ServiceNow

Week – 2

Koushik Mondra

KL University

2110030040cse@gmail.com

ServiceNow Platform:

- ➔ ServiceNow provides application platform as service (APaaS).
- ➔ It is cloud-based platform which provides infrastructure needed to develop, manage and run applications.
- ➔ Organizations can utilise ServiceNow in various departments such as IT, HR , Customer Service Management and also security.
- ➔ Organizations can automate different business processes using ServiceNow platform and increasing their speed of delivery.
- ➔ ServiceNow uses a multi-instance architecture, unlike many cloud platforms that rely on multi-tenant setups where data can be shared among companies. Each ServiceNow instance is separate, keeping your data and configurations completely distinct.
- ➔ In these multi-instance architecture, organizations data, applications and customizations reside in a unique software stack called instance. Every organization might have more than one instance but are isolated with each other and can be communicated.

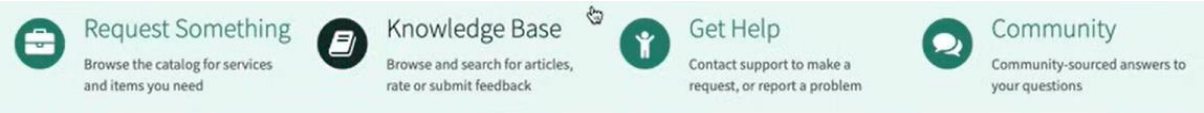
ServiceNow User Interface:

- ➔ There are 3 ways to interact with ServiceNow User Interface. One is Native UI basically the web page of ServiceNow. Second is Mobile apps. Third is Service Portal. All the data is same in all these types of interfaces.

- ➔ ServiceNow provides 3 different Apps. They are ServiceNow Agent, Now Mobile, ServiceNow Onboarding. Administrators can use them as per the business requirement.

- ➔ The Native UI is the primary UI. It is the best used on desktop and laptops and is accessed via a web-browser and the instance URL.

- ➔ We can access service portal by adding sp in the URL. After the domain name just add sp in the URL and you'll be redirected to service portal.



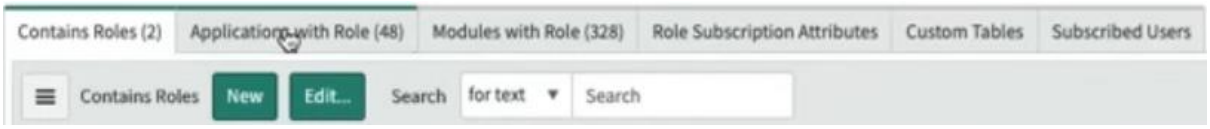
- ➔ These are all the options and many more are available in the service portal. But it mainly depends on the access that user gets.

- ➔ Roles are assigned to users depending on their functions and responsibilities they have. Not everyone needs to access all the information all the time, they only need some specific information which ServiceNow uses role-based access to provide them what's needed.

- ➔ ServiceNow has Users and Groups table. You can easily navigate them by typing them in application navigator. Users table show the data of all the users and each individual user has many fields like name, ID, Password, and many more details. Passwords can be set by admin and after setting password user can access instance by logging in using those credentials.

- ➔ In the similar way groups table shows all the groups that exist. Each group shows all the information in detail. It displays Group name, description, manager, roles and group members.

➔ Roles table also displays all the roles which are out of the box in ServiceNow. Each role describes its name, application (whether it is global or local), description, and it has other options as well which are shown in below image.



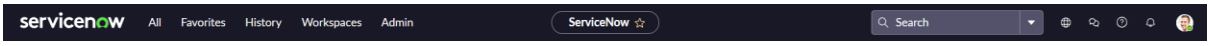
➔ Roles can be assigned to individual users or groups. But the best practice is to assign roles to groups rather than assigned them to individual users.

➔ ServiceNow provides one of the useful feature called ‘impersonate user’. You can get the view of any user if you have proper permissions and rights.

➔ If a user encounters an issue, the "impersonate user" feature would be very useful. It allows us to see exactly what the user sees, enabling us to identify and resolve the problem more easily.

➔ UI’s has versions, users can switch to versions depending on which UI they’ll work. Go to settings and you can find switch UI button, when clicked it changes UI version and in the same way you can go back to previous version as well.

➔ ServiceNow main screen is divided into 4 main elements. That are banner frame, Application Navigator and content frame.

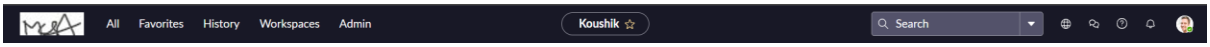


➔ Banner frame appears at the top of every page and contains logo, user menu, system settings, global search, favourites, notifications, chat, help and history.

➔ Application Navigator which provides easy access to ServiceNow modules and application instances

➔ Content frame shows the information of application or module selected, you’ll see form, list and other pages in content frame.

➔ Below image shows example of branding , I changes banner image and name tag

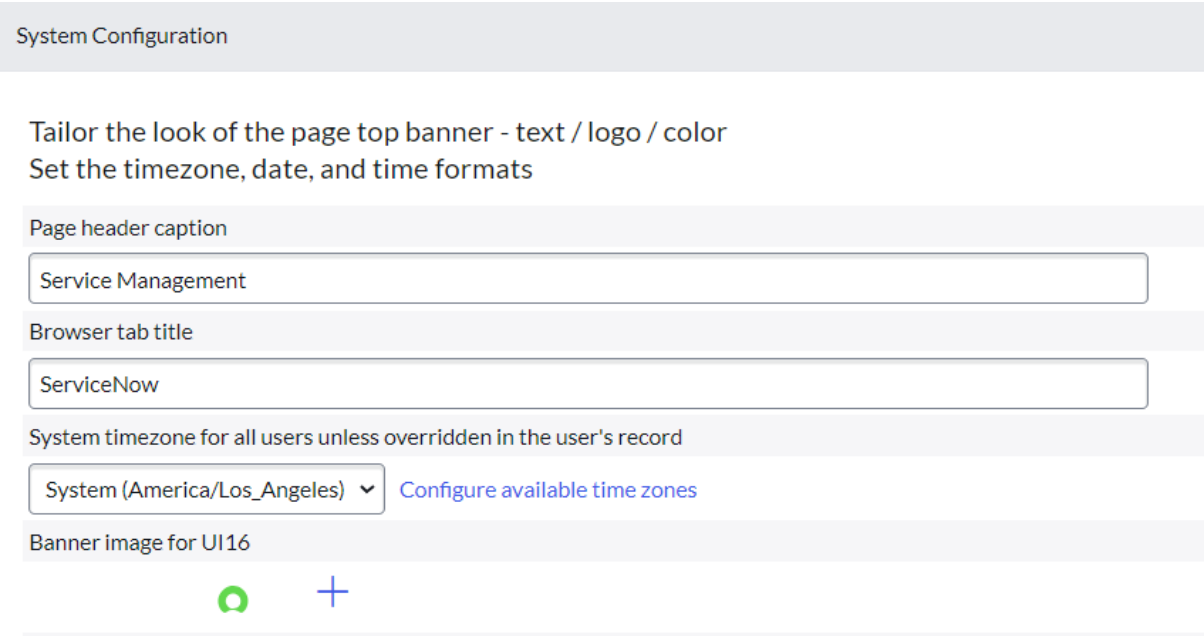


➔ Branding in ServiceNow refers to the customization of the platform's user interface (UI) to reflect an organization's brand identity.

➔ This involves modifying the look and the ServiceNow instance to align with the company's colours, logos, fonts, and other visual elements.

➔ Branding enhances user experience by providing a familiar and consistent visual environment across the platform.

➔ In the application navigator, navigate to UI16 you can see the below image the UI looks this way. In this you can enhance your branding by adding your organizations name, caption and adding a banner image. We can also enhance colours according to organization.



➔ For the Now Platform UI, navigate to System UI > System Properties > Basic Configuration UI16. Here you can change the banner, colours, and logo properties.

➔ In ServiceNow we change also change the login page , Welcome page and also the text in it.

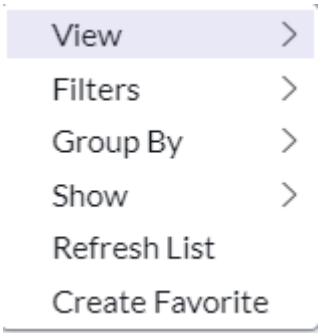
➔ Lists in ServiceNow is a table of records, it might be any table. Example incident table which shows all the incidents that are available.

➔ There are several elements in lists. Those are title bar, filters, breadcrumbs, Column and Field, List Menu, List Context Menu.

➔ We select multiple filters for a list. Breadcrumbs shows all the filters that are selected.

All > Caller = System Administrator > Active = true > Universal Request is empty

➔ List Menu has several options such as view, filters, group by, show , refresh list, Create favourite. View is used to change the view of the table. Filters are used to apply any filter. Group By is used to group the data using any field or column. Show is used to change the records per page. Refresh List is used to reload the page. Favorite is used to make any filter as favorite.



➔ List context menu also has multiple options such as sorting, shows visual task board (VTB), bar and pie chart, etc.

➔ When we use a filter on regular basis, ServiceNow provides an option where we can save that filter and also, we can enable it to group, everyone or just you. This saved filter will appear in the Filters option.

➔ ServiceNow allows you to configure branding settings for multiple languages to ensure a consistent experience for global users. This can be set up using the Multi-Language Support feature.

➔ Forms in ServiceNow are the primary way users interact with data. They provide a structured interface for entering and managing records within tables.

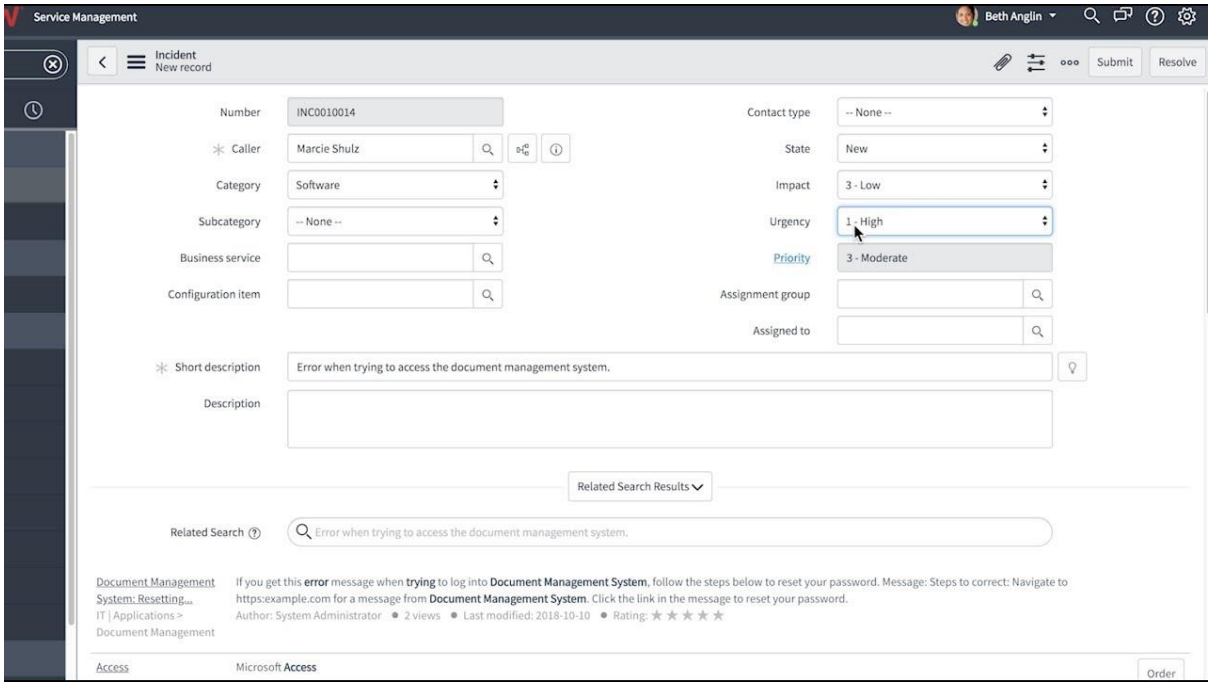
➔ Forms in ServiceNow are essential elements that allow users to interact with records and data within the platform.

