Cloud

The term Cloud refers to a Network or Internet. Cloud can provide services over network i.e on Public Networks or on Private Networks i.e. WAN – Wide Area Network, LAN – Local Area Network.

Applications such as Email, Web Conferencing, Customer Relationship Management (CRM), all run in cloud.

Cloud Computing

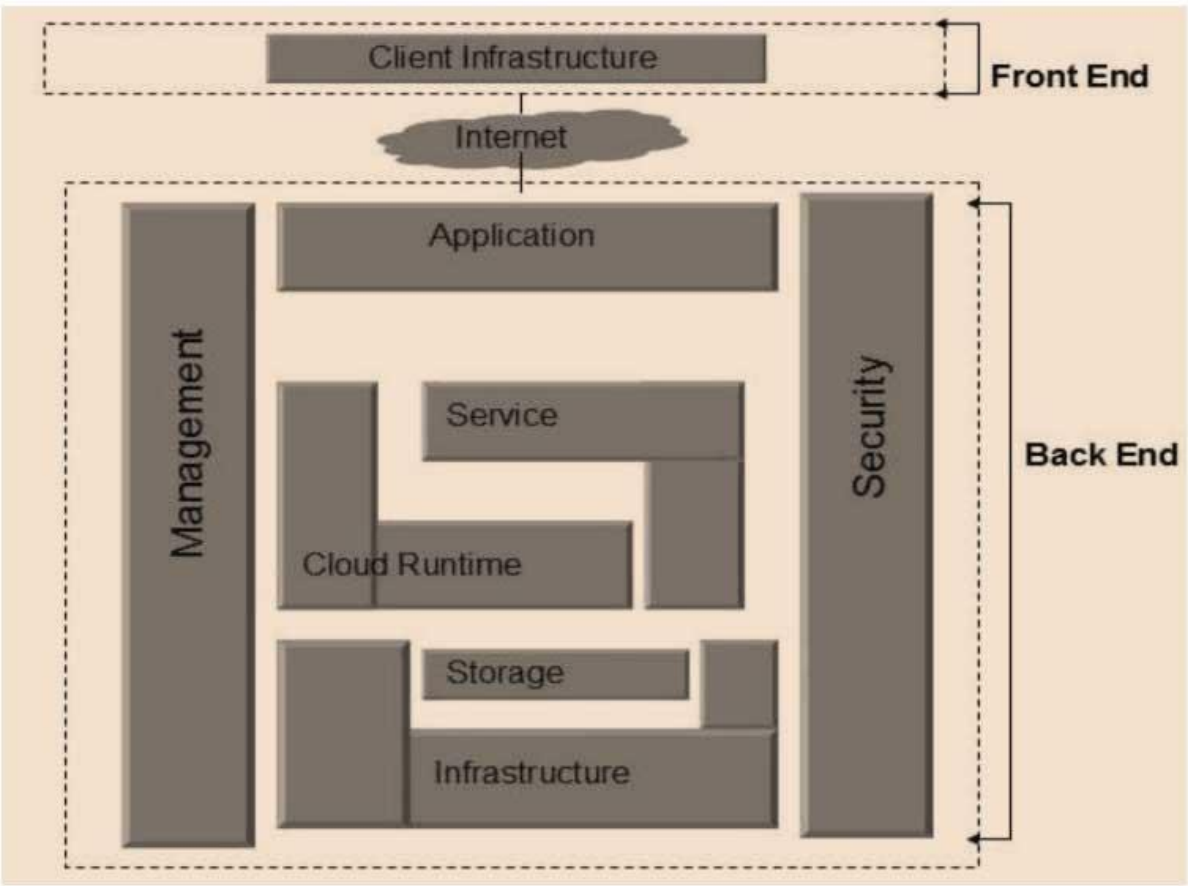
Cloud computing is a method of providing software services or applications services over the internet without the need for any kind of applications to be setup or installed on your actual local machines.

Cloud Computing Architecture

Cloud Computing Architecture comprises of many cloud components, each of them are loosely coupled. We can broadly divide the cloud architecture into two parts:

Front End:
Front End refers to the client part of the computing system. It consists of interfaces and applications that are required to access the computing platforms. Example: Web Browser

Back End:
Back End refers to the cloud itself. It Consist of all the resources required to provide cloud computing services. It comprises of huge Data Storage, Virtual Machines, Security Mechanism, Services, Deployment Models, Servers etc.



Deployment Models of Cloud Computing

Deployment Models of Cloud define the type of access to the cloud, i.e. how the cloud is located?

There are 4 types of deployment models of cloud computing namely:

1. Public Cloud Model.
2. Private Cloud Model.
3. Community Cloud Model.
4. Hybrid Cloud Model.

Public Cloud Model:

The Public Cloud Model allows systems and services to be easily accessible to general public. Example: Google App Engine, Amazon EC2.

Private Cloud Model:

The Private Cloud Model allows systems and services to be accessible within an organization. It is operated only within a single organization.
However, it may be managed internally or by third-party. Example: IBM Cloud.

Community Cloud Model:

The Community Cloud Model allows systems and services to be accessible by a group of organizations. It shares the infrastructure between several organizations from a specific community. Example: Salesforce
It may be managed internally or by the third-party.

Hybrid Cloud Model:

The Hybrid Cloud Model is a mixture of Public Cloud and Private Cloud. Non-critical activities are performed using the public cloud while the critical activities are performed using private cloud. Example: AWS(Amazon Web Service).

Service Providers

- Google Cloud.
- AWS(Amazon Web Services).
- Microsoft Azure.
- IBM cloud.
- Alibaba Cloud etc.

Cloud Service Models

There are basically 3 types of cloud service models:

Software as a Service (SaaS)

Software as a Service (SaaS) makes the software available over the internet. This model allows software applications as a service to the end users. It refers to a software that is deployed on a hosted service and is accessible via the internet.

There are several SaaS applications such as Billing and Invoicing System, Customer Relationship Management (CRM) applications, Help Desk Applications, Human Resource (HR) Solutions.

Characteristics:

● Available on demand.

● The software is are maintained by the vendor rather than where they are running and hence making it cost effective.

● The license to the software may be subscription based or usage based and it is billed on recurring basis.

● Scaled up or down on demand.

● Automatically upgraded and updated.

● SaaS offers a Shared Data Model.

● All users are running the same version of the software.

Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS) provides access to fundamental resources such as Physical Machines, Virtual Machines, Storage etc. All of the above resources are made available to the end user via server virtualization. Moreover, these resources are accessed by the customers as if they own them.

Characteristics:

● Virtual machines with pre-installed software.

● Virtual machines with pre-installed operating systems such as Windows, Linux, and Solaris.

● On-demand availability of resources.

● The computing resources can be easily scaled up and down.

Platform as a Service (PaaS)

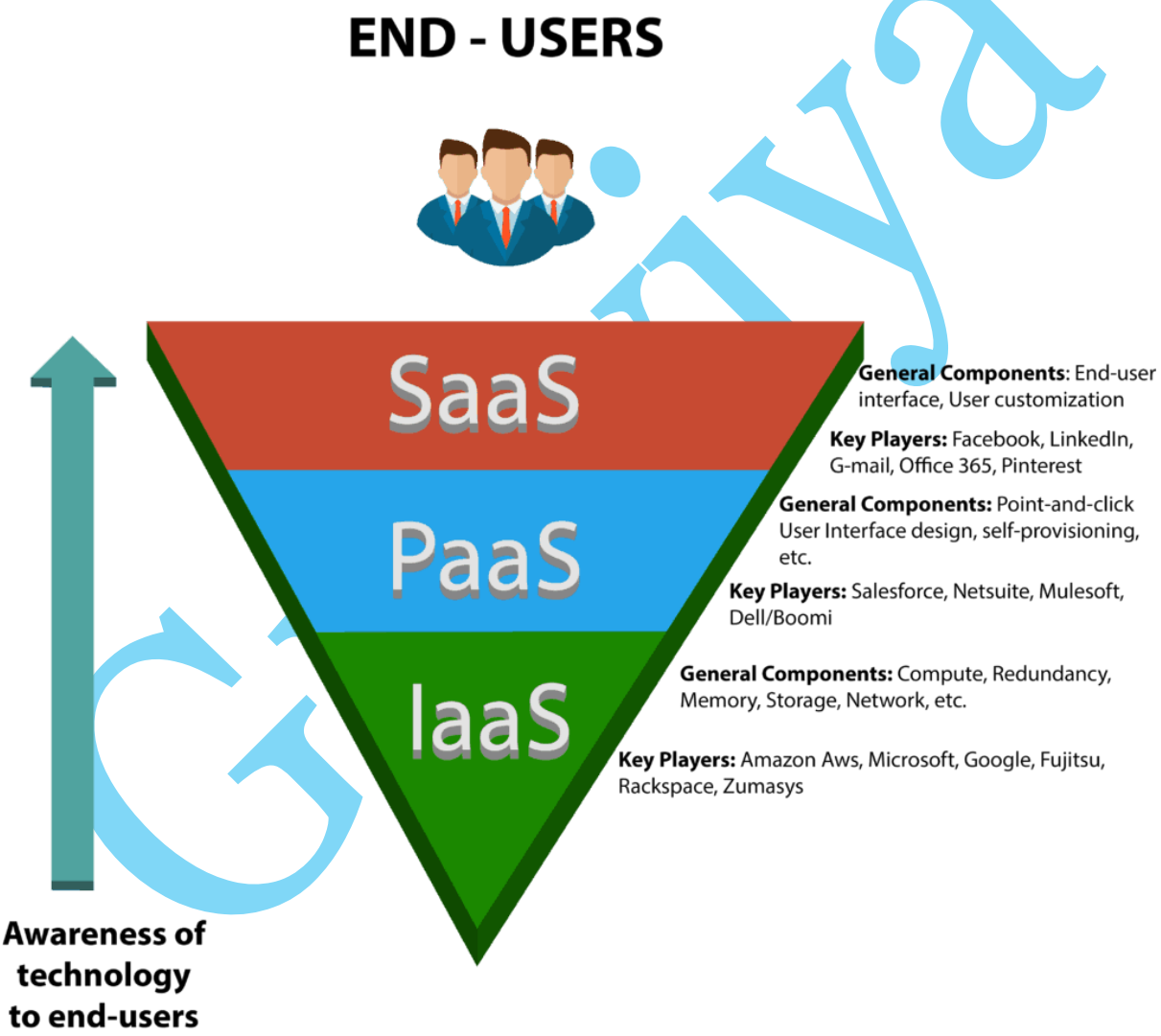
Platform as a Service (PaaS) offers the run time environment for applications. It also offers development & deployment tools, required to develop applications. PaaS has a feature of point-and-click tools that enables non-developers to create web applications.
Force.com platform is an example of PaaS in salesforce.

Characteristics:

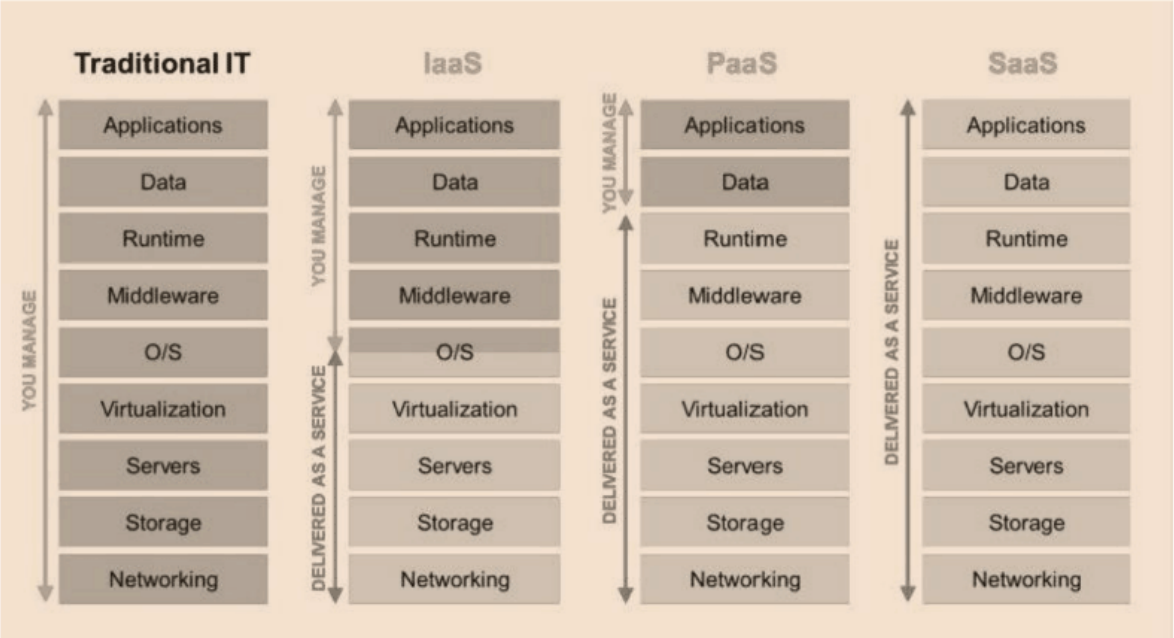
● PaaS offers a browser-based development environment. It allows the developer to create a database and edit the application code either via Application Programming Interface or Point-and-Click tools.

● PaaS provides built-in security, scalability, and web service interfaces.

● PaaS also provides web services interfaces that allow us to connect the applications outside the platform.

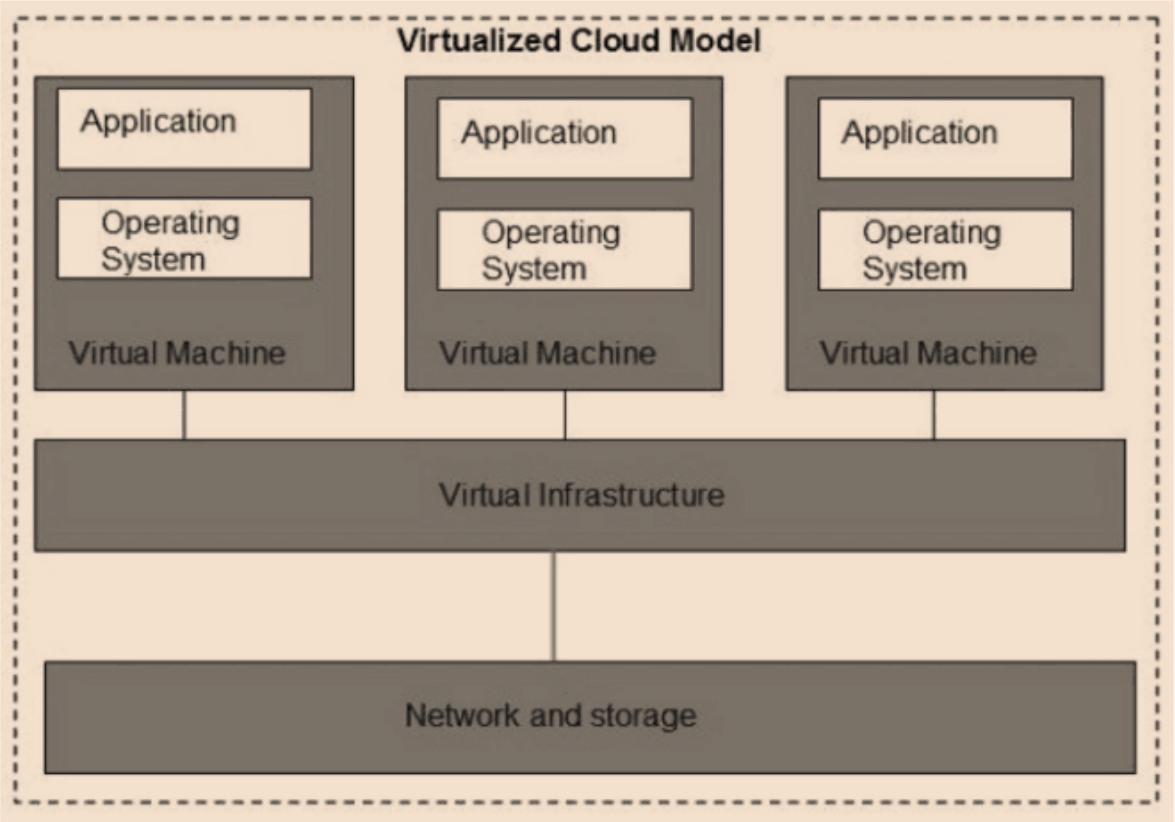


Differences between IaaS, PaaS, SaaS



Virtualization

Virtualization in Cloud Computing is a technique, which allows to share a single physical instance of an application or resource among multiple organizations or tenants (customers). It does this by assigning a logical name to a physical resource and providing a pointer to that physical resource when demanded.



The Multi-tenant architecture offers virtual isolation among the multiple tenants. Here multiple clients share the same infrastructure under different logical IDs according to their requirement. Hence, the organizations can use and customize their application as though they each have their instances running.

Benefits of Cloud Computing:

- Lower cost
- Lower risks
- Higher profit margins
- Higher security
- Collaborative
- Better performance
- Reliability
- Simplicity
- Scalability

Some Misconceptions about Cloud Computing:

- Cloud is not secure.
- You lose control with the cloud.
- We won't need PCs any more with cloud computing.
- Cloud is not reliable.
- The cloud will give you performance problems.

Salesforce

Salesforce is a global web-based application and cloud computing company best known for its Customer Relationship Management (CRM) product. They specialize in Software-as-a-Service (SaaS) to help users handle all of their business needs.

Salesforce.com is a customer success platform, designed to help users sell, service, market, analyse, and connect with their customers.

Salesforce has everything required to run a business from anywhere. Using standard products and features, users can manage relationships with customers, collaborate and engage with employees and partners, and store their data securely in the cloud.

CRM

CRM stands for Customer Relationship Management. It is a process or methodology used to learn more about customer's needs and behaviours in order to develop stronger relationships with them.

This technology allows users to manage relationships with their customers and track data related to all of their interactions. It also helps teams collaborate, both internally and externally, gather insights from social media, track important metrics, and communicate via Email, Phone, Social, and other channels.

The more useful way to think about CRM is as a process that will help bring together lots of pieces of information about customers, sales, marketing effectiveness, responsiveness and market trends.

Benefits of CRM:

- Discover new customers
- Increase customer revenues
- Provide better customer services
- Sell products more effectively
- Help sales staff close deals faster
- Make call centre's more efficient
- Simplify marketing and sales processes

Salesforce Architecture

Salesforce CRM service is broken down into several broad categories which are:

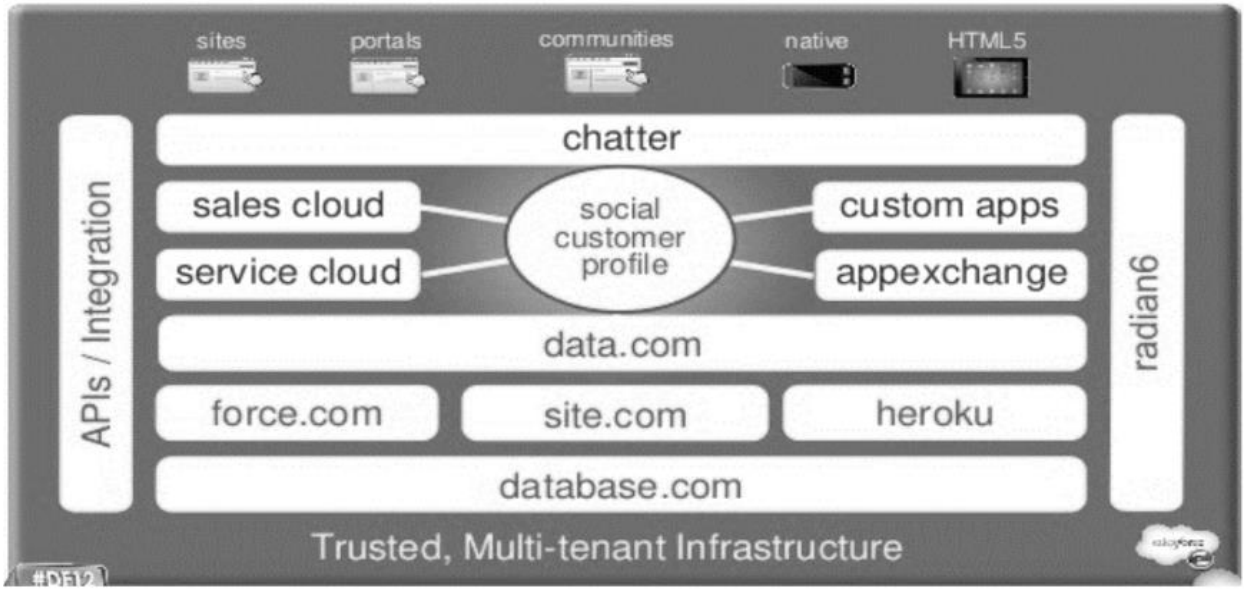
- Sales Cloud
- Service Cloud
- Data Cloud
- Marketing Cloud
- Collaboration Cloud
- Analytics Cloud
- Custom Cloud including Force.com

Chatter:

Chatter makes the business social and facilitates connections. It provides collaboration features and capabilities to any application built on the Force.com platform.

Sales Cloud:

Manage sales process end-to-end. When organizations develop products in force.com for sales then it comes in Salesforce Sales Cloud.



Service Cloud:

Support customers after the sale. When organizations want to provide some facility and also provide support to the clients then it comes in Salesforce Service Cloud.

AppExchange:

AppExchange is Salesforce's cloud computing marketplace, through which end users can access, download and install custom applications.

Data.com:

Data.com is a cloud-based automated system for acquiring and managing CRM records within a user's Salesforce.com account.

Heroku:

Heroku is a cloud platform as a service (PaaS) supporting several programming languages. It supports development in Ruby on Rails, Java, Node.js, PHP, Python, Scala and Clojure.

Radian6:

Radian6 provides social media monitoring and analysis, in an automated solution that tracks conversations across different online channels from social media to websites.

Site.com:

Site.com is a content management system available on cloud. It follows the concept of "Create Once and Publish Anywhere". It allows you to create your own websites, social channels like Twitter, Facebook. It provides WYSIWYG-style tools to build and deliver web sites.

Site.com provides WYSIWYG-style tools to build and deliver web sites.

Database.com:

Database engine for cloud application developers.

Benefits of Salesforce:

- It is highly customizable.
- It is the one which comes with three major releases adopting to the user's requirement, which is increasing the user's adaptability rate.
- It has got the highest user adoptability rate.

Force.com

Force.com is a platform-as-a-service (PaaS) product designed to simplify the development and deployment of cloud-based applications and websites. Developers can create apps & websites through the cloud IDE & deploy them quickly to Force.com multi-tenant servers.

It is used to create and deploy next-generation cloud apps. Because there are no servers or software to buy or manage, developers can focus solely on building apps that include built-in social and mobile functionality, business processes, reporting and search. Apps run on a secure, proven service that scales tunes and backs up data automatically.

Benefits of Force.com:

- Proven
- Agile
- Social
- Mobile

Salesforce Environments

An environment is an instance of Force.com platform which can also be called an organization. Salesforce Environments lets users access, deploy or create applications with various feature sets, depending on the configuration of the environment.

A good way to think about an instance of Force.com, is to think about Gmail: when users sign up for Gmail, they get a unique username and access to their private email account in the cloud. Similarly, when users sign up for a Force.com organization, they get a unique username and private access to their Force.com cloud computing environment.

Characteristics of an Environments:

- Contains data (records) and customizations.
- Each environment is based on an edition, which contains specific functionality and limits.
- Can be accessed through a web browser.

Types of Salesforce Environments:

There are three types of Environments:

- Production Environments.
- Development Environments.
- Testing Environments.

Production Environments

Salesforce Environments that have active paying users accessing business critical data are known as Production Environments. It stores the live data that is actively used to run the business.

Custom Force.com application can be developed for:

- Own production use
- Existing salesforce.com customers
- Any business or organization in the world

Note:

- *Users or Developers can easily build a custom object, field or validation rule directly to their production environment without the need of a development or test environment. However some features like Apex code can only be created in development environments.*
- *Salesforce always recommends that one should use a development environment when building a Force.com app.*

Developer Environments

Salesforce Environments where users/developers can extend, integrate and develop on Force.com without affecting their production environments.

Development Environments are used strictly for developing and testing apps. These environments contain test data that are not business critical. Development can be done inside the browser or with the Force.com IDE, which is based on Eclipse.

There are two types of Development Environments:

1.Developer Edition(DE) Environment.

It is a free, fully-featured development environment with limits on data and users. It is a logically separate environment, ideal for initial development on Force.com. Users/Developers can sign-up for as many DE organizations as they require. This allows them to build an application designed for any of the Salesforce Production Environments.

2.Sandbox

Sandbox is a nearly identical copy of the Production Environment. It can include data, configurations, or both. It is possible to create multiple Sandboxes in Production Environments for a variety of purposes without compromising the data and applications in the Production Environment.

Different types of Developer Edition and Sandbox Environments

Development Environment	User Licenses	Data Storage	Notes
Developer Edition	2 full CRM licenses 3 Force.com Platform licenses	5 MB	Sign-Up is free
Partner Developer Edition	20 full CRM licenses 20 Force.com Platform licenses	250 MB *(about 125,000 records)*	This is a DE org with more storage, features and licenses. Free for enrolled partners.
Full Sandbox	Same as production environment		This is a copy of your production org including data and customization. You may order up to a maximum of 3 full sandboxes.
Partial Copy Sandbox	Same as production environment	5 GB data *(about 2.5 million records)* 5 GB file storage	This copies the customization, and a sample of your production organization's data. You may order up to a maximum of 6 partial copy sandboxes.
Developer Sandbox	Same as production environment	200 MB data *(about 100,000 records)* 200 MB file storage	This copies customization (metadata), it doesn't copy production data.
Developer Pro Sandbox	Same as production environment	1 GB data *(about 500,000 records)* 1 GB file storage	This copies customization (metadata), it doesn't copy production data, but has more storage than Developer sandbox.

Testing Environments

Before deploying an app to production or release it to customers, developers migrate apps to a dedicated Testing Environment where they can perform integration tests with large sets of data, do security checks for multiple users and profiles, and uncover bugs or discover enhancements they would like to add.

Automated test scripts can't determine a user's level of satisfaction with the app, so there is a need to train several new users to evaluate the application in a real-world setting without affecting the business data. In essence, developers need to create another development environment that mimics the Production Environment, and have these trained users/testers use it.

It can be Production Environment or Development Environment specifically used for testing application's functionality before deploying to production or releasing to customers.

Developers have multiple options for creating a dedicated Testing Environments.

- One is to create a Sandbox copy of their Production Environment and deploy directly from the Development Environment to their sandbox. Using this way, developers will not only test the finished application, but also the procedure used to deploy the completed application as well.
- If there is no sandbox then they can use a Developer Edition environment as their testing environment. The standard Developer Edition environment may be limiting because of license and storage limits.
- There is also Partner Developer Edition for partners who are looking for a production-like environment with more users and storage to run real-life tests.

