Admin Guide

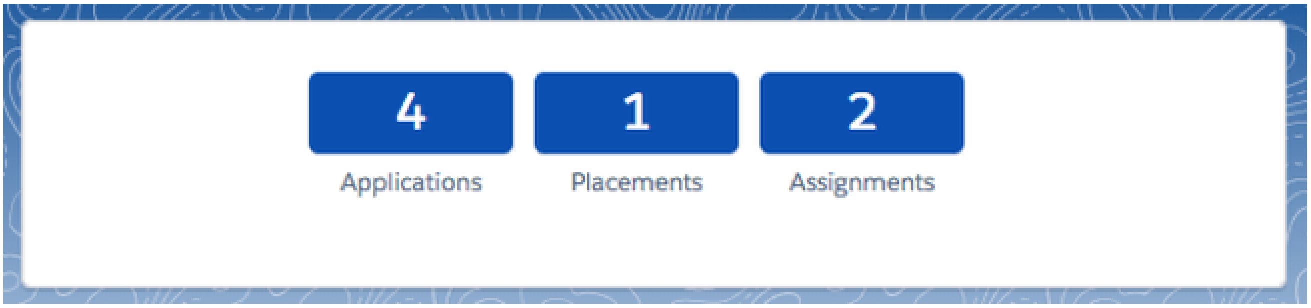
VERSION 1.5

About this Guide

The Status Components feature is being released in Beta as part of the Winter 2018 Business Release. This guide is intended for Jobscience administrators and implementation professionals. It includes the necessary steps required for initial configuration and deployment, as well as instructions for creating new components.

Status Components Overview

Knowledge workers today are faced with overwhelming amounts of information, especially in the staffing industry. Recruiters must review hundreds of candidates a week when sourcing for a job. With all the details captured on the typical candidate's record, it is difficult to get the summary needed to make a quick decision on whether to dive deep or pass.



Status Components enable an organization to summarize the most important insights for any record in the Jobscience, or other Salesforce application, so users can instantly understand the essence of the record they are viewing.

Note: Activity objects may be used to define components. This is especially important in the Sales and Business Development world when the number of Calls and In Person Meetings are important metrics to keep track of.

Jobscience provides a library of sample components, as well as examples in this document, to make it easy to create new components or to modify existing components to match the needs of your organization.

Note: Status Components are **Lightning Components** and as such are only available in Lightning Experience.

Application Examples

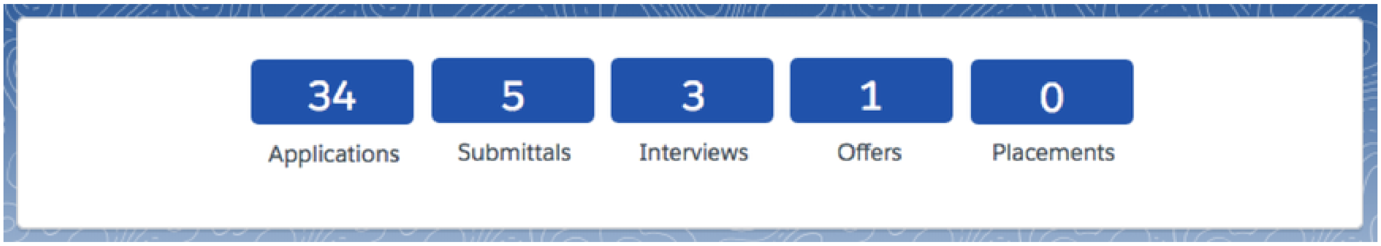
A concrete example will make it easier to understand.

The application process varies from organization to organization, but in general the major application stages are:

- Application
- Submittal
- Interview
- Offer
- Placement

When viewing a Job, it is important to see at a glance how many applications have been created for this Job, if any candidates have been submitted yet, if any have had interviews, and if any offers have been made.

By providing a component for each of these stages, we can provide an instant understanding of the current situation of the Job Search.



In this example, it is easy to see that 34 total candidates applied, five were submitted, and three of those five were interviewed. There was one offer made, but it has not been accepted since there are zero placements.

If you want an indication of active status in the Beta Release, you must create a separate component for that or a Check Component which indicates the status. See types of Components below.

This same set of components can be added to an Account record that represents a client to show their current situation across all jobs. The component set used on an account would likely include a component for number of total jobs and perhaps the number of active jobs.

