Field Conventions

All Naming Conventions Are [RFC 2119]() and [RFC 6919]() compliant.

General Conventions

1. All field `API` names ***MUST*** be written in English, even when the label is in another language.
2. All field `API` names ***MUST*** be written in [PascalCase]().
3. Fields ***SHOULD NOT*** contain an underscore in the fields name, except where explicitely defined otherwise in these conventions.
4. Fields generally ***MUST (but you probably won't)*** contain a description.
5. In all cases where the entire purpose of the field is not evident by reading the name, the field ***MUST*** contain a description.
6. If the purpose of the field is ambiguous, the field ***MUST*** contain a help text. In cases where the purpose is clear, the help text ***COULD*** also be defined for clarity's sake.

Field API names should respect the following prefixes and suffixes. \\2* Prefixes and Suffixes ***SHALL NOT*** be prepended by an underscore.

Field Type	Prefix	Suffix
MasterDetail		`Ref`
Lookup		`Ref`
Formula		`Auto`
Rollup Summary		`Auto`
Filled by automation (APEX) \\1*		`Trig`
Picklist or Multipicklist		`Pick`
Boolean	`Is` or `IsCan` \\3*	
Created for use in DLRS	`DLRS`	

Grouping fields

1. If the organization is home to multiple services, the field `API` name ***SHOULD*** be prepended with the name of the service that required the field, followed by an underscore.
    - This ***MUST NOT*** be the case if there is only one service using the object.
2. If an Object has a large number of fields, Field `API` Names ***SHOULD*** be prepended by a category, followed by an underscore.

3. If several services use the field, or the field was originally required by a service before being used by others: the field `API` name ***MUST NOT*** be prepended with the name of the service. The Description of the field ***MUST*** indicate which services use the field. \\4*
4. In the case the field is use differently by different services, the Description of the field ***MUST*** contain an explicit description of each use.
5. If a field is created to host a value for technical reasons, but is not or should not be displayed to the users, the `API` name ***MUST*** be prefixed with TECH and an underscore.
6. If more than 50 fields are created on an object, a consultant ***SHOULD*** consider using prefixes to group fields in the same manner as technical fields, in the format of $GROUPNAME followed by an underscore.

Examples

Object	**Field type**	**Comment**	**Field Label**	**Field API Name**	**Field Description**
Case	Lookup	Looks up to Account	Service Provider	ServiceProviderRef__c	Links the case to the Service Provider who will conduct the task at the client's.
Account	Formula	Made for the Accounting department only	Solvability	Accounting_SolvabilityAuto__c	Calculates solvability based on revenue and expenses. Sensitive data, should not be shared.
Contact	Checkbox		Sponsored ?	IsSponsored__c	Checked if the contact was sponsored into the program by another client.
Contact	Text		Secondary Street	Adress_SecondaryStreet__c	Street of the secondary adress.

> \1 Workflows, Process Builders and Flows are not included in this logic because these automations either allow field name modifications with no error, or can be modified by an administrator. If fields are created for the sole purpose of being filled by automation (*e.g.* fields that will be used in roll-up summaries), a consultant ***WOULD PROBABLY*** use the Trig suffix anyway, to indicate that users cannot set the data themselves.*
>
> \2 while norms for other field types were considered, e.g. to make sure number, currency and percentage fields were easily recognizable, they were discarded as being too restrictive for an admin. fixing type mismatches in this case is easily solved by casting the value to the correct type using either TEXT() or VALUE() functions.*
>
> \3 `IsCan` replaces "Can", **e.g.** `CanActivateContract` becomes `IsCanActivateContract`. This is to enable searching for all checkboxes on a page withasingle query.*

> \4 While modifying API names post-deployment is notoriously complicated, making sure that field are properly recognizable is better in the long term than aviding a maintenance during a project. Such modifications ***SHOULD*** be taken into acccount while doing estimations.*