➔ There are many different forms in ServiceNow, but these forms have similarities such as header, fields, sections, lists. Below image shows that in detail.

➔ While creating a new record there are 2 options to insert that record into table. Submit and Save there are both the options that can be used to create a new record.

➔ Both the functionalities are different from each other, submit saves the record and exits the form where as when saved that record the record will be inserted into table and also the form wont exit, it still displays the form.

➔ The red Asterix mark indicates that as mandatory field, in the below image you can observe that.



➔ Forms also have activity streams where they show what are all the changes made by a user or administrator and the timeline.

➔ There are 2 ways to change the form configuration. Form layout and Form Design. This is only available for admins. Form layout is used to create new fields and adjust the layout of the form.

➔ Form Layout determines the order of fields and sections displayed on the form. You can add new fields, rearrange existing fields, or create form sections to group related fields.

➔ In the Form Designer, you can drag and drop fields, add sections, and configure various UI elements like form views, tabs, and field types (e.g., text, choice, date).

➔ Forms can include related lists at the bottom of the page to show related records from other tables. This helps provide a comprehensive view of related data, such as tasks related to an incident.

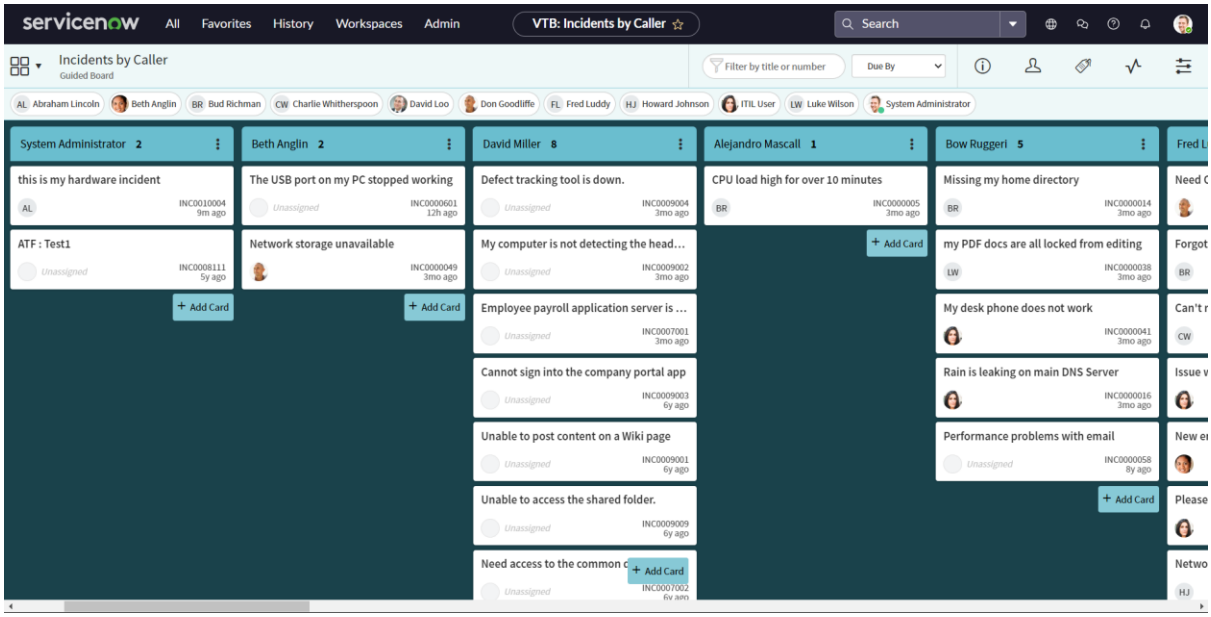
➔ Toggle template bar when clicked a template bar will be displayed in the bottom, which shows all the templates that are available, where we can also add our custom templates.

➔ When incident call type template is applied, some fields are automatically filled. If there's a requirement of creating new forms every day or frequently, using these templates makes it useful.

➔ Visual Task Board (VTB) is a drag-and-drop interface that provides a visual representation of tasks. VTBs allow users to manage tasks more intuitively by dragging tasks between lanes (columns) representing different states or priorities.

➔ Users can customize boards to suit their needs by adding or removing columns, changing labels, and configuring card types. Just by dragging them you can do many things.

➔ Tasks can be integrated with the Knowledge Base to provide helpful articles and documentation related to the task at hand. This helps users resolve tasks more efficiently.



➔ ServiceNow provides out-of-the-box notifications for task updates, assignments, and state changes. These notifications can be customized or created as per business requirements to keep users informed.

➔ ServiceNow includes powerful Reporting and Analytics tools that allow you to track task performance, workload, bottlenecks, and SLA compliance. Dashboards can be configured to provide a visual representation of key task metrics.

➔ Task Management in ServiceNow involves creating, assigning, tracking, and managing various tasks within different workflows, such as Incident Management, Problem Management, Change Management, and other business processes. ServiceNow provides several tools and features for effective task management across departments and teams.

➔ In ServiceNow, a **task** is any record that requires action and can be assigned to a user or group. Tasks are records in the task table or any child table that extends from it, like Incident, Change Request, Problem, Request, etc.

➔ You can track the progress of any task assigned to a user or group.

➔ The three most common tasks in ServiceNow are Change Request, Incident, and Problem. These three are different tables and are extensions to the task table.

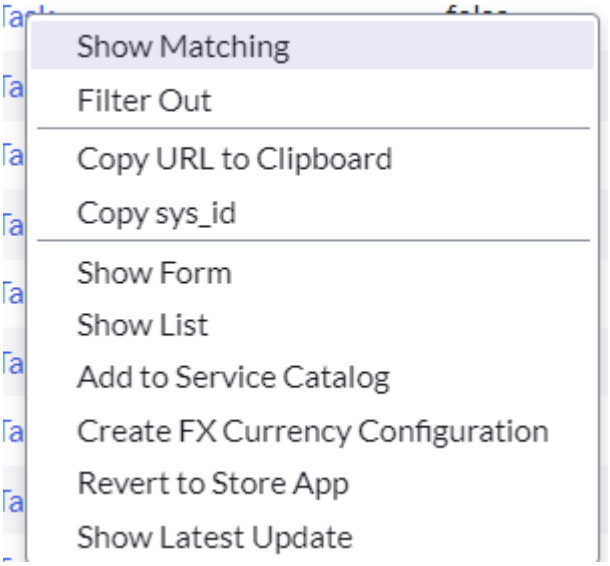
➔ Tasks can be assigned to an individual User or a Group of Users. We can assign tasks using Assigned to or Assignment group. Tasks can be automatically assigned or can also be manually assigned.

➔ Users can create, view, edit, and manage tasks directly from various applications or modules. For example, an Incident can be created under Incident Management, or a Change Request can be created under Change Management.

➔ Functionalities associated with tasks are Approvals, Assignments and SLA(Service Level Agreement).

➔ When opened tables in ServiceNow you can see all the tables that are there in that instance. You can also see a column called extends table, which basically tells you that from which parent table it is extended.

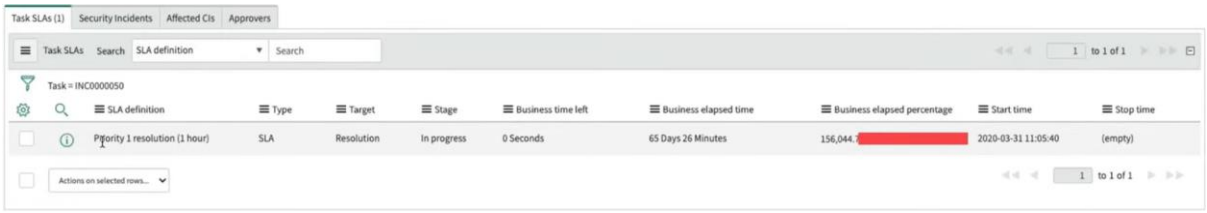
➔ If the extends table is task, then all those tables are task tables. One can easily navigate to all the task tables just by right clicking on Task and show matching.



➔ Show matching shows all the tables that are of task tables, this is an easy way to find all the task tables that are there.

➔ Every task must be assigned to a group and user. Where user has to a part of that group.

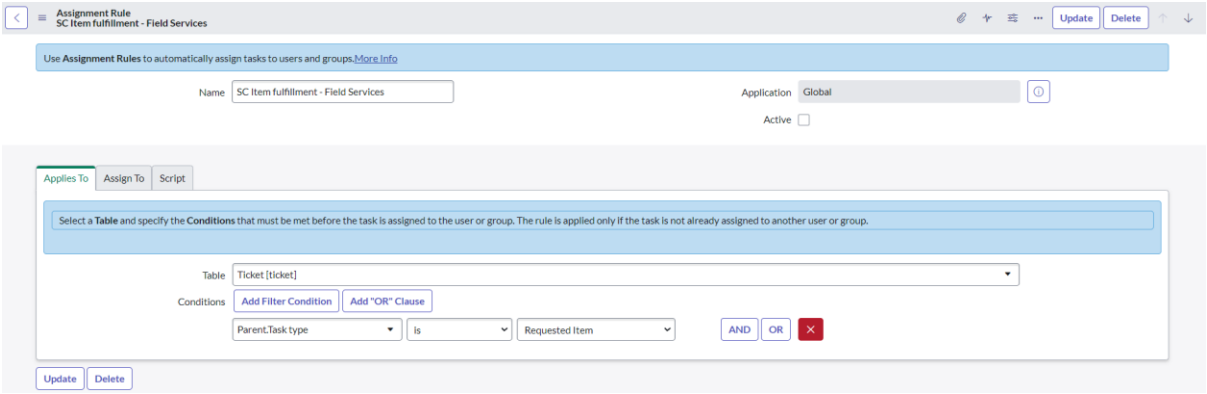
➔ Tasks have SLA's, in general terms with an example, consider there's a request you to solve all the incident tasks within 1 hour. If those tasks are not solved in given time, then the SLA will show as breached. So, it's just another condition to perform any task.



➔ Taks can be assigned in many ways such as manual assignment, using assignment rules.

➔ We can manually select the groups or users for a particular task and can assign that task.

➔ Navigating to Assignment and then selecting the table and conditions for which we are assigning, then selecting user and group.



➔ Above image clearly shows how the conditions apply and next to it we have assign to where we select users and groups for assigning. Scripting is advanced methodology. If the conditions won't follow, then we write script.

➔ Service Desk in ServiceNow shows different modules such as Callers, incidents, Knowledge, My Work, My Group Work, My Approvals.

➔ Callers: View and manage user profiles and contact details to provide personalized support.

➔ Incidents: Track and manage incidents reported by users to restore normal service operations.

➔ Knowledge: Access a repository of articles and solutions to help resolve common issues and enable self-service.

➔ My Work: View and manage tasks and incidents specifically assigned to the logged-in user.

➔ My Groups Work: Access tasks and incidents assigned to any group the user belongs to, facilitating team collaboration.

➔ My Approvals: Manage and process pending approvals for changes, incidents, or service requests.

➔ Even if we send an email in a form, it is also captured in the activity stream, it only appears when the user notifications are enabled. It won't show in activity stream if the notifications are disabled.

➔ Outbound Notifications are communications sent from ServiceNow to users or external systems to provide updates, alerts, or confirmations. These are typically automated and triggered by specific events or actions within the platform.