A key feature of these components is the ability to define a single component and define a different set of rules for how the count is calculated based on the page it is being hosted on.

The same component definition is used for the Application Count across several objects, but the filter definitions are different when the page is an Account Page, Candidate Page, Contact Page, or Job Order Page. We will show how this is accomplished in the sections that follow.

Detailed Capabilities

Types of Components

The example above was provided for a **Count**, the first type of Component. These Components *count* the number of records that match the filter criteria defined.

Sum and **Average** are typically used with currency fields to total values across the records that match the filter criteria. **Sum** will accumulate the value found in a single field across all of the records that qualify for the filter. **Average** will calculate the same sum, but divide it by the number of records, which provides a simple average value. You can create a component, for example, that displays the average order value from a given client or total lifetime business.

The final type of component is a simple **Check Box**. This type of component is used to pull a simple check box field into the component set making it easier to see, or to combine multiple conditions from the records into a single, highly visible status.

Do not call
ActiveSearch

For example, if there are three separate activities that must be performed to vet a new candidate, a component called “Qualification” could be created which is checked only when all three of the other status values are complete.

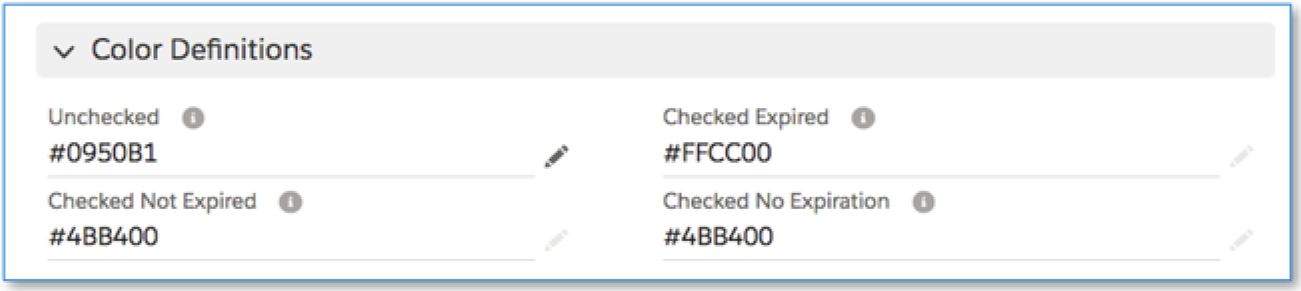
There are four potential states for a Check Box:

- Unchecked
- Checked Not Expired
- Checked Expired
- Checked No Expiration

The last one refers to a Checked Status where no expiration rules have been specified, or put a different way, never expires.

We have defined default colors for each of these states but recognize that there are times when a component deserves a special color treatment for each of the four states. For example, you might want to make the Checked versions of “Do Not Call” Red to emphasize the danger. Maybe you want to have the component completely fade away when inactive, so you change the color to White and make it disappear.

To configure these changes, simply look for the Color Definitions Section on the Check Box record type. Colors are in Hex RBG format and require the leading hash. So, our default Checked Green color is entered as #4BB400.



Color Definitions

Unchecked
#0950B1

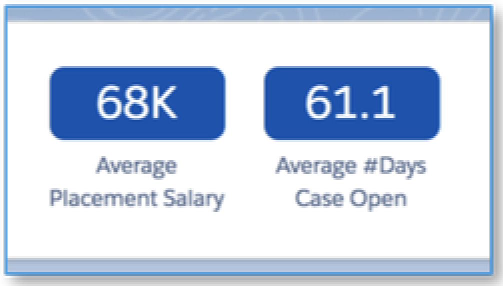
Checked Expired
#FFCC00

Checked Not Expired
#4BB400

Checked No Expiration
#4BB400

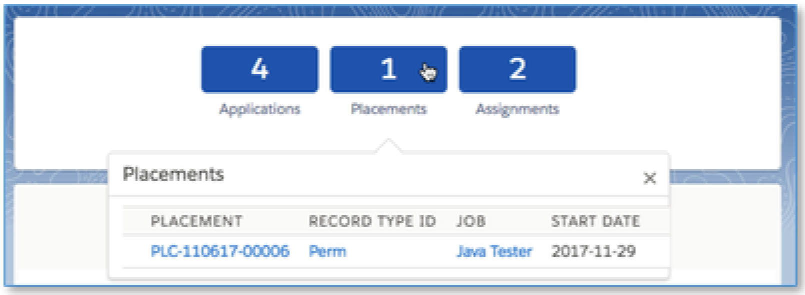
Display Formatting

The examples given above are simple counts, but what happens when a component provides a large sum or average. Revenue numbers can be in the millions of dollars. The Display formatting features give you control over how the information is displayed.