Salesforce License:

- **Starter- $25 per user per month**
- **Professional- $75 per user per month**
- **Enterprise- $150 per user per month**
- **Unlimited- $300 per user per month**
- **Developer Edition-Free**

Signing up with developer Edition:

First Create a developer Account by using this link

- https://developer.salesforce.com/

After that login into your developer edition by using this link

- https://login.salesforce.com/

If you want to learn or study salesforce so you can take the help of trailhead in which everything is present like as theory, modules.

- https://trailhead.salesforce.com/today

Limitation of developer Edition:

- 2 users(salesforce License)
- Expires if its idle for 6 months.
- Maximum Data Storage is 5MB.
- No Sandbox.

APPROACH IN SALESFORCE

1.point & click tool(Declarative Approach):

In simpler terms, declarative programming denotes the kind of click or drag-and-drop solutions that allow someone without coding knowledge to build an application. The blocks of code are prewritten and packaged into components that you can just select as needed.

2.Code(Programmatic Approach):

Programmatic customizations are made with code . programmatic customizations require coding skills and allow developers to extend beyond the declarative Salesforce.com capabilities.

Salesforce Classic vs Salesforce lightning:

1.UI(User Interference)

Salesforce Lightning offers a new and improved interface with additional features that make it easier and more engaging for users, whereas Salesforce Classic offers the original, albeit updated, interface that is lacking in some key areas. Generally speaking, Lightning just offers more.

2.Architecture

Lightning Experience is based on the concept of a **Single-Page Architecture** (**SPA**). In SPAs, once the page is loaded, for any other subsequent request of data from the server, an Ajax callout is made and the page is re-rendered with new data. The aim is to avoid the whole page refreshing. Classic Experience is based on the concept of **Page-Based-Architecture(PBA).** In PBA, when the page is loaded ,so the whole page start refreshing.

Spreadsheet

A spreadsheet is a computer application for organization, analysis and storage of data in tabular form.

Why we use Database instead of spreadsheet.

1.Spreadsheets are unstructured , database is structured(hide data).

2.filtering and sorting data is a mess in spreadsheets.

3.Data validation is not there in spreadsheets.

4.Referncing data from other tables is hard in spreadsheet.

5.Pulling Data from third party service(app) is not straight forward in spreadsheets.

Database

A database is **a collection of information that is organized so** that it can be easily accessed, managed and updated. Computer databases typically contain aggregations of data records or files, containing information about sales transactions or interactions with specific customers.

ID	First Name	Last Name	Email	Year of Birth
1	Peter	Lee	plee@university.edu	1992
2	Jonathan	Edwards	jedwards@university.edu	1994
3	Marilyn	Johnson	mjohnson@university.edu	1993
6	Joe	Kim	jkim@university.edu	1992
12	Haley	Martinez	hmartinez@university.edu	1993
14	John	Mfume	jmfume@university.edu	1991
15	David	Letty	dletty@university.edu	1995

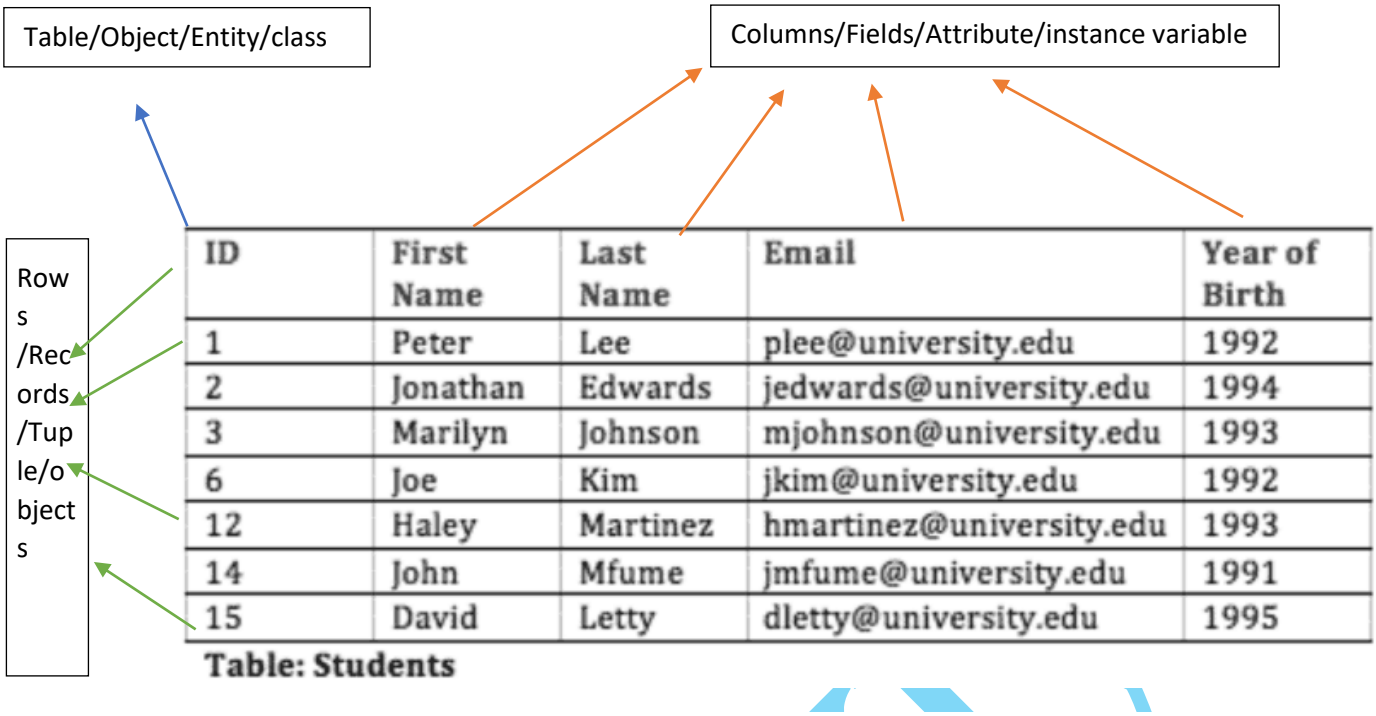
Table: Students

In a relational database:

- Data is stored in tables.
- Each table comprises a number of columns of a particular data type such as Text, Number or Date.
- Information is stored in the rows of tables.
- Tables can be related to other tables using the concept of Primary Key and Foreign Key.

Whereas in Salesforce:

- It uses objects to store data.
- Each object comprises a number of fields which correspond to columns in a database.
- Data is stored in records of objects which corresponds to rows in a database.
- Objects can be related to other objects using relationship fields.



ID	First Name	Last Name	Email	Year of Birth
1	Peter	Lee	plee@university.edu	1992
2	Jonathan	Edwards	jedwards@university.edu	1994
3	Marilyn	Johnson	mjohnson@university.edu	1993
6	Joe	Kim	jkim@university.edu	1992
12	Haley	Martinez	hmartinez@university.edu	1993
14	John	Mfume	jmfume@university.edu	1991
15	David	Letty	dletty@university.edu	1995

Table: Students

Objects

Objects in Force.com are represented in the form of a table and it is here referred as what an entity in a database.

There are 2 types of objects:

Standard Objects:

Standard Objects are included with the Salesforce by default. Example: Accounts, Leads and Opportunities.

Custom Objects:

Custom Objects are created to store information unique to an organization. Custom objects extend the functionality that standard objects provide.

Note:

Each object in Force.com has a built-in feature like user-interface, security and sharing model and much more.

Standard Object	Custom Object
Standard Object are out of box.	Custom object is user defined.

API names does not end with __c.	API names ends with __c.
Limited customization	Fully customization.
Example: Account, Contact & user.	Example:Student__C,EMPLOYEE__C or DEPARTMENT__C,Book__c.

External Objects

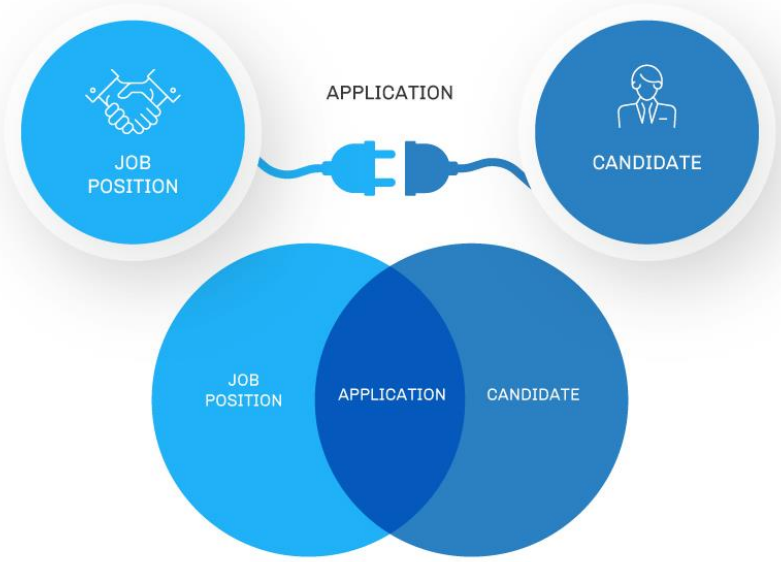
The objects which you create map to the data stored outside your organization.

Junction Objects

Salesforce Junction Objects give you a way to create a many-to-many relationship between Salesforce objects. They are created using a custom object and then relating two other objects via two master-detail relationships. This is necessary when modelling certain data schemas, let's dive into a examples…

Junction Objects Example

Take a recruitment process where you have a job position and candidate object. As it is a requirement for candidates to apply to multiple job positions, a standard parent-child relationship would not work. You need a Junction Object, called something along the lines of 'Application' in order for many candidates to apply to multiple job positions.



Apps in Salesforce

Apps in Salesforce is little more than a container for all of the objects, tabs and other functionality.

It is a group of tabs that works as a unit to provide application's functionality.

It consists simply of a name, a logo and an ordered set of tabs. The simplest app contains only one tab i.e. the Home Tab and a default logo.

There are basically 2 types of Apps in Salesforce:

Standard Apps:

The apps which come with every instance of Salesforce by default. It includes App Launcher, Call Centre, Community, Content, Marketing, Sales, Salesforce Chatter and Site.com.

These apps can be customized according to the needs and requirements of an organization.

The label, description and logo of a standard app can't be changed.

Custom Apps:

The apps which are built to meet the specific business needs & requirements of an organization.

Custom apps can be made by grouping standard as well as custom tabs.

Logo in custom apps can be added and they can be changed after that also.

Console Apps

Console apps are designed and developed to provide a superior UI experience so that users can use the application in a productive manner.

It is mostly used for fast paced sales and service environments where users can use many functionalities at the same time on a single window without the need to refresh the whole page. It provides partial refresh which refresh the tabs updated without refreshing the whole page.

It can include Live Agent (Chat), Softphone (Call) and several other tabs including the Visualforce Pages.

It provides the access to data and features frequently used by the user. Users can see records as well as items related to them on one screen. It also allows us to work on multiple items simultaneously.

Tabs

Tabs in Salesforce help users view the information at a glance. It displays the data of objects and other web content in the application.

There are mainly 4 types of tabs:

Standard Object Tabs:

Standard object tabs display data related to standard objects.

Custom Object Tabs:

Custom object tabs display data related to custom objects. These tabs look and function just like standard tabs.

Web Tabs:

Web Tabs display any external Web-based application or Web page in a Salesforce tab.

Visualforce Tabs:

Visualforce Tabs display data from a Visualforce Page.

Lightning Tabs

Lightning Tabs Display data from lightning component.

Tab Visibility

Default On:

If the tab visibility is Default On then it would appear on the top bar along with other visible tabs.

Default Off:

If the tab visibility is Default Off then user will still be able to access the tab but by clicking on "+" symbol at the end of all visible tabs on top bar.

Tab Hidden:

If the tab is hidden then it will neither be visible on the top bar nor by clicking on the "+" symbol.

Note:
However users can edit this according to their preferences.

Fields In Salesforce

Fields in Salesforce represents what the columns represent in relational databases. It can store data values which are required for a particular object in a record.

There are 2 types of fields:

Standard Fields

There are four standard fields in every custom object that are Created By, Last Modified By, Owner, and the field created at the time of the creation of an object.
These fields cannot be deleted or edited and they are always required. For standard objects, the fields which are present by default in them and cannot be deleted from standard objects are standard fields.

Custom Fields

The Custom fields which are added by the administrator/developer to meet the business requirements of any organization. They may or may not be required.

Datatypes of fields:

Auto Number	Automatically assigns a unique number to each record.
Checkbox	Allows users to check a box, indicating a true or false attribute of a record.
Currency	Allows users to enter a currency amount.
Date	Allows users to enter a date or pick a date from a popup calendar.
Date/Time	Allows users to enter a date or pick a date from a popup calendar and enter a time of day. They can also add the current date and time by clicking the date and time link next to the field.
Email	Allows users to enter an email address of up to 80 characters, which is validated to ensure proper format.
Geolocation	Allows users to specify a location by its latitude and longitude.
Number	Allows users to enter any number. This is treated as a real number and any leading zeros are removed.
Percent	Allows users to enter a percentage number as a decimal

Phone	Allows users to enter any phone number. Character limit is 40.
Picklist	Allows users to select a single value from a list that you define.
Picklist (Multi-select)	Allows users to select more than one picklist value from a list that you define.
Text	Allows users to enter any combination of letters, numbers, or symbols.
Text (Encrypted)	Allows users to enter any combination of letters, numbers, or symbols that are stored in encrypted form.
Text Area	Allows users to enter characters that display on separate lines similar to a Description field.
Text Area (Long)	Allows users to enter up to 131,072 characters that display on separate lines similar to a Description field. You can set the length of this field type to a lower limit, if desired. Any length from 256 to 131,072 characters is allowed. The default is 32,768 characters.
Text Area (Rich)	With the use of a toolbar, users can format the field content and add images and hyperlinks. The toolbar allows the user to undo, redo, bold, italicize, underline, strike-out, add a hyperlink, upload or link to an image, modify alignment, add a numbered or non-numbered list, indent, and outdent.
URL	Allows users to enter up to 255 characters of any valid website address.

Global Picklists

Global Picklists can be used for sharing a single list of values across many custom picklist fields. It is a restricted picklist. The list of values in the global picklist gets locked and cannot be edited on custom picklist fields. To change the values, the global picklist should be edited.

Field Dependency

Field Dependency in Salesforce are filters that allow us to change the content of a picklist based on the value of another field checkbox/picklist. There are two terms used in Salesforce Field Dependency:

Controlling Field:

The value of this field controls the value of the dependent field.

Dependent Field:

The value of this field depends on the value of the controlling field.

Datatype	Controlling	Dependent
Standard Picklist	Y	N
Custom Picklist	Y	Y
Custom Multi Picklist	N	Y
Standard check box	Y	N
Custom check box	Y	N

Relationship in salesforce

Relationship in Salesforce is a 2-way association between 2 objects. Using relationships we can link objects with each other and we can make connections and display data about other related objects.

Why we need to create two table

1.data Redundancy

Data redundancy is defined as the storing of the same data in multiple locations.

Example - Problem Of Data Redundancy In Single Tabale Database

Employee Number	First Name	Last Name	Date of Birth	Department Code	Department Name	Department Head
1001	Steave	Jakson	25-09-1985	SA001	Sales	Paul Colgan
1002	Kitty	Mathew	06-04-1998	ACC008	Accounts	Jerry Mathew
1003	Meena	Patel	11-05-1992	SA001	Sales	Paul Colgan
1004	Nancy	Samual	02-12-1996	ACC008	Accounts	Jerry Mathew
1005	Michael	Smith	28-03-1995	SA001	Sales	Paul Colgan
1006	James	Garcia	22-01-1994	SA002	Sales	David Smith
1007	Nancy	Samual	11-02-1996	ACC008	Accounts	Charles Williams

Redundant Data In Table

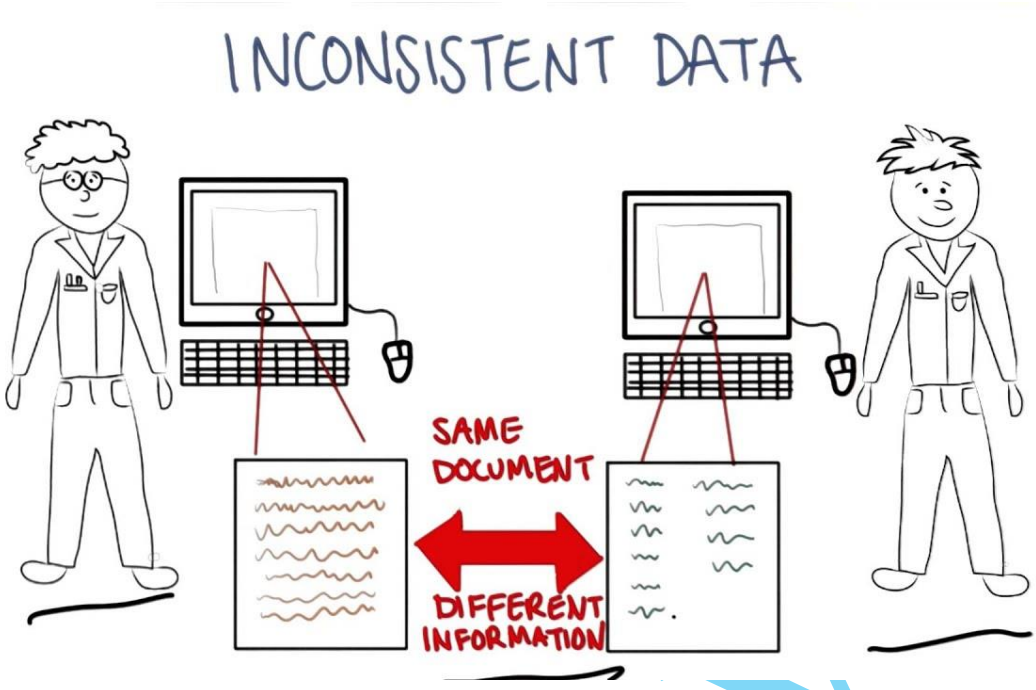
SA001	Sales	Paul Colgan
ACC008	Accounts	Jerry Mathew

www.learncomputerscienceonline.com

2.Data Inconsistency

Data inconsistency is a situation where there are multiple tables within a database that deal with the same data but may receive it from different inputs.

Example: If we have an address of someone in many tables and when we change it in only one table and in another table it may not be updated so there is the problem of data inconsistency may occur.



3.another reason why keep data in different tables is security of data. Take an example of employ data. Employee details like team, Designation , Manager , email id and contact number can be shared with everyone. However we cannot share salary , Projects ,Rating ,Aadhaar No. , Account No. and other data in the same table that is another reason we use different tables.

In Relational Database

To create a relationship between two tables we use the concept of Primary Key and Foreign Key.

Primary Key:

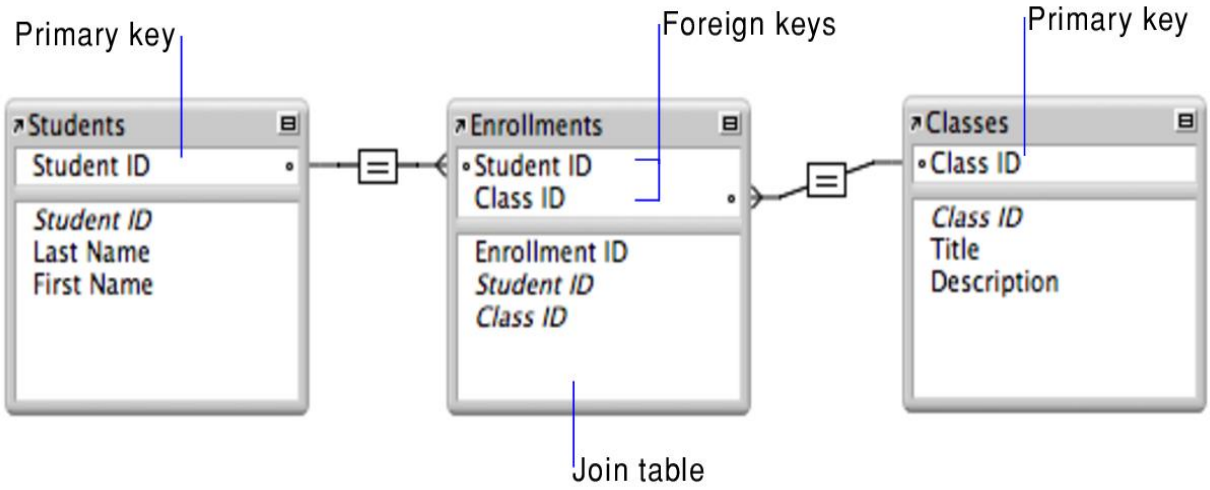
A primary key is a **column -- or a group of columns -- in a table that uniquely identifies the rows in that table**. For example, driver license number, telephone number (including area code), or vehicle identification number (VIN). A relational database must always have one and only one primary key.

It resides on the table which is one in a many to one relationship.

Foreign Key:

A foreign key is a column or group of columns in a relational database table that provides a link between data in two tables. It acts as a cross-reference between tables because it references the primary key of another table, thereby establishing a link between them.

It resides on the table which is many in a many to one relationship.



In Salesforce

To create relationships between two objects, we use relationship fields. The field is created on the many side object and it is related to the one side object.

In salesforce there are basically 2 types of relationship fields

1. Master detail relationship.
2. Lookup relationship.

Master Detail Relationship

Master-Detail Relationship in Salesforce is a parent-child relationship in which the master object controls certain behaviours of the detail object.

- When a record of the master object is deleted, its related detail records are also deleted.
- The Owner field on the detail object is not available and is automatically set to the owner of its associated master record. Custom objects on the detail side of a master-detail relationship cannot have sharing rules, manual sharing, or queues, as these require the Owner field.
- The detail record inherits the sharing and security settings of its master record.
- It is required on the page layout of the detail record.
- By default, records can't be re-parented in master-detail relationships. Administrators can, however, allow child records in master-detail

relationships on custom objects to be reparented to different parent records by selecting the Allow reparenting option in the master-detail relationship definition.

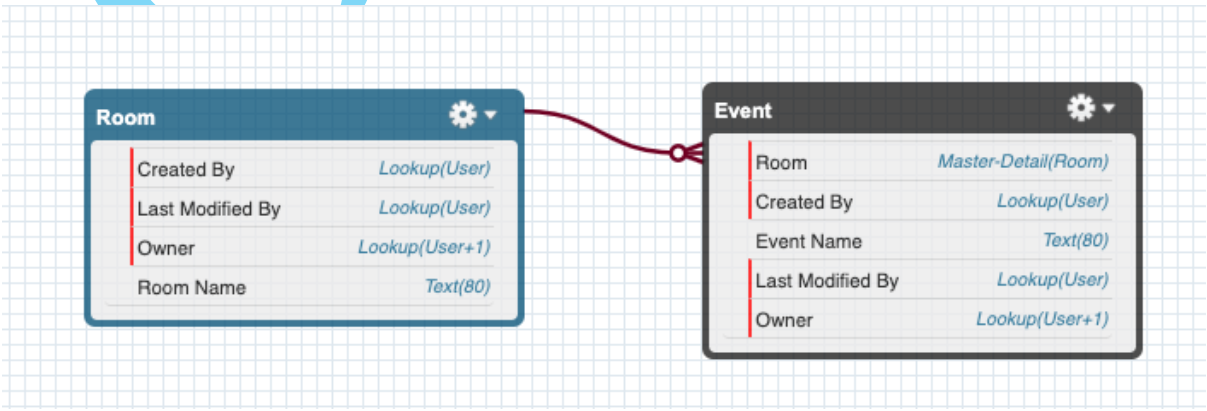
- It can be defined between custom objects or between a custom object and a standard object. However, the standard object cannot be on the detail side of a relationship with a custom object.
- The data related to the object appears on a related list.
- When we undelete the master record then all its related details records also get undeleted.
- if we want to create master detail relationship so how both object should be or what will happen if we will take that object

1. SO and CO (Correct)
2. SO and SO (Incorrect)
3. CO and CO (Correct)
4. CO and SO (Incorrect)

- Rollup summary field available
- Only 2 MD fields are allowed.

Note

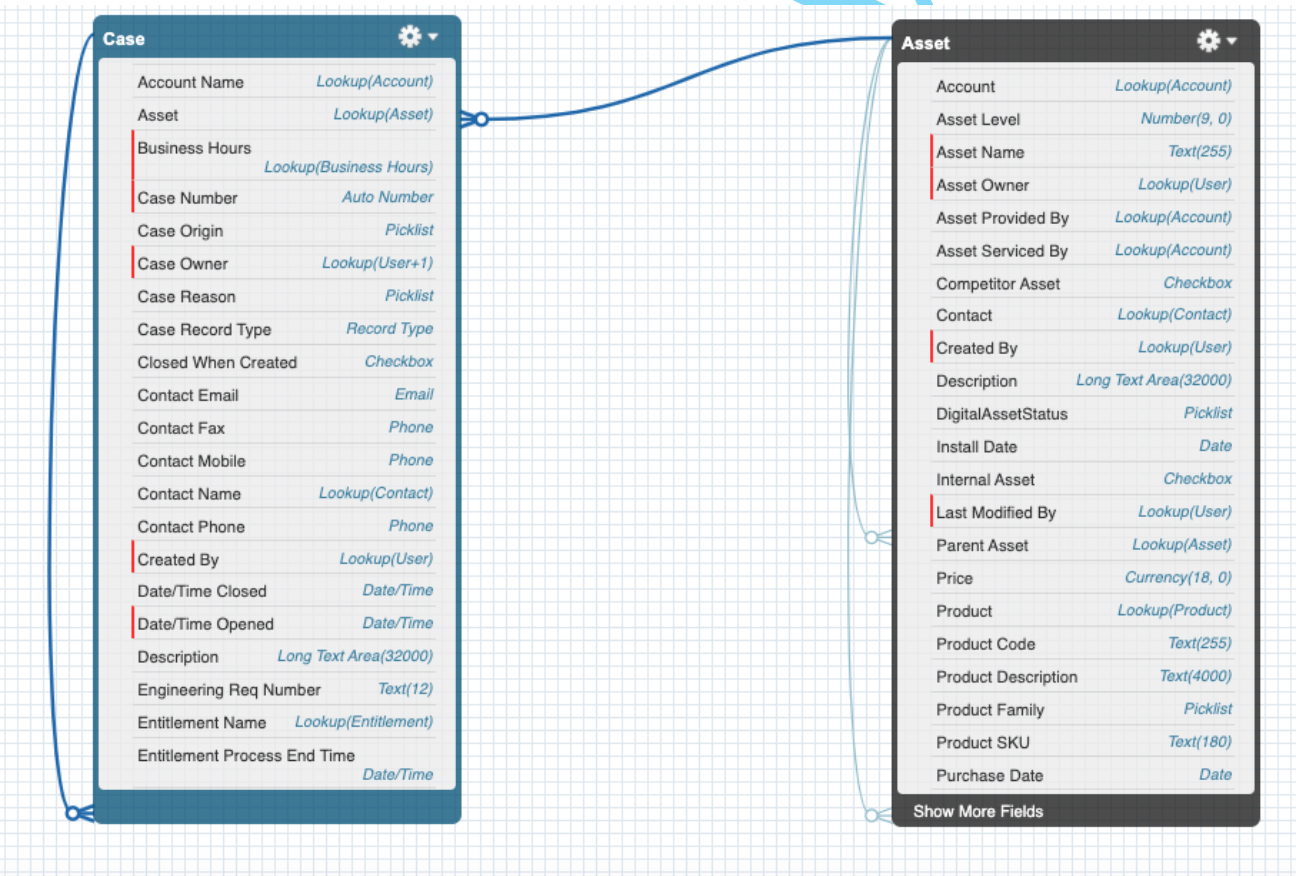
Master-Detail relationship cannot be created where User or Lead objects are the master.



