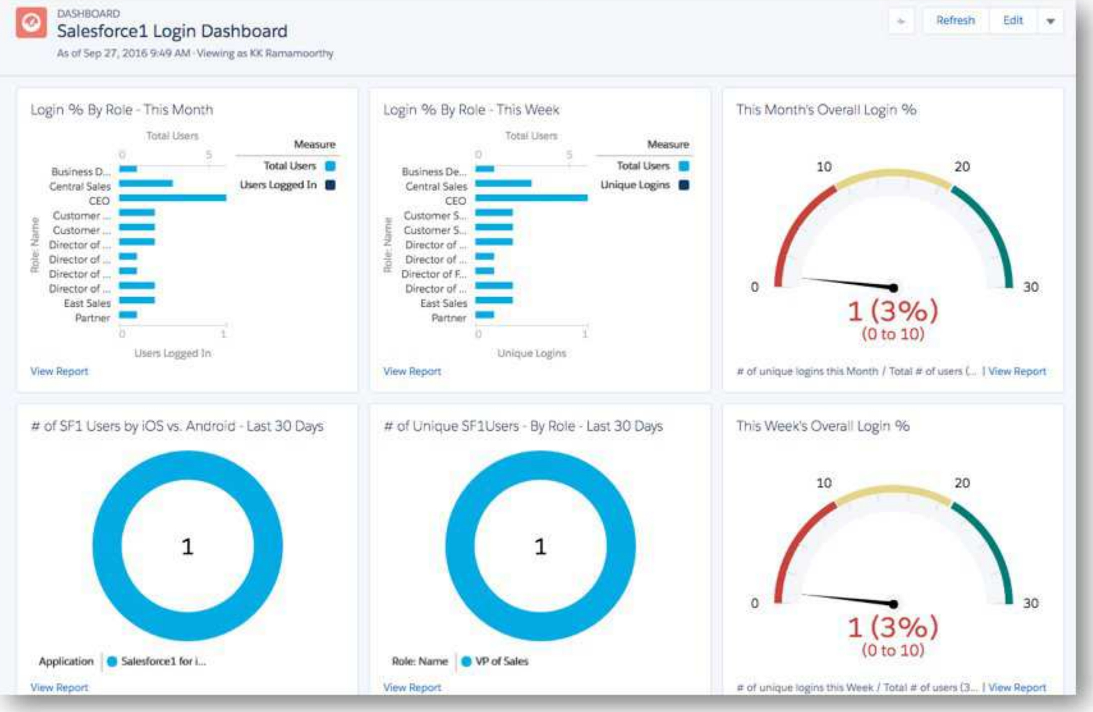
Monitor Adoption of Salesforce1 app

Salesforce1 empowers you to run your business on the phone. However, it is only as powerful as the number of users actually using it. While building and rolling out compelling mobile use cases is important, it is also very important to monitor its adoption and the user engagement. To that point, this document provides you with tips and tricks to measure adoption and engagement of Salesforce1 in your org.

Who is using the Salesforce1 app?

Salesforce stores various login events in a standard object called 'Identity Event Logs'. Among other things, this object tracks the application that is used by the user to access Salesforce and if the user uses Salesforce1 mobile app, the 'Application' field will contain the value as 'Salesforce1'. So, an administrator can create custom reports on this object to monitor usage of Salesforce1 app. But wait, Salesforce has made this easier for you. Instead of having to reinvent the wheel, go ahead and install this free AppExchange package that comes prepackaged with a bunch of useful reports and dashboards. Once installed, navigate to "SF1 Login Reports" and "SF1 Login Dashboards" folders to check out these reports and dashboards. You can monitor various key adoption metrics like 'Overall Login %', 'Login % by Role', '# of Users by iOS vs Android' etc.