- The **Shorten Numbers** checkbox indicates that the component should express large numbers as thousands (K), or millions (M) to save room such as the 68K above in Average Placement Salary.
- The **Decimal Places** field indicates the number of digits to the right of the decimal to display with regular and abbreviated numbers like the single decimal place in 61.1 above average days open.

Hovers



PLACEMENT	RECORD TYPE ID	JOB	START DATE
PLC-110617-00006	Perm	Java Tester	2017-11-29

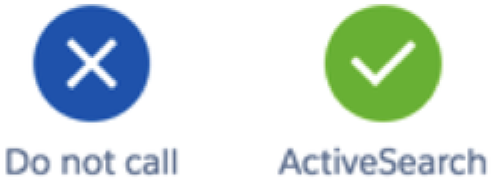
Hovers take these components to the next level by giving the user the detail behind the summary. As the name suggests, hover the mouse cursor over the component to

review the detail. Click on the line items to navigate to that item. In the example above, click on the Placement name to open the Placement, or the Job name to open the Job Order.

The columns displayed in this hover are configurable by the admin.

Expiration

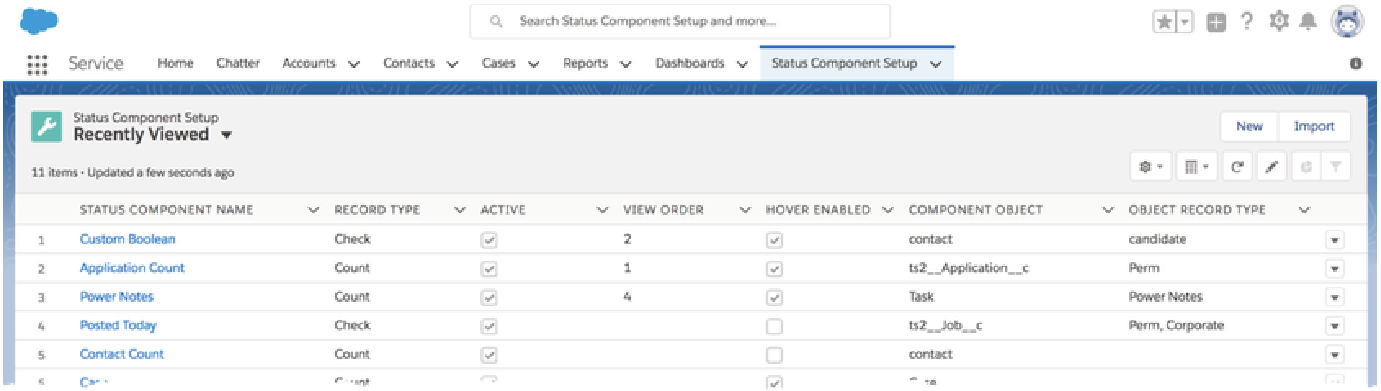
Sometimes a status is only good for a limited time. For example, a valid driver's license must be renewed periodically. Components with Check as their Type have a feature called **Expiration** that controls the background color of the component.



This example shows two components. The first is not checked and shows and X. This indicates that the Candidate is OK to call. The second shows that an Active Search is in progress, checked and Green.

Component Definitions

Components are defined using Custom Objects with Jobscience provided Page Layouts. The best way to access them is from a custom Tab called "Status Component Setup".