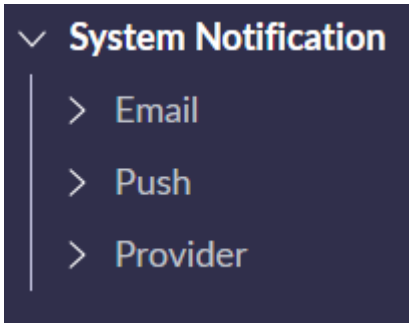
➔ ServiceNow can send email notifications to users when certain conditions are met (e.g., incident creation, task assignment, change approval, SLA breach).

➔ Outbound notifications can be configured to send different types of messages, such as status updates, approvals, escalations, or task completions.

➔ Inbound Notifications refer to incoming communications that ServiceNow processes. These are generally triggered by external users or systems sending messages to ServiceNow. Inbound notifications are mainly associated with Inbound Email Actions.

➔ Common actions include creating or updating records (e.g., incidents, change requests), triggering workflows, or even replying to the sender with a confirmation or status update.

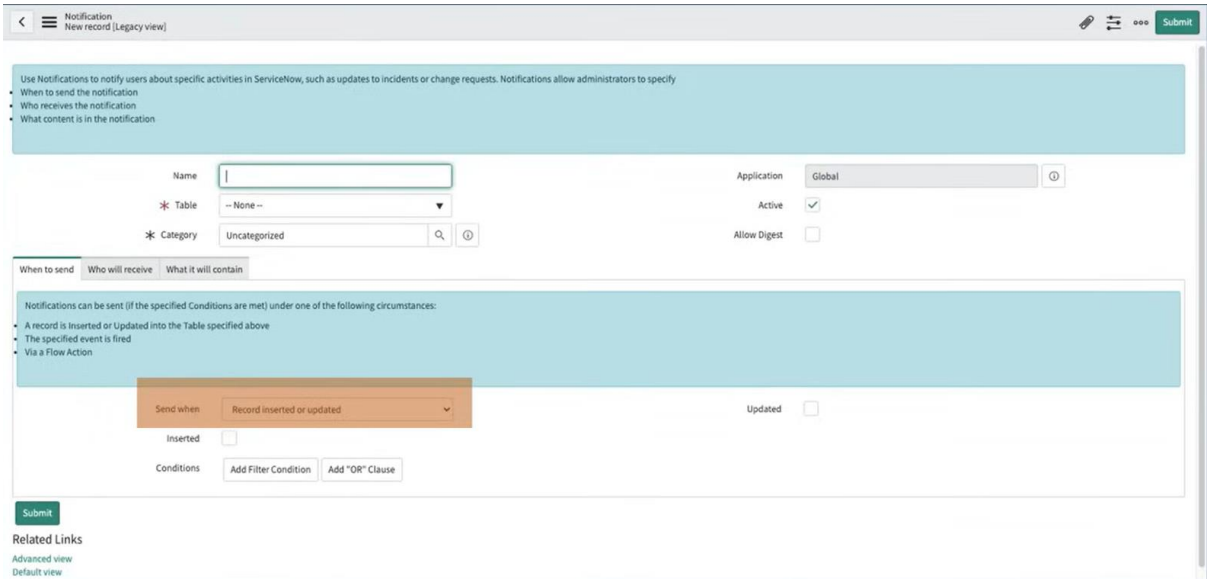
➔ ServiceNow provides an out of the box module to manage notifications, users can create and manage different types of notification with this module.



➔ Email has Digest Intervals which controls the number of notifications a user should receive within the time period we mention, Notifications where we can see all the notifications that are available in the instance and also we can add new notifications.

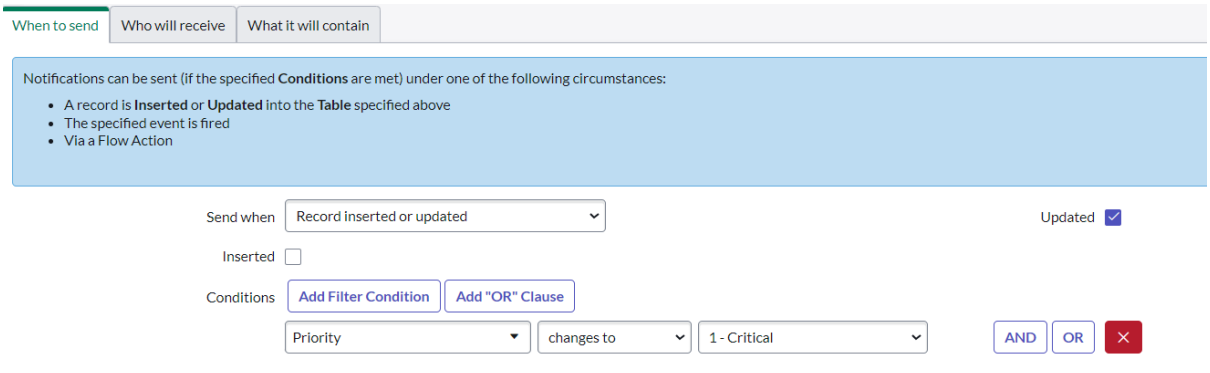
➔ Notification Email Scripter, it is used to send emails using scripts. Notifications category shows all the categories of the emails in the instance.

➔ Notification Form: ServiceNow admins can create new notifications using Notification Form.



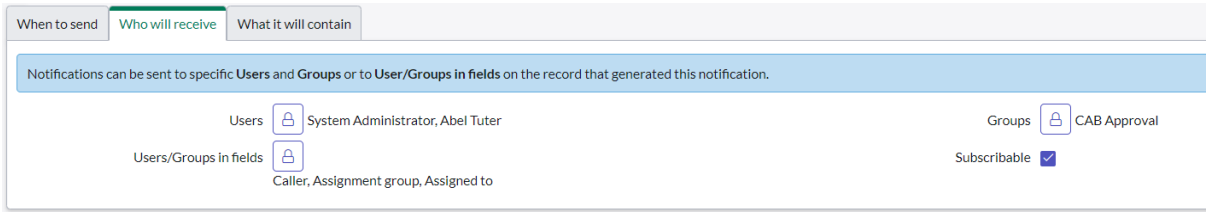
➔ Above image shows the Notification Form and its fields. We have several fields like name, Table, Category, Application, when to send, who will receive, what will it contain.

➔ When to send is used to set the conditions when the notification has to be sent.

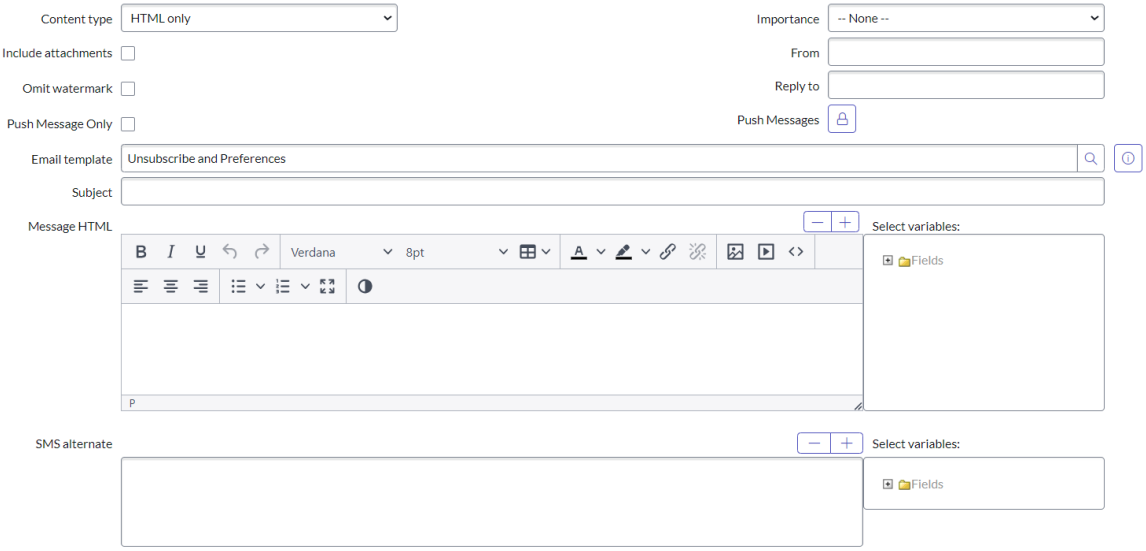


➔ Who will receive is used to set the groups and users who has to receive this notification.

➔ The checkbox for "Subscribable" is selected, indicating that additional users or groups can subscribe to this notification.



➔ What it will contain is used to describe the notification body, where we can set message text, email template, and also variables. ServiceNow provides another feature called variable. Where we can select a particular field value and then include that in message which will be like ${number} it'll automatically send the incident number with the mail.



➔ ServiceNow admins can configure inbound action for different tables to create or update records of those tables when email is received.

➔ Inbound email action form contains name, target table, action type, stop processing. The conditions for this form are when to run, actions, Description.

➔ There are 3 types of emails. New, Reply and forward. We can trigger the inbound action based on these mails.

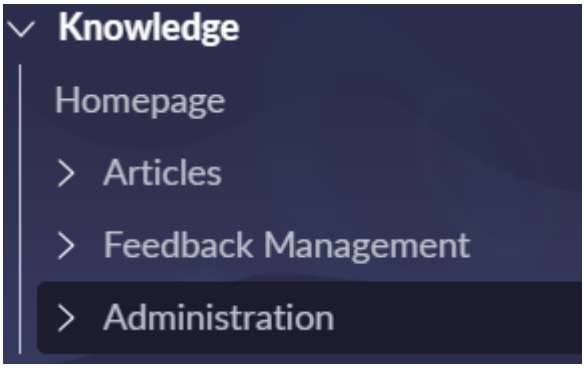
➔ A Knowledge article is a record in Knowledge base that provides information to users. A knowledge article can be a policy, self-help tips, troubleshooting and resolution steps.

➔ Knowledge Management provides a single place for creating, categorizing, and viewing articles. Where users can easily access information and find solutions to their queries. This reduces the time spent searching for answers across various resources.

➔ Knowledge Management systems store valuable information in structured(categories) knowledge bases, making it easy to manage, retrieve, and share data. This enhances the organization's ability to leverage existing knowledge effectively.

➔ To manage knowledge management and its process, ServiceNow has out of the box knowledge management application, users with right roles can access this application and the modules under this application. They can create and maintain knowledge articles in this.

➔ Knowledge application has following modules included in it. Homepage where it shows all the knowledge articles that are available.



➔ Articles module where we can create an article. Import an article, and we can also see the articles which are published, unpublished, Retired.

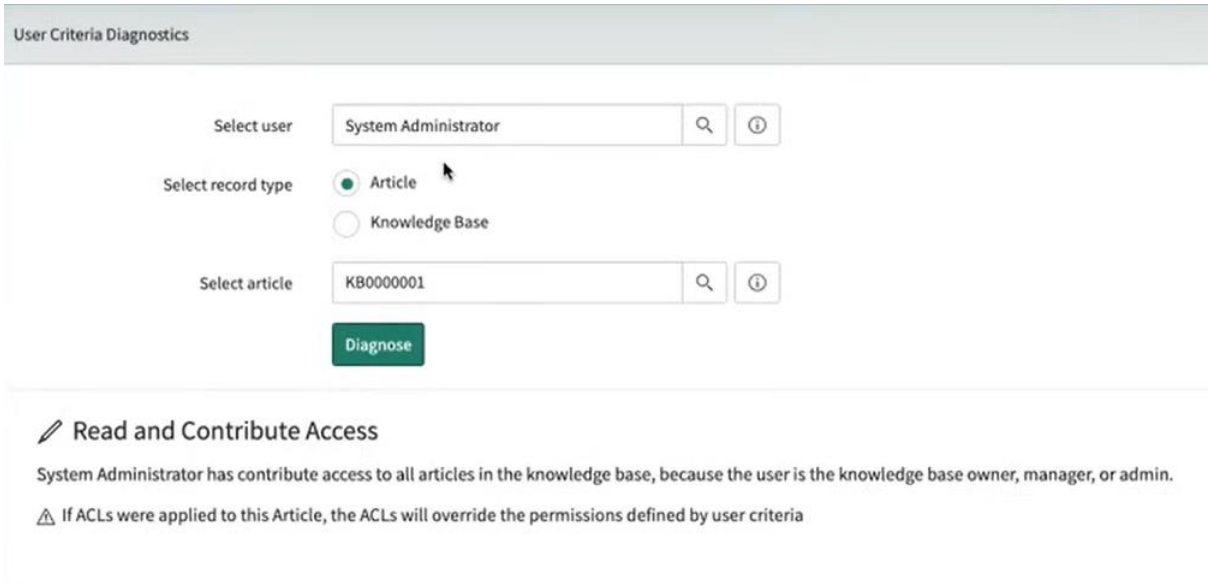
➔ In Feedback Management we can see all the feedbacks that are given by user, flagged articles, and all other tasks which are opened, closed.