Lookup Relationship

Lookup Relationship in Salesforce links two objects together but has no effect on deletion or security. Unlike master-detail fields, lookup fields are not automatically required.

- Simple relationship.
- Standard object record can be on the detail side of a custom object in a lookup relationship.
- Independent of security and sharing setting.
- Roll up summary field are not available
- Up to 40 lookup fields allowed.

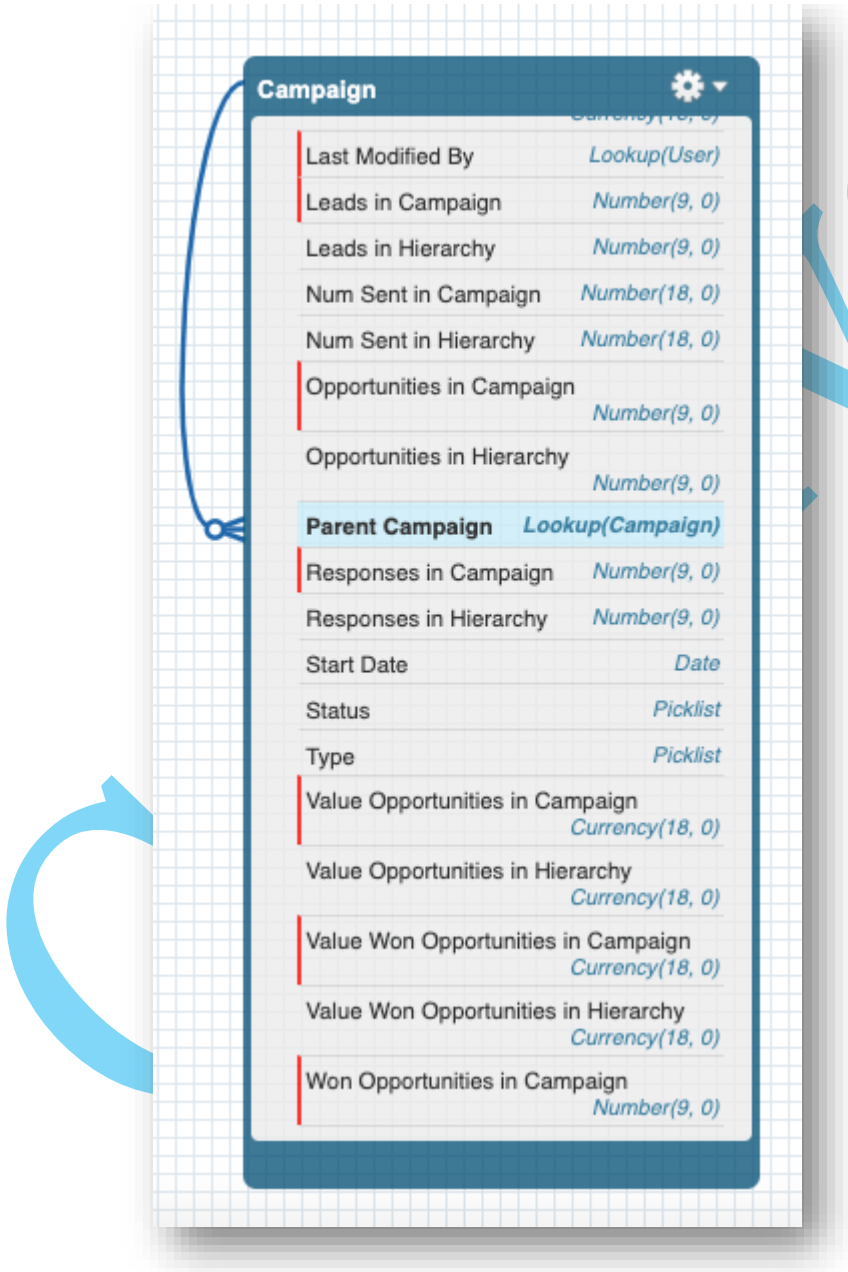


There are derived into 2 subtypes:

Self:

When an object has a lookup with itself, it is a self-relationship.

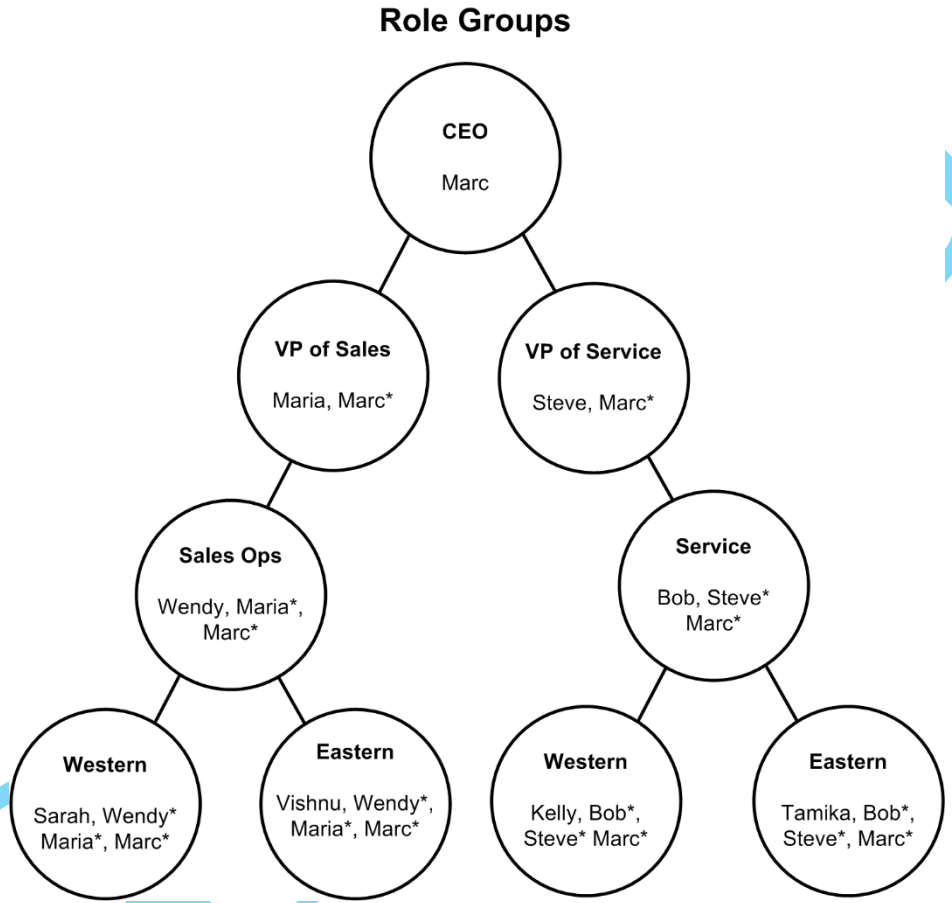
Example: you have a Campaign. This campaign is part of a bigger campaign. You can use a lookup field from campaign to… campaign!



Hierarchical

A special lookup relationship is available for only the user object. It lets users use a lookup field to associate one user with another that does not directly or indirectly refer to itself.

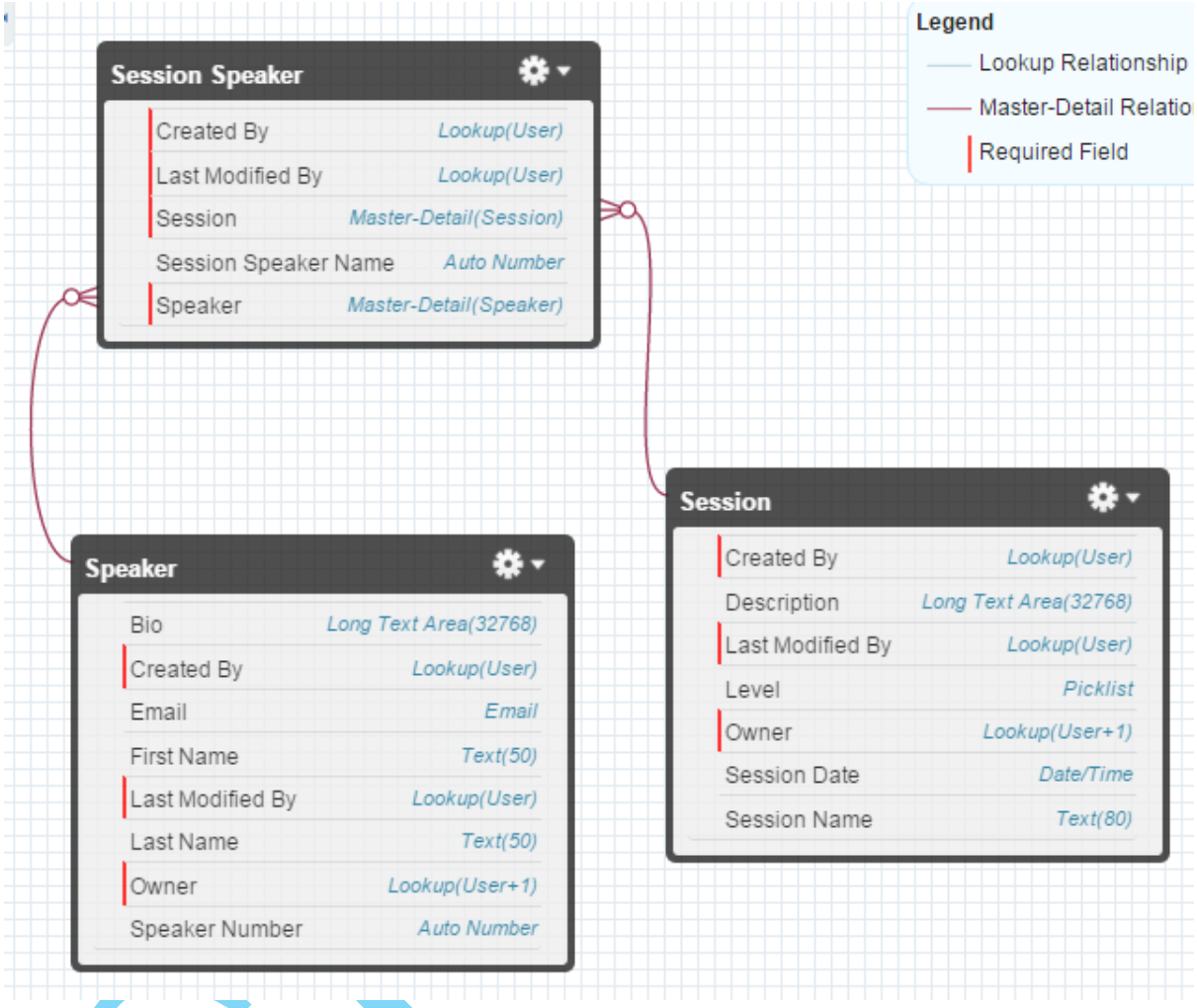
For example: You can create a custom hierarchical relationship field to store each user's direct manager.



Many-to-Many:

Many-to-Many relationships can be modelled using master-details relationships between any two objects. It allows each record of one object to be linked to multiple records from another object and vice versa.
To create a many-to-many relationship, simply create a custom junction object with two master-detail relationship fields, each linking to the objects needed to relate.

Example: a session can have multiple speakers but also speakers can present at multiple sessions. A perfect place to add that junction object of 'Session Speaker' in between to handle this relationship!



Primary Relationship:

It is the relation created first with any of the two master objects.

- The object with which the relationship is created first is called the primary master object.
- The detail and edit page of the junction object will use the colour and any associated icon of the primary master object.
- The junction object records will inherit the value of the owner field.
- The sharing and security settings of junction objects will depend on the associated master record.
- Deleting a record of the primary object will delete the associated records of the junction object as well.

Secondary Relationship:

The second master-detail relationship created with another master object.

- The object with which the relationship is created second is called the secondary master object.
- This relationship does not affect the look and feel of the junction object.
- The security and sharing setting of the junction object depends on secondary relationships as well.
- Deleting a record of a secondary object also deletes the associated records of the junction object.

Note:
Junction objects cannot be created as a master object in a master-detail relationship with another object as detail.

External Relationships

External relationship to be formed between a Salesforce object and an external object. If your Salesforce org was the family, then the objects would be related but not in the same family.

A great **example** of such is relating payment records to an account from an ERP system – the ERP system is external to Salesforce but the records are important. The way they relate is via an external ID to match the payments to the account.

Lookup Filter in Salesforce

Salesforce Lookup Filter limits which records can be associated within an object relationship. It can be applied to Lookup, Master Detail, and Hierarchical Relationship Fields.

Formula

A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.

Cross Object Formula Field in Salesforce

Cross Object Formula Fields span two related objects and reference merge fields on those objects. It can be referred to merged fields of parent objects on the child objects.

Cross Object Formula Field is available on both master-detail as well as lookup relationship.

Fields can be referred up to 10 relationships away. Cross Object formulas can be used everywhere except when creating default values.

Difference between custom formula field and Cross Object formula Field

Custom Formula Field	Cross Object Formula Field
1.Only Refers fields of object on which it is created.	Refers fields of related object also.
2.Can be Used in other custom formula field and cross object formula field.	Cannot be used in other custom Formula field.
3.Custom Formula Field can be referred in roll up summary fields.	Cross Object Formula Field Cannot be referred in roll up Summary Fields.
4.It allows security and sharing settings.	It by-passes security and sharing settings.

Roll-Up Summary Fields

Roll-Up Summary Fields in Salesforce summarize data from a set of related detail records and automatically display the output on the master record.

These can be used to display the sum, maximum, minimum value of a field in a related list, also the count of all the records listed in a related list. They are only available to master objects in a master-detail relationship.

After creating a Roll Up Summary Field on an object, the master-detail relationship cannot be converted into a lookup relationship.

Roll up summary field can only be created on the master object in a master detail relationship.

Validation Rule

Validation rules in Salesforce verify the data a user enters in a record. The data should meet the standards specified by the organization. It can contain a

formula or expression that evaluates the data in one or more fields & returns a value true or false.

Validation Rules also include an error message to display to the user when the rule returns a value true due to an invalid value/data.

Schema Builder

Schema Builder provides a dynamic environment for viewing and modifying all the objects and relationships in an app. This greatly simplifies the task of designing, implementing, and modifying the data model, or schema.

It can be used to view existing schema and interactively add new custom objects, custom fields, and relationships, simply by dragging and dropping. This eliminates the need to click from page to page to find the details of a relationship or to add a new custom field to an object in the schema.

Schema Builder provides details such as the field values, required fields, and how objects are related by displaying lookup and master-detail relationships. The fields and relationships can be viewed for both standard and custom objects.

Schema Builder is enabled by default and lets administrators add the following to the schema:

- Custom objects
- Lookup relationships
- Master-detail relationships
- All custom fields except: Geolocation

Page Layout in Salesforce

Page Layout in Salesforce allows us to customize the design and organization of detail and edit pages of records in Salesforce.

Page layouts can be used to control the appearance of fields, related lists, and custom links on standard and custom object's detail and edit page. It also controls which standard and custom buttons will be visible on detail pages and related lists.

Fields can be set as visible, read-only, and required on page layouts.

Note: Page layouts should not be the sole means to restrict access to sensitive data a user should view or edit because it controls only a record's edit page and detail page on other parts of the platform from where the records can be created or updated.

Mini Page Layouts in Salesforce

Salesforce Mini Page layouts contain a subset of the items in an existing page layout. When we hover on the record on recent item we see the fields which are present in mini page layouts.

Each page layout has its own mini page layout.

Here the field access settings and profile associations to page layouts matter.

Compact Layouts

Compact layouts specify the group of fields that are visible on the highlights panel of Salesforce1 for a quick glance on key field values of that record.

It can have a maximum of 10 fields. Compact layouts support all field types except text area, long text area, rich text area, and multi-select picklist.

Salesforce1(Salesforce)

Run your business through your phone. This app is for business users who actively pay for the organization and run their business through the phone.

Note: The changes or customizations made in this app from the Salesforce site gets reflected only after re-logging into the app.

Salesforce A

It only allows administrators to manage the users on the go through the phone. No more functionality is provided for administrators until Spring16.

In simple term we can add the user, we can freeze the user, we can activate the user , we can deactivate the user ,we can change the password of user.

List Views

List view allows us to filter the list of records on an object's tab. List views present already can be edited and new list views can also be created to meet the organization requirements.

It can be user-specific as well as organization-wide.

Salesforce Search Layouts

Search Layouts in Salesforce are used to customize the fields displayed for users in search results, search filter fields, lookup dialogs, recent records list on tab home pages, and in lookup phone dialogs for Salesforce CRM call centre.

By default all lookup dialogs and related lists that result from new relationships only display record names or numbers. To add fields in these

related lists and lookup dialogs, the administrator needs to add fields in search layouts.

Search layouts in Salesforce are an ordered group of fields that are displayed when a record is presented in a particular context such as in search results, a lookup dialog, or in a related list.

Salesforce Search Layout consists of:

Search Results:

The search result originates from searching for a record on the left sidebar or an advanced search.

Lookup dialogs:

The lookup dialogs result that originated from clicking next to a lookup field on an edit page.

Lookup Phone Dialogs:

The lookup dialog result that originates from clicking next to the lookup field with a phone datatype on an edit page.

Object Tab:

The list of recent records that appears on the homepage of a tab and in a related list on another object's detail page.

Object List View:

The layout is not for specifying fields, instead, use it to specify the buttons that appear on the list view page for an object.

Search Filter Field:

The filters that can be applied to search results.

Note: Encrypted, formula, lookup, and roll-up summary fields are not searchable.

Actions

Actions add functionality to Salesforce. We have standard actions such as create and update records. Also, we can create actions based on our company's requirements.

Actions enable users to do more in Salesforce and Salesforce1.

There are 2 types of actions in Salesforce:

- Global Action
- Object Specific Actions

Global Actions in Salesforce:

These can be added to any page that supports action. For example home page, chatter tab, and object pages.

Global create actions enable users to create object records, but the new record has no relationship with other records.

Use global actions in salesforce to let users record call details, create or update records, send an email, or create a task from the Salesforce1 Feeds page, Groups pages, and any other page driven by the global publisher layout.

Action Layouts

Action layouts are similar to page layouts.

Just as object record pages have page layouts that can be customized, actions have action layouts that can be customized. When we create an action, Salesforce populates its layout with a default set of fields. You can add, remove, or reorder fields on the action layout to present only the essential items your users need when they're taking the action.

The first time you view the layout for an action you've created, certain fields are pre-populated target object default fields, standard required fields, and any custom universally required fields.

Action layout can be created for object-specific actions as well as for global action.

Global Publisher Layouts

Global publisher layouts determine the global actions that appear in the various Salesforce interfaces

Salesforce Classic

These layouts customize the actions in Chatter publishers on global pages (like the home page) and on the Chatter page.

Salesforce1

These layouts drive the actions that appear in the action bar on the Feed and People pages. Global publisher layouts can include global actions only.

Note: Global publisher layouts can include global action only.

7

Object Specific Actions Salesforce

Object Specific Actions Salesforce lets users quickly create or update records, log calls, send emails, and more in the context of a particular object.

There are 5 types of object-specific actions:

Create Records:

It creates records that are automatically associated with related records.

Update Actions:

It makes it easy to edit the records. We can define the fields that will be available for updates.

Log a Call:

It lets users enter notes about calls, meetings, or other interactions that are related to a specific record.

Custom Action:

These are Visual Force Pages or canvas app that let users interact with or create records that have a relationship to an object record. The Visualforce page for an object-specific custom action must include the standard controller for the relevant object.

Send email actions:

These are available only in cases. It gives users access to a simplified version of the case feed e-mail action on Salesforce1.

Note: *Action Type cannot be edited once an action is created.*

Record Types in Salesforce

Salesforce Record Types allow us to specify a category of records that display different picklist values and page layouts.

Administrators can associate record types with profiles so that different types of users should see different picklist values and page layouts in the record's detail page.

Note:
When we create a record type on an object then a new standard record id field gets added to its field list. It stores the record type using which the record is created and used whenever the particular record is edited next time.

Data Management

Data management in Salesforce deals with Import/Export of data or records to/from a Salesforce organization.

Salesforce Data Import

Data can be easily imported into Salesforce through various tools provided by Salesforce. Supported data sources include any program that can save data in the comma delimited text format (.csv).

There are 3 operations possible while Importing Data in Salesforce:

1. **Insert:** It simply creates new records in Salesforce.
2. **Update:** It modifies existing records in Salesforce with the help of record id or external id.
3. **Upsert :** It is a combination of insert and update. It modifies the existing records and if the record is not present in the org, it creates a new record for it.

Salesforce offers two main methods for importing data.

Data Import Wizard

This tool, accessible through the Setup menu which lets us import data in common standard objects, such as contacts, leads, accounts, opportunities, as well as data in custom objects.

- It can import up to 50,000 records at a time.
- It provides a simple interface to specify the configuration parameters, data sources, and field mappings that map the field names in your import file with the field names in Salesforce.

Data Loader

This is a client application that can import up to five million records at a time, of any data type, either from files or a database connection.

- It can be operated either through the user interface or the command line.
- In the latter case, you need to specify data sources, field mappings, and other parameters via configuration files.
- This makes it possible to automate the import process, using API calls.
- Use it when you want to save.

Note:

1. *Data Import Wizard only imports Accounts, Contacts, Solutions, leads, and all the custom objects whereas Data loader imports all standard as well as custom objects.*
2. *With both methods, the number of records you can import depends on your permissions, the type of data you're importing, and the overall data storage limits for your organization.*

Data Import methods Comparison:

	In browser import wizards	Data loader
No of records	up to 50000 records	50000+
Catches duplicates	Yes	No
Export data	No	Yes
Import data	Yes	Yes

Important points about Data Import:

This information can help you integrate your imported data into Salesforce.

New Values for Picklists and Multi-Select Picklists:

If our import file contains data to be displayed in picklists or multi-select picklists, the wizard warns you when you attempt to import a new picklist value that does not match any valid picklist values. If you ignore the warning, the new value is automatically added to the imported record. You can later edit the field to add the necessary values.

Multi-Select Picklists:

To import multiple values into a multi-select picklist, separate the values by a semicolon in your import file.

Checkboxes:

To import data into a checkbox field, use 1 for checked values and 0 for unchecked values.

Default Values:

For picklist, multi-select picklist, and checkbox fields, if you do not map the field in the import wizard, the default value for the field, if any, is automatically inserted into the new or updated record.

Date/Time Fields:

Ensure that the format of any date/time fields you are importing matches how they display in Salesforce per your locale setting.

Formula Fields:

Formula fields cannot accept imported data because they are read-only.

Field Validation Rules:

Salesforce runs validation rules on records before they are imported. Records that fail validation aren't imported. Consider deactivating the appropriate validation rules before running an import if they affect the records you are importing.

Universally Required Fields:

You must include universally required fields in your import files or the import will fail.

Salesforce Data Export

Data can be easily exported from Salesforce either manually or on an automatic schedule. The data is exported as a set of comma-separated values (CSV) files. Salesforce Data Export provides a convenient way to export the data, either for backup or for importing into a different system.

Salesforce offers two main methods for exporting data.

Data Export Wizard:

It is an in-browser wizard, accessible through the Setup menu. It allows us to export data manually once every six days (for weekly export) or 28 days (for monthly export). We can also export data automatically, at weekly or monthly intervals using "Schedule Export".

Data Loader:

It is a client application that needs to be installed separately. It can be operated either through the user interface or the command line. The latter option is useful if you want to automate the export process or use APIs to integrate with another system.

Salesforce creates a zip archive of CSV files and emails you when it is ready. To download the zip file follow the link on an email or click on Data Export. Exports will complete as soon as possible, however, Salesforce does not guarantee the date and time the export will get complete. Large exports are broken up into multiple files. Zip files are deleted 48 hours after the email is sent.

Security Token

Accessing Salesforce from outside the trusted IP range of the organization using the desktop client or the API requires a security token to log in. It is a case-sensitive alphanumeric code that needs to be appended to the password or entered as a separate field in the client application or the API. Security token gets reset automatically every time the password is changed.

External ID

External ID in Salesforce is a custom field that has the "External ID" attribute checked meaning that it contains unique record identifiers from a system outside of Salesforce. When we select this option the import wizard will detect existing records in Salesforce that have the same External Identification.

This operation is case-insensitive but if the custom field has a separate "Unique" attribute then the case sensitive option for that field is selected which means Uppercase and Lowercase letters will not be considered identical.

An object can have at most 25(3,7) External IDs' fields. The field type should be any one of auto-number, email, number, or text. Custom fields marked as unique also count against an object's limit of 25 External IDs' fields.

Record ID

Each record in the Salesforce.com system has a unique ID field assigned to it which is known as Record ID. It is system generated and cannot be edited or deleted. It is generated every time a new record is inserted into the application.

Entering a known Record ID in Salesforce will direct you straight to the details page of that record. The first 3 characters of an ID indicate the object type.

For example: Accounts – 001, Contacts – 003, Leads – 00Q, etc.

These prefixes can never be changed for objects.

Custom objects are assigned 3-char prefixes based on internal SFDC rules that we cannot predict.