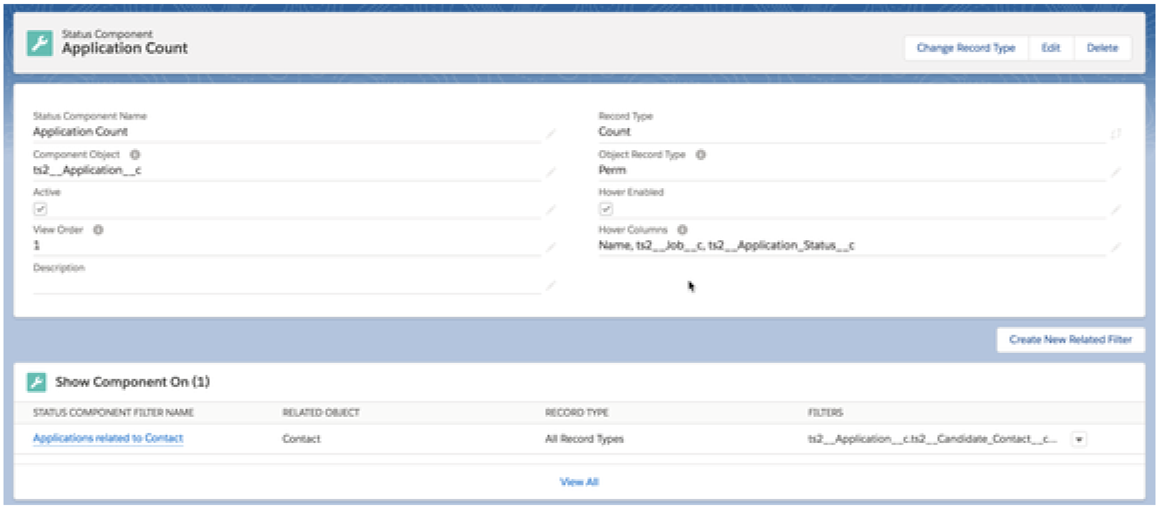


Status Component Setup
Recently Viewed

11 items • Updated a few seconds ago

	STATUS COMPONENT NAME	RECORD TYPE	ACTIVE	VIEW ORDER	HOVER ENABLED	COMPONENT OBJECT	OBJECT RECORD TYPE
1	Custom Boolean	Check	☑	2	☑	contact	candidate
2	Application Count	Count	☑	1	☑	ts2__Application__c	Perm
3	Power Notes	Count	☑	4	☑	Task	Power Notes
4	Posted Today	Check	☑		□	ts2__Job__c	Perm, Corporate
5	Contact Count	Count	☑		□	contact	

Clicking on the Component Name launches the following detail page.



Status Component
Application Count

Status Component Name Application Count	Record Type Count
Component Object ts2__Application__c	Object Record Type Perm
Active ☑	Hover Enabled ☑
View Order 1	Hover Columns Name, ts2__Job__c, ts2__Application_Status__c
Description	

Show Component On (1)

STATUS COMPONENT FILTER NAME	RELATED OBJECT	RECORD TYPE	FILTERS
Applications related to Contact	Contact	All Record Types	ts2__Application__c.ts2__Candidate_Contact__c...

Step 1—Basic Component Definition

The basic information required is presented in the detail section at the top.

The unique **Component Name** is just the name for this component. It should be concise but descriptive, usually an object and Component Type such as Application Count, or Case Average Days Open.

The Object used to calculate the component value is called the **Component Object**. It is formatted as the API Name for that Object such as “ts2__application__c”.

The Status Component often applies to a subset of record types available for the Component Object. In the recruiting use cases, Permanent Placements and Temporary Placements have very different types of information. **Object Record Type** acts as a filter for the Component Object above. If this field is left blank, all record types will be considered. If a single record type is listed, only Objects of that type are considered. If multiple Record types are required, they are provided in a comma separated list.

The **Active Flag** is used to enable or disable this component. When Active, this component will display on any page that includes the Status Component and matches one of the Filters provided. To disable the component, uncheck this box.

If several components are active for a given page, **View Order** will control which is displayed first (far left) through last (far right). The numbers should be integers starting with 1. They do not have to be sequential and in case of a tie, the system will choose the order.

The **Record Type** of this Component determines the behavior of the Component as well as the fields displayed on the page layout of the Component detail page. These record types correspond to the types presented above:

- Count
- Average
- Check
- Sum

When the Average or Sum Component Record Type is selected, an additional field is revealed in the page layout. **Sum Field** should contain the API name of a numeric or currency field on the Component Object. The data in this field will be summed or averaged as dictated by the Sum or Average record type. This field is required for Sum and Average Component Record Types.

Description provides the space to document more about the behavior of this component. It is intended for use by Admins and is not displayed to end use

Step 2—Basic Formatting

The remaining fields of the **Status Components Detail** page contains the selections for formatting the Component and Hovers.