➔ In Administration we have Guided setup, this module provides step-by-step guidance to implement knowledge management in your organization.

➔ We also have knowledge bases , this module shows all the knowledge bases that are created in your instance, users can also create knowledge bases for different departments as well.

➔ Ratings, Search log, Navigation Add-ons, Messages, Properties, Overview, User Criteria these are also the part of Administration. Overview is a dashboard where it shows analytics of all the articles.

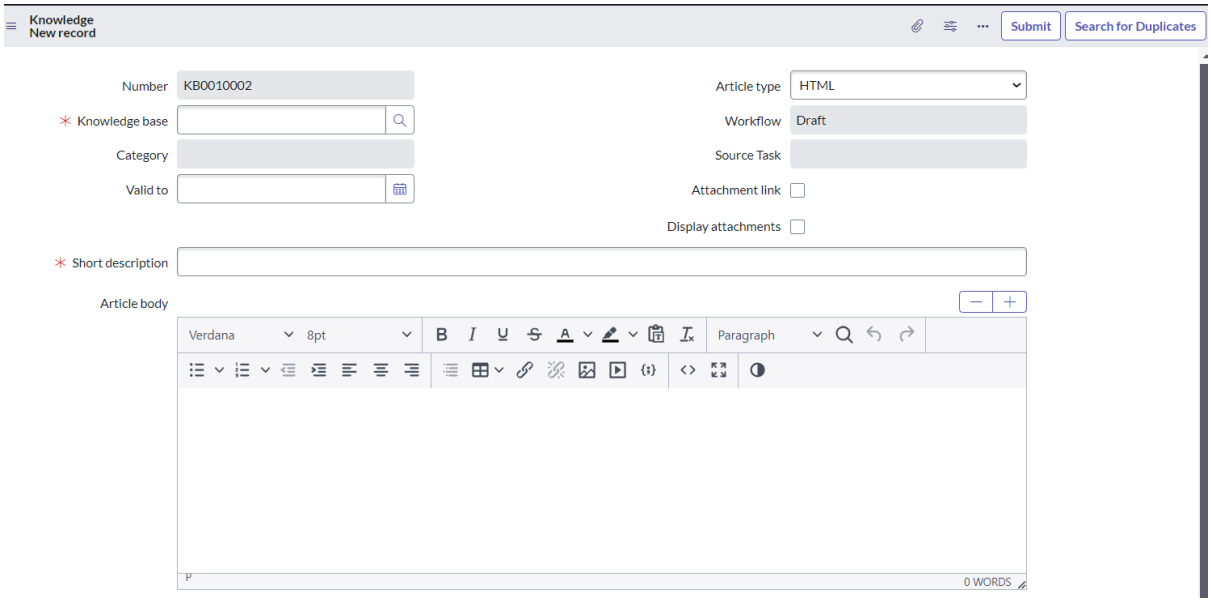
➔ User Criteria Diagnosis , what kind of access control is applied on knowledge base or an article to any user.



➔ Above image shows us how actually diagnosis work. We selected a user and record type. Then the article which is needed to be diagnosed and that's it. It shows the access that user has.

➔ Knowledge base is also available in-service portal, which is similar to the native UI knowledge base. Knowledge base can also be customised based on requirements. We can add or delete icons and all related to that.

➔ To create a new article in knowledge base we have knowledge form which is used to create an article.



➔ There are several fields in Knowledge Form. Name, it is autogenerated. Knowledge Base, where we specify which part it is. Category. Valid to, Article type are other fields in this.

➔ Attachment link is another useful ServiceNow feature, where if we click on checkbox, it creates a link where when presses it downloads the entire article.

➔ Display Attachments, if some attachments are required to show in article, then we can add them using this.

➔ Short Description, this will be the title of that article that user is creating. This will be displayed in bold and bigger text.

➔ Article body, this is the actual article that appears in the knowledge base. Here we provide the text related to the article.

➔ Knowledge management also has some roles, by which users are restricted to just some functionalities. These makes the knowledge bases more efficient.

➔ Knowledge management has a workflow, an article cannot be created without any review to ensure that its in the correct format.

➔ When a user creates an article it goes it draft stage, now user can push to the next stage by clicking on publish button then article will be sent for approval.

➔ This approval goes to knowledge base managers or owners, or it can be routed to different approvers as per the workflow creates by the organization.

➔ Service Catalog is a request ordering system. To request services and products offered by different departments of any organization.

➔ Service Catalog is a one stop solution to request different services provided by all the departments in the organization. All the services are categorized which helps users to choose the right service.

➔ Multiple catalogs can be implemented by an organization. Multiple requests can be raised and also can track that, when exactly you're going to get that service.

➔ In service catalog application, where users with catalog roles can edit, create and manage services, products and different configurations provided by different departments.

➔ Services and products of service catalog are divided into logical groups which are called categories. For example, we have hardware, software, office etc.

➔ Service Catalog has 3 major components. First one is order, which is a service or a product requested by user. Then we have order form which is submitted by the user after filling up the form. The last one is ordering process which works when the request is created.

➔ Each catalog item can have a predefined workflow that handles the process from submission to fulfillment. Workflows can include approval steps, task assignments, notifications, and more, ensuring that the request is processed efficiently.

➔ There are 3 different tasks tables and records associated with service catalog, when user places an order to request service or product, it creates records related to the request.

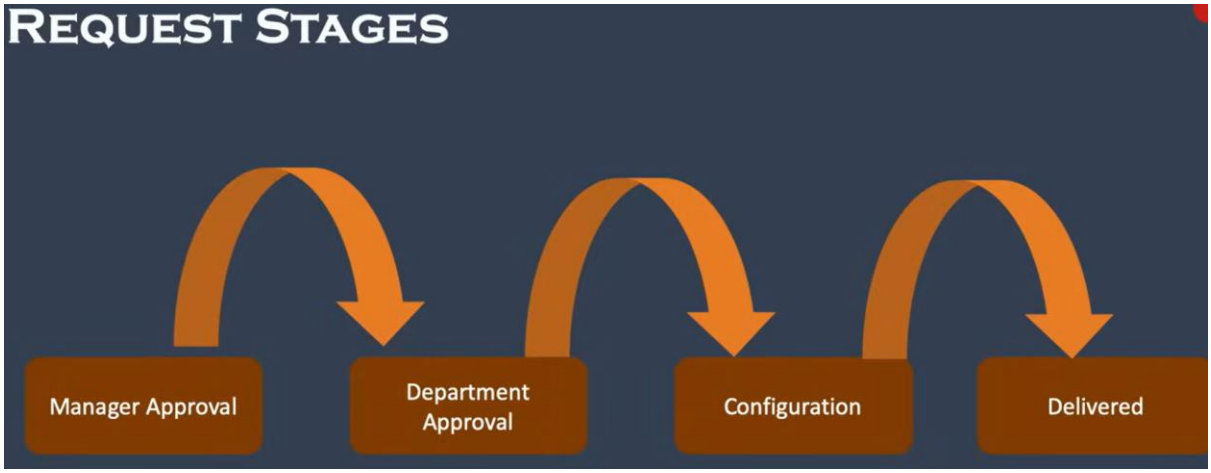
➔ When form is submitted it creates request record, which gets created in sc_request table. It is the first request when user submits the form.

➔ Next after submitting the form, item record is created in Request item table(sc_re_item). Every request item will have catalog task records(sc_tasks) each task is assigned to different teams so that they can process the request.

➔ Try it button in another useful feature by ServiceNow , this exactly shows us the form which is visible to the user.

➔ When request is submitted by the user, then users can track the request by knowing the stages of the requested item.

➔ These stages can be created and customized by the admins for different items as per the requirements.



➔ These stages are not static they are dynamic, if you have access in ServiceNow service catalog you can also create these stages.

➔ A Variable in ServiceNow is a field that is used to gather user inputs when submitting a request. Each variable is associated with a catalog item and can collect different types of data, such as text, numbers, dates, or selections from a list.

➔ A Variable Set in ServiceNow is a collection of variables that can be reused across multiple catalog items or services. Instead of manually adding the same variables to multiple catalog items, you can group them in a variable set and then apply the set wherever needed.

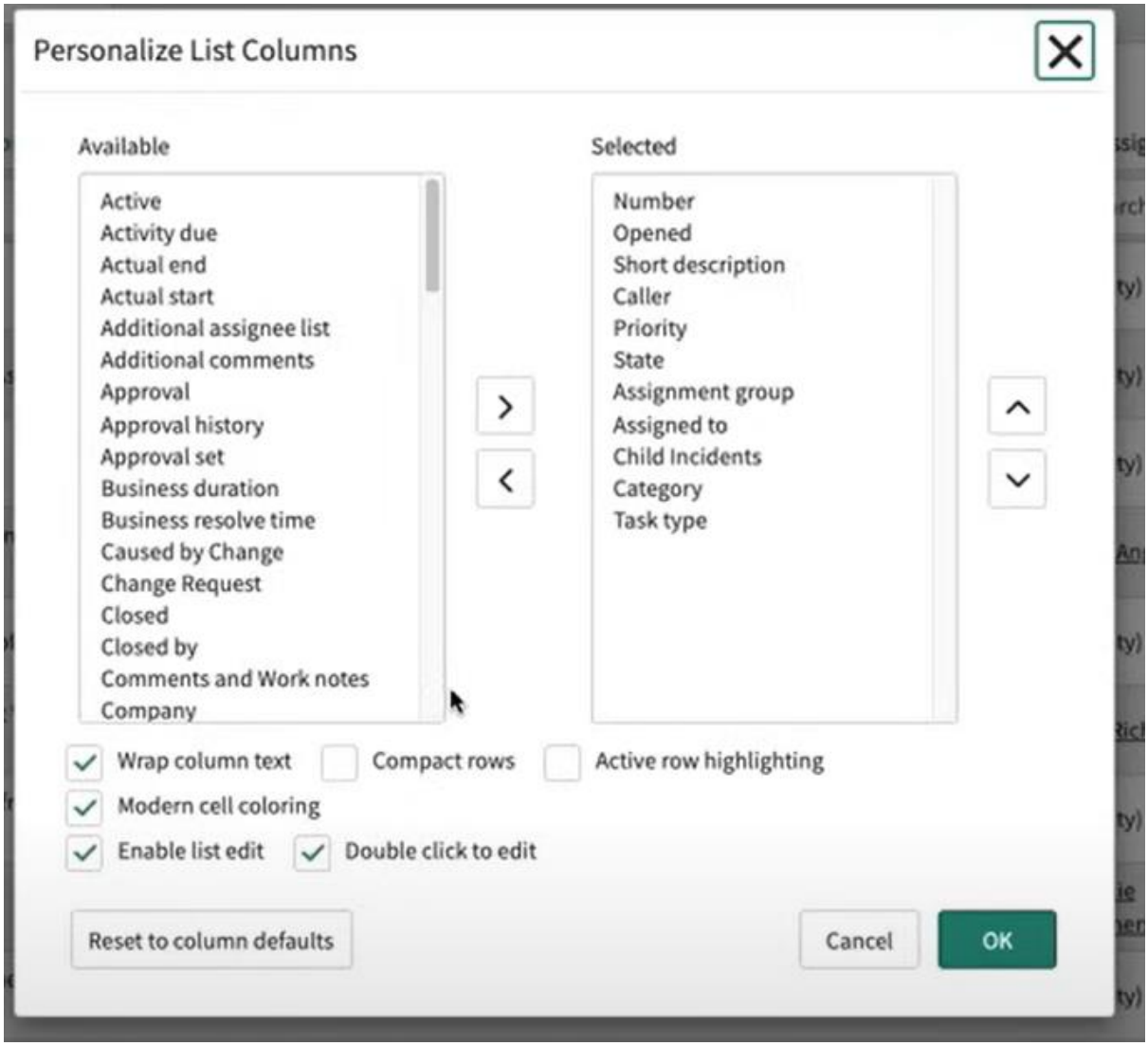
➔ Everything is managed via database structure in ServiceNow. We have tables which is a database component which stores records.