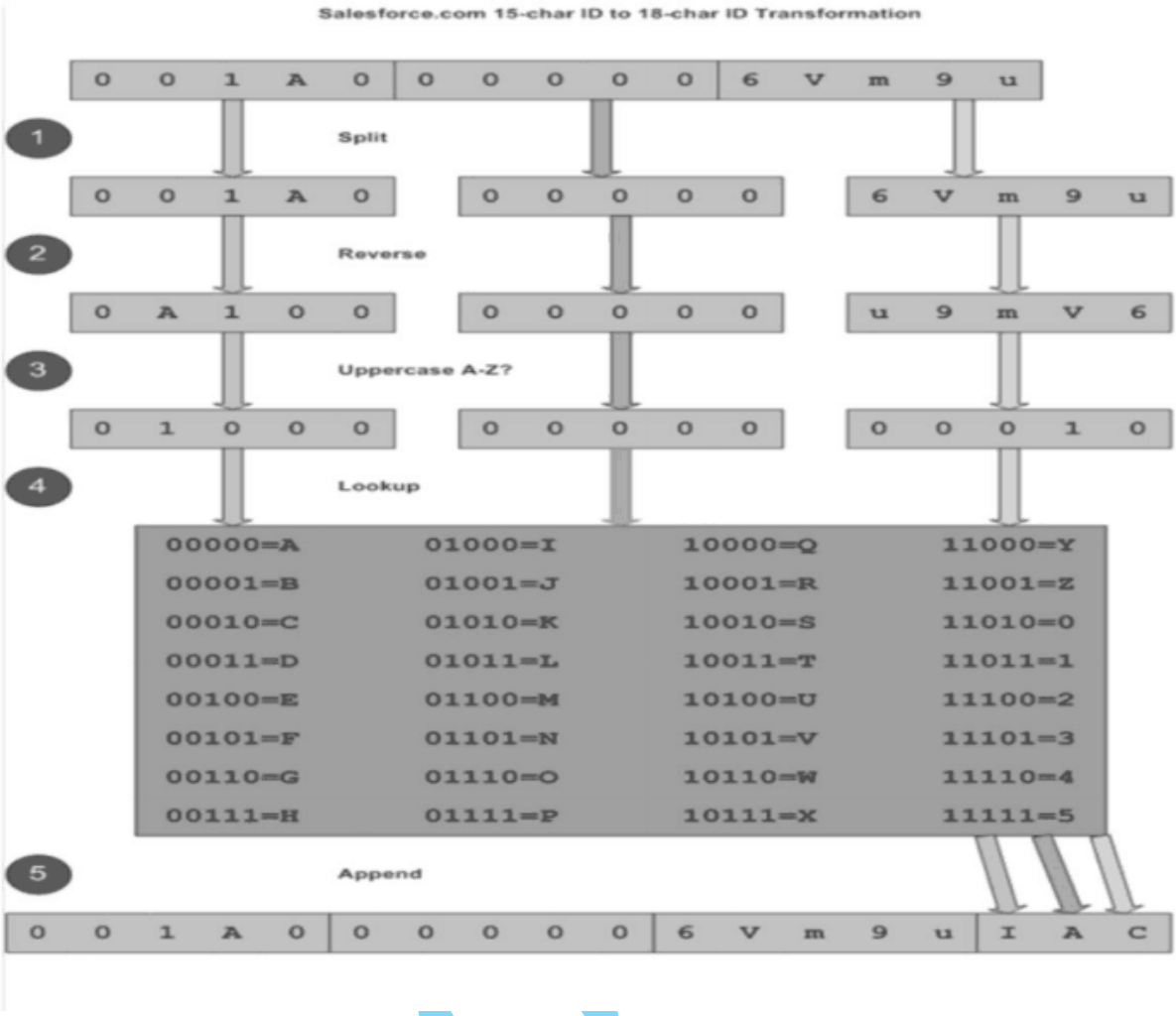
There are two versions of Record IDs:

1. **15-digit case-sensitive version:** It is referenced in the user interface.
2. **18-digit case-insensitive version:** It is referenced through the API. The last 3 digits of it are the checksum of the capitalization of the first 15 characters.

A 15-digit id can be converted into an 18-digit id using CASESAFEID() function and to convert an 18-digit id into 15-digit simply remove the last 3 digits from it.

Note:

Salesforce Record IDs are identical only in the Full Copy Sandbox and Production environments.


Salesforce.com 15-char ID to 18-char ID Transformation
0 0 1 A 0 0 0 0 0 0 6 V m 9 u
1
Split
0 0 1 A 0 0 0 0 0 0 6 V m 9 u
2
Reverse
0 A 1 0 0 0 0 0 0 0 u 9 m V 6
3
Uppercase A-Z?
0 1 0 0 0 0 0 0 0 0 0 0 0 1 0
4
Lookup
00000=A 01000=I 10000=Q 11000=Y
00001=B 01001=J 10001=R 11001=Z
00010=C 01010=K 10010=S 11010=0
00011=D 01011=L 10011=T 11011=1
00100=E 01100=M 10100=U 11100=2
00101=F 01101=N 10101=V 11101=3
00110=G 01110=O 10110=W 11110=4
00111=H 01111=P 10111=X 11111=5
5
Append
0 0 1 A 0 0 0 0 0 0 6 V m 9 u I A C

Data Security in Salesforce

Salesforce Data security deals with the security or sharing settings of data and visibility between users or groups of users across the organization. Force.com platform provides a flexible, layered sharing model that makes it easy to assign different data sets to different sets of users.

Security and Sharing model can be configured entirely using the user interface yet it is implemented at the API level which means any permissions specified for objects, records and fields apply even if a user query or update the data via API calls.

Level of Data Access

The data access on Salesforce is configured in four levels, the following are:

1.Organization Level

The access to the whole organization is secured at this level by maintaining a list of authorized users, setting password policies, and limiting login access to certain hours and certain locations.

2.Object Level

Object-level security provides the simplest way to control which users have access to which data. By setting permissions on a particular type of object, you can prevent a group of users from creating, viewing, editing, or deleting any records of that object.

3.Field Level

Field Level security restricts access to certain fields, even for objects a user already has access to.

4.Record Level

Record Level security lets users access some records but not others. It is used to control data access with greater precision. Users can have access to view an object, but can be restricted to the individual records.

Note:

Always make a table for various types of users and the level of access of data that each user has in your organization to implement a security and sharing model.

Control Access to the Organization

Access to organization (Salesforce User Management) can be restricted by four means:

1. Allowing only authorized users to access Salesforce
2. Setting Password Policies
3. Restricting IP ranges for Users.
4. Restricting Login Hours for Users.

User Management in Salesforce (User Management)

A user is anyone who logs in to Salesforce. Users are employees in your organization. Every user in Salesforce has a user account. The user account identifies the user, and the account settings determine what features and records the user can access.

Each user account contains at least the following:

Usernames:

It must be unique across all Salesforce organizations.

User Licenses:

It determines which features the user can access in Salesforce. For example, you can allow users access to standard Salesforce features and Chatter with the standard Salesforce license. But, if you want to grant a user access to only some features in Salesforce, you have a host of licenses to choose from. For example, if you have to grant a user access to Chatter without allowing them to see any data in Salesforce, you can give them a Chatter Free license.

Profiles:

It determines what users can do in Salesforce. Profiles should be selected based on a user's job function.

Roles:

It determines what users can see in Salesforce based on where they are located in the role hierarchy. These are optional but each user can have only one.

Alias:

An alias is a short name to identify the user on list pages, reports, or other places where their entire name doesn't fit. By default, the alias is the first letter of the user's first name and the first four letters of their last name.

User records in Salesforce can't be deleted, it can only be deactivated or frozen.

Deactivate a User	Freeze a User
User record cannot be deleted so in order to stop the user from logging in to Salesforce organization administrators need to deactivate them.	A user record cannot be deactivated immediately such as when a user is selected in a custom hierarchy field. So to prevent the user from login in to the organization while administrators perform the steps to deactivate them, they can simply freeze that user's record.
Deactivating user records frees up the license assigned to the user. So that now new users can use that license in order to access Salesforce platform features.	Freezing a user does not frees the license assigned to the user.

Managing Salesforce Password Policies (Control Access to Organization)

Password policies in Salesforce are configured to ensure that the user's password is strong and secure. Managing Password Policies is important.

There are several settings to ensure this:

Password policies:

Set password and login policies, such as specifying an amount of time before all user's passwords expire and the level of complexity required for passwords.

User password expiration:

Expire the passwords for all the users in your organization, except for users with "Password Never Expires" permission.

User password resets:

Reset the password for specific users.

Login attempts and lockout periods:

Specifies the number of attempts a user can make and if a user is locked out due to too many failed login attempts, the administrator can unlock its access.

Password Policy can be set into two levels

1. *Organization level*
2. *Profile level(it is for all the user who is in a single profile).*

Restrict Login Access by IP Address (Control Access to Organisation)

By default, Salesforce doesn't restrict the location for login access. However, for added security, administrators can restrict login access by IP.

Administrators can specify an IP address range for the entire organization as well as for specific user profiles, but the behaviour is very different for each option.

If login IP range is set at:

Organization level:

Users who log in outside the set range are shown a login challenge. If they complete the challenge question, typically by entering an activation code sent to their mobile device or email address, login access is granted. This method does not restrict access, entirely, for users outside of the set IP range. Here the set IP range is called the "trusted" IP range.

Profile level:

Users outside the permitted range are always denied for the access.

Restrict Login Access by Time (Control Access to Organization)

By default, Salesforce doesn't restrict the time for login access. However, for added security, administrators can restrict it.

Restricting login access by time can only be achieved at the profile level. For each profile, administrators can specify the hours when users can log in. For example, for employees who only need to access customer data during business hours, you can deny login access during evening hours and weekends.

If users are logged in when their login hours end, they can continue to view their current page, but they can't take any further action.

Object Level Security in Salesforce

Salesforce Object Level Security provides the simplest way to control data access. It prevents a user or group of users from creating, viewing, editing, or deleting any records of an object by setting permissions on that object.

There are two ways of setting object permissions:

1. Profiles:

It determines the objects a user can access and the permissions a user has on any object record.

2. Permission Sets:

It provides additional permissions and access settings to users.

Profiles

Profile is a collection of settings and permissions that determine which data and features in the platform users have access to.

Settings determine what users can see for example apps, tabs, fields, and record types whereas Permission determine what users can do for example create or edit records of a certain type, run reports and customize the app.

Profiles Control:

- Object Permission
- Field Permission
- User Permission
- Tab Settings
- App Settings
- Apex class access
- Visualforce page access
- Page Layouts
- Record Types
- Login Hours
- Login IP Ranges

Profiles are typically defined by a user's job function but anything that makes sense in an organization can be created as a profile. The platform includes a set of standard profiles. Each of the standard profiles includes a default set of permissions for all of the standard objects available on the platform.

Some of them are:

1. Standard User:

Standard User profile has Read, Edit, and Delete permissions to most standard objects.

2. Read Only:

Read-only user had permissions exactly similar to standard user but limits the access to read-only.

3. Marketing User:

Permissions of Standard User+ Additional Permissions.

4. Contract Manager:

Permissions of Standard User + Additional Permissions.

5. Solution Manager:

Permissions of Standard User + Additional Permissions.

6. System Administrator:

The System Administrator profile has the widest access to data and the greatest ability to configure and customize Salesforce. The System Administrator profile also includes two special permissions namely “View All Data” and “Modify All Data”.

When a custom object is created most profiles except those with modify all data permission do not give access to that custom object.

Note:

- *Object permissions on the Standard profile cannot be edited.*
  -So to overcome through this it is good to make copies/clones of standard profiles and then customize the copies to fit the needs of the organization.
  -The profile functionality in an organization depends on the user license type.
- *Every profile should have at least one visible app.*
- *If an app is visible, its tab won’t show up unless a profile has permissions to view the associated objects.*
- *A profile can be assigned to many users but the user can be assigned to only one profile at a time.*

Salesforce Permission Sets

Permission sets in Salesforce are also a collection of settings and permissions that determine user’s access to various tools and functions on the platform.

Settings and permissions available in permission sets are also found in profiles but permission sets extend the functionality of users without changing their profiles.

Use permission set to grant additional access to specific users on top of their existing profile permissions, without having to modify an existing profile, create new profiles, or grant an administrator profile where it’s not necessary.

Permission Set Control:

- Object Permission
- Field Permission
- User Permission
- Tab Settings
- App Settings
- Apex class access

- Visualforce Page access

There are a couple of ways to use the Permission Set in Salesforce:

1. To grant access to custom objects or entire apps.

2. To grant permissions-temporarily or long term-to specific fields

Permissions are additive which means we can't remove a user's existing permissions by assigning a permission set we can only add permissions. To limit access for a user or group of users, ensure that their base profile as well as any of their permission set limits this type of access.

The license type cannot be changed once assigned. And it is not mandatory while creating the permission set.

Profile	Permission Sets
Profiles have most restrictive settings and permission a user assigned to this profile should have.	Permission Sets extend the access settings and permissions provided by the profile.
A user can have only one profile assigned.	Users can have more than one permissions sets.
Profiles are restrictive.	Permission Sets are additive.
Every user must be assigned to a profile.	It is not necessary for every user to have a permission set.

Salesforce Field Level Security

Field level security in salesforce controls whether a user can see, edit or delete the value for a particular field on an object, unlike page layouts which only control the visibility of the field on detail and edit pages of an object. It secures the visibility of fields in any part of the app including related lists, list views, reports, and search results.

Field level security can be applied to multiple fields on a single profile or permission set and can also be applied to a single field on all profiles.

Salesforce Record Level Security

Record Level Security in Salesforce determines which individual records users can view and edit in each object they have access to in their profile.

The permission on a record is always evaluated according to a combination of object, field, and record-level security permission. When object- versus record-level permissions conflict, the most restrictive settings win.

To implement it the administrator need to answer the following questions:

- Should the users have open access to every record or a subset?
- If it's a subset then what rules should determine whether the user can access them?

Salesforce provides 4 ways to implement it:

- Organization-Wide Default.
- Role Hierarchy.
- Sharing Rule.
- Manual Sharing.

Organization-Wide Default

Organization-Wide default or Organization-Wide sharing settings determine the baseline level of access for all records of an object. Organization-wide defaults can never grant users more access than they have through their object permissions.

Organization-Wide defaults should be most restrictive in record level security because other record-level security implementations only grant additional accesses, they cannot restrict the access of records provided by Organization-Wide defaults.

Organization-Wide defaults can be set to any of the 3 below:

1. Public Read/Write:

All users can view, edit, and report on all records.

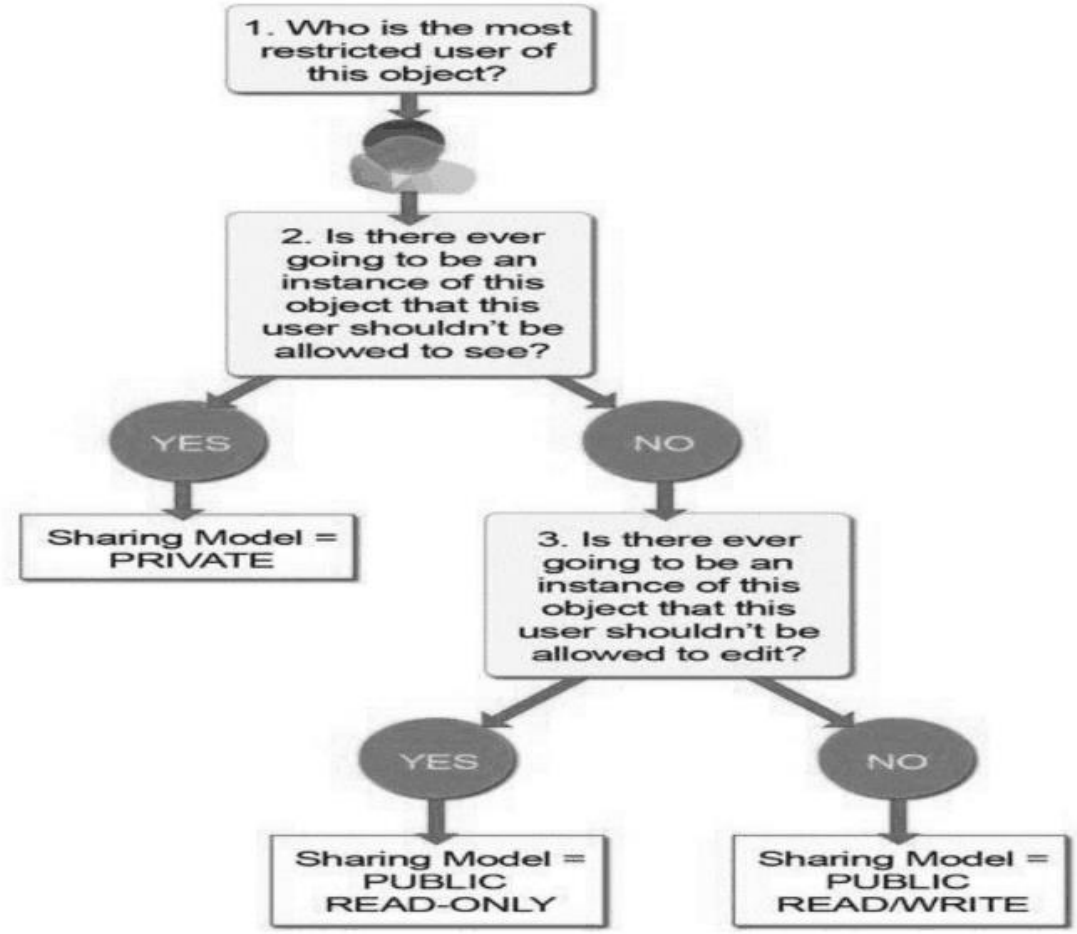
2. Public Read-Only:

All users can view and report on records but not edit them. Only the owner, and users above that role in the hierarchy, can edit those records.

3. Private:

Only the record owner, and users above that role in the hierarchy, can view, edit, and report on those records.

To determine the Organization-wide default of an object consider the below diagram:



The data may be too restrictive for some users according to org-wide defaults but it can be opened for users who need it using role-hierarchies, sharing rules, and manual sharing. A sharing recalculation gets started to apply access changes to records whenever an update is made for Organization-Wide Default settings. An email is sent by Salesforce whenever it gets completed or we can see the update on Setup Audit Trail as well.

Salesforce Role Hierarchy

Every Salesforce organization maintains a role hierarchy for the organization using Salesforce. This role hierarchy defines the hierarchy of the users working in the organization.

Salesforce Role Hierarchies can be used to extend the record access automatically so that:

Manager:

A Manager will always have access to the same data as his/her employees regardless of the organization-wide default settings.

Users:

Users who tend to need access to the same types of records can be grouped. These groups can be used as Roles & Sub-Ordinates in Sharing Rules.

Role hierarchies don't have to match your organization chart exactly. Instead, each role in the hierarchy should just represent a level of data access that a user or group of user's needs.

Depending on the organization's sharing settings, roles can control the level of visibility that users have into the organization's data. Users at any given role level can view, edit, and report on all data owned by or shared with users below them in the role hierarchy, unless the organization's sharing model for an object specifies otherwise.

Note:

- *If the "Grant Access Using Hierarchies" option is disabled for a custom object then only record owners and users granted access by the organization-wide defaults have access to the object's records. However users such as with the "View All" and "Modify All" object permissions and the "View All Data" and "Modify All Data" system permissions can still access records they do not own.*
- "Grant Access Using Hierarchies" option is enabled for all objects and it can only be changed for custom objects.

Public Groups

A public group consists of a set of users. It can contain individual users, other groups, or the users in a particular role or territory plus all the users below that role and subordinates in the hierarchy.

Salesforce Sharing Rules

Sharing rules in Salesforce are used to create automatic exceptions to the Organization-Wide Default settings for the users who do not own the record.

They should be applied to the objects whose org-wide defaults are set to Public Read-only or Private because sharing rules can only extend the access they cannot restrict the access provided by Organization-wide defaults.

There are 2 types of Sharing Rules in Salesforce based on which records to be shared:

1. Owner Based:

Owner based shares the records owned by certain users. Owners can be identified through public groups, roles and roles, and sub-ordinates.

2. Criteria Based:

Criteria based shares the records that meet certain criteria.

Before creating them, administrators need to answer these 3 questions:

1. Share which records?

This identifies the records that need to be shared. They can be categorized based on the owner of the records or the criteria that records met.

2. With whom the records need to be shared?

Records can be shared with public groups, roles, and roles & subordinates.

3. What kind of access should be provided for these records?

The users with whom the records are shared should have Read-Only or Read/Write access is decided by this question.

Note:

- *As they cannot restrict access this is the reason it gives Read-Only and Read/Write as access parameters in sharing rules.*
- *They work best when they are defined for a particular group of users that can be determined or predicted in advance rather than a set of users that frequently changes.*

Manual Sharing

In Salesforce Manual Sharing, records are shared individually with other users by using the share button on the record. Sometimes it is not possible to define a consistent group of users who need access to a particular record that is where manual sharing comes in. Salesforce Manual Sharing allows the users to share the record to users who would not have access to the record any other way.

Only these 4 users can share the record:

- Record Owner
- A user in a role above the owner in the role hierarchy.
- Users granted “Full Access” to record.
- Administrator

Sometimes granting access to records also includes access to its associated records. The sharing button is available when your sharing model is either “Private” or “Public Read-Only” for a type of record or related record. Records can be shared manually with groups, roles, roles & subordinates, and individual users.

There are 4 access levels that determine the access provided to users

1. Full access:

Users with full access can view, edit, delete, and transfer the record. These users can also extend sharing access to other users. Users cannot grant full access to other users.

2. Read/Write:

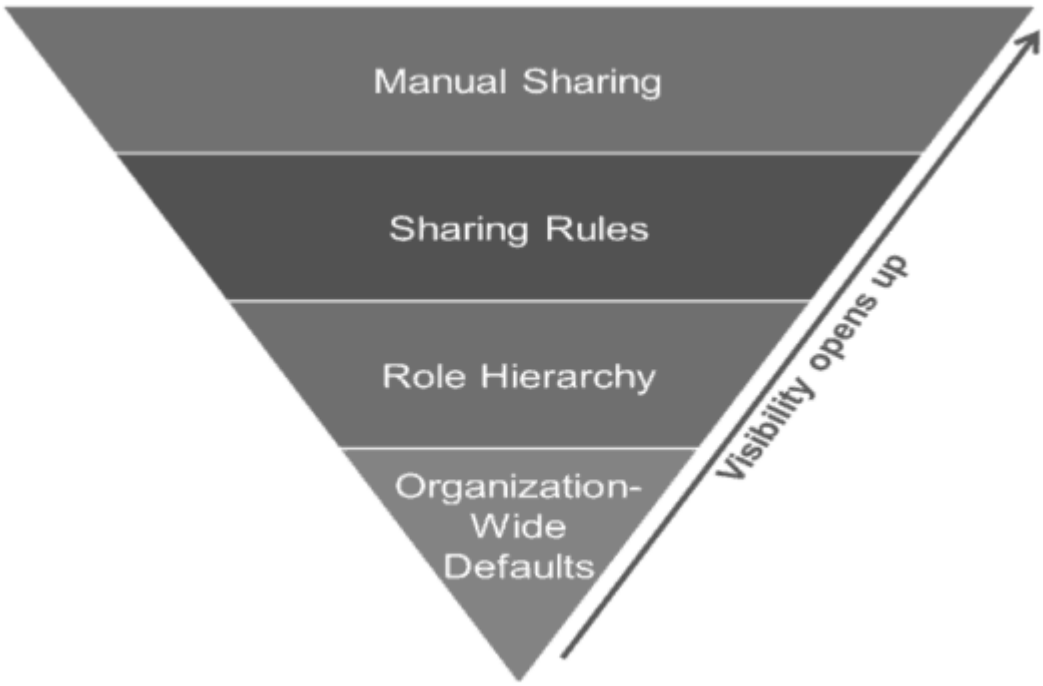
Users can view and edit the record and add associated records, notes, and attachments to it.

3. Read Only:

Users can view the record and add associated records to it. They cannot edit the records or add notes and attachments.

4. Private:

Users cannot access the record in any way.



Different Types of groups, users, roles & territories:

Type	Description
Managers Groups	All direct and indirect managers of a user.
Manager Subordinates Groups	A manager and all direct and indirect reports who he or she manages.
Public Groups	All public groups defined by your administrator.
Personal Groups	All personal groups defined by the record owner. Only the record owner can share with his or her personal groups.
Users	All users in your organization. Does not include portal users.
Roles	All roles defined for your organization. This includes all of the users in each role.
Roles and Subordinates	All of the users in the role plus all of the users in roles below that role in the hierarchy. Only available when no portals are enabled for your organization.

Queues in Salesforce

Salesforce Queues are a collection of records that doesn't have any owner. Users who have access to the queue can examine every record that is in it and claim ownership of the one's they want. Queues in Salesforce help to prioritize, distribute, and assign records to teams who share workloads.

Queues are traditionally used in sales and support organizations to distribute new leads and support cases to the employees who have the most availability because the platform natively supports queues for leads, cases, service contracts, and any custom objects.

Records can be added in the Queue manually by changing the record's owner. Also, assignment rules can add cases or leads to a queue based on criteria. Records remain in a queue until they are assigned an owner or a queue member volunteers to own them. Any queue member or users higher in the role hierarchy can take ownership of records in a queue.

Users of a Queue can be changed by

1. Admin
2. The person with “Manage [Object Name]” permission privileges

The OWD sharing model for an object determines the access users have to that object’s records in queues.

Public Read/Write/Transfer

Users can view & take ownership of records from any queue.

Public Read/Write or Public Read-Only

Users can view any queue but only take ownership of records from the queue of which they are a member or depending on the sharing setting if they are higher in a role hierarchy than a queue member.

Private

Users can only view and accept records from Queues of which they are a member or depending on sharing settings if they are higher in the role hierarchy than a queue member.

Note:

- *Regardless of the sharing model users must have the edit permissions to take ownership of records in queues of which they are the member.*
- *Admin, users with “Modify All” object-level permission for the respective object and users with “Modify All Data” permission can view and take records from any queue regardless of the sharing model or their membership in the queue.*
- *Before deleting a queue, reassign its records to another owner and remove it from any assignment rules.*

Process Automation

In Salesforce several tools automate the organization’s business processes.

Automating the business process is crucial so as identifying the tool from which it should be implemented.

There are several tools for automating the processes in Salesforce:

Workflows in Salesforce

Workflow in Salesforce is a force.com platform business logic engine that allows you to define rules to automate certain actions when a particular criteria is met.

Workflow rules are considered as a container that consists of two components:

a. Criteria:

It consists of the condition which if found true then the actions should take place.

b. Actions:

It consists of the actions that should be performed when the criteria for workflow rule are met.

Every Workflow rule must be based on a single object because when a rule is defined this object influences the fields that will be available to set the criteria.

There are 3 types of evaluation criteria for Workflows in Salesforce:

Evaluate the rule when a record is	Description
1. Created	Runs the rule if the rule criteria is met, only when a record is created. Using this option the rule runs single time only.
2. Created & Every time its edited	Runs the rule if the criteria is met every time a record is created and edited. With this option the rule criteria runs repeatedly as long as the rule criteria is met. **Note:** Time Dependent actions cannot be added in a workflow rule with this evaluation criteria.
3. Created and Every time its edited to subsequently meet the criteria	Runs the rule if the criteria is met: • Always when a record is created and meets the criteria. • Only when a record not meeting the criteria is updated and now it meets the criteria. By default this criteria is selected.

Criteria for rules can be represented as a condition or a formula.

Actions in workflow rules are of 2 types based on when they are executed:

1. Immediate Actions:

These actions are executed immediately when the criteria for a rule is met.

2. Time-Triggered Actions:

These actions execute at a specific time which is specified when creating this type of action.

In these types of actions when the date or time comes on which the action needs to be executed, it again verifies that the record is still meeting the criteria or not. If yes then it executes the respective action otherwise not.

There are 4 actions which can be specified in a workflow rule:

1. Email Alert:

It sends automated emails using an email template.

