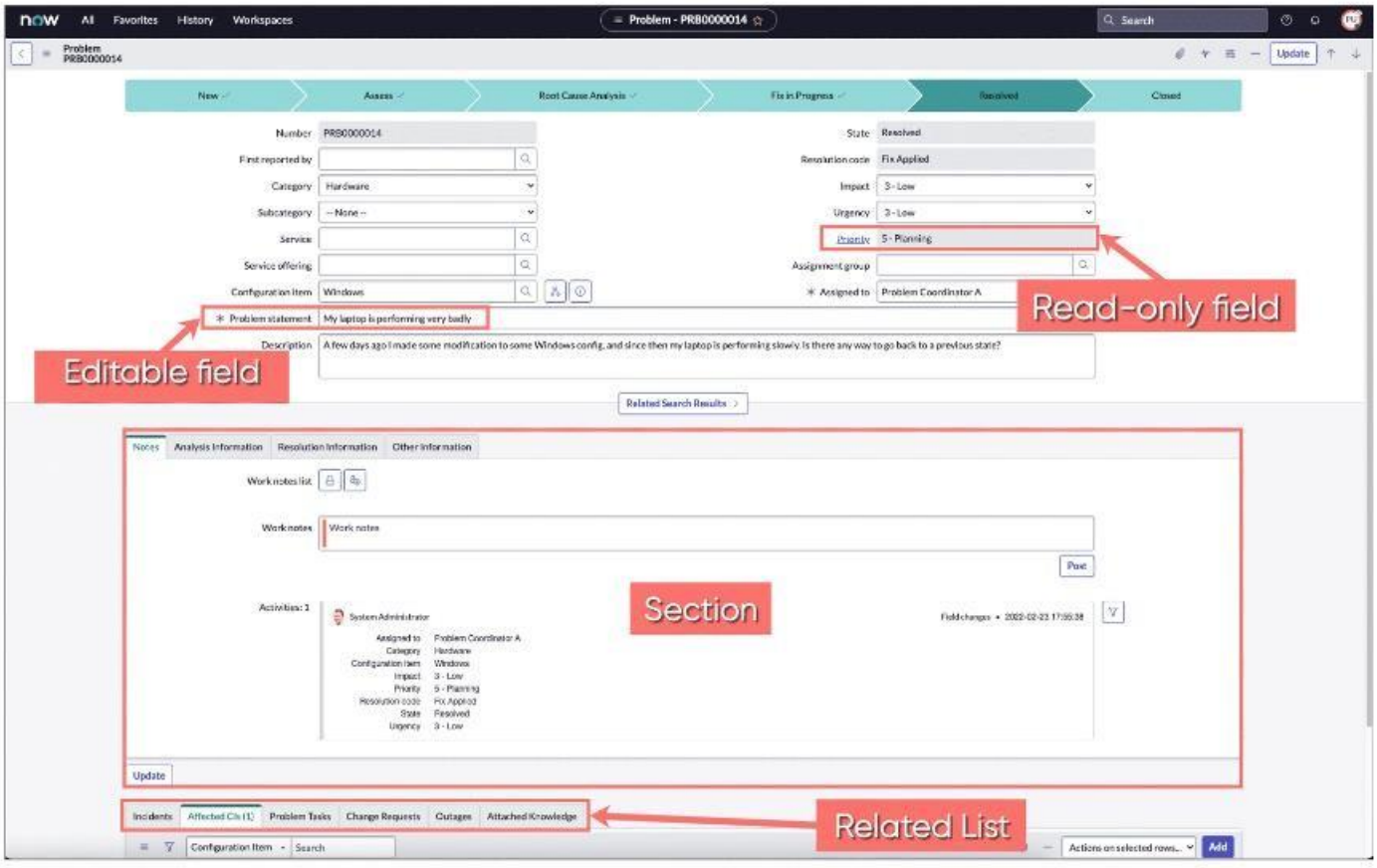
This prompt will result as → Before refreshing the list, the refresh icon displays an indicator with the number of records changed and the list header prompts to check the refresh icon for updates. After the list gets refreshed, the affected rows are displayed in bold text with an indicator badge.



Forms

A form displays fields from one record. The specific information on a form depends on the type of record displayed. Users can only view and edit the fields in a record they have access to. In addition to fields, the form can also contain sections (ex. Notes, Analysis Information, Resolution Information, and Other Information sections seen below) and Related lists. Related lists display records from tables that have a relationship to the current record. Related lists will only appear on a form if the record has already been saved to the database.