➔ Records: These are stores in a table and these records have fields which show information about that particular record.
➔ List: Which shows the records in a table.

➔ To get all the tables that are available in the instance, you can simply navigate to system definition module and under that you can find tables module which shows all the tables.

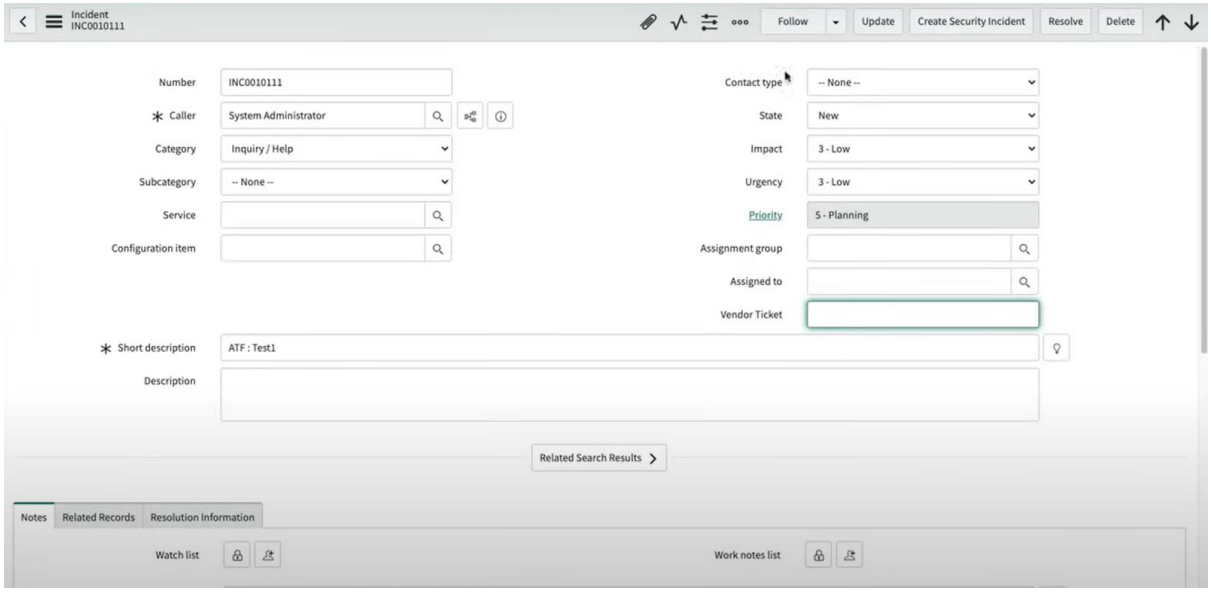
➔ A table is a collection of records in the database, where information can be entered. Tables have individual rows which correspond to a record. Table columns correspond to a field on record or row.

➔ Field is a column of table which stores the actual data. It can be of different types like string, choice, reference, true or false. The major attributes of fields are field label, field name and field value.



➔ Users can personalize the view of the list , they can add or remove the columns and can do indexing to them.

➔ Below image shows fields of a record, where the actual data is located.



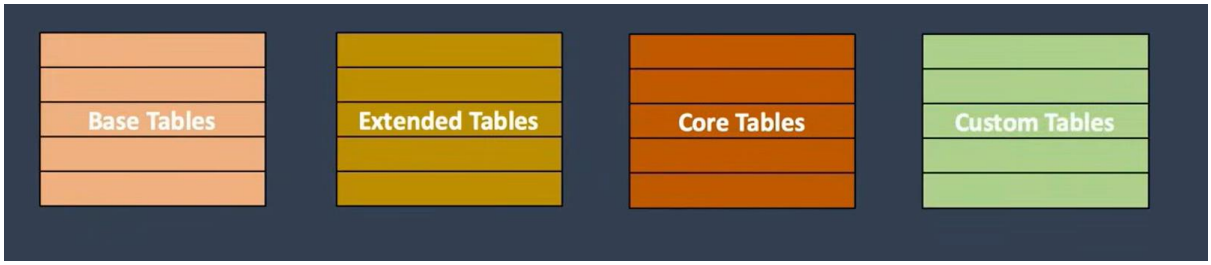
➔ As there are many tables in ServiceNow these tables can also be related to each other in different ways. First relationship is ‘one-to-many’. Second is ‘many-to-many’ and third is ‘extended’.

➔ One-to-many: one table has some fields which are referencing to other tables, like reference fields. For example, the Assigned to field in the Incident table refers to a user in the User table.

➔ Many-to-many: two or more tables are related to each other, for example groups and roles table which are related to group roles table.

➔ Extended: where one or more tables can be extended another table also inherit the fields from parent table. For example, the Incident table is extended from the Task table, meaning it inherits fields such as State, Assigned to, and others from Task.

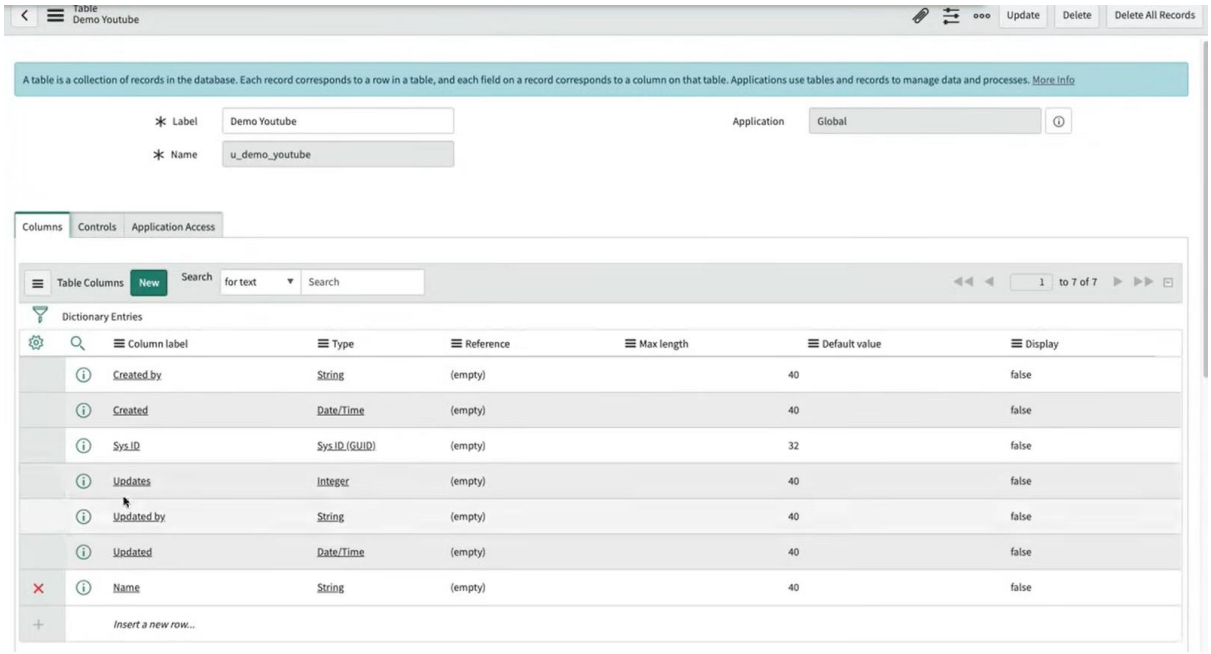
➔ ServiceNow has different types of tables , which are categorized as per the structure they have. ServiceNow administrators and developers need to know which type of table they want to meet the business requirement before starting the development or creation of that table.



➔ Above image shows us the types of tables. A base table is a table which is not extended from any other table. A table which is extended from another table is called extended table. Best example is change, incident tables are extended from task table. Where task table is called parent class, change and incident are called child class.

➔ Core tables are created by ServiceNow which are in the base system and custom tables are the tables which are developed by the developer or administrator.

➔ Examples of core table are Task, Change, Incident, and for custom tables they are leave, mapping table, matrix table.



➔ While creating a new table, when you give label then the name will be auto filled by adding u at the starting. Also, the columns are created by default, in the above image you can see that there are 6 columns which are added automatically expect the one with red cross mark this indicates that we have added this column.

➔ The Schema Map graphically represents the relationships between tables, especially showing parent-child relationships in extended tables.

➔ Schema Map shows inherited fields from a parent table to its child tables. This helps users understand which fields are available in child tables due to inheritance.

➔ The Schema Map provides visibility into customizations, including which fields, scripts, or rules are added to a table as part of a custom solution or modification.

➔ ServiceNow has different levels of security to access data into ServiceNow, this accessibility of data can be controlled by Access Control List (ACL).

➔ The first security is login which is basically controlled on the basis of user groups and roles. Login is mandatory to be authenticated before interacting with UI.

➔ The next level of security is applications and modules, which will be dependent on roles assigned to the user.

➔ Next comes tables and records, if user has access to the modules, then records of that table are visible, even though user should have access to those then only he can edit or view them. These accesses for tables and records are based on access control list (ACL).

➔ Access Control is a kind of security rule, which is defined to restrict the permissions of a user to interact with tables and records.

➔ Above image has some operations. In ServiceNow these are the operations you can apply restrictions using ACL's.