Salesforce provides functionality to create email templates that can be used to send emails.

There are 4 types of email templates:

a. Text:

All users can create or change text email templates.

b. HTML with letterhead:

Administrators and Users with "Edit HTML Templates" permission can create HTML email templates based on a letterhead.

c. Custom HTML:

Administrators and users with "Edit HTML Templates" permission can create HTML email templates without using a letterhead.

d. Visualforce:

Administrators & Developers can create templates using Visualforce. Visualforce email templates allow for advanced merging with a recipient's data where the content of a template can contain information from multiple records.

Email templates are stored in folders. There are 2 folders already present in every Salesforce org:

Unified public Email Templates

Email Templates in this folder are available to every user in the organization.

My Email Templates

This folder stores the email templates which are used personally by a particular user.

Note:

If another object's field is added in an email template then it will be left blank as workflows only have access to the object for which it is applied.

2. Field Update:

Field updates allow automatic change of a field value on the record that initially triggered the workflow rule. If "Re-evaluate Workflow Rules after Field

Change" is enabled for a field update action then Salesforce re-evaluates all workflow rules on the object if the field update results in a change to the value of the field.

Only workflow rules that didn't fire before will be retriggered. Cross Object field updates are also available for changing the values of fields on a related master record. But it is not available when re-evaluation is checked.

3. Tasks:

It simply assigns a task to a single user, owner, or a role. We can also make sure that a notification email is sent to the assignee when a task is automatically assigned. Tasks should be assigned to roles if they have only one user assigned to that role. If there are more than one user assigned to that role then the task automatically gets assigned to the owner of the workflow rule Users who triggered the workflow rule.

4. Outbound Message:

An outbound message sends particular information to a designated endpoint such as an external service. These types of messages can be listened to using a SOAP API.

All the above actions can be used as an immediate action as well as time trigger actions. Time-triggered actions which are already triggered are visible in Time Based Workflow Queues.

Administrators can also set default workflow users. Default Workflow users are the users that will be visible when the user that triggered the rule is not active.

Notes:

- *Time trigger actions cannot be added to active salesforce workflows. To add a time trigger action first deactivate the workflow rule then add an action.*
- *Another time trigger cannot be added if there is already a time trigger action scheduled for that object and present in a time-based workflow queue.*
- *The action a workflow rule takes can also trigger the execution of other workflow rules.*

Approval Process

Approval Process in Salesforce is an automated process that an organization uses to approve records in Salesforce. Records submitted for approval are approved by the users in the organization. These users are called as Approvers. It is bound to a single object because when a rule is defined this object influences the fields that will be available to set the criteria.

There are 2 options to set the criteria:

- As a condition where criteria are met.
- As a formula that evaluates to true.

They can be left blank if all the records that are submitted for approval should enter the approval process in salesforce. Automated Approval routing routes the approval request automatically to user-specified in Manager Field or any hierarchical relationship field in submitter's user record.

Note:

If we do not select Manager Field or any hierarchal field of user record at the time of approval process definition then the automated approval routing for any approval step in this process will not be enabled.

If the "Use Approver field of [Object] owner" is checked then in Approval steps the Manager Field or any hierarchical field whichever may be selected will be of record owner instead of the submitter.

Records when submitted for approval automatically get locked which means they cannot be edited. However, administrators can edit the records during an approval process in salesforce, and also when defining the approval process administrators can select whether the approver is allowed to edit the record or not.

We can also assign a notification email template which will send an email notification to approvers whenever a record is submitted for approval.

Similar to Action Layouts in Actions, approval processes also have Approval Page Layouts in which fields that will be visible to approver are specified.

Note:

Standard Name field and Owner field cannot be removed from Approval Page layouts.

Approval History information can be displayed as a related list within a record. Users who can submit the records for approvals are called as submitters. We can specify the submitters in the approval process definition. Also, there is an option that gives additional functionality using which the submitters can recall the approval requests.

Actions in the Approval Process:

There are 4 actions present except the approval steps which completes an approval process, following are:

1. Initial Submission Actions:

Initial submission actions are the actions that occur when a user first submits a record for approval. By default, an action to lock the record runs automatically

on initial submission. Initial submission actions can include any approval actions such as email alerts, field updates, tasks, or outbound messages.

2. Final Approval Actions:

Final Approval actions are the actions that occur when a record is approved from all the approval steps. It also locks or unlocks the record, as specified. It can include any approval actions such as email alerts, field updates, tasks, or outbound messages.

3. Final Rejection Actions:

Final Rejection actions are the actions that occur when a record is rejected from any of the approval steps. It also locks or unlocks the record, as specified. It can include any approval actions such as email alerts, field updates, tasks, or outbound messages.

4. Recall Actions:

Recall actions are the actions that occur when a record is recalled after submission for approval. It can include any approval actions such as email alerts, field updates, tasks, or outbound messages.

Note: *Initial submission default actions cannot be edited but final approval & rejection default actions can be edited.*

Approval Steps:

The approval process consists of one or more than one approval steps. Every approval step contains its criteria and approvers.

Approval Steps are given step numbers to identify the hierarchy in which the approval steps should execute. Every step consists of a criteria which qualifies that the record needs to get into this step for approval or not.

If the record does not fulfil the criteria for any step except the first step then the record gets approved and executes Final Approval actions after it.

In approval step 1 we can choose whether the record should get approved or rejected if does not fulfil the criteria.

Approvers of approval step can be:

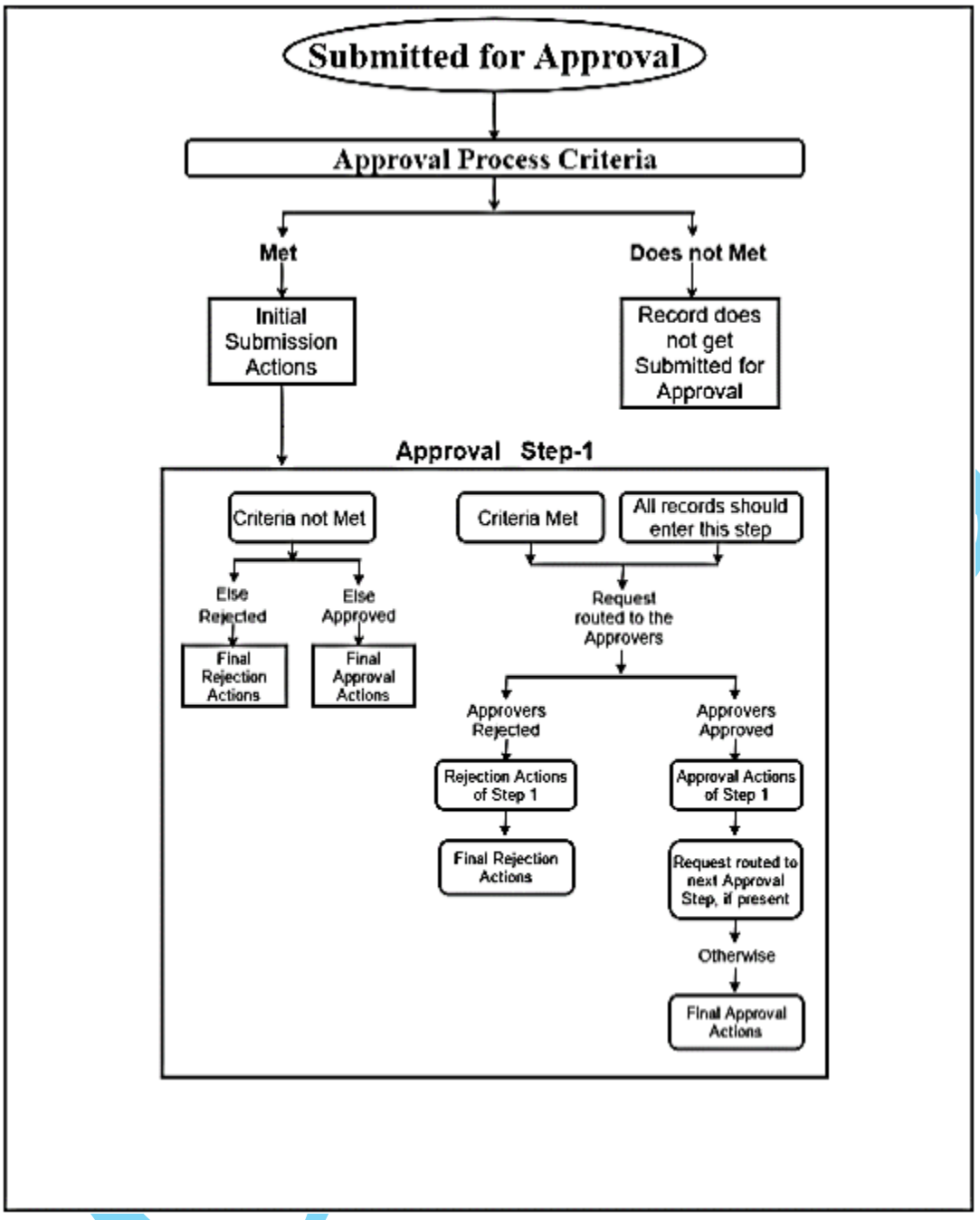
- Manager Field of user record or any other hierarchal field.
- Submitters can also choose manually the approver of that step.
- The record can automatically be assigned to a queue for approval.
- Or we can set a particular user or a queue for a particular approval step. In this option, there are 2 options that decides how the approver/approvers will decide whether the record is approved or not.

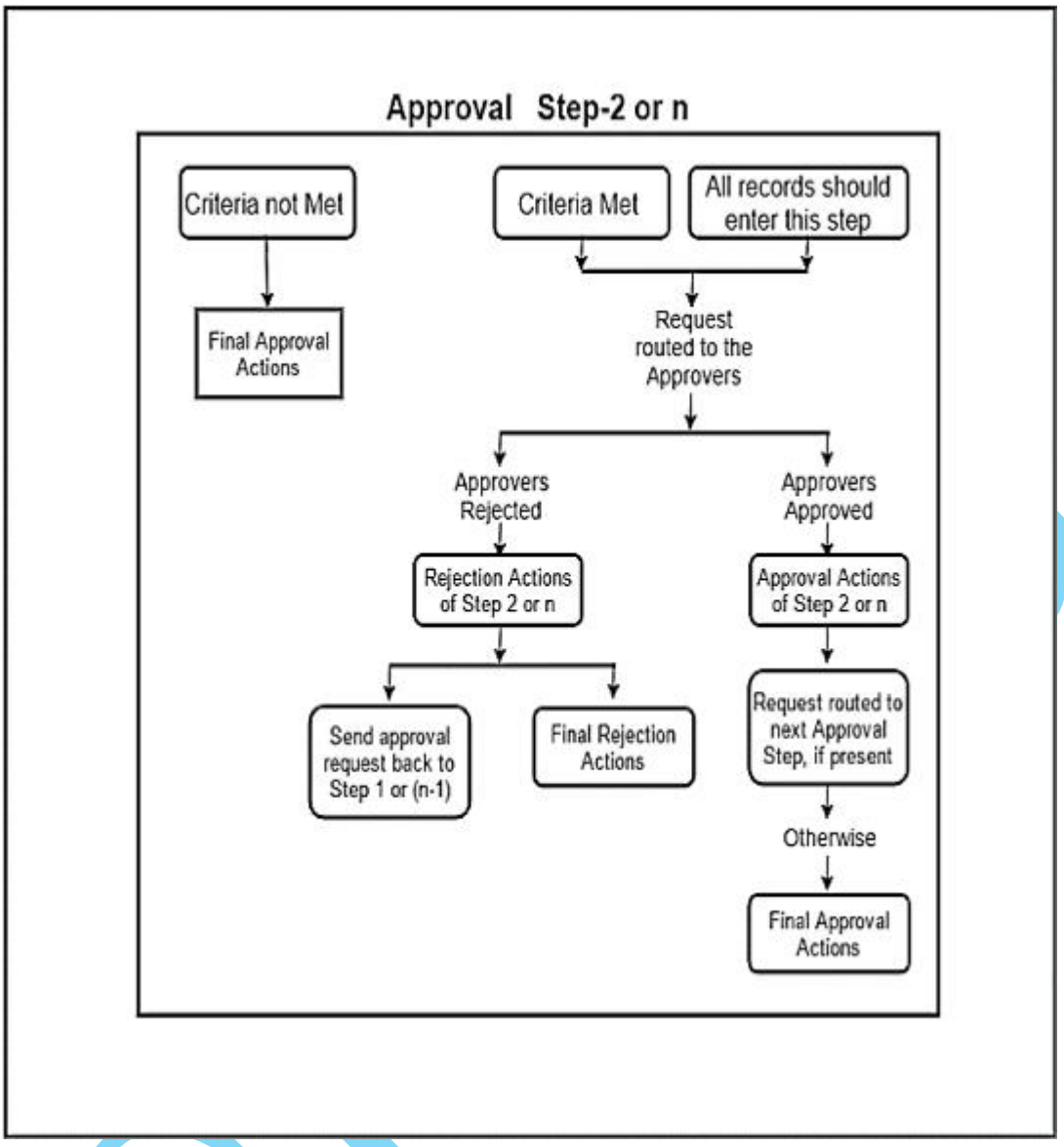
1. Approve or reject based on the FIRST response. Any of the approver if approves the record it is considered to be approved.
2. Require UNANIMOUS approval from all selected approves. If any of the approves rejects the record it is considered to be rejected.

Delegated approvers can also approve the record if they are allowed to approve explicitly in the approval step definition. Approvers can also reassign the approval request to another user. Every approval step contains its approval and rejection actions. These actions can be any amongst Email Alert, Task, Field Update, and outbound message. In order to activate an approval process, there should be at least one approval step.

Note:

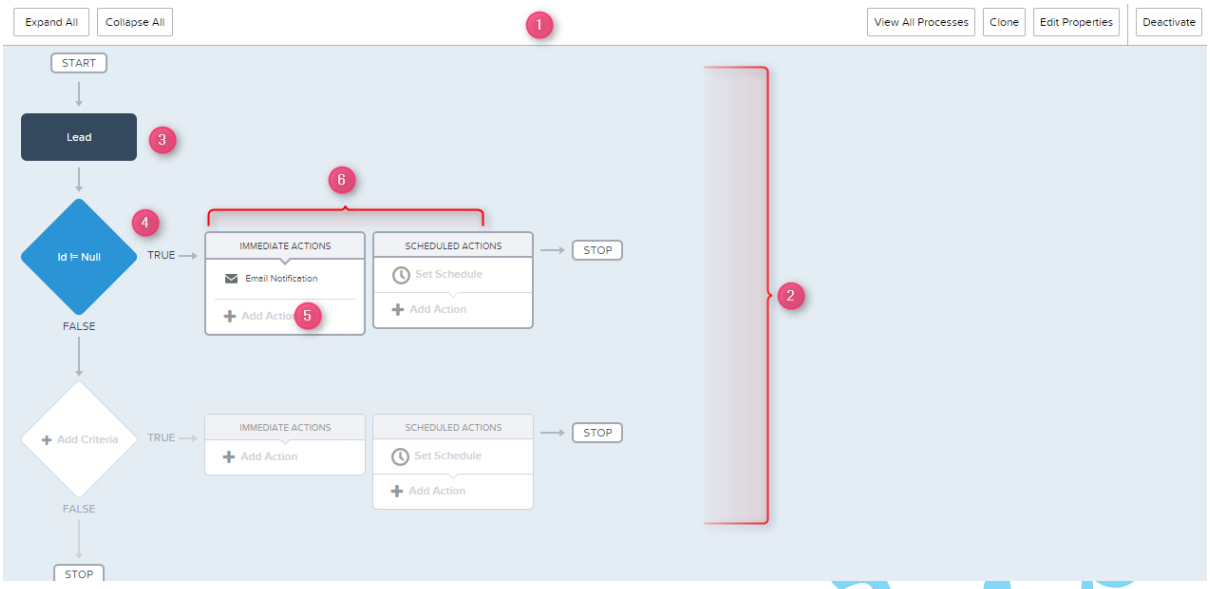
Once an approval process is activated no more steps can be added into it.


Submitted for Approval
Approval Process Criteria
Met
Does not Met
Initial Submission Actions
Record does not get Submitted for Approval
Approval Step-1
Criteria not Met
Criteria Met
All records should enter this step
Else Rejected
Else Approved
Request routed to the Approvers
Final Rejection Actions
Final Approval Actions
Approvers Rejected
Approvers Approved
Rejection Actions of Step 1
Approval Actions of Step 1
Final Rejection Actions
Request routed to next Approval Step, if present
Otherwise
Final Approval Actions



Process Builder

Process builder is used to automate more complex business process just by a few point and click actions. It provides a user – friendly representation for building a process in Salesforce.



1. Button Bar

Use the button bar to collapse and expand actions, clone the process and activate/deactivate the process.

2. Canvas

Canvas is a main workplace for the process

3. Add Objects

The records that the process should evaluate

4. Criteria nodes

Entry criteria node includes conditions that are used to evaluate the records.

5. Actions.

If the criteria met for the record that starts the process, the actions start immediately or at a scheduled time.

Process Builder is the next version of a workflow tool. You can implement all the workflow actions except outbound message, using Process Builder. Also, Process Builder provides some additional features to accomplish complex flows.

1. Create a Record

- Creates a record by manually entering values or by using the values of related records.

2. Update any related record

- Updates one or more records—that are related to the record that started the process—by manually entering values or by using the values from related records.

3. Quick Actions
   - Create a record, update a record, or log a call by using an object-specific or global action that you or another administrator created for your organization
4. Launch a Flow
   - Launches a flow from your process to automate complex business processes—creates flows to perform logic and enables events trigger the flows via processes—without writing code.
5. Send an Email
   - Easily send an email from a process by using an email alert. Email alerts are configured outside of the Process Builder and contain the standard text, list of recipients, and template for the email.
6. Post to Chatter
   - Post to the feed of a user, a Chatter group, or the record that started the process.
7. Submit for Approval
   - Submit the record that started the process for approval.
8. Apex
   - Now, you can call an Apex from process builder for a particular scenario without going for trigger and Visualforce.
   - And also that Apex should have @invokeablemethod.

SCENARIO:

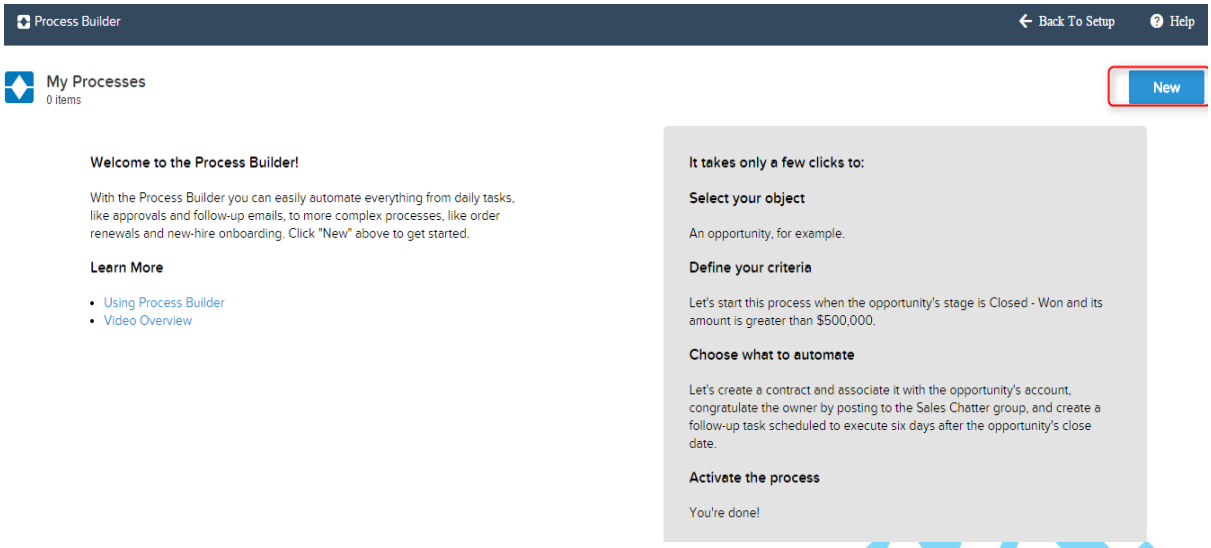
If an Opportunity is closed Won, create a Contract Record and Submit this record for manager review. If the Opportunity is Closed Lost, Send an Email notification to VP.

In this scenario, the following actions will take place:

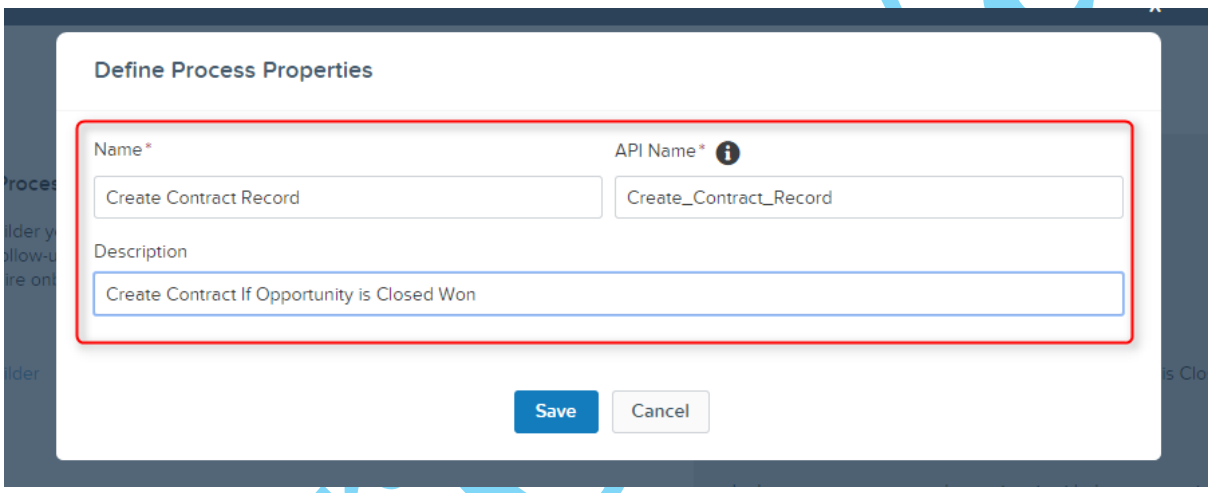
1. Create a New Record
2. Submit for Approval
3. Email Alert.

Please follow the below steps to accomplish the scenario.

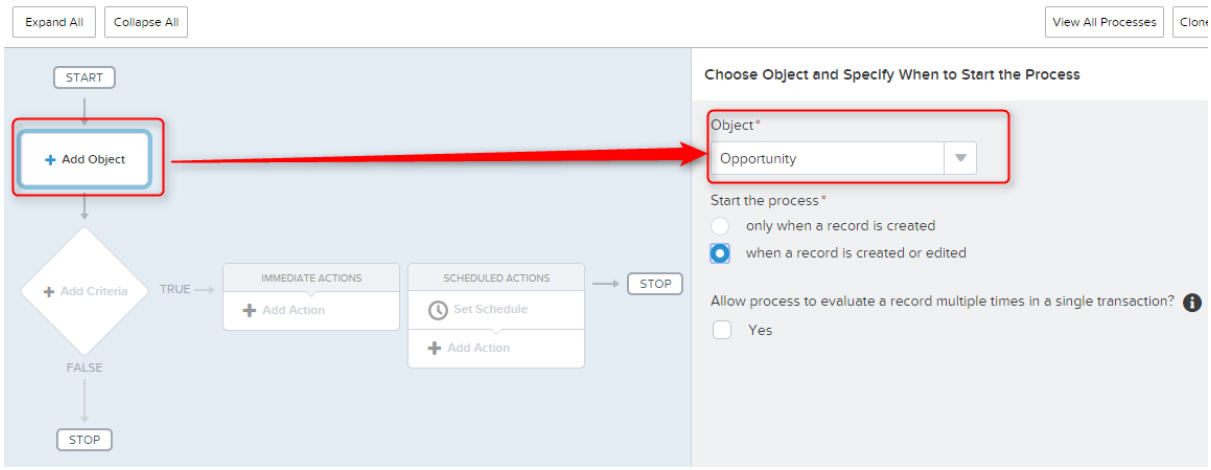
1. Click Setup -> Create -> Workflow & Approvals -> Process Builder
2. Click [New] Button.



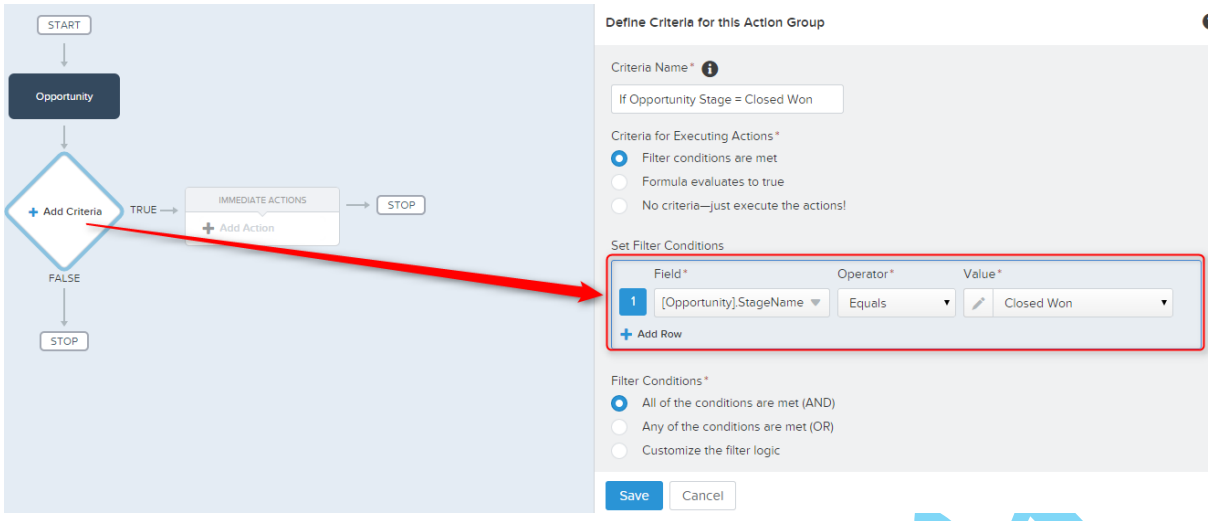
3. Fill in the Name and Description fields and Submit [Save] button.