Form Layout:



Back: If you are viewing a list and select a record to navigate to a form, this back button will lead you back to the original list of records.

Form Context Menu (Additional actions): provides additional options specific to the form. **Save** can be found in the Form Context Menu and is used to save a form while remaining on a page. Depending on which type of form you viewing, options may vary. You may also right-click on a form header in order to save a record.

Record title: displays the name of the record. If the record is anything other than the Default view, the view name will also display in the form title.

Manage Attachments: Add or remove files by selecting the file attachment (paperclip) icon.

Activity Stream: Displays the list of activities, or history on a form. It provides an easy way to track items not displayed within a field in the record (for example, journal fields like comments and work notes).

Personalize Form: Allows you to select which form fields display on the record. A cleared grey checkbox indicates form fields that you previously hid. **NOTE:** Only the fields that the form is configured to display are available in the Personalize Form menu (the options will vary depending on the type of form).

More options menu: Select the 3 dots (**...**) to have more options on a form. The options available may vary based on the form. For example, the More options menu on an incident form may display the option to **Email** users straight from the record, select the **Toggle Template Bar** (manually apply, create, or edit templates), **Toggle Annotations on/off** (display guidance/more information about certain fields on a form) and create/add **Tags**.

Submit/Update: If you are working on a new (not yet saved) record, you will see the **Submit** button. If a record has been previously saved and you have made more updates, you can select **Update**.

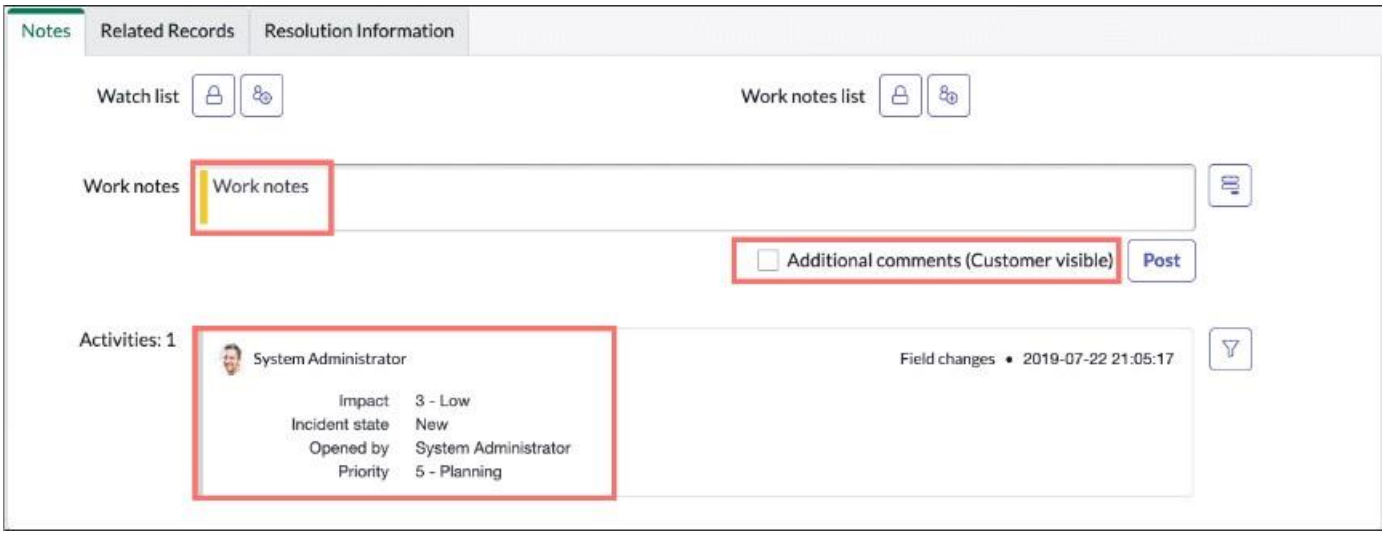
Resolve: Since the view above is from an Incident form, you can **Resolve** the incident all by the click of a button.

Related Lists:

Related lists show records from tables that have a relationship to the current record. Related lists are presented as tabs at the bottom of the form view. From a related list, a user can select records using the **Edit...** button or create new records using the **New** button. As with other lists, a user can personalize columns, filter, or refresh the list using the **gear** icon, **list controls**, or the **funnel** icon.

Formatters:

A **formatter** is an element used to display information that is not a field on a record. For example, the Activity Stream or Work Notes/Customer Comments are examples of a formatter as it displays a list of activities.