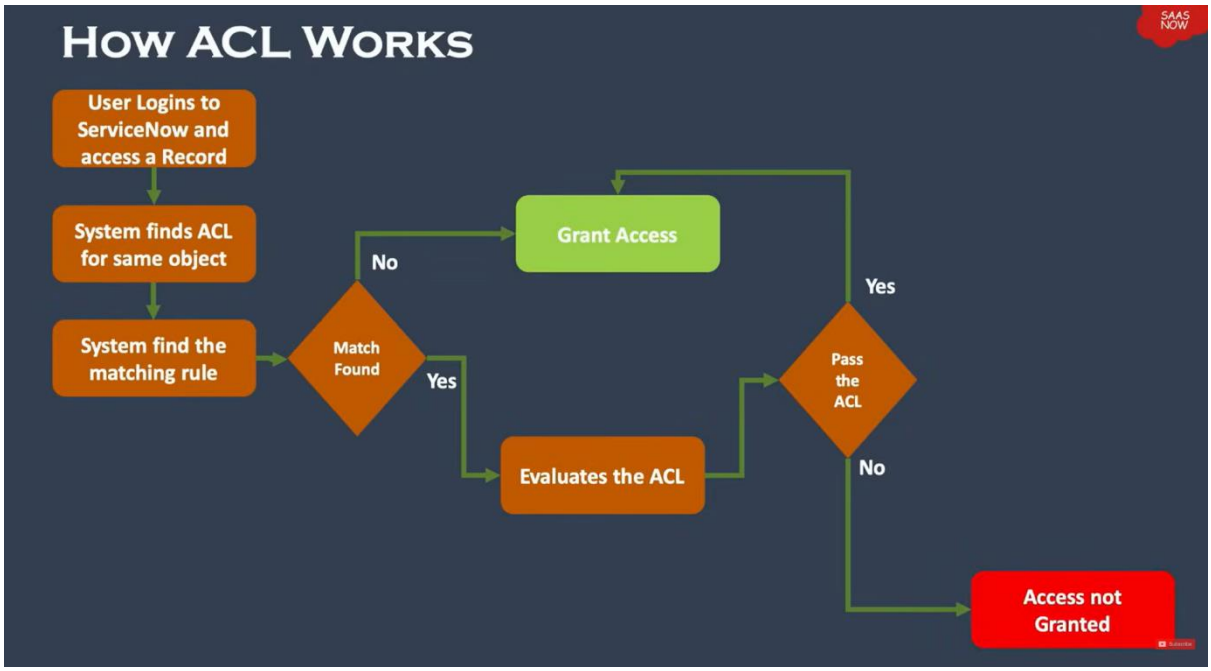
➔ Access control list is a list of defined rules, these rules are created at table, records and field level. ServiceNow system administrator can edit or add access control list rules.

➔ While creating an ACL user needs to provide name where it specifies which object needs to be restricted. There are 3 types of ACL.

➔ Table.None
This specific configuration means that the ACL applies to the entire table without specifying any fields. It controls access to the records in the table as a whole rather than individual fields or specific conditions within the records.

➔ Table.*
The asterisk (*) in this context means that the ACL applies to all fields within the table. For example, selecting "Incident.*" would imply that the ACL rule is meant to cover all fields in the "Incident" table.

➔ Table.Field
This is a combination where you specify a table and a specific field. For example, "Incident.priority" would be an ACL rule that applies only to the "priority" field of the "Incident" table.



➔ The above image shows how actually ACL works,flowchart clearly tells us how it goes through each step.

➔ When a user logs into the ServiceNow platform and attempts to access a specific record (e.g., an incident record), ServiceNow begins the process of evaluating whether the user has the necessary permissions.

➔ The system then searches for any ACL rules that are defined for the object (table or record) the user is trying to access. ACLs in ServiceNow are defined for specific objects, such as tables, records, or fields.

➔ The system then attempts to match the user's request with the defined ACL rules. Each ACL rule specifies conditions under which access should be granted or denied. If a matching rule is found, the system proceeds to the next step; if not, the system directly grants access because there are no restrictions defined for that object.

➔ If a matching ACL rule is found, the system evaluates the rule's conditions and scripts to determine if the user meets the requirements for accessing the record. This may include checking user roles, group memberships, or specific field values.

➔ If the user passes the ACL evaluation (i.e., meets all the defined conditions and requirements), access to the record is granted, and the user can proceed with their intended actions on the record.

➔ If the user does not pass the ACL evaluation (i.e., fails to meet the required conditions), access is not granted. The system denies the user access to the record, preventing them from viewing or modifying it.

➔ To create new ACL's or to edit you need to elevate your role to security_admin then you can create or edit ACL's. Elevating roles can only be done by admins.

➔ While creating a table if we tick the check box of access control users and assign roles, some of the access roles will be created automatically (CRUD).

➔ Users with the correct roles can only access any particular table or records, without roles they can't view tables. The UI shows Security Constraint prevent access to requested page.

➔ ServiceNow administrators and users can import data into different tables.

➔ There are many ways that you can use and import data into ServiceNow, some of them are import XML, in this we can only import XML files, and this is most used by developers and administrators. We can find this option in the list context menu. When clicked it will display a new screen where it tells you to upload, and we can upload it from local system.

➔ Another way of importing is ‘import’, we can find this in the same context menu. Import uses the functionality of import sets and transform map which is the core feature of import data into ServiceNow. Only excel sheet can be imported in this.

➔ One more way of importing is ‘importing sets’, in this data can be imported from various sources and map that data to different servicenow tables. A user with admin role or import_admin role can import data.

➔ ServiceNow has a separate application called system import sets which is used to manage the imports.

➔ Import set has some components, through which the data will reach the target table via these components. Components are Data Source, Import Set table, Transfrom map, Mapping Assit, Coalesce, Target Table.

➔ Data Source: a source of records from where the data needs to be imported it can be a file, an attachment, any kind of JDBC connection.

➔ Import Set Table is a staging area for records to be imported from source before data is inserted or updated into the actual target table. It stays in staging table which has fields related to source data and the name of this import set table is based on the filed you try to import.

➔ Transform map is used to create the relationship between fields of source data and target table.

➔ Mapping assist provides a visual way to map fields of source data and target data. Mapping can also be done automatically.

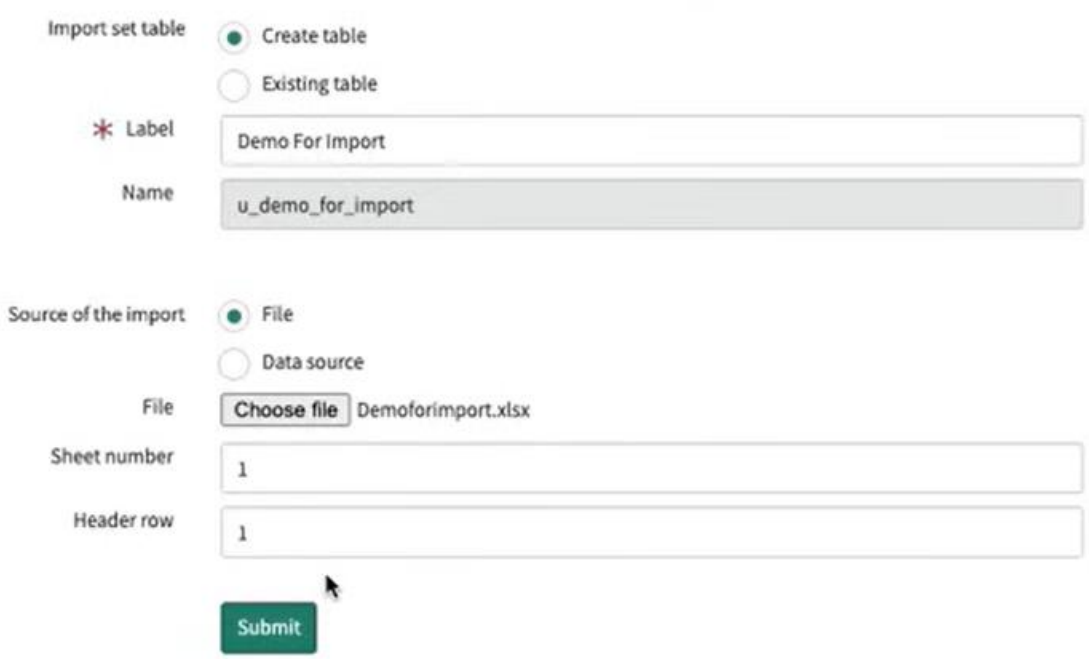
➔ Coalesce is used to trigger a check before importing of data, if it is enabled on field then system will for existing record, if match found then the same record is updated else new record is inserted.

➔ After all these processes data will be imported into the target table.

➔ Every data being imported into ServiceNow via Import set functionality it is always tracked. It creates a record and shows all the details about that.

➔ We can also check the progress and transform history and transform error of all the imports that have been done.

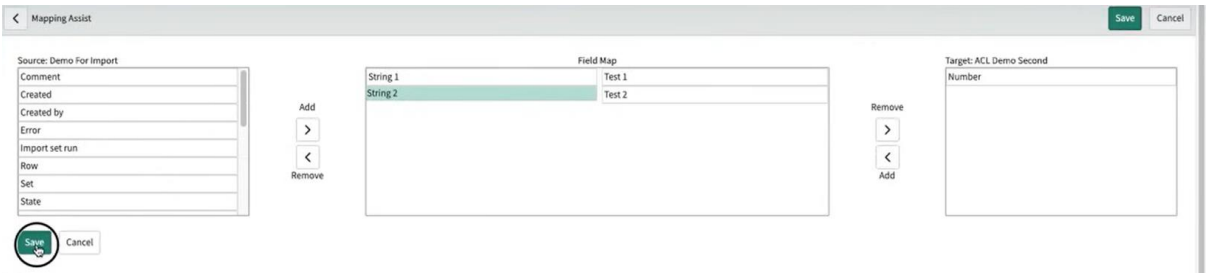
➔ To import any data using import set we have 3 steps included, first is loading the data, we can load the data by creating a new table or using an existing table after that attaching file or giving any other data source and then submitting it.



➔ This is how the UI looks like, where we selected option of creating a table and giving it a name and choosing a file from our local system.

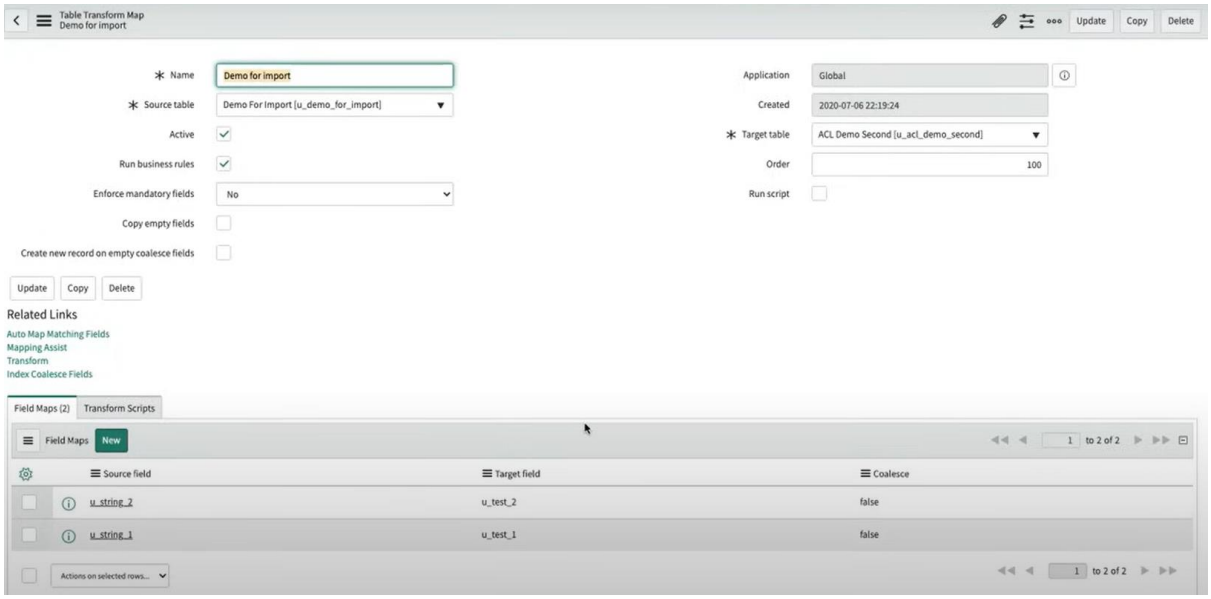
➔ After submitting the import process will begin and displays the status of that import.

➔ It also shows the next steps which are needed to be done. Now this data is imported to staging table. Next step is to create a transform map to send the data to actual target table.



➔ We have done mapping using mapping assist, you can observe that in the above image clearly where it shows the fields that are mapped to each other.

➔ We can confirm this in transform map and another important thing is we can only map the fields in mapping assit only after selecting the target table in transform map.



➔ Now after everything is done click on transform button which is available in the related links. After that we can see the table in the actual target table.