4. Click [+ Add Objects] in the flow chart and Select Opportunity Object.



5. Click the [Save] button.
6. Click [+ Add Criteria] in the flow chart and enter the evaluation criteria.

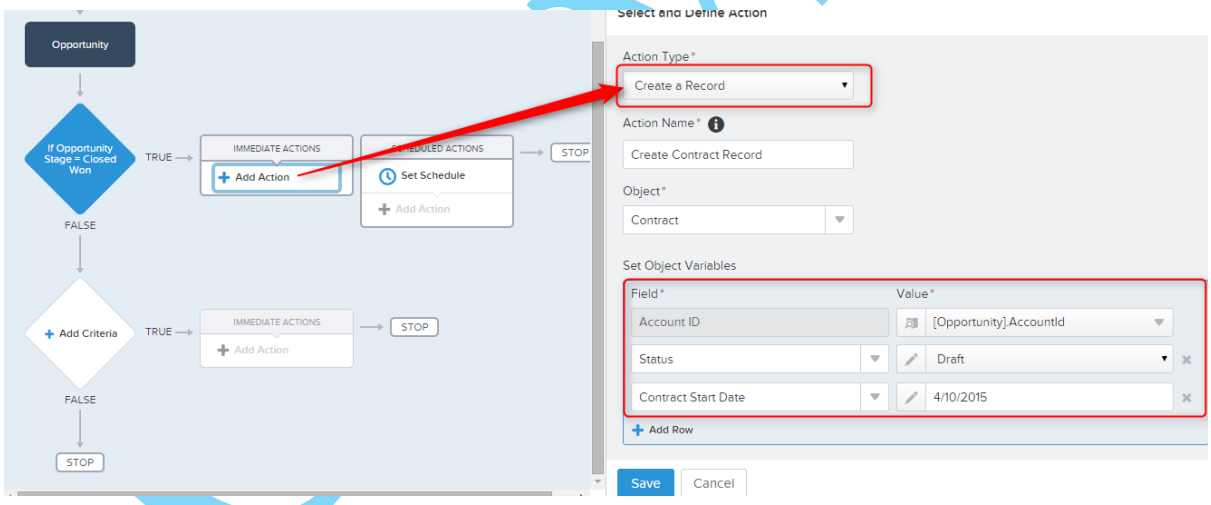


7. Click the [Save] button.

8. Now, you can define actions that are executed when criteria are met. You can add up to 10 immediate actions and 10 scheduled actions.

9. Click [+ Add Action] in Immediate Actions.

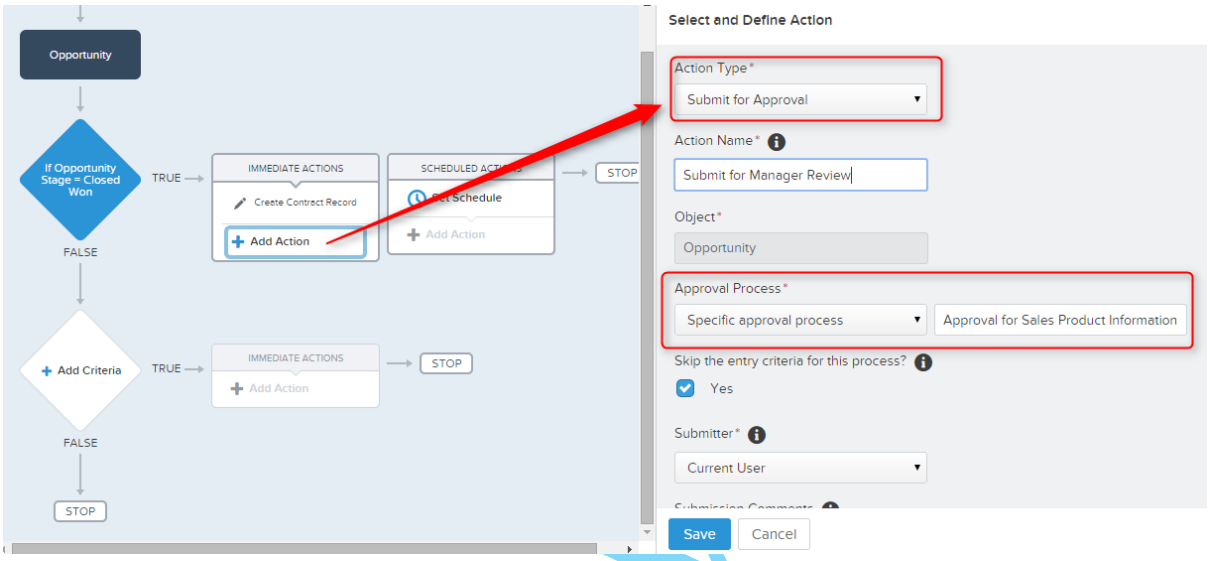
10. Select Action Type as “Create a Record” and Select Object as “Contract” from the drop down and set the values for Object variables. Then, click [Save].



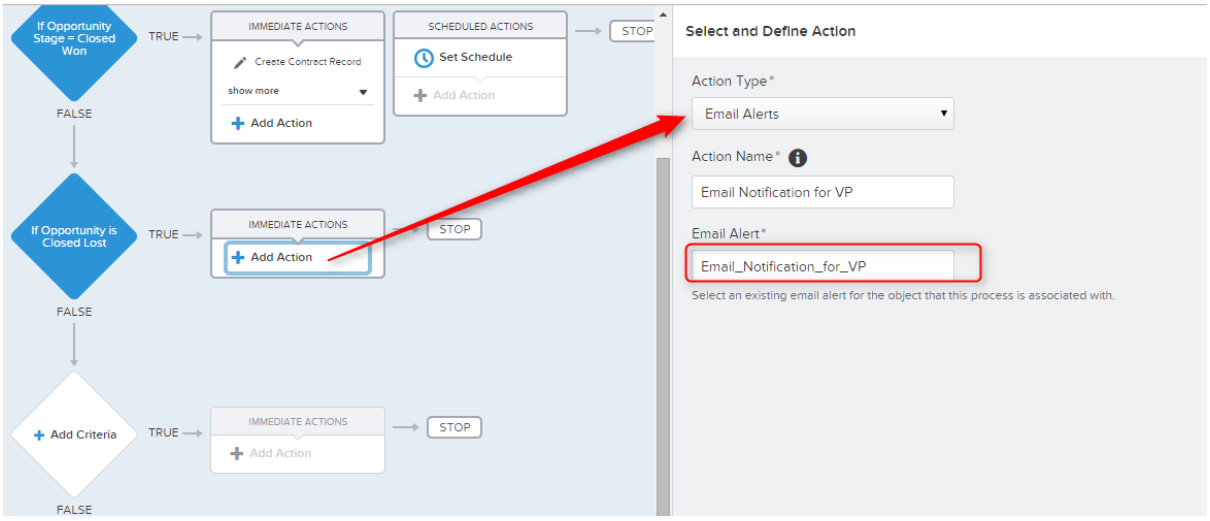
Click [+ Add Action] and

1. Select Action Type as “Submit for Approval from dropdown.
2. Enter the value for Action Name
3. Object is “Opportunity” by default because the process is created for the Opportunity object.
4. Select Approval Process as “Default Approval Process or Specific Approval Process”. If you select the “Specific Approval Process”, you should enter your exiting opportunity approval process name in the next text box.

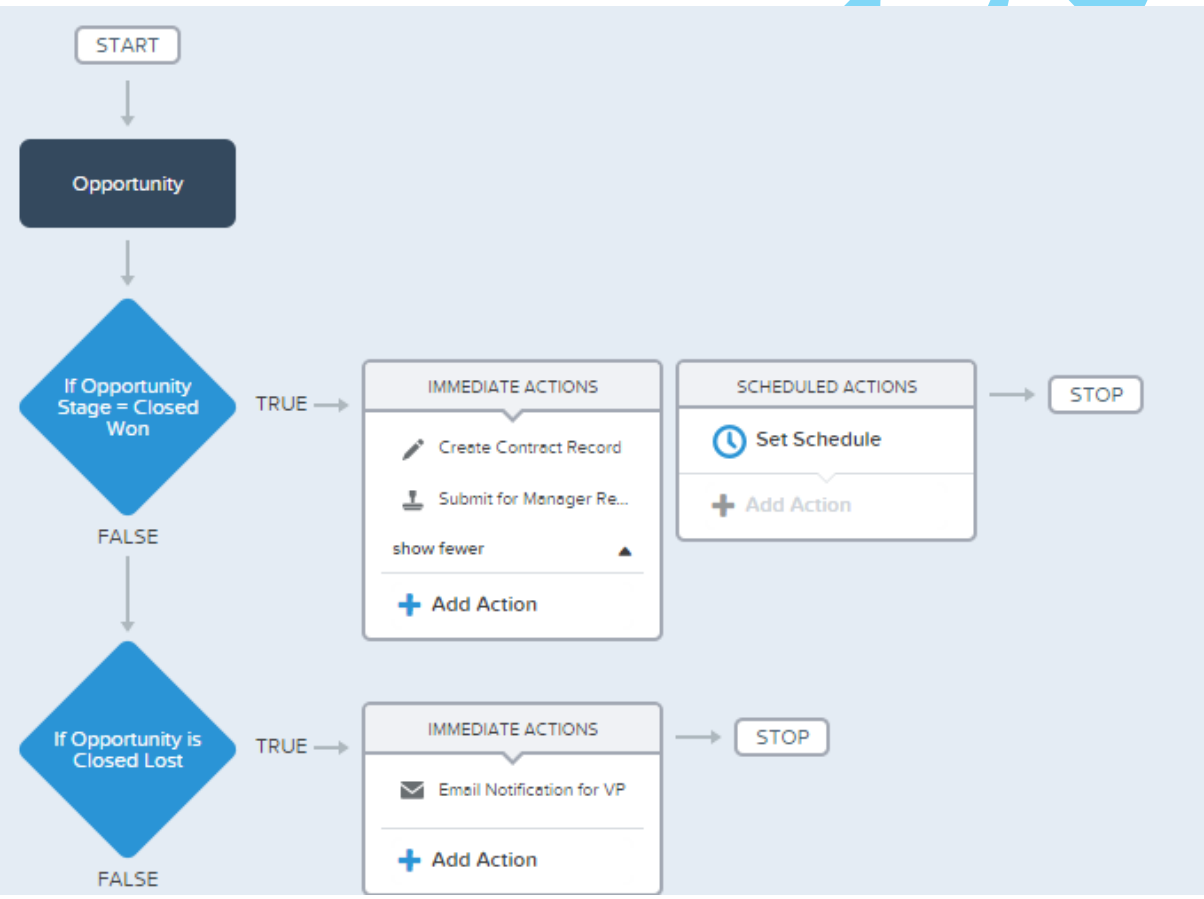
5. If you want skip the entry criteria for approval process , you can check the checkbox or you can leave it as unchecked.
6. Select Submitter from the dropdown. Submitter may be “Current User (or) User field from the record (or) Other User”. If you select “Other User”, you have to enter username in the next textbox.
7. Submission Comments is optional.
8. Click the [Save] button.



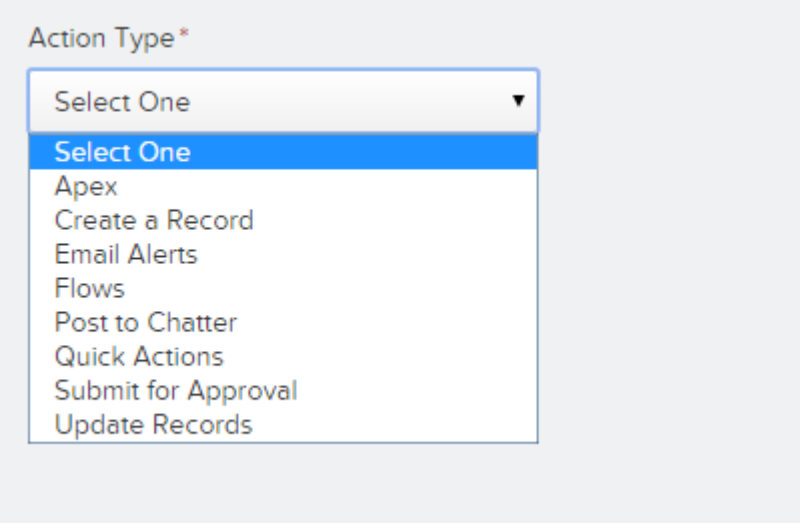
So far, the process is completed for Opportunity Closed Won. Suppose, if the Opportunity’s Stage is Closed Lost, you need to Add criteria in the “Else” part and create Email Notification Action.



Overall Flow is:



Likewise, based on your business requirements, you can create your flow and action using the Process builder. You can select one of these actions for the "Action Type" dropdown.

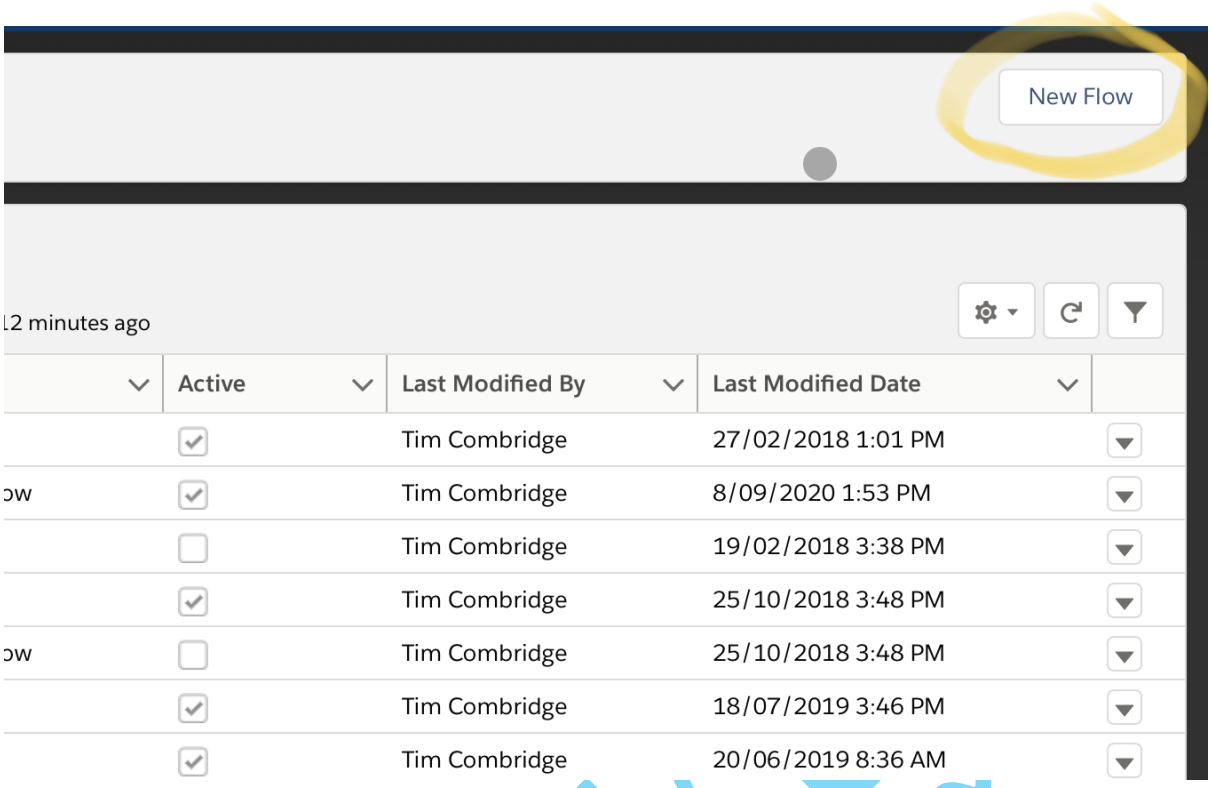


Visual Flows

Flows allow you to build complex business automation using clicks instead of code. As an admin, Flows are going to be your best friend because you will be able to handle the majority of complex business requirements without the help of a Salesforce developer!

The benefit of Salesforce Flow is that they are easy to maintain because anyone (assuming they know Flows) should be able to follow along with what you built.

Flows are accessible through the Setup menu. Simply enter ‘Flows’ into the Quick Find box, and create a new Flow to get started.



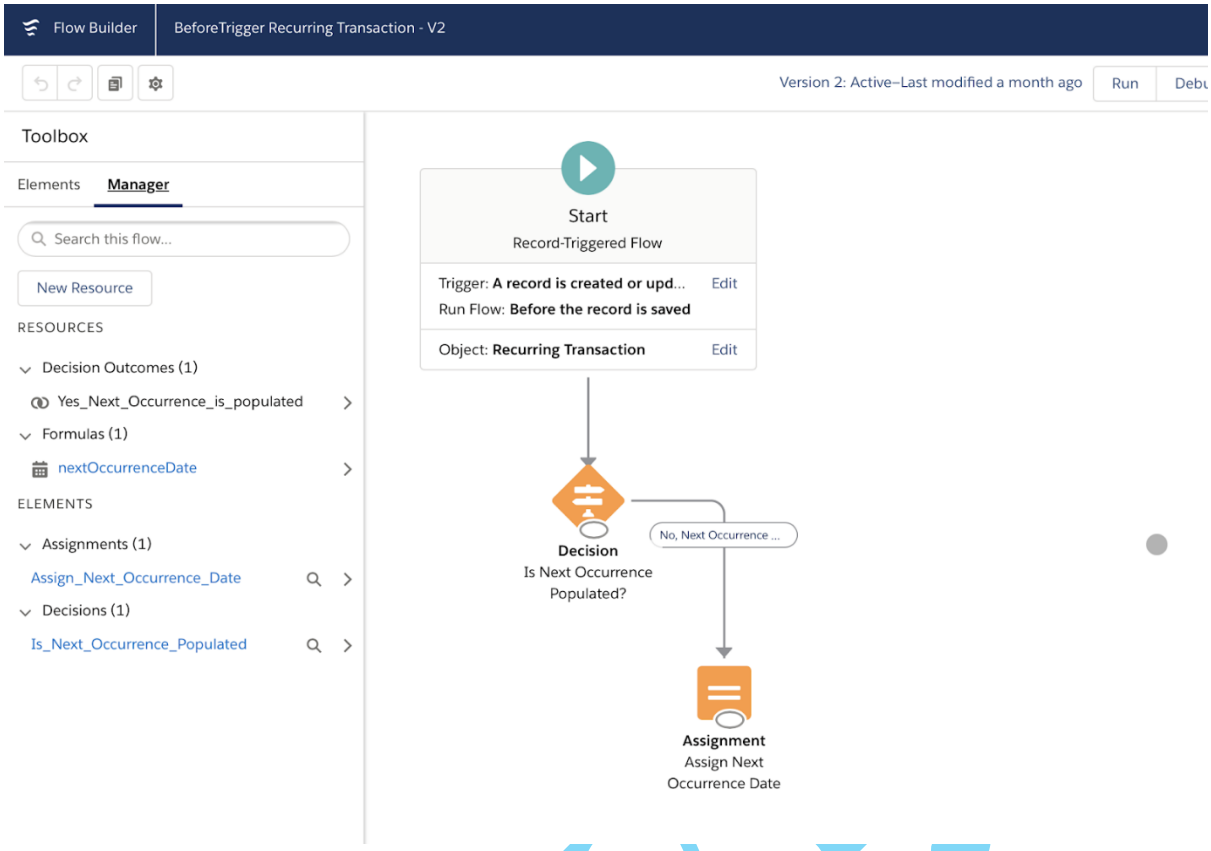
There are 3 main "building blocks" of any Flow:

1. **Elements** are the individual building blocks of the Flow. These perform logical actions such as assignments, decisions, or loops. There are also data elements that will query the database or commit record changes.

2. **Connectors** determine which element leads to which. Winter '21 enables Auto-Layout, and connects the Elements together automatically.

3. **Resources** are the individual variables of data that are to be used in a Flow – these can be strings of text, numbers, records, formulae, or collections.

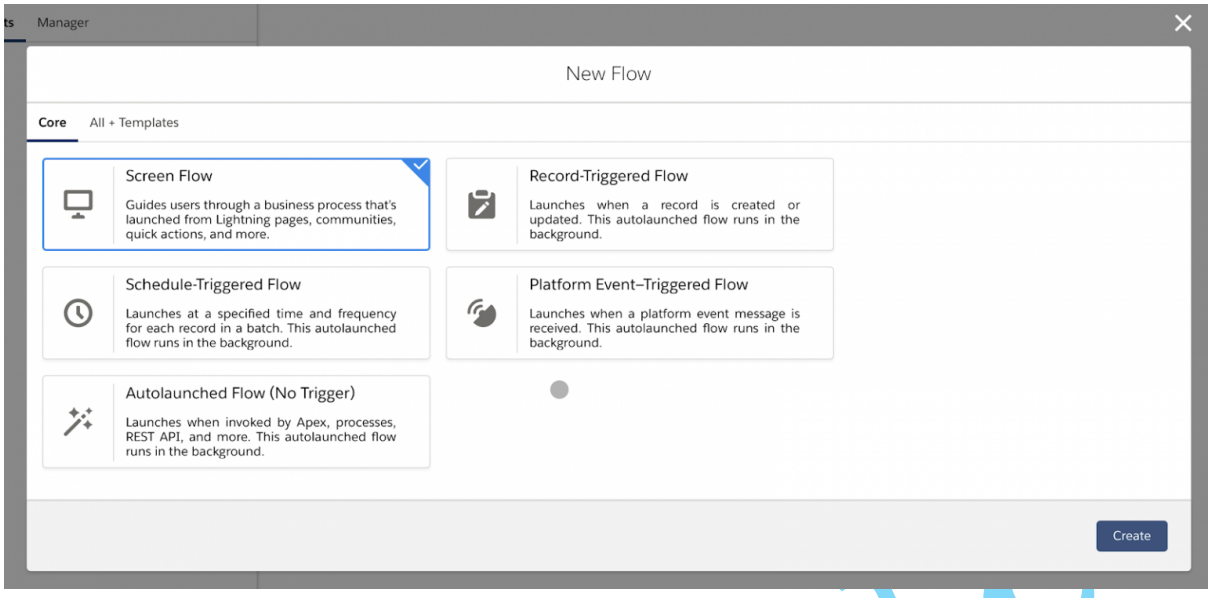
You can see these in action on the Flow Canvas below. This example Flow below is a declarative replacement for a 'before' trigger. In the 'Start' element at the top you can see that the trigger is when a record is created or edited, and it should run before the record is saved.



To 'call' a Flow means that something happens in order to kickstart the Flow process. This could be a Salesforce record change, from another process in Apex/Process Builder, or automated on a recurring schedule.

When you create a new Flow, you're prompted to select the type of Flow you wish to create.

- *A Screen Flow is called through a button or action, or displayed in a Lightning Page or the Utility Bar, and appears as a screen to the user to interact with. This cannot be automatically called.*
- *A Schedule-Triggered Flow runs automatically to a recurring schedule. This is handy for tasks that need to be performed daily on a set of records, or to handle jobs that run frequently*



- *Auto launched Flows are called through Apex, Process Builder, or another Flow. They can be used to perform actions automatically behind the scenes.*
- *Record-Triggered Flows begin when a record is created or updated, very similar to Process Builder (more on this later).*
- *Platform Event-Triggered Flows are called when a platform event is received, similar to an Auto launched Flow*

Reports and Dashboards in Salesforce

Reports and Dashboards provide the answers to the questions which are important for an organization/business.

For example,

- How many leads got converted into accounts?
- Which sales team has generated maximum leads?
- How many employees got a bonus of more than 20% this year?
- Etc.

The answers to these questions are provided in Salesforce using Reports & Dashboards which automatically gives answers to these questions by using the data (records) present in the different objects in the organization.

Reports in Salesforce

Reports in Salesforce is a list of records that meet a particular criteria which gives an answer to a particular question. These records are displayed as a table that can be filtered or grouped based on any field.

Report types

Report type determines which set of records will be available in a report. Every report is based on a particular report type. The report type is selected first when we create a report. Every report type has a primary object and one or more related objects. All these objects must be linked together either directly or indirectly.

Note:

- *A report type cannot include more than 4 objects.*
- *Once a report is created its report type cannot be changed.*

There are 2 kinds of report types in salesforce:

1. Standard Report Types:

Standard Report Types are automatically included with standard objects and also with custom objects where "Allow Reports" is checked.

Standard report types cannot be customized and automatically include standard and custom fields for each object within the report type. Standard report types get created when an object is created, also when a relationship is created.

Note:

Standard report types always have inner joins.

2. Custom Report Types:

Custom report types are reporting templates created to streamline the reporting process. Custom Reports are created by an administrator or User with "Manage Custom Report Types" permission. Custom report types are created when standard report types cannot specify which records will be available on reports.

In custom report types we can specify objects which will be available in a particular report. The primary object must have a relationship with other objects present in a report type either directly or indirectly.

Object relationships that are supported by Custom Report Types can:

1. Include all records that have children.

2. Include all records that may or may not have children.

Note:

- *Once a report type is saved then its primary object can't be changed.*
- *If the primary object on a report type is a custom object, and the custom object is deleted, then the report type and any reports created from it are automatically deleted.*
- *If you remove an object from a report type, all references to that object and its associated objects are automatically removed from reports and dashboards based on that type.*

After selecting the report type we switch to the report builder to create a report.

The report builder is a visual editor for reports. It lets administrators or users create reports in a very easy way. It contains 3 main components:

1. Fields Pane:

Fields Pane displays the fields that are available from a particular report type.

2. Filters Pane:

Filters Pane allows us to set additional filters that limit the records available in reports. Criteria here include view, time frame, and custom filters.

3. Preview Pane:

Preview Pane shows how the report will look like. This pane allows us to add, reorder, and remove columns, summary fields, formulas, groupings, and blocks. Also, it allows us to change the report format and display a chart based on the report.

Note:

Preview pane only shows a limited number of records. Run the report to see all the results.

There are 4 types of report formats in Salesforce:

1. Tabular Reports:

This is the most basic report format. It just displays the row of records in a table with a grand total. While easy to set up they can't be used to create groups of data or charts and also cannot be used in Dashboards. They are mainly used to generate a simple list or a list with a grand total.

2. Summary Reports:

It is the most commonly used type of report. It allows grouping of rows of data, view subtotal, and create charts.

3. Matrix Report:

It is the most complex report format. Matrix report summarizes information in a grid format. It allows records to be grouped by both columns and rows. It can also be used to generate dashboards. Charts can be added to this type of report.

4. Joined Reports:

These types of reports let us create different views of data from multiple report types. The data is joined reports are organized in blocks. Each block acts as a sub report with its own fields, columns, sorting, and filtering. They are used to group and show data from multiple report types in different views.

When we save a report only the parameters of the report get saved. The data is always evaluated in real-time (when a report is run). The report can be run manually as well as it can be scheduled to run automatically based on the running user.

The data which is displayed in reports is based on the running user's sharing and security settings. But when we schedule a report we have to select a running user based on whose security and sharing settings the report is generated.

Every report is saved in a particular folder. Users who have access to the report folders can run the report.
Each user, group, or role can have its own level of access to a folder.

There are 3 types of access levels of folders:

1. Viewer:

With this access level, users can see the data in a report but cannot make any changes except cloning it into a new report.

2. Editor:

With this access level, users can view and modify the reports it contains and can also move them to/from any other folders they have access level as Editor or Manager.

3. Manager:

With this access level, users can do everything Viewers & Editors can do, plus they can also control other user's access levels to this folder. Also, users with Manager Access levels can delete the report.