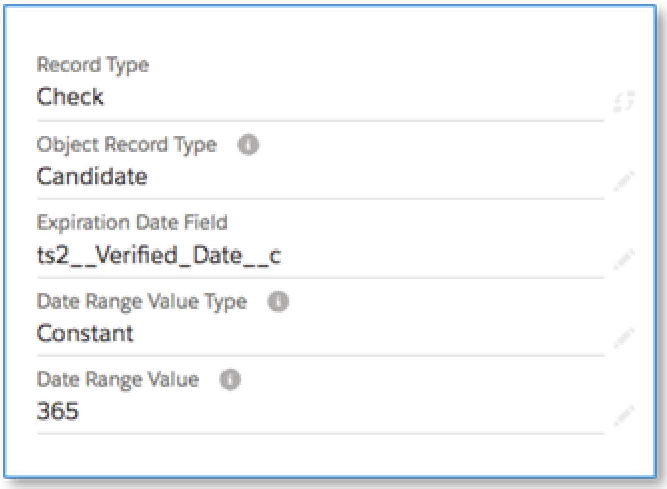
To abbreviate thousands to K and millions to M, check the **Shorten Numbers** checkbox.

For **Shortened Numbers** and calculated fields like **Sum** and **Average**, select the number of decimal places you want to display to the right of the decimal.

We recommend you enable hovers for most components by selecting the **Hover Enabled** checkbox and select the important columns to display in the **Hover Columns** field. Provide a comma-separated list of the API names for the fields you want to display in the Hover Columns field.

Step 3—Expiration Settings

Expiration Tracking is set with three controls in the bottom right section of the **Status Components Detail** page for Check Type Components.



Record Type
Check

Object Record Type
Candidate

Expiration Date Field
ts2__Verified_Date__c

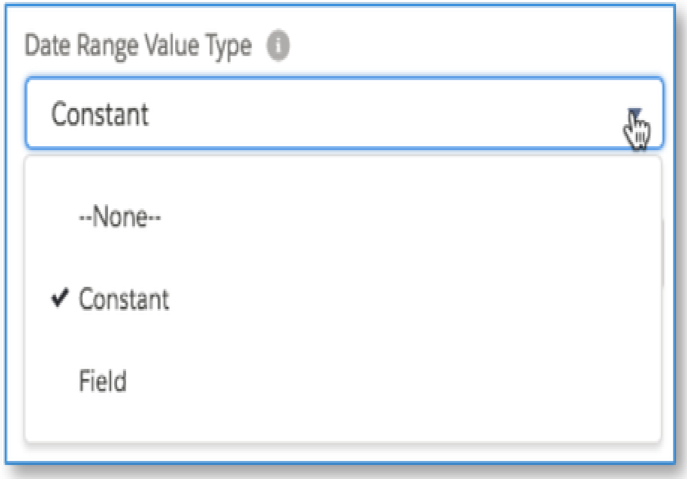
Date Range Value Type
Constant

Date Range Value
365

The first required field is the **Expiration Date Field**, which indicates the date that this activity was last performed. Supply an API name for a date field on the Component Object.

The system calculates the days elapsed since this date to compare with the **Date Range Value**. There are two ways to specify Date Range Value.

The first is as a constant within this Component Definition. Select ‘Constant’ as the **Date Range Value Type** then set the **Date Range Value** to the number of days that this value is valid.



Date Range Value Type

Constant

--None--

✔ Constant

Field

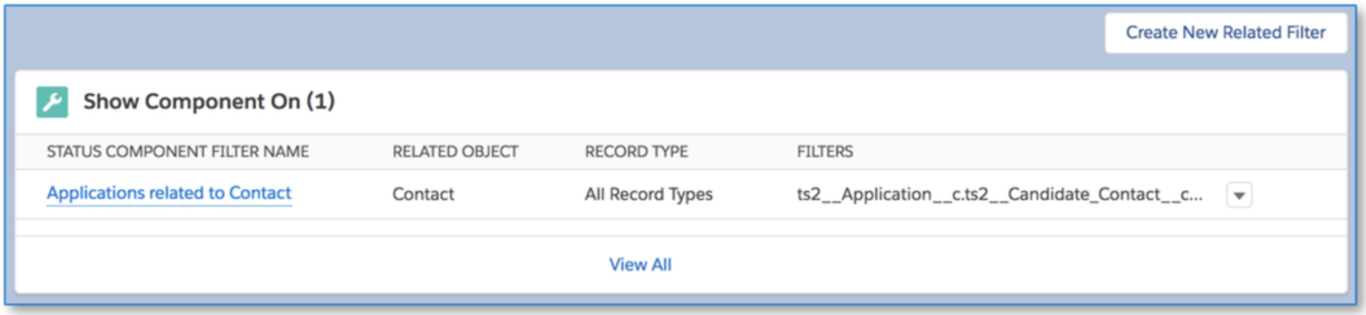
With these settings in the example above, the system determines the elapsed days from the date entered in the fields specified and if greater than 370 sets the background to Blue. If less than or equal to 370, the background is set to Green.