Provided below is a sample screenshot of "SF1 Login Dashboard"

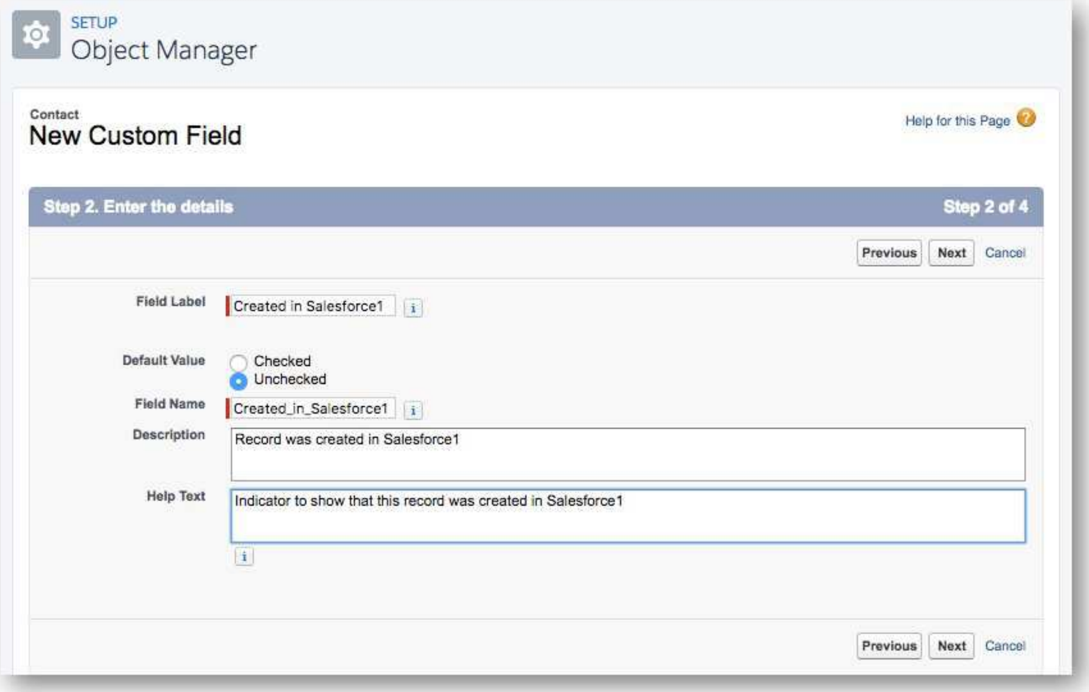


Great, how about engagement?

While the adoption reports and dashboards provide you a great view of who is using the Salesforce1 app, it doesn't really tell you what they are using it for. Let's say, for example, you created this awesome quick action for creating contacts using Salesforce1 app and want to know which users are using it, how often are they using it and when they are using it? You can achieve this using some simple declarative capabilities of the platform. Let's take the example of contact creation quick action to walk through the approach.

STEP 1:

Create a custom field of type 'checkbox' in the contacts object. Call it 'Created in Salesforce1'. While at it, make sure you don't add this field to any page layouts, you don't want your users manually updating this field



Field Label: Created in Salesforce1

Default Value: ○ Checked ● Unchecked

Field Name: Created_in_Salesforce1

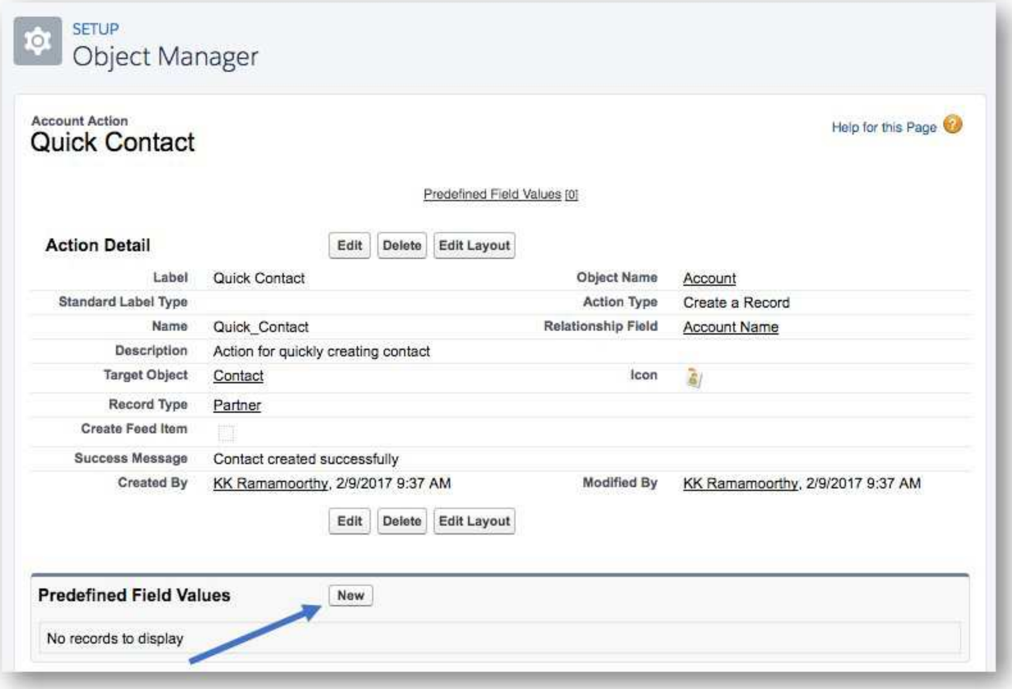
Description: Record was created in Salesforce1