Note:

By default, a person who creates a folder is the Manager of that folder.

If a folder does not have Manager Access then it is public and users with "View Reports in Public folders" can view it. Depending on their object access these users can run the report. Reports present in public folders can be emailed to Salesforce users.

If a user is not ready to share his/her report then he/she should save it in "My Personal Custom Reports" folders. Reports can be exported in .xlsx format or.csv format, The report displays up to 2000 rows of data. Larger reports can be emailed to Excel.

Note:

A user cannot create a report on an object he/she does not have access to.

Salesforce Dashboards

Dashboards in Salesforce are a graphical representation of Reports. It shows data from source reports as visual components. These components provide a snapshot of key metrics and performance indicators of the organization at a glimpse.

There are 5 types of components in Salesforce Dashboard :

1. Charts:

Used for showing comparisons. These are further divided into 6 types:

- Line Chart
- Vertical Bar Chart
- Horizontal Bar Chart
- Donut
- Pie
- Funnel

2. Table:

Tables are used for showing lists. For example top five or bottom five opportunities.

3. Gauge

Gauge is used to show progress towards a goal.

4. Metric

Metrics are used to show a single number like a grand total from a report with a label specified.

5. Visualforce

Visualforce is used to show a Visualforce component as a dashboard. It is used to pull data from other data sources.

Dashboards display data as per the last time report was run. Each dashboard can have up to 20 components. Access to dashboards is determined by 2 things:

1. Folders:

The visibility of dashboards is also based on the folders in which they are stored. Only the users who have access to the folder can run the dashboard.

2. Running User:

The data which will be available on dashboards are based on running user's security and sharing settings. Running User can be set to:

a. Run as specified user:

It shows the data on the dashboard according to the specified user's security and sharing settings irrespective of the security and sharing settings of the user running it. All users who have access to the dashboard can see the same data.

b. Run as logged in user:

It shows data on the dashboard based on the running user's security and sharing settings. It makes the dashboard dynamic.

The dashboard data is based upon the report's data but when a user views the drill-down report for a dashboard component, running user's access permissions determine what data is displayed on the drill-down report. Hence it is possible that the data in the drill-down report does not match the associated dashboard data.

Dashboards also support automatic refresh and email. It can be scheduled on a daily, weekly, and monthly basis.

Dynamic Dashboards

It enables each user to see the data they have access to. With dynamic dashboards, we can control data visibility without creating separate dashboards. A single dynamic dashboard can display a standard set of metrics across all levels of an organization.

Note:

- *Dynamic dashboards cannot be saved to personal folders.*
- *Dynamic dashboards cannot be scheduled for refreshes.*

Dashboards	Dynamic Dashboards
1. It displays data/information based on a single user's perspective.	1. Displays data/information tailored according to the security and sharing settings of running user.
2. Used when we need to show organization wide data to a set of users.	2. Used when we need to show data based on running user's security and sharing settings.

Sales Cloud

Salesforce Campaign Management

Campaign object is used to store information about a campaign organized usually for lead generation or brand building.

Lead generation campaign includes direct mail, email blasts, web seminars, conferences, and trade shows whereas brand-building campaigns include different types of advertisements.

Important Fields:

Active:

It indicates whether the campaign is Active or not. If the campaign is not active then it cannot be related to other object's records.

Expected Revenue:

Amount of money expected to generate from the campaign.

Expected Response:

Percentage of responses expected to receive for the campaign.

Num Sent:

Number of individuals targeted by the campaign. For example, the number of emails sent.

Parent Campaign:

Allows multiple individual campaigns to be associated with a larger initiative.

Related Lists:

Campaign Members

It is an object in itself which allows Leads/Contacts to be associated with multiple campaigns. These Leads and Contacts are tracked using a status field having values sent (targeted to the campaign) or responded to (responded to the campaign). Campaign members help to track the response rates of a campaign.

Campaign Influence

Opportunities are usually influenced by more than one campaign. Campaign influence allows one to associate one opportunity to multiple campaigns.

However, an opportunity can have only one parent campaign, and only the parent campaign influences campaign roll-up summary fields on the related campaign.

In short, campaign influence will allow you to display and report on an opportunity's relationship with multiple campaigns, but the opportunity is only attributed to a single campaign for roll-up summary and ROI calculations.

To use campaign influence in your organization simply add "Campaign Influence" related list on Opportunity page layout. Also to setup Automatic Campaign Influence navigate to:

Setup –> Customize –> Campaigns Influence –> Campaign Influence Settings.

Note:

While enabling Campaign Influence first add the related list to page layout and then enable Campaign Influence.

Limitations with Campaigns

- To create, edit, or delete a Campaign the "Marketing User" needs to be checked on the User record.
- The maximum number of leads/contacts that can be added from a view is 250 and from an import file is 50000.

Salesforce Accounts

An Account is a business entity, it's a company. We need insight into our business and our data and that starts with the people we're doing business with.

We can store information about our customers using accounts and contacts. Accounts are companies that we're doing business with.

An Account is an organization that is a qualified potential customer, an existing customer, partner, competitor or has a relationship of similar significance. Accounts should be created for organizations (companies, non-profits, foundations), Individuals.

The account is the glue that holds the relationships (contacts) and interactions (activities, opportunities, cases, etc.) with an organization. The standard Salesforce account is configured for B2B (business to business) relationships.

One account = one organization

A business account is a combination of contacts and accounts.

We also have Person accounts in Salesforce which is used by B2C organizations. Person accounts are oriented towards doing business with persons while others are with the companies. To use Person accounts we have to ask Salesforce to enable it for our org and we also cannot revert it back.

One person account = one person

- Person and business account have a few important differences:
- Person accounts are forever. After they're turned on, we can't turn them off.
- If our organization uses both business accounts and person accounts, we'll have to select which type of account we're creating whenever we add an account.
- Person accounts can't have contacts.
- Person accounts don't have an account hierarchy.
- We can't get account insights for person accounts (but we can associate person accounts with social network profiles by using the Social Accounts, Contacts, and Leads feature).

Important Fields:

1. Parent Account:

This is a self-relationship field. It relates 2 accounts with each other. The parent account is selected from the lookup dialog and the one on which the field exists is the child account. A parent account can't be a child of itself.

2. Account Name:

The name of the company or the person (in case of person accounts) with whom we are doing business.

Related Lists:

Account Team

An account team is a team of users who work together on an account. Account Teams help multiple users to better collaborate on accounts by defining a role for each team member, setting record-level access individually, and viewing teams in list views and reports. Account team can be built for each account.

To display Account Team related lists on accounts we need to enable it from the setup menu. When selecting an account team member, specify the role that the person plays on the account.

Depending on our sharing model, we can specify each account team member's level of access to the account and any contacts, opportunities, or cases associated with that account. That way, we can give some team members read-only access and others read/write access.

We can also set up a default account team that includes the people who we normally work with on our accounts. We can edit the Team role picklist according to the assigned roles.

We can only add users as team members on the account team.

Partners

It shows all the partners (other accounts) involved with that account. Account can't be a partner with its own account.

One account can be a partner with different accounts with their respective roles. These roles can be edited by the system administrator in the setup menu.

Notes:

1. An account can't be deleted if it is associated with some case or contract.
2. If we delete the account, then contacts & opportunities related to it will also be deleted.
3. If we delete the parent account, then the child account will not be deleted.

Salesforce Contacts

Contacts in Salesforce store an individual's demographic information, such as phone numbers and email addresses, and are linked to accounts. If a contact is not linked to an account then it is a "private" contact and only viewable by the contact owner or Salesforce administrator.

One contact = one person

Important Fields:

Lead Source

1. The record Source: for Example, Advertisement, Partner, or Web.

2. The entry is selected from a picklist of available values, which the administrator sets.

Reports To

1. The name of the contact's manager to which this particular contact will report.

Related Lists:

Campaign History

1. Campaign and Contact are related through junction objects i.e. campaign members.

2. It describes that in which campaign this particular contact has been added as a member and also its status for that campaign.

Contact Role

Contact Roles define the role that a contact or person account plays in an account, case, contract, or opportunity. When contact roles are defined, our sales team has more information for determining who to contact and when. Up to 50 contacts that are associated with the record's account are displayed. Optionally, designate a primary contact for the record.

Contact Role fields are as follows:

1. Contact:

Name of the contact or person account. Select an existing account or create a new one.

2. Primary:

When selected, identifies the person as the primary contact for the record. The Primary option isn't available for cases. Instead, the contact listed in the Contact Name field on the case record is automatically the primary contact. There can only be one primary role in this list for one account at a time.

3. Role

The role of the contact for the record. The administrators can customize the selections for this picklist on each object.

Lead in Salesforce

Lead object is used to store information about a person interested in the product or service we are delivering. In business terms, leads are the people who are your potential customers. It contains some information about a company and the information of a person working in that company. Hence Lead in Salesforce can be considered as a business card.

- Leads are prospects who've expressed interest in our product but aren't qualified to buy it, hence they can also be called as an unqualified prospect.
- Lead conversion means qualifying a lead for the sales process.
- Lead management is a process that allows us to measure and monitor lead conversion. It helps inside opportunities or telesales teams qualify leads before passing them on to sales.
- Sales reps get instant access to the latest prospects and ensure that leads are never dropped.

Important Fields:

Company:

Name of the company with which the lead is affiliated.

Title:

Position of lead within his/her own company.

Lead Status:

This field specifies the status on which the lead is. For example, Open, Contacted, Working, Closed-Converted, or Closes-Not Converted.

Lead Score:

This field specifies the source from which the lead is captured. It can be anything like Web, Purchased List, and Partner Referrals etc.

Related Lists:

Campaign History

Campaign & Leads are related through a junction object i.e. Campaign Member.

Campaign History related list describes that in which campaign this particular lead has been added as a member and also its status for that campaign. When we initially create a lead record and add the campaign through a lookup dialog then also it is added to campaign history, not as a field.

Lead Process

Lead processes allows to create different process cycle a Sales representative follow for leads. We can set up different lead processes to control the steps our users follow for leads because there are different stages through which a lead gets in the process of qualifying and converting which varies from company to company & department to department.

In a Lead process, we define the picklist values of the Status field which will be available to the user using that lead process. By implementing different lead processes, we can have different lead life cycle for each kind of lead. Lead processes are included in record types which can be assigned to the user's profile to have a different lead life cycle.

Setting up each process takes several steps:

- Step1: Create the lead status needed in the lead process.
- Step2: Name the lead process, and select what lead status are included in that particular process.
- Step3: Create a record type for the sales process. Record types link the sales process to the page layout that goes with it.

Lead Assignment Rules in Salesforce

Salesforce Lead Assignment Rules are used to automatically assign lead records to a particular user or queue based on different conditions. It can contain many rule entries that determine the assignee of a lead.

Rule entry specifies the:

- Sort Order which determines the order of evaluation of rule entries.
- Entry criteria which determine that the response will be sent through which rule entry.
- Name of the user/queue to which the record should be assigned.
- Email template which will be used to send the response.

We can create as many assignment rules as we want but only 1 can be active at a time. “Don’t Reassign Owner” determines if the user whose process stack is in use becomes the owner of the rule entity, or if it remains owned by its creator user.

Web to Lead

The process of capturing leads from a website is called Web-to-Lead. It allows organizations to generate a web form that they can add in their company’s website using which people can register themselves as a lead and show their interest in your product, the entries in that webform when submitted, creates a lead record within Salesforce.com.

To allow the leads to be submitted through the web we need to enable web-to-lead then we need to create a web form. While generating the web form we can specify the fields we would like to capture from leads.

In web-to-lead form fields like Name and email should always be set explicitly. It asks for the return URL also where the page will be redirected when the lead will be submitted through the web form. Web-to-Lead is limited to receiving 5000 leads per day.

Salesforce runs field validation rules before creating records submitted via Web-to-Lead and only creates records that have valid values.

Auto Response Rule

Lead Auto response rules are used to automatically send an email response to the leads captured through the web. We can create as many auto response rules as we want but only 1 can be active at a time. A rule can contain many rule entries.

Rule entry specifies the:

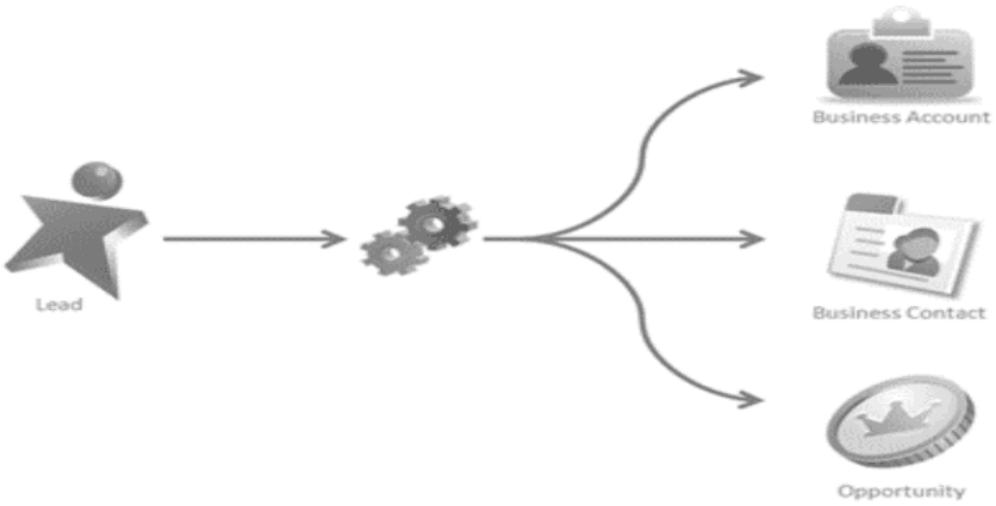
- Sort Order determines the order of evaluation of rule entries.

- Entry criteria which determine that the response will be sent through which rule entry.
- "From" address & Name mentioned in the response.
- Email template which will be used to send the response.

Salesforce processes each rule until it finds a match between the lead attributes and the rule entry criteria. When it finds the first match it sends the response according to it and other rule entry criteria after it are not evaluated. If none of the rule entry matches the criteria then the response is sent according to "Default Response Template" specified in Web-to-Lead settings to the leads registered from Web.

Salesforce Lead conversion

Lead conversion in Salesforce is a process in which a lead record is converted into Accounts, Contacts & Opportunities. This happens when a lead is identified as a qualified Sales prospect.



When a lead is converted:

- A contact, account & opportunity are created and populated with the lead's data (unless otherwise specified during conversion).
- The lead field "Converted" is changed from false to true.

After the conversion, the lead record cannot be viewed or edited as a lead but it can be viewed in reports as a lead that means the record is still present. The data

within standard lead fields are automatically transferred to contact/account and/or opportunity. The data within custom lead fields to transfer to the contact/account/opportunity during lead conversion, the administrator must map the custom lead fields.

Note:

A custom lead field can only be mapped to a single field on either contact, account, or opportunity.

If the account/contact/opportunity already exists then

- There is no way to convert a lead to an existing opportunity.
- When we convert a lead Salesforce attempts to find an account with the same name as the field "Company" on the lead record. If an account name contains the company's name, then you will have the option to use the existing record.
- If we attach the lead to an existing account and the lead name matches the name of an existing contact, then you will have the option to use the existing contact record also.

Notes:

- *Converted leads cannot be modified at all.*
- *It is not always necessary to create an opportunity when converting a lead.*
- *The default record type for the user performing the lead conversion is selected automatically for records created in the conversion process.*
- *Once a lead is converted it cannot be reverted to an unconverted state.*
- *Salesforce does not overwrite existing Account/Contact data on lead conversion.*

Salesforce Opportunities

Opportunities in Salesforce represent a transaction between your company and an Account. Typically this is a potential sales transaction that would include information about the specific products and/or services one of your sales representatives is presenting to a prospective customer.

An Opportunity typically represents either:

- Completed sale
- A lost potential sale
- A potential for a future sale

In other words, we can say that salesforce opportunities are deals. In Salesforce you can create opportunities for existing accounts or by converting a qualified lead.

Important Fields:

Opportunity Name:

This required text field represents the name of the specific deal as you want it to appear on your list of opportunities or on a pipeline report.

Amount:

This field displays the total cost of opportunity.

Close date:

This field describes when you'll close a deal. It is a required field. This field is commonly used to track the date on which the deal is closed.

Expected Revenue:

This is a read-only field that is automatically generated by multiplying the Amount field by the Probability field.

Opportunity Owner:

This person in your organization owns the opportunity. Though an opportunity record has only one owner, many users can still collaborate on an opportunity with the help of opportunity team related list.

Private:

If you want to keep an opportunity private, select this checkbox to render the record accessible to only you and the system administrator.

Stage:

This is a required field that allows you to track your opportunities, following your company's established sales process. Salesforce provides a set of standard drop-down list values common to solution selling, but you can modify these values.

Probability:

It is a percentage field. The probability is the confidence factor associated with the likelihood that you'll win the opportunity. Each sales stage that your company defines is associated with a default probability to close. When you win a deal, the probability is 100% because you won it, if you lost it then it is 0%. System administrator can modify these percentages and also add new stages.

Primary Campaign Source:

It defines the name of the campaign responsible for generating the opportunity. For opportunities created during lead conversion, this field is automatically filled in with the campaign name from the lead. If the lead has multiple associated campaigns, the campaign with the most recently updated member status is inserted into the opportunity.

Related Lists:

Opportunity Teams:

- An opportunity team is a group of Salesforce users who work together on a sales opportunity.
- We can build different teams for each opportunity that we own. While selecting an opportunity team, we can give different access to different users according to our sharing model.
- Opportunity teams show who's working on the opportunity and what each team member's role is. Using opportunity teams makes it easier for you to track the work of a team, too.
- We can define a default team also which can be added to multiple opportunities.

Competitors:

- It is a related list that describes the competitors associated with that particular opportunity.
- This is the list to show the strength and weaknesses of the competitors for that opportunity.
- We can track competitors in a pending sale by listing the competitor's names in the opportunity.

Orders:

It is an object that stores the customer's request for products.

Subscriptions:

It is an object that is available on the opportunity so that we can provide a subscription for the products those require renewal after a certain period of time.

Partner:

It shows all the partners (other accounts) involved in that opportunity. These Partners are stored as an account in your org. We can have different partners for different opportunities. These partners are stored with their respective roles that can be edited by the system administrator.

Stage History:

- Stage history lets you track the changes made to an opportunity, so you can see who changed the opportunity and when.
- This is not an editable related list.
- Every time a user changes the Amount, Probability, Stage, or Close Date fields, a new entry is added to the Stage History related list.
- All entries include details of the change and who made it.

Sales Processes:

Sales processes allows to create different sales process cycle a Sales representative follow for opportunities. We can set up different sales processes to control the steps our users follow for opportunities because there are different stages through which an opportunity gets in the process of sales.

In a Sales process, we define the picklist values of the Status field which will be available to the user using that Sales process. By implementing different sales processes, we can have a different lead life cycle for each kind of sale. Sales processes are included in record types which can be assigned to the user's profile to have a different sales life cycle.

Setting up each process takes several steps:

- Create the opportunity stages needed in the sales process.
- Name the sales process, and select what opportunity stages are included in that particular process.
- Create a record type for the sales process. Record types link the sales process to the page layout that goes with it.

Big-Deal Alert:

An organization can use alerts that automatically send an email notification for opportunities with large amounts. Customize this alert to send an email when an opportunity reaches a threshold. Our threshold consists of an opportunity amount and probability.

For example, you may want to send an email to your team that an opportunity of $500,000 has reached a probability of 90%.

Products in Salesforce

Products in Salesforce is a service or item the company sells to the customers. After defining the products, you can associate them with the prices you have established in the price book.

- Products are available through a related list on the Opportunities tab.
- With Salesforce you will have layouts and custom fields to track which products are successful. Products can then be added to a Salesforce opportunity allowing you to ensure that inventory and forecasting are accurate.

Important Fields:

Product Name:

It is a required field used to describe the label of the product. The name should be given in a way so that it is easily recognized.

Product Code:

Its field stores the product code which uniquely identifies the code. The code can only be numeric or it can be a mix of digits & alphabets.

Active:

It is a checkbox field, if checked then only the users will be able to associate this product with an opportunity.

Product Family:

It is a picklist field used to categorize the products. To add/remove categories, edit the Product family picklist field.

Related Lists:

Standard Price

Every product has a baseline price, that baseline price is stored as a standard price.

Product History

Product History tracks the field history of a particular product.

Price books

A price book is a list of products and their associated prices. Each product and its price are called a price book entry. The price book will provide a list of the products and the different prices for the products. Salesforce will provide standard and custom price books, making it easy to have multiple prices for a product.

In other words we can say that sometimes your prices for products will differ per customer – it can depend on who is buying, when they are buying, or in what quantity. To accommodate this flexibility, a Product can exist in Salesforce with many different associated prices, using a feature called Price Books and Price Book Entries.

A Product can exist in multiple Price Books, for every different price on offer to a customer. The combination of the Product and the price together is what we call a Price Book Entry. Price book is divided into two parts:

1. Standard Price Book:

The Standard Price Book is the full list (i.e. the master list) of all the products and services your company sells. It is already present and new products can be added to it using the related list present at the product's detail page.

2. Custom Price Book:

These are additional price books created as per the requirements of the company. The price for each Product in the Custom Price Book may be different from the price in the Standard Price Books. The price for the same Product in each Custom Price books does not need to be the same – it can vary from one Custom Price Book to another.

Important Fields:

Active:

If checked, then the price book is available to users for adding a product to an opportunity otherwise not.

Related Lists:

Products

It shows the list of products added to this price book.

Price book History

It describes the field history of a price book.

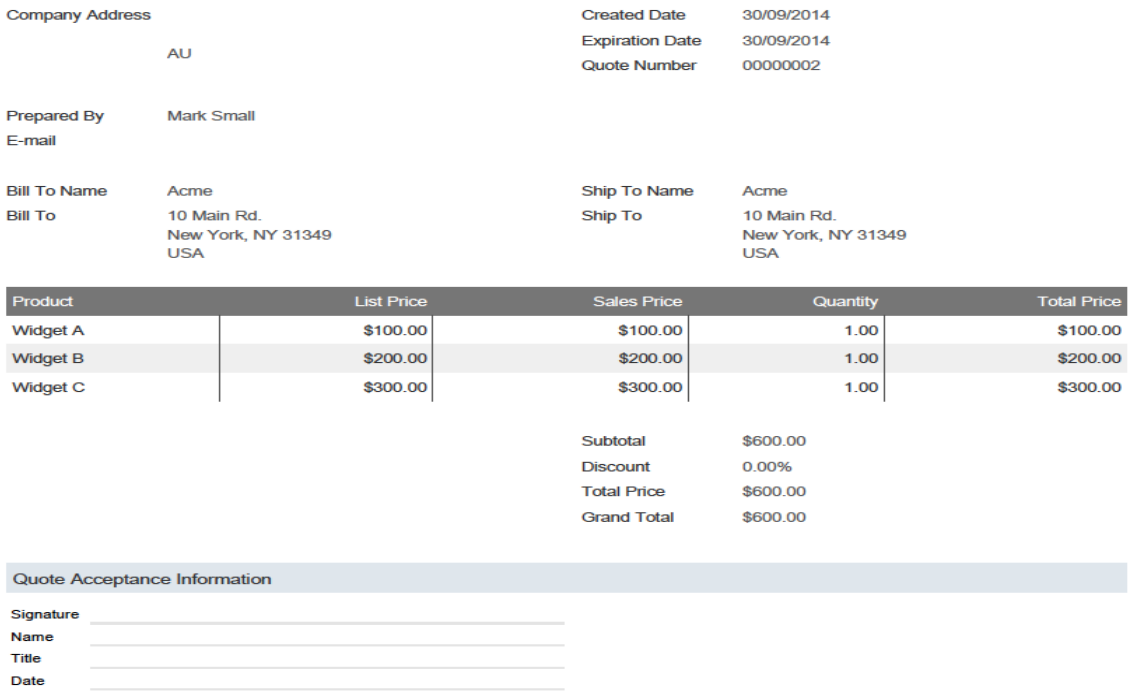
Notes:

Products are related to opportunities using an object called an Opportunity Product.

An opportunity can have only one price book associated with it and only the products present in that price book are available to add in the opportunity. If some products are already added to an opportunity using a particular price book and if you want to change the price book then all the products already added to the opportunity will be removed automatically. The cost of the products added to an opportunity is aggregated and displayed as Amount in opportunity record.

Salesforce Quotes

A quote is a specific combination of Products, Quantities and Pricing. It's the specific group of products and their associated prices that you've quoted to a customer.



Company Address

AU

Created Date 30/09/2014
Expiration Date 30/09/2014
Quote Number 00000002

Prepared By Mark Small
E-mail

Bill To Name Acme
Bill To 10 Main Rd.
New York, NY 31349
USA

Ship To Name Acme
Ship To 10 Main Rd.
New York, NY 31349
USA

Product	List Price	Sales Price	Quantity	Total Price
Widget A	$100.00	$100.00	1.00	$100.00
Widget B	$200.00	$200.00	1.00	$200.00
Widget C	$300.00	$300.00	1.00	$300.00

Subtotal $600.00
Discount 0.00%
Total Price $600.00
Grand Total $600.00

Quote Acceptance Information

Signature
Name
Title
Date

Service Application

Activity Management in Salesforce

Activity Management in Salesforce are special, they consist of:

- Tasks (To-Dos)
- Events (meetings)

Task

Task is an object in itself that is used to assign a business activity such as making a phone call or other TO-DO items which have a target end date.

Event

Event is also an object which represents a calendar item such as a meeting or conference which has Start & End Date/Time and also it usually involves other persons who here are called as invitees.

Both of these objects come under Activities in the user interface but they are completely different objects in terms of API. Even in the setup menu under Activities, they both have their own field list, validation rules, buttons & links, page layouts, etc. Yet we can add a field to the activity field list which will get added to both Task & Event's page layout.

Types of Activities:

- Tasks
- Call Logging
- Events
- Email

There is no tab for Activities (Tasks or Events) in Salesforce. Activities can be created on Contract, Order, Campaign, Account, Opportunity, Product, Asset, Case, Solution, Coaching, Goal, and Metric standard objects. Also, the custom objects whose "Allow Activities" checkbox is checked can have activities on them.

Important fields in Tasks & Events:

Subject:

The subject field in activities is a combo box that can be described as a picklist that looks like a lookup field and behaves like a text field.

Related To:

This field's API name is 'WhatId' and it defines which object's record a particular task or event is related to. These objects can be any of the custom objects which allow activities on them and all the standard objects on which they can be created. This field is polymorphic.

Assigned To:

This field's API name is OwnerID and it defines to which user the task or event has been assigned.

Due Date(On Task):

This field's API name is ActivityDate and it represents the date by which the task should get completed.

Status(On Task):

This field describes the status of the task such as "In Progress", "Not Started" or "Completed".

Name:

This field's API name is WhoID and using this field tasks can be related to a lead or contact. This field is also polymorphic.

Start & End Date/Time (On Event):

This field indicates the start & end date/time of the event.