The second way is to state the **Date Range Value Type** as ‘Field’ and **Date Range Value** as an API name for the field that contains the number of days. In this case, the system will look up the number of days contained in the **Date Range Value** field and compare the elapsed days to this value, which could be different for every record.

Step 4—Define Related Filters

The steps above determine how to calculate the number and format the display of the information in general. But which records are used to calculate the number? The answer is that it may vary depending on which page the component is embedded in. This is where the **Related Filters** come in to play.

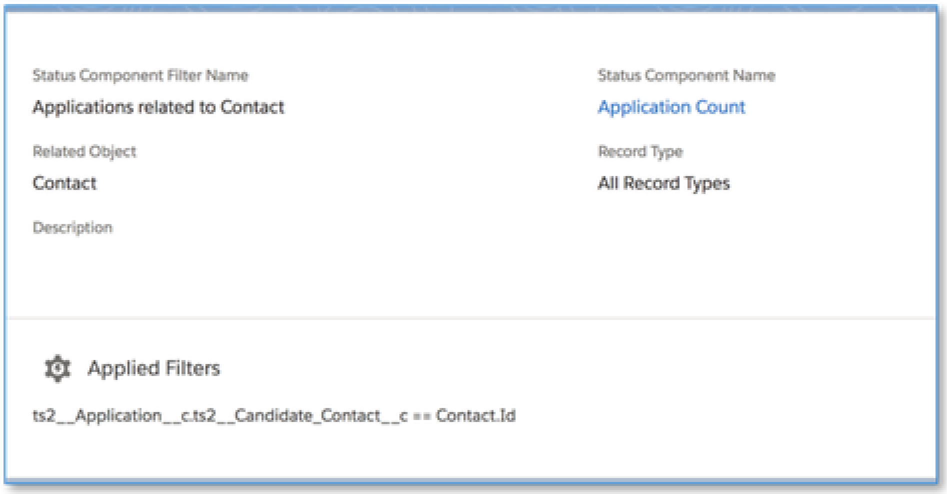
You will find the filter definitions on the **Filters** related list at the bottom of the **Status Components Detail** page.



Show Component On (1)

STATUS COMPONENT FILTER NAME	RELATED OBJECT	RECORD TYPE	FILTERS
Applications related to Contact	Contact	All Record Types	ts2__Application__c.ts2__Candidate_Contact__c...

Clicking on the Component Filter Name will open the Filter Detail Page.