➔ If we select the coalesce field as true it checks whether the record is available or not and if yes it updates that instead of inserting a new record in the table. If there's no match, then it will create a new record. So, this is why coalesce is a very important feature in ServiceNow.

➔ Data Policies in ServiceNow are used to enforce data consistency and data quality across the platform. They help ensure that data entered into the system meets specific standards and requirements without requiring the use of scripts.

➔ You can define conditions specifying when the data policy should be applied. For instance, it can target specific tables, forms, or certain conditions on data fields. Set specific fields to be mandatory, read-only, or visible.

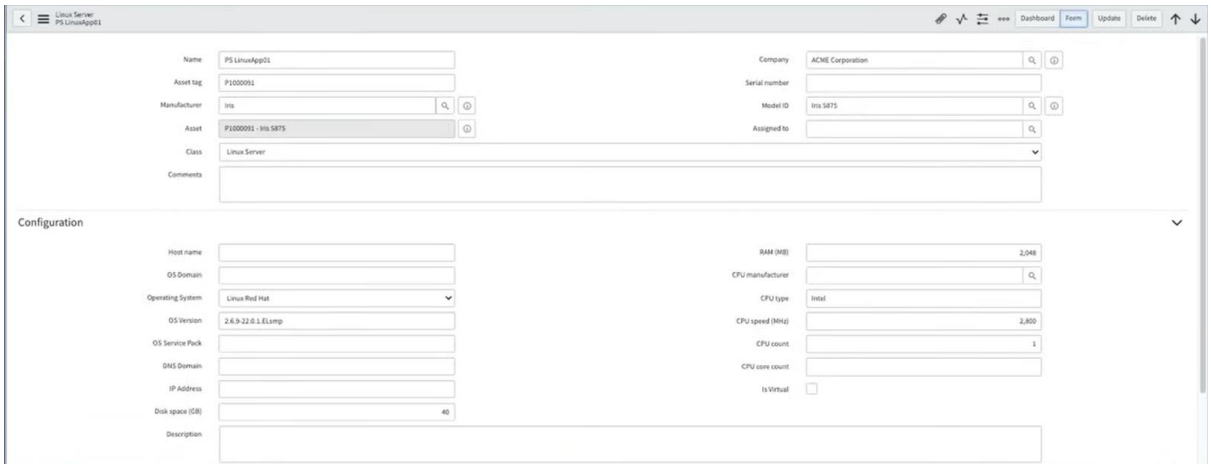
➔ If the data doesn't meet the policy's requirements (e.g., a mandatory field is missing), the system will prevent the record from being saved or imported and will display an error message or log the error.

➔ Data Policies in ServiceNow enforce data quality and consistency by defining rules such as mandatory or read-only fields. This helps maintain data integrity across the platform without the need for complex scripting.

➔ Configuration management database (CMDB). CMDB stores information about configuration items. It also stores the relationship between configurations items,

➔ Configuration Item (CI) is any component within an IT environment that needs to be managed to deliver a service. CIs can include hardware, software, network components, documentation, services, and even individual users or locations.

➔ ServiceNow has an out of the box application to manage CMDB called configuration. We have lot of modules under this application.



➔ CI Form in ServiceNow is a user interface used to view, create, and manage details of a Configuration Item (CI) within the Configuration Management Database (CMDB).

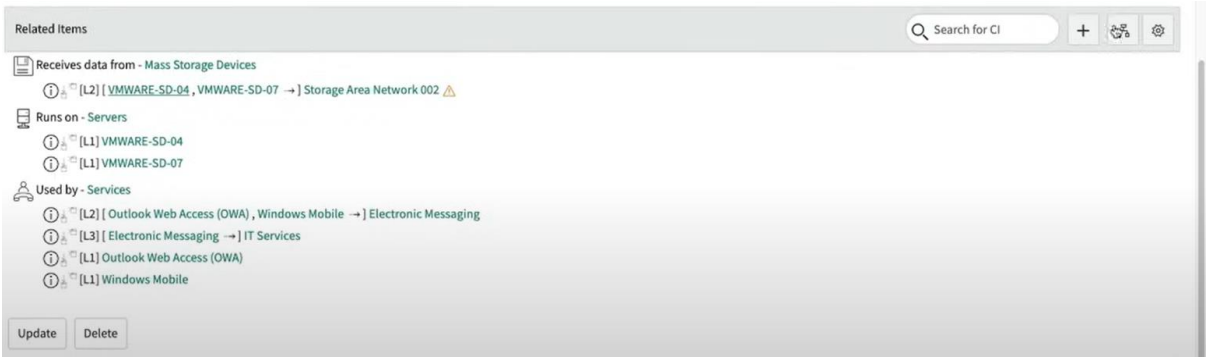
➔ The CI Form captures various attributes and properties of the CI, such as its name, type, status, serial number, location, owner, and other relevant information.

➔ We can also see more details about configuration item in CI Dashboard view. A CI dashboard provides real-time visibility into a project's build and test status.

➔ It typically shows build results, test outcomes, code quality metrics, and deployment status. The dashboard helps track progress and quickly identify issues. It often includes notifications, historical data, and pipeline visualizations.

➔ There are 3 key CMDB tables. They are Base Configuration Item, which is main parent table of whole CMDB. Next is configuration item, which is extended from base configuration item. The third table is CI Relationship table, which store the relationship between different CI Records.

➔ CI Class manager displays the entire CI class available in the instance in hierarchal structure like tree, showing all CI class definitions in one place. It is an easier way to view, modify or extend a CI class.



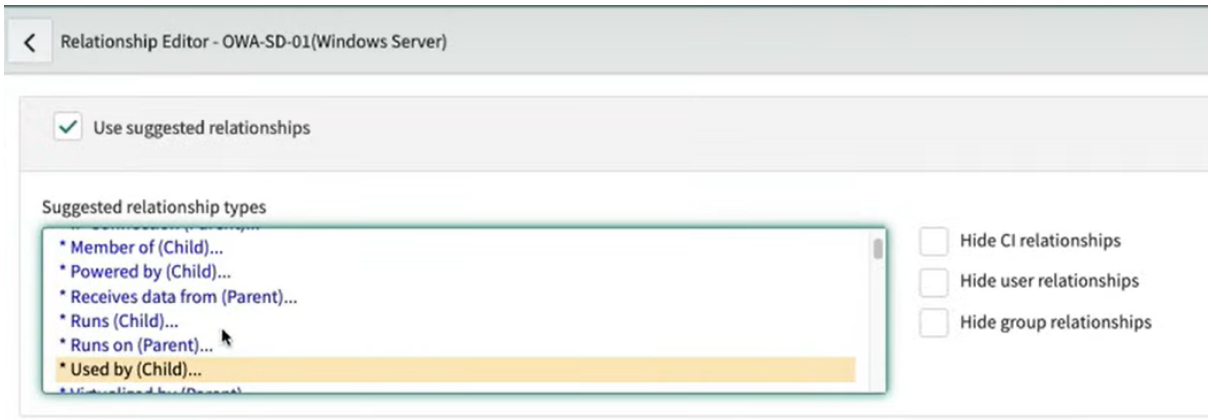
➔ Above image shows a section of a Configuration Management Database (CMDB) interface related to a Configuration Item (CI). It outlines the relationships and dependencies between this CI and other components in the IT environment.

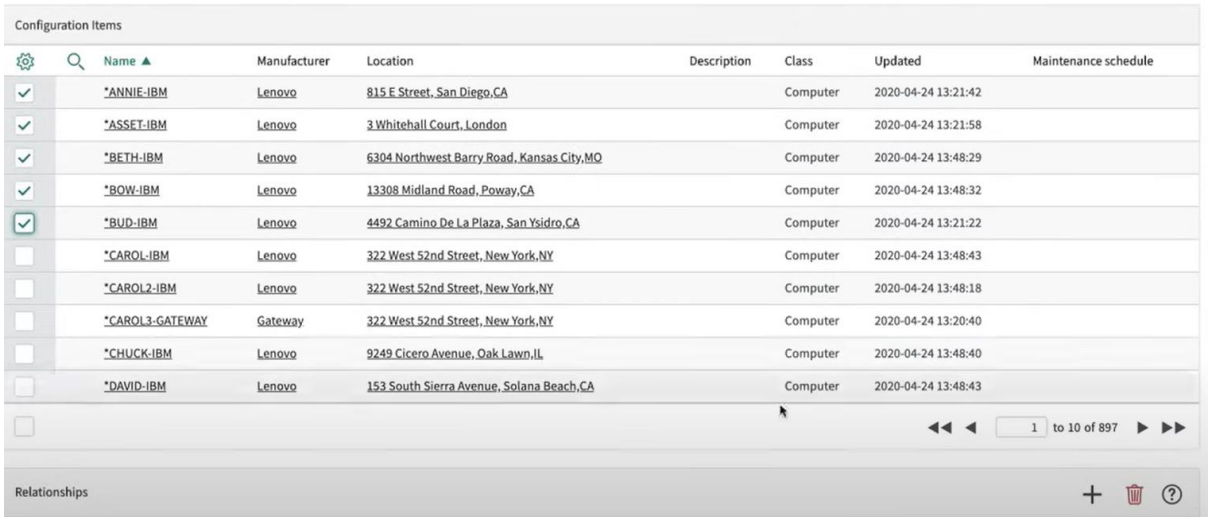
➔ Receives Data From - Mass Storage Devices: Indicates that this CI receives data from mass storage devices, such as storage area networks (SANs). The image shows specific storage devices or systems (e.g., VMWARE-SD-04, VMWARE-SD-07).

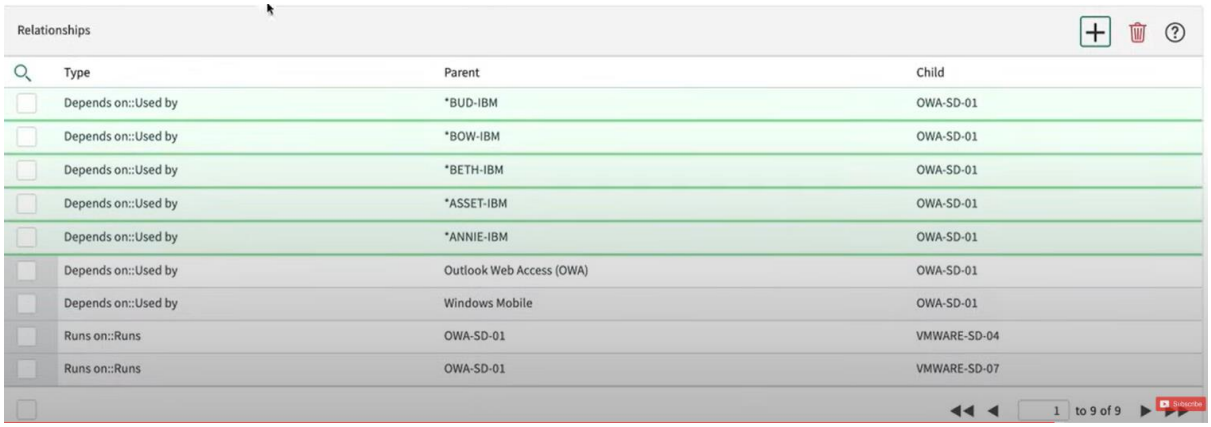
➔ Runs on - Servers: Lists the servers on which this CI operates. For example, this CI runs on servers like VMWARE-SD-04 and VMWARE-SD-07.

➔ Used by - Services: Shows which services depend on this CI. In this example, services such as Outlook Web Access (OWA), Windows Mobile, and Electronic Messaging rely on this CI.

➔ If you want to add any new relationship, you can just click on the plus button which on top right corner in the related lists field. A new screen relationship editor will be displayed.