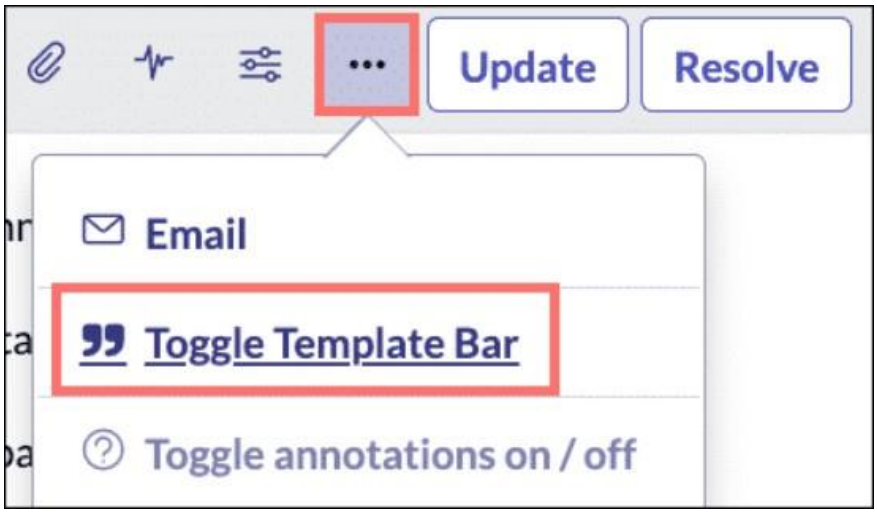


Form Personalisation:

By selecting the Personalised form icon (gear icon) on any record to choose which fields need to be displayed in a form. There are two ways to remove a field in a form: (1) A cleared (unchecked) box which indicates a field that is previously hidden. (2) A Hide icon (red circle with a dash through it). Additionally, by selecting the Personalize Form icon andselecting Reset.

Form Templates:

Templates allow fields to be populated automatically, simplifying the process of generating new records. These can be personalised by selecting one, adding the appropriate fields and saving/updating the changes.



Task Management

Tasks are essential building blocks within ServiceNow, used to track activities and progress within various processes:

- Create and manage tasks: Creating tasks manually or through workflows, assigning them to specific users or groups, setting due dates and priorities.
- Task routing and automation: Leveraging automation rules to automatically route tasks based on pre-defined criteria, which improves efficiency and reduces manual work.
- Task tracking and reporting: Monitoring task progress, identifying bottlenecks, and generating reports to analyse task completion rates and identify areas for improvement.

Notifications

Knowledge Management

The Knowledge Management match all check boxes which is used to determine whether all the elements from each populated criteria field must match. If selected, only users who match all criteria are given access.

Knowledge bases use user criteria records to determine which sets of users can read or contribute knowledge within that knowledge base. If a knowledge base has no user criteria selected, articles within that knowledge base are available to all users.

The knowledge management guided setup provides a sequence of tasks that help Administrators configure knowledge management for your instance. Knowledge managers can define knowledge categories to pre-populate the list of available categories, and knowledge contributors can select categories, and add or edit categories, if enabled for a knowledge base. Knowledge articles within a knowledge base are grouped by category. With knowledge management, each organisation can have their own knowledge base with flexible controls over who can see the information and who can help develop its content. This allows users to create, categorise, review, approve and browse important information in a centralized location that is shared by the entire organisation. Knowledge content exists within a knowledge base managed by one or more knowledge managers.

Service Catalog

The ServiceNow platform supports multiple service catalogs. Users with the admin or catalog_admin role can manage multiple Service Catalogs and provides services to different teams within the organisation. Examples include IT services, human resources and facilities management. Views can be defined for groups that view a Catalog, and Catalog items can be shared by multiple catalogs. This results in the ability to dynamically control the ordering options from user to user.

Service catalog variables are global by default and provide options to tailor a catalog item to the customer's need.

A record producer is a specific type of catalog item that allows end users to create task-based records such as incident records, from the service catalog. Use of record producers is to provide a better end-user experience instead of using the regular task-based form for creating records. The look and feel of a record producer is similar to that of a catalog item. However, the record producer generates a task record such as incident, instead of a requested item.

Catalog builder is used to create or edit a catalog item or record producer using a visual and guided experience along with specified restrictions. The catalog builder experience enables you to delegate the creation and maintenance of the catalog.

Tables and Fields

Tables are a data structure or database component, which contain records. Records are the data stored in tables, which contain fields. Fields are the individual pieces of data within a record. Values are individual data elements that can be accessed through a field in a record.