Status Component Filter Name
Applications related to Contact

Status Component Name
Application Count

Related Object
Contact

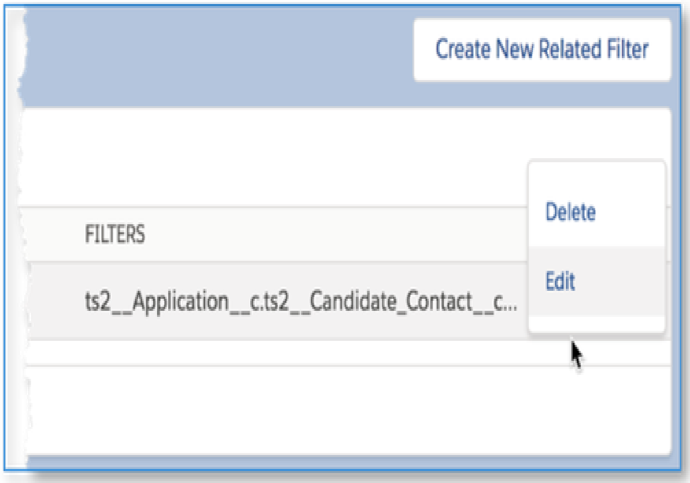
Record Type
All Record Types

Description

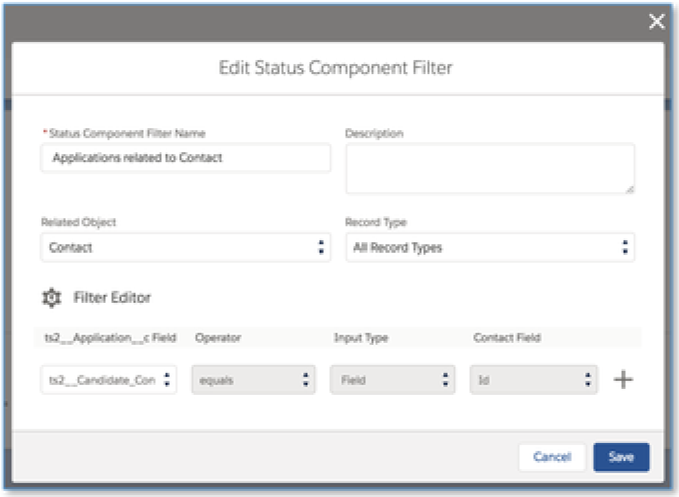
Applied Filters

ts2__Application__c.ts2__Candidate_Contact__c == Contact.Id

This is only marginally better that the related list view for most simple filters.

Clicking on the arrow on the far right of any existing filter will give you the options to Edit or Delete the related filter.	 FILTERS ts2__Application__c.ts2__Candidate_Contact__c...

Clicking on **Edit** will provide the following edit screen:



Edit Status Component Filter

*Status Component Filter Name: Applications related to Contact

Description:

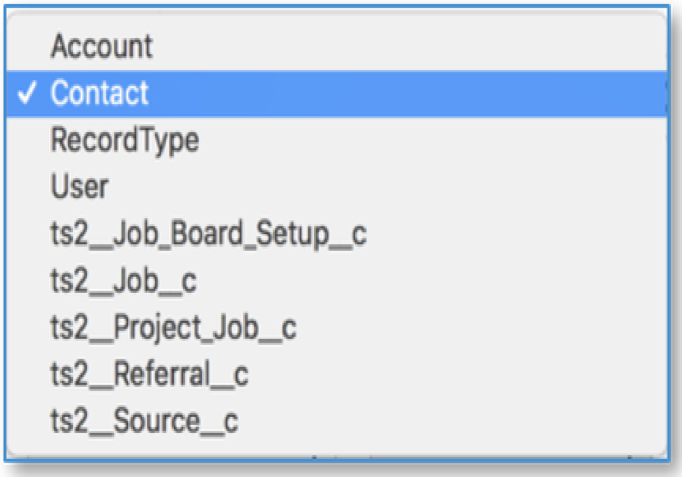
Related Object: Contact

Record Type: All Record Types

Filter Editor

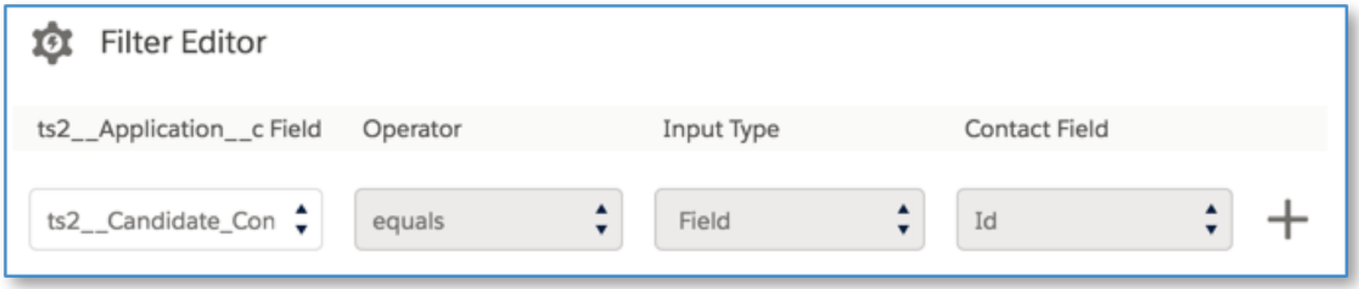
ts2__Application__c Field	Operator	Input Type	Contact Field
ts2__Candidate_Con	equals	Field	Id

The **Status Component Filter Name** should be descriptive and include some notion of the type of filters applied.

 Account ✓ Contact RecordType User ts2__Job_Board_Setup__c ts2__Job__c ts2__Project_Job__c ts2__Referral__c ts2__Source__c	The Related Object identifies the Object associated with this Filter. The only objects presented in this pick list are objects which have a Lookup or Master Detail relationship with the Parent Component Object. In this case the Application Object.