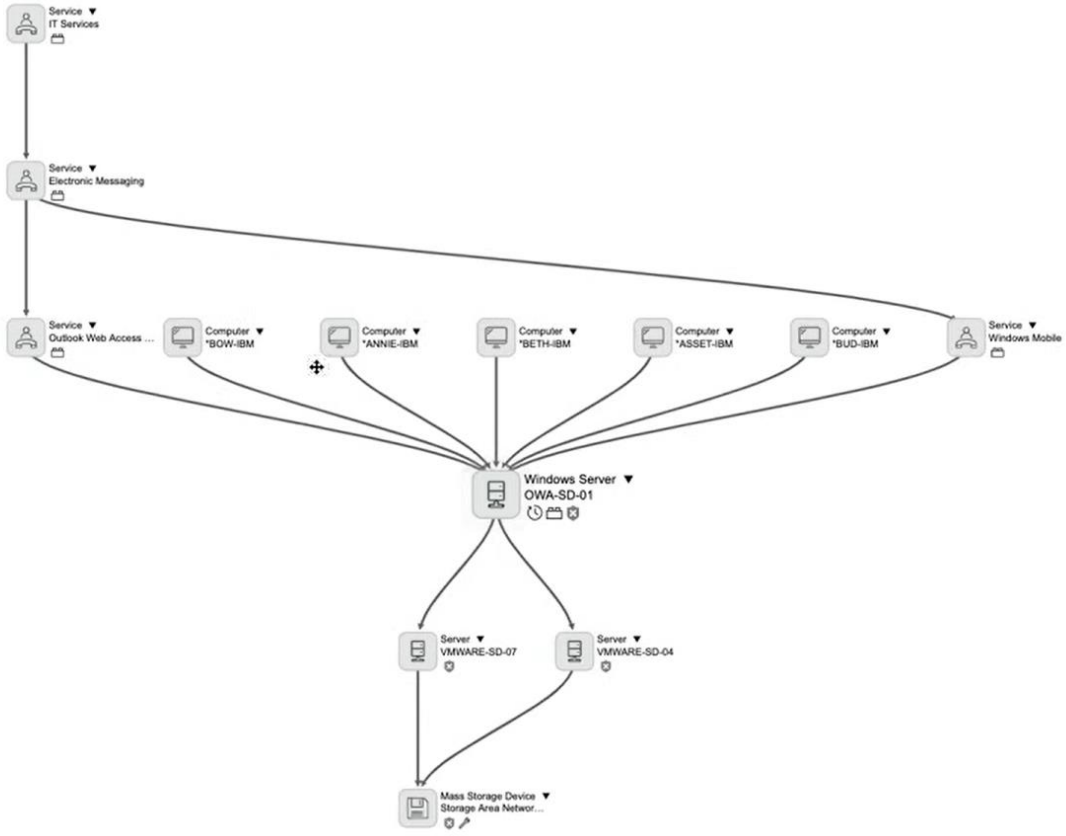


➔ First image shows which relationship we are selecting. In the second image we are selecting the items which are to be configured and in the third image those are added, and we can see it in the relationships table.

➔ A dependency view in a CMDB provides a visual representation of the relationships and dependencies between different configuration items (CIs) within the IT environment. It helps understand how various components interact, how they depend on each other, and the potential impact of changes or failures.

➔ To see dependencies open any CI and under the related items navbar you can see plus icon, right to the plus icon is dependencies icon. When clicked on that you can view all the dependencies.

➔ Example image of a dependency is below



➔ At the centre is a Windows Server (OWA-SD-01), which is connected to multiple computers (e.g., BOW-IBM, ANNIE-IBM) and other servers (VMWARE-SD-07, VMWARE-SD-04).

➔ ServiceNow can share data with third party applications or external systems with integration. Some of the integrations that are used in ServiceNow are

- SSO
- LDAP
- Monitoring
- Notifications
- Events

➔ Common integrations are done with CMDB, Incident management, Problem management, Change management, User management, Login via SSO

➔ Another useful feature in ServiceNow is Integration Hub which provides a single solution to quickly integrate with the third-party application to share the data with ServiceNow or other systems.

➔ It provides capability to perform integration without any coding as it used flow designer functionalities.

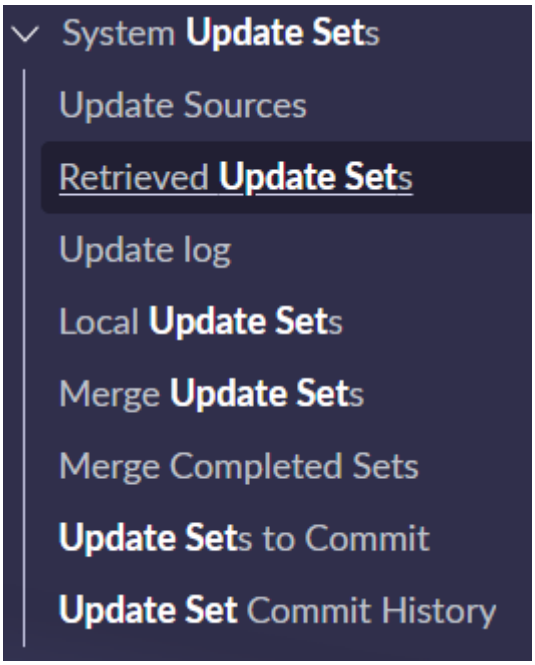
➔ ServiceNow Integration Hub comes with different subscription levels, each offering varying capabilities. Starter, Standard, Professional, Enterprise.

➔ Enterprise: The most comprehensive level, providing all features, including unlimited integration possibilities, premium support, and enhanced security and compliance options.

➔ Update Set in ServiceNow is a container that captures changes made to the platform so they can be moved between different ServiceNow instances. It's an essential tool for managing and deploying customizations and configurations across ServiceNow environments.

➔ This feature allows administrators to group a series of changes into a named set and then move them as a unit to other systems for testing or deployment.

➔ There are several update set applications in ServiceNow



➔ Update Sources: Used to define and manage remote instances from which Update Sets can be retrieved.

➔ Retrieved Update Sets: Displays a list of Update Sets that have been imported from other instances. Once imported, these Update Sets can be reviewed and committed to the current instance to apply the changes.

➔ Update Log: Provides a detailed log of the changes recorded in Update Sets. It helps in tracking modifications made to different configurations, which is useful for troubleshooting and audit purposes.

➔ Local Update Sets: Lists all the Update Sets created and being worked on within the current instance.

➔ Merge Updated Sets: Allows you to combine multiple Update Sets into one. This is useful when you have related changes split across different Update Sets and want to consolidate them before moving to another instance.

➔ Merge Completed Sets: Similar to "Merge Update Sets," but specifically focuses on merging Update Sets that have been marked as "Complete." This helps in preparing a single, comprehensive Update Set for migration.

➔ Update Sets to Commit: Displays a list of Update Sets that have been imported into the instance but not yet committed.

➔ Update Set Commit History: Shows the history of all committed Update Sets in the instance. This allows you to track which Update Sets were applied, by whom, and when, which is essential for auditing and change management.

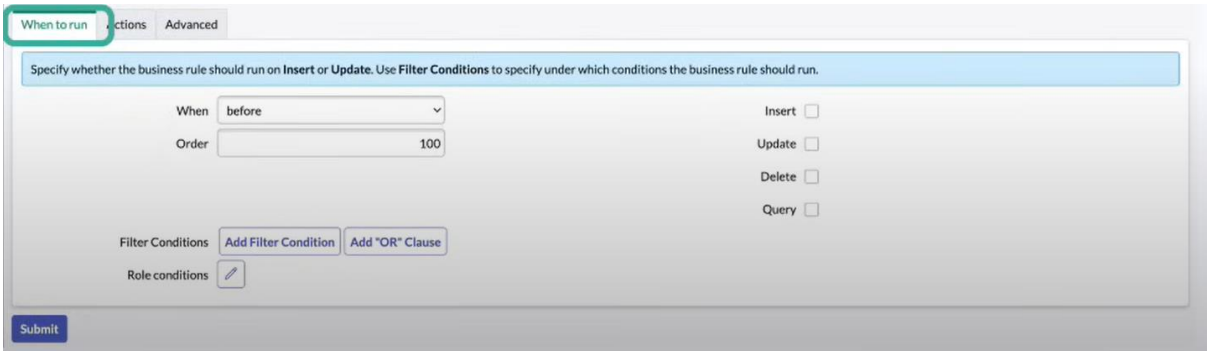
➔ Above are some best Practices for Update Set Management in ServiceNow.

➔ Event: Events are special log records the system generates when something notable has happened or certain conditions occur.

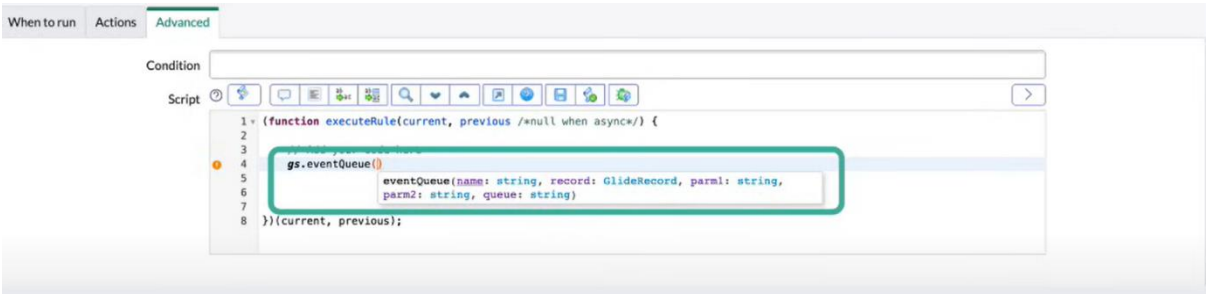
➔ Events can be generated through

- Business Rules
- Event Queue Scripting API
- Flow
- Workflow

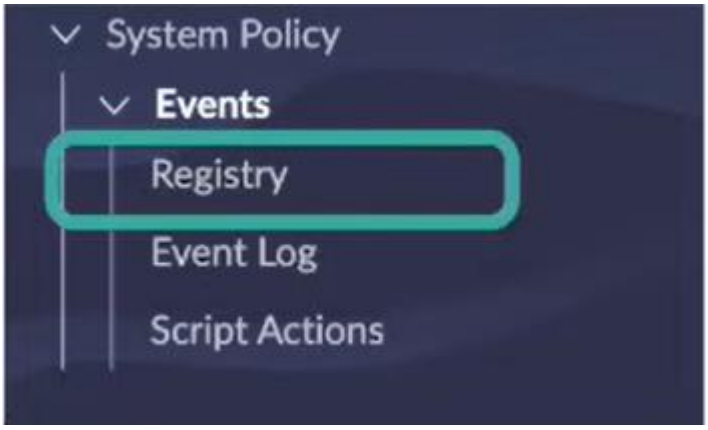
➔ Business Rule: A Business Rule in ServiceNow is a server-side script that runs when records are inserted, updated, deleted, or queried in a database table.



➔ Event Queue Scripting API: This API is used to generate and process events. These events can be used to trigger notifications, workflows, or other business logic asynchronously without affecting the immediate user action.



➔ This method (gs.eventQueue(name, record, parm1, parm2, queue)) is used to queue an event for processing. It is often utilized when something happens in the system (like an update or an insert), and you want an event to be fired based on that activity.



➔ If you have to generate an event that event should be first captured in event registry. So you have to go to registry, create that event, use that naming convention (table name).

➔ In ServiceNow, event logs help track the events that have been triggered in the system. These logs provide a record of all events queued and processed, allowing you to monitor event-driven actions like notifications, workflows, and custom event handling.

➔ Event Actions in ServiceNow refer to the specific responses or activities that are triggered when an event occurs. These actions are configured to automate processes, notify users, or integrate with other systems based on predefined conditions.

➔ Platform statistics in ServiceNow help monitor the health, performance, and usage of the platform. These statistics are essential for administrators to ensure that the platform is running optimally and that resources are being used efficiently.