All Day Event (On Task):

It is a checkbox which if checked, disables the End Date/Time field for the event.

Recurring series of Tasks:

This field helps them to create recurring events or tasks daily/weekly/monthly/yearly.

Invitees are the Users, Leads, or contacts invited for a particular event. When we save the event and send them the update they get an email using which they can accept or decline the request which then will be visible in the Accepted & Declined related list on the event's detail page.

Related Lists:

There are 2 related lists which appear on the page layout of the object which allows activities on it, following are:

1. Open Activities:

This related list displays all open tasks and events for the record and its associated records.

2. Activity History:

This related list displays all the completed tasks, logged phone calls, expired events, outbound emails, mass emails, and merged documents for the record and its associated records.

Tasks assigned to users are visible on My Tasks related list on their homepage and events scheduled for in which a particular user is an invitee is visible on calendar related list on the home page.

Limitations with activities

- Notes & Attachments cannot be added to activities.
- Custom fields such as long text areas or rich text areas cannot be added to activities.
- Tasks cannot be assigned to a queue.

Service Cloud in Salesforce

"Service Cloud" refers to the "service" (as in "customer service") module in salesforce.com. It also encompasses features such as the Public Knowledge Base, Web-to-case, Call Centre, and the Self-Service Portal, as well as customer service automation (e.g. escalation rules, assignment rules). It is designed to allow you to support past, current, and future clients' requests for assistance with a product, service, billing, etc.

Service Cloud in Salesforce allows users to automate service processes, streamline workflows and find key articles, topics, and experts to support the agent. The purpose is to foster one-to-one marketing relationships with every customer, across multiple channels, and on any device. Service cloud can "listen" and respond to customers across a variety of social platforms and automatically route cases to the appropriate agent.

Social customer service is integrated with the Salesforce Customer Success Platform, which allows the social team to gather a comprehensive picture of the customer to inform responses. Service Cloud built on a SaaS model.

Important objects in Service application

- Case
- Solution

Case Management in Salesforce

Cases are used to track support issues. If a customer purchases a product or service and calls in for support, this would get tracked using a case record. A case can be a customer's question or feedback. An open case is unresolved while a closed case is resolved.

Support agents can review access to see how they can deliver better services. While sales representatives can use cases to see how they affect the sales process as responding to your cases keeps your customers happy and enhances your brand.

Important fields:

Status:

It is a required field which describes the status of a Case such as New, Working, Escalated, or Closed.

Case Origin:

It is also a required field and describes the origin of the case that can be Phone, Email, or Web.

Contact Name:

It stores the ID of the associated contact which describes that this case is related to this contact/customer.

Account Name:

It stores the ID of the associated Account to the case which describes that this case is related to this account/company. It is auto-populated when we select Contact but can be changed later also.

Parent Case:

It stores the ID of the parent case in Case Hierarchy.

Assign using Active Assignment rule:

It is only available when submitting cases manually. When checked it uses the active assignment rule to assign the case to a user/queue. Otherwise, the case is assigned to the user submitting the record.

Send notification email to contact:

This checkbox when checked sends a notification to the Contact/Customer stating that their case has been registered within our organization and will be resolved soon.

Web Email/Phone/Company/Name:

All these fields get filled automatically when a case is submitted through our Web-Form or Email. These fields are different from the Contact Email/Phone/Name and Account Name. Also, these fields are only visible on the records submitted through other channels like web or Email.

Case Number:

Assigned automatically when each case is inserted. It can't be set directly, and it can't be modified after the case is created. It is used to uniquely identify a record.

Related Lists:

Case Comments

Case comments let support agents add, edit and delete public and private comments on a case. All comments appear in this related list and can be private or publicly available to a case's contact on the customer portal, self-service portal, or chatter answers. Comments can be made private or public depending on the sharing model.

Case History

This related list tracks the changes to the case. Anytime a user modified any of the standard or custom fields whose history is set to be tracked on the case, a new entry is added to the Case History related list. This list is not editable.

Case Teams

A case team is a team of users that work together on a case. For example, it may include support representatives, support manager, and a product manager.

Every member in the case team has a particular role assigned that determines access to the case, like Read Only or Read/Write, and also whether the user is visible to users in the customer portal or not.

Salesforce Support Processes

Support processes in Salesforce allow to create different process cycle a Support executive follows to resolve a case. We can set up different support processes to control the steps our users follow to resolve a case because there are different steps through which a case gets in the process of resolving which varies from company to company & department to department.

In the Salesforce Support process, we define the picklist values of the Status field which will be available to the user using that support process. By implementing different support processes, we can have different resolving process cycle for each kind of case. Support processes are included in record types which can be assigned to the user's profile to have a different case resolving process.

Setting up each process takes several steps:

Step1: Create the case stages needed in the case resolution process.

Step2: Name the support process, and select what stages are included in that particular process.

Step3: Create a record type for the support process. Record types link the sales process to the page layout that goes with it.

Auto Response Rule

Case Auto-response rules are similar to lead auto-response rules. They automatically send an email response to the case submitters who submit the case from Web or Email or Customer Portal or Self-Service Portal. We can create as many auto-response rules as we want but only 1 can be active at a time. A rule can contain many rule entries.

Rule entry specifies the:

- Sort Order determines the order of evaluation of rule entries.
- Entry criteria which determine that the response will be sent through which rule entry.
- Email template which will be used to send the response.

- "From" address & Name mentioned in the response.
- Reply-to address where can give the reply of the response.
- "Send response to all recipients" checkbox which allows you to send the response to all those mentioned in the To or CC of the original message.

Salesforce processes each rule until it finds a match between the case attributes and the rule entry criteria. When it finds the first match it sends the response according to it and other rule entry criteria after it is not evaluated. If none of the rule entries matches the criteria then the response is sent according to "Default Response Template" specified in Web-to-Case settings to the cases registered from Web.

Salesforce Web-to-Case

The process of capturing cases from a website is called Web-to-Case. Salesforce Web-to-Case allows organizations to generate a web form which they can add in their company's website using which their Support & Service customers can easily create a case within Salesforce.com.

To allow the cases to be submitted through the web we need to enable it then we need to create a web form. While generating the web form we can specify the fields we would like to capture from our customers while submitting a case.

In Salesforce Web-To-Case form fields like Name and email should always be set required explicitly. It asks for the return URL also where the page will be redirected when the case will be submitted through the web form.

The values entered through the web will be stored in the newly created Case in the Web Name and Web Email fields. If that email address happens to be associated with a Contact in your system, then it will automatically associate that case with the contact who has that email address, and with the account associated with that contact.

If that email address is not found, or it discovers more than one contact with that email address, then it will not know which contact to associate to the case. In that instance, it will leave the Contact and Account fields on the case blank and allow you to fill them (which you can generally find using those Web Name and Web Email fields). It is limited to receiving 5000 cases per day.

Salesforce runs field validation rules before creating records submitted via Web-to-Case and only creates records that have valid values.

Assignment Rules

Case assignment rules are similar to lead assignment rules. It automatically assigns case records to a particular user or queue based on different conditions. A case assignment rule can contain many rule entries that determine the assignee of a case.

Rule entries contain the conditions a case must meet to be assigned to a user or a queue.

We can create as many assignment rules as we want but only 1 can be active at a time.

"Don't Reassign Owner" determines if the user whose process stack is in use becomes the owner of the rule entity, or if it remains owned by its creator user. Case Teams can also be assigned to cases which will help to close the cases more effectively.

Salesforce Email-to-Case

Email-to-Case in Salesforce allows us to generate a case automatically when a customer emails in. (generally on support email). It also populates some case information automatically saving manual effort.

Salesforce Email-to-Case should be used when you have one or more support email addresses which agents handle manually. We can configure more than one Email-to-Case channel if our organization has multiple support email addresses & processes. There are 2 methods to implement it:

Email to Case:

It uses an agent on the machine behind the organization's network firewall.

On-Demand Email to Case:

It uses Apex email services to convert email to cases without having to install an agent behind an organization's network firewall.

Major Differences b/w Email-to-Case & On-Demand Email to Case:

Email-to-Case	On-demand Email-to-Case
Email to case requires an agent to be installed behind an organization's network firewall.	On demand email to case is a simpler version of email to case which does not required an agent to be installed.
It does not contribute to API usage when creating cases.	It contributes to API usage when creating cases.
It accepts emails larger than 25 MB including header, message attachments.	It refuses emails larger than 25 MB.
It keeps email traffic within your firewall.	It does not keep email traffic within your firewall.
It handles attachments more than 10MB.	It only handles attachments from customers up to 10 MB.
When it first came it was known as Email-to-Case as a service.	

Enable Email-to-Case to start accepting emails that will be converted in cases. If we select On-Demand Email-to-Case there are 3 options that can occur in failure of response:

Bounce Message

The email service returns the message to the sender or to the Automated Case User for On-Demand Email-to-Case, with a notification that explains why the message was rejected.

Discard Message

The email service deletes the message without notifying the sender.

Requeue Message (Over Email Rate Limit Action Only)

The email service queues the message for processing in the next 24 hours. If the message is not processed within 24 hours, the email service returns the message to the sender with a notification that explains why the message was rejected.

Routing address specifies email ids from which the emails need to be converted as cases. There can be more than 1 routing address for an organization. After verifying the email id in the routing address Salesforce provides an Email Service Address on which we can get emails that will be converted in cases for our organization.

Note:

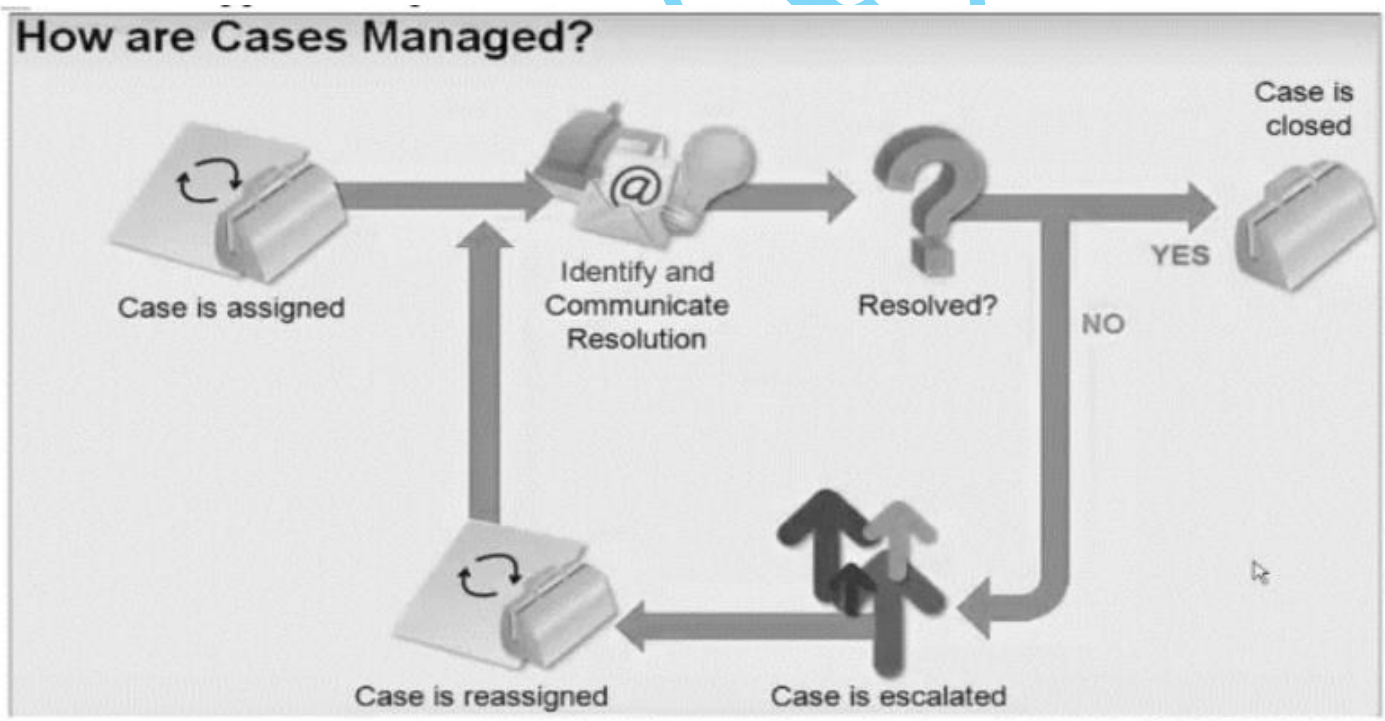
Salesforce will automatically relate the contact and its associated account to the case if the from email id matches the email id from any of the contact record's email id.

Salesforce Escalation Rules

Case escalation rules in Salesforce are used to reassign and optionally notify individuals when a case is not closed within a specified time period.

Salesforce Case Escalation rules are used to:

- Ensure that cases do not go unresolved by support agents or
- Ensure that the customer support team is meeting customer service-level agreements or
- Prioritize the support of important customers.



Case Management Lifecycle

We can create as many escalation rules as we want but only 1 can be active at a time. A rule can contain many rule entries. Rule entry specifies the:

- Sort Order which determines the order of evaluation of rule entries.

- Entry criteria which determine that through which rule entry the case is eligible for escalation.
- Business Hours criteria for the rule entry.

It specifies the hours on which the support team is available to serve customers. It makes the escalation process more accurate. Business hours help organizations set the working time of support teams of different locations and different time zones which makes it easy to calculate the time related to cases.

An organization can have many different business hours for different time, teams and locations. Every rule entry can have its own business hours set according to which the time for escalation gets evaluated. Also, there is an option which ignores the business hours and calculates the time irrespective of business hours.

Evaluation criteria of escalation times:

It helps to calculate the Age Over time of a case. It has 3 options according to which it is calculated:

- "When the case was created" calculates the age over time from the created date/time of case.
- "When the case was created and disable after the case is first modified" does not escalate the case if the case record is modified after creation and the Age Over time is calculated according to created date/time.
- "Based on last modification" calculates the age over time from the last modified date/time of case.

Note:

The calculation of age over time depends on the business hours also, if selected.

Escalation Actions

These define that after how much time the case should escalate and to whom the case should be reassigned. Also, we can specify to whom the notification email should be sent of its escalation with respective email templates.

Note:

We cannot set the same Age Over time for 2 actions in a rule entry.

Case Escalations can be monitored under Setup -> Monitor -> Case Escalations.

Solution Management in Salesforce

Solution is a detailed description of a customer's issue and the resolution of that issue. Solutions can streamline the communication of common case resolutions, improving support agent productivity and response time to customers. Thus, Salesforce Solution Management is very important.

Solutions are used to document and communicate common resolutions to cases.

For instance:

- If customer A calls in with a problem and steps 1-5 must be taken to resolve the issue, these steps can be entered into a solution.
- When customer B calls in with the same problem, instead of retyping steps 1-5, the customer support agent can reference the solution to quickly communicate the resolution.

Solutions tab displays a homepage that lets you quickly locate and manage solutions. If your organization uses solution categories, you can browse for and find solutions by category.

Important Fields:

Status:

This field defines the status of a solution whether it is saved as a Draft or as an Un-reviewed solution or as a reviewed solution.

Public:

It indicates whether the solution has been published or not as a public solution.

Visible in Knowledge Base:

It makes the solution visible in the public knowledge base where customers can see the solution without logging in or without registering the case.

Title:

Specifies the title for the solution.

Description:

Steps that will help to resolve the case using this solution.

Solution Management – Permissions

Solutions do not use sharing rules instead, object-level security and the "Manage Published Solutions" privilege determine what access the user has to solutions. Each solution can be optionally marked as "Reviewed". To edit/delete solutions in a "Reviewed" status, the user must have the "Manage Published Solutions" privilege.

Solution Category

Solution categories allows us to group similar solutions together. Create solution categories to keep solutions organized and to allow users to easily navigate your public solutions. Users can then browse and search solutions based on categories.

To add solutions under a category, add the solution category related list on the page layout and add the category on the solution.

Solution Management Process

You can set up different solution processes to control the steps your users follow for solution reviews because there is a different level of knowledge in each solution type, the review processes and values in the Status field could be different. By implementing different solution processes, you can have different review cycles for each kind of solution.

Setting up each process takes several steps:

Step1: Create the opportunity stages needed in the sales process.

Step2: Name the sales process, and select what opportunity stages are included in that particular process.

Step3: Create a record type for the sales process. Record types link the sales process to the page layout that goes with it.

Knowledge Articles (Salesforce Knowledge)

Salesforce Knowledge lets us create and manage our company information and securely share it when and where it is needed. Salesforce Knowledge base is built from knowledge articles, which are documents of information.

Articles can include information on process, like how to reset your product to its defaults, or frequently asked questions like, how much storage your product supports. Experienced service agents and internal writers write the articles. The

articles are then published to a range of channels: internal database, customer and partner communities, or public websites.

Portals and Communities in Salesforce

Salesforce Portals and Communities in Salesforce empower your customers and partners by providing a social forum directly related to your internal business processes so that they can connect with the right information and the right people at the right moments.

Portals in Salesforce:

Portals were a thing of the past they are not available for new organizations. When it was available there were different portals for organizations like

Customer Portal:

It allows customers to manage their cases, view solutions/knowledge, contribute to communities (questions, answers, ideas), and access data within custom objects.

Partner Portal:

The key difference between the customer portal and partner portal is that partner users can access leads and opportunities. This allows your organizations and its partners to collaborate on your organization's sales pipeline. Partners can also manage cases, view solutions/knowledge, contribute to communities (questions, answers, ideas), and access data within custom objects.

Self-Service Portal:

It allows customers to manage cases and view solutions/knowledge.

Salesforce Communities:

Communities in Salesforce replace these portals. Portals essentially gave external users (partners, customers, etc.) the ability to access Salesforce whereas Communities is aimed at connecting the right people (whether internal users, partners, or customers) together within Salesforce.

The structure of the features is similar, but there are several differences as well:

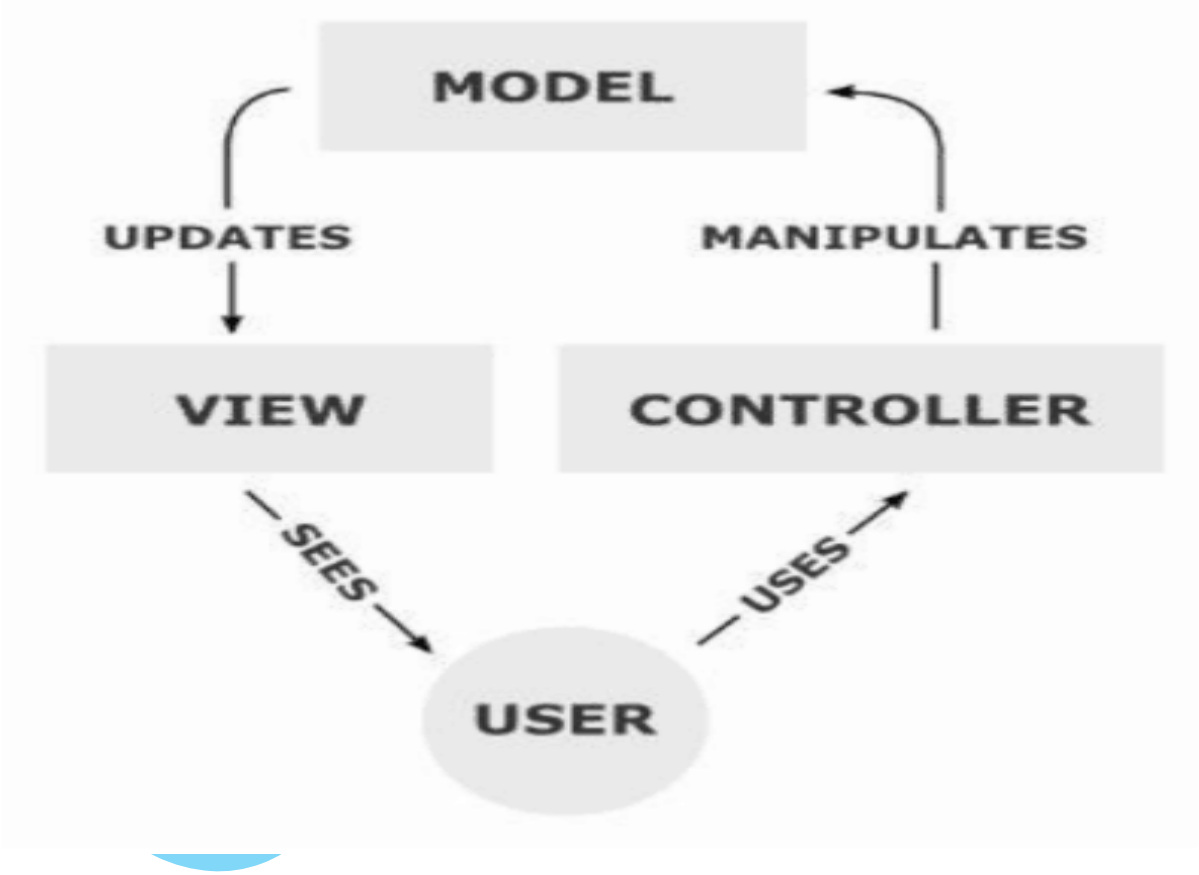
1. External users (partners/customers) can communicate via Chatter in Communities. Portals do not support Chatter.

2. The standard Communities user interface is very close to that of a regular internal Salesforce user. The out of the box portal user interface looks dated.

3. The licensing model for each is similar; however, Communities may have a slightly higher fee.

MVC Architecture in Salesforce

Model View Controller (MVC Architecture in Salesforce) is a software pattern that separates the representation of information from the user's interaction with it.



SFDC MVC:

- We can write our VIEW pages using SFDC visual force (VF pages).
- Each VF page is associated with a Controller.

- We can make use of already built Standard controllers or we can write our own controller using Apex language. Apex is OO and very much similar to JAVA.
- We can also write Model Classes using Apex.

MVC architecture in Salesforce pattern contains below three modules:

Model (Database Layer):

The schema and data salesforce use to represent the system completely. In Salesforce, we can say that sObjects are the model as every entity in salesforce is mapped to some sObject. (sObjects are any objects that can be stored in the Force.com platform database)

View (User Interface Layer):

How the schema and data are represented. Visualforce is used to present the data to users.

Controller (Business Logic Layer):

How the interface actions. Controllers are used to perform the actions whenever users interact with visual force.

In SFDC:

- Visual Force pages, Page Layouts, Tabs comes under View Layer of Model View Controller in Salesforce.
- Workflows, Apex Classes, Triggers comes under Controller part in Model View controller in Salesforce.
- Objects, Fields, Relationships come under Model Layer of Model View Controller in Salesforce.

Benefits of using MVC Architecture in Salesforce:

Reliability:

The layers have clear separation so it allows greater reliability if we want to change any layer. For example, if can change the look and feel of an application without recompiling Model or Controller code.

High Reuse and adaptability:

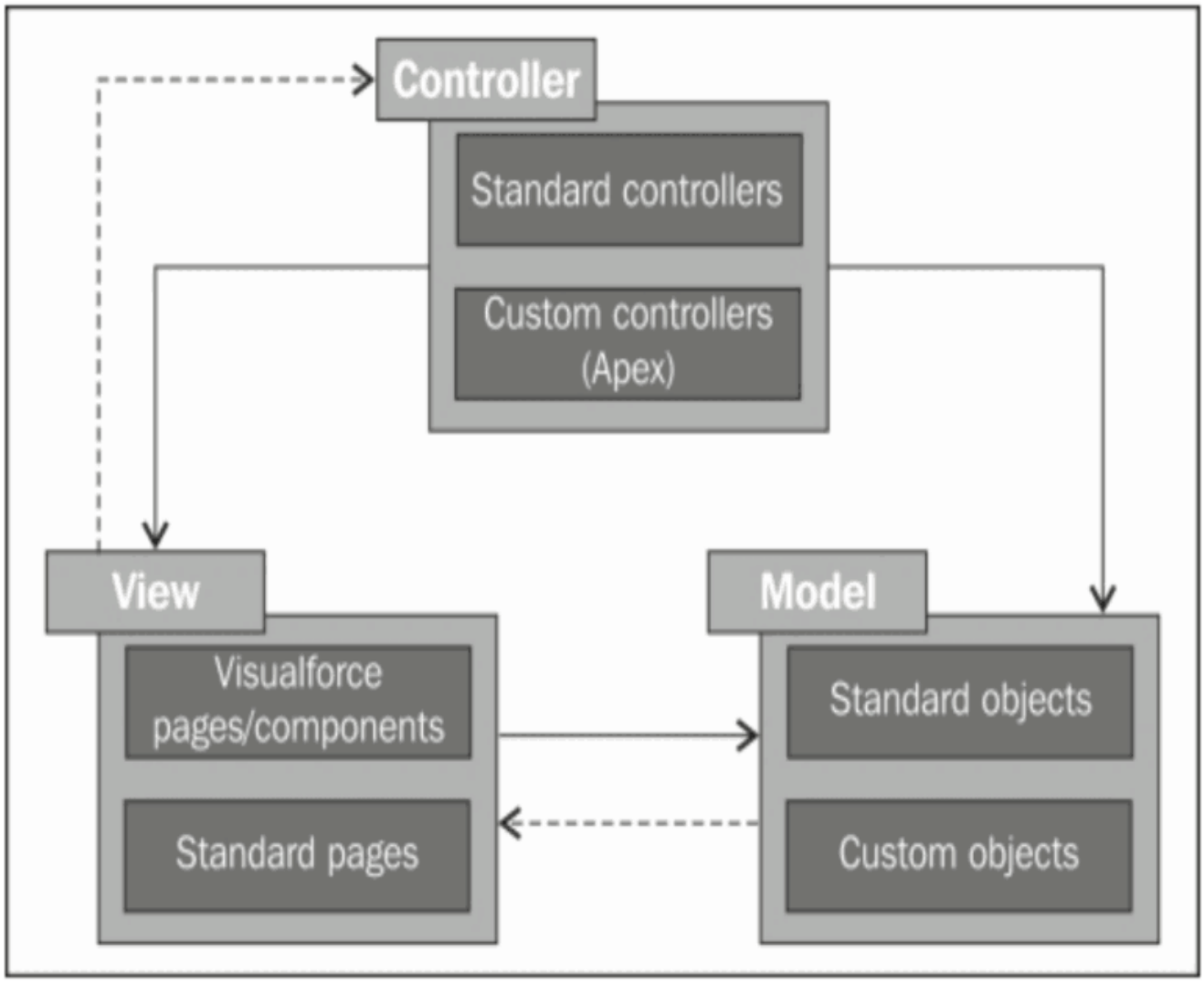
MVC provides multiple types of views all accessing the same code.

Very low development and life cycle cost:

It makes it possible to have lower-level programmers develop and maintain user interfaces. Development time can be significantly reduced because Controller programmers focus solely on transactions and view programmers focus solely on presentation.

Easy to maintain:

The separation between layers makes it easier to maintain, test, and deploy apps.



Implementation of MVC Architecture in Salesforce