In our example, we want to define how an Application Count works on a Contact Object, so we select Contact as the **Related Object.** There is a **Record Type** field on the filter to restrict the types of Contacts we care about.

The heart of this filter is the List of one or more conditions specified under the Filter Editor.



Filter Editor

ts2__Application__c Field	Operator	Input Type	Contact Field
ts2__Candidate_Con	equals	Field	Id

The meaning of the filter shown here is:

"When the Application Component is displayed on the Contact Record,

Only count Applications where the ts2__Candidate_Contact__c field

Equals the ID Field of this Contact Record."

In other words, only count Applications made by this Candidate. This is a common pattern for any object that appears on a related list, like Application related lists on a Candidate record.

Note that when the Component is based on the Activity object, the **What** relationship is used to determine the filters.

Additional filter rows can be added to further qualify applications using any field on the Application object. For example, the Stage and Substage can be used to provide a component that counts only the number of Rejected Applications for this candidate.

Almost any field from the Application Record can be used to select which Applications to Count.

Input Type can be set to either Field or Value.

1. *Field* enables matching fields on the Contact record.
2. *Value* enables matching Constants. For example, Application Stage = Interview could be used if I wanted to count the number of Applications this contact has in the Interview stage.
3. *Value* has preset logic for **true**, **false**, and **null** to check the value of checkbox field and see if a field is empty.
4. Value has preset logic for the following date comparisons **Yesterday, Today, Tomorrow, Last_Week, This_Week, Next_Week, Last_Month, This_Month, Next_Month, Last_Quarter, This_Quarter, Next_Quarter, Last_Year, This_Year, Next_Year, Last_Fiscal_Quarter, This_Fiscal_Quarter, Next_Fiscal_Quarter, Last_Fiscal_Year, This_Fiscal_Year, Next_Fiscal_Year**

Step 5—Add the Component to the Page Layout

Jobscience has defined a Lightning Component named “Status Component” that should be added to any page where you want the Status Components to show up. The component has a “Justification” option that controls alignment of individual Status Components on the page. The options are “Left, Right, Center, and Spread.”

We recommend adding it as the top most component, but if can be added anywhere on the page. This top-level container is the same for every page. The actual components that appear in this space depend on three rules:

1. The **Active Status** checkbox for each component must be checked in order for the component to appear.
2. **Check Type Components** only appear on pages that match its object and record type on the Main Component Definition.
3. **Count**, **Sum**, and **Average Component** only appear on pages that match the Related Object Filters, including Record Type.

If we define an Applications Component and define Filters for Job Orders, Candidates, and Accounts, you will see this component on the Job Order, Candidate, and Account pages if the Status Component has been added to the page layout. It will not show up on the Contact page unless there is a rule for a Contact Filter.

Tips and Tricks

When designing components, check out our existing components and try to find one that is close to what you need. Then make a copy as a starting point to further refine it.

Be very careful to understand the data model and the difference between fields and objects which are transient in nature and those that are more permanent.

For example, the Stage and Status field on objects like Opportunities, Cases, and Applications are by nature transient and reflect the current status. If you want to provide a count of Opportunities currently in the Negotiating stage, then your filter can easily identify these.

“Closed - Lost” and “Closed - Won” are terminal stages. The Opportunity Stage is expected to remain in this state once it is reached. So, using these as filters to determine the total wins and losses for a given account is easy. Cases with a Closed Status are likewise in a Terminal state.

But what if you want to track the number of Opportunities that reached the Negotiating Stage before being Closed? Or how many Applicants made it to the Internal Interview, Submittal, and Client Interview Stage before being rejected? These stages were once valid but have now changed. You must find a permanent artifact in the database to record the fact that this stage was reached.

In Jobscience, we have several objects that were built to serve this purpose, but not everyone uses them. The best way to keep track of status/stage history is to create a completed task record to memorialize the event. These may be created with Business Process logic or triggers if you have development resources. Our professional services organization can create these as well. Use the Subject along with the Who and What lookups to log information in the record and use these as your filters.

So, in the case of counting Applicants which reached the Submittal stage, if you are not using our Submittal Records, you should create a Completed Task when the Applicant hits the Submittal Stage and count these Tasks instead of the Applications which are currently in the Submittal stage.