System dictionary contains the definition for each and every table and field in the database. Navigate to All > System Definition > Dictionary to access the system dictionary to modify table and field attributes.

Reference fields store a unique system identifier (known as sys_id) of a record on another table which is what establishes the reference relationship. Admins can create new reference fields and configure several options for reference fields.

Table extensions is about deriving fields from one table and replicating them to another. It is used for tables that contain similar data. For example, child tables extended from the Task table include change, request, incident, and problem. Core tables is a table that exists in the ServiceNow base system. In other word, core tables come with the systems they are there from start. This is probably the easiest distinction of this table type. If it is a table that comes with ServiceNow, its core table. But it is important to understand that a core table can also be a parent table (e.g. Task), a child table (e.g. Incident) or a base table (e.g. task). What enables a base table different from a core table is that the base table is not an extension of another table (i.e. it has no parent table). Although custom tables are not in the base system, they can still interact with existing core tables or other custom tables. Related tables can be a combination of multiple core tables and/or multiple custom tables.

Access Control Lists

There may also be times when a system admin needs an elevated role to accomplish higher-security tasks. Elevated roles grant modification access, allowing users to modify the Access Control lists (ACL) which restricts access to data by requiring users to pass a set of requirements before they can interact with the data. To end the elevated role, uncheck the

box for the selected role and select Update. Access control is a security rule defined to restrict the permissions of a user from viewing and interacting with data.

ACL can restrict CRUD (Create, Read, Update, Delete) operations, and in addition to restricting this rule can restrict ServiceNow-specific operations on tables and fields. Additional ServiceNow operation examples include:

Execute- user cannot execute scripts on a record or UI page

Edit_ci_relations- User cannot define relationships between Configuration Item [cmdb_ci] tables.

Save_as_template- Controls the field that should be saved when a template is created.

Report_on- User cannot right-click a choice list field and Configure Choices.

Not even users with the admin role can impersonate a security_admin role (an elevated privilege role) and elevate privileges while impersonating to access higher security functionality. The base system, only the default System Administrator has the security_admin role. ACL rules for parent table apply to any table that extends the parent table, and ACL rules for fields in a parent's table apply to any table that extends the parent table.

Data Import

The importing of data into ServiceNow via integrations actually involves three data entities:

- Source- The entity containing the data to be imported into ServiceNow.
- Staging- A table that ServiceNow automatically creates as part of import process to temporarily store data pulled from the source prior to transforming and adding to the target.
- Target- The ServiceNow table into which data is to be imported.

Integration in ServiceNow start with the creation of a DataSource depending on its type (File, JDBC, OIDC, LDAP, REST, Custom). Common DataSource fields which appears while creation is:

- Name- A unique name for this data source.
- Import set table label- A label for the import set staging table to use.
- Import set table name- The Now Platform uses the label you entered to construct a unique table name. This prevents namespace collision with an already existing table.
- Type- The type of data that can be imported:

o From a remote **File**

o From a **JDBC** database

o Using **LDAP**

o Using **OIDC**

o Using **REST** to import data through the Integration Hub

o From a **Data Stream** action through the Integration Hub

o Using a **Custom** script

• Use Batch Import- If selected, specifies that loading data should be batched when you click Load data or Test load 20 records.

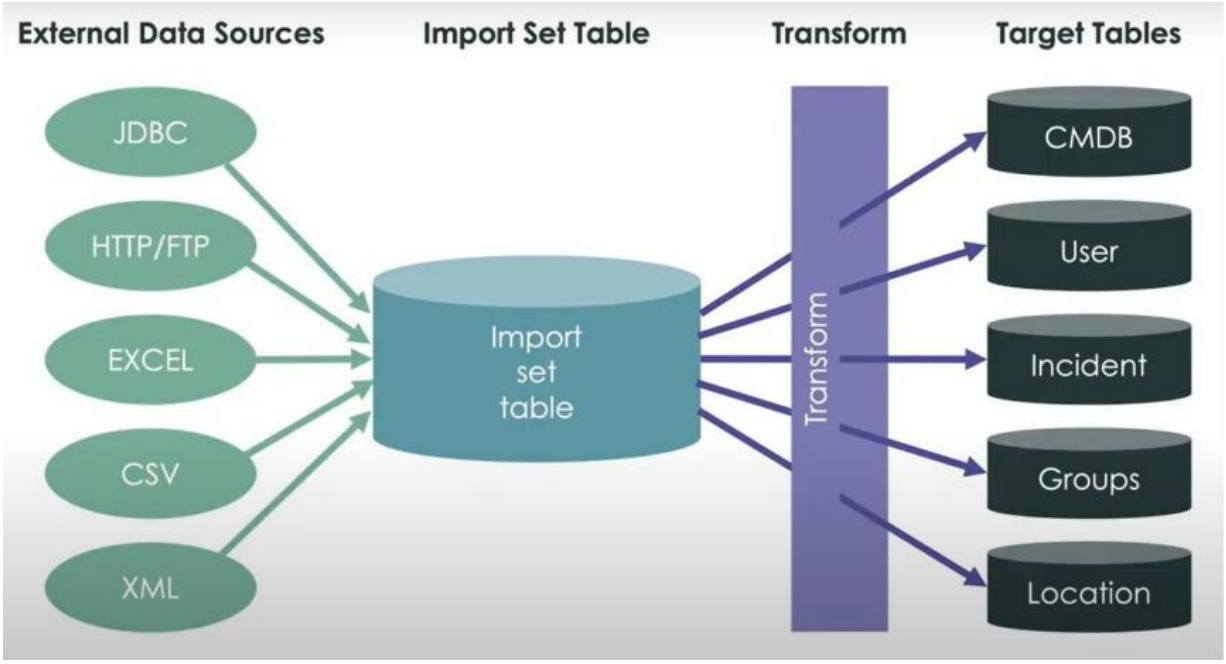
• Batch Size- Specifies how big the batches should be. Appears only when the Use Batch Import field is selected.

• Application- Application scope for this record.

Collection and importing into the ServiceNow platform:

Importing enables us to configure, run, and schedule your data imports all through a single simplified interface. It consolidates multiple data integration capabilities into a single environment, eliminating the need to create and manage several forms throughout the platform. Its step-by-step experience guides through the configuration of data sources, target tables, and data mapping. Data imports can be scheduled and run them on demand.

Import sets enable administrators to import data from various data sources, and then map that data into ServiceNow tables.



Data comes from the external data sources; that may be a spreadsheet or a web service, JDBC connection etc. This imports the data to Import set table that represents the ServiceNow tables. The job of the transform map is to clean up the raw data from the data sources.

When a row in the spreadsheet matches the asset in the record, the record is updated. If it doesn't match any asset in the table, a new record is created.

If multiple fields are set to coalesce: true- then all fields must match for an update.

Transform maps can process data between the import set table and target table. The process of doing so could involve Field Mapping and Data Transformation. The Transform Maps module enables an administrator to define destinations for imported data on any tables. Transform mapping can be as simple as a drag and drop operation to specify linking between source fields on an import set table and destination fields on any table. Use transform mapping to map source and destination fields dynamically.

Transform considerations Auto-mapping

Double-check that fields the system maps automatically are actually required. For example, encrypted passwords probably should not be mapped.

Mapping reference fields

If you are mapping reference field data and the sys_id does not exist, the sys_id could potentially appear in the target record as the DisplayValue, and this may be undesirable. Field mapping a large number of reference fields incurs additional performance overhead because the system checks that the referenced sys_id actually exists before performing choice actions at the field level.

Using multiple transform maps

Multiple transform maps can be applied to a single data source. One import set row is created per transform map, which can cause a large number of temporary records to be generated.
Run multiple transforms off a single import set

Users can select multiple transform maps during data import. The selected transform maps will be executed on the same import set in the order specified.

Transformation events occur during the process of transforming an import set table onto a table. These events modify the transformation behavior from any type of mapping specification. Transformation Event scripts modify the processing of the events at various stages of the transformation. For example, the processing of a mapping operation defined using the Mapping Assist Utility can be manipulated using the event scripts. There are a number of import set JavaScript objects that are accessible during these events. These objects represent tables or portions of tables. It is important to note that what these objects refer to varies depending on the context of the event in which they are referenced. Some transform events are as follows:

- onStart
- onComplete
- onBefore
- onAfter
- onForeignInsert
- onChoiceCreate
- onReject

CMDB Integration

Integration in ServiceNow refers to connecting the platform with other external systems to streamline workflows and exchange data. This allows for functionalities like:

- Automated incident creation from emails or ticketing systems
- User provisioning across different applications
- Data synchronization between ServiceNow and other platforms

The different integration methods in ServiceNow, such as:

- REST APIs: APIs (Application Programming Interfaces) that allow communication between ServiceNow and external systems.
- Web Services: A communication protocol for exchanging data between applications.
- Plugins: Pre-built extensions that enable integration with specific external systems.