Help Text: Indicator to show that this record was created in Salesforce1

STEP 2:

Go to the quick action and create a new pre-defined field using the custom field 'Created in Salesforce1' that you created in the above step. For the field value, use the formula below

```
IF( $User.UIThemeDisplayed == "Theme4t", TRUE, FALSE)
```



Account Action

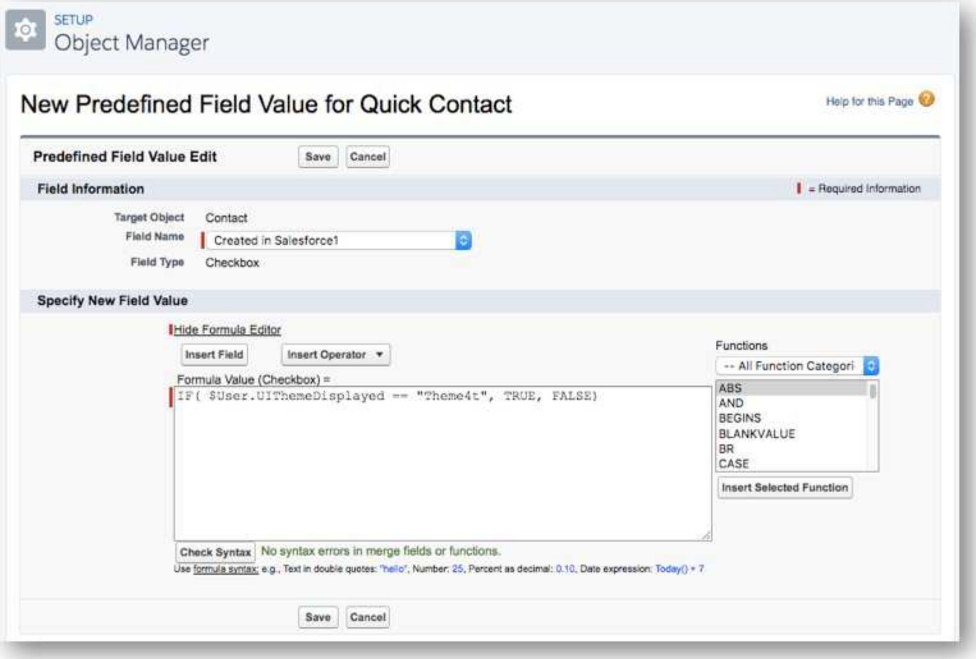
Quick Contact

Action Detail

Label	Quick Contact	Object Name	Account
Standard Label Type		Action Type	Create a Record
Name	Quick_Contact	Relationship Field	Account Name
Description	Action for quickly creating contact		
Target Object	Contact	Icon	
Record Type	Partner		
Create Feed Item	□		
Success Message	Contact created successfully		
Created By	KK Ramamoorthy, 2/9/2017 9:37 AM	Modified By	KK Ramamoorthy, 2/9/2017 9:37 AM

Predefined Field Values

No records to display



New Predefined Field Value for Quick Contact

Predefined Field Value Edit

Field Information | = Required Information

Target Object	Contact
Field Name	Created in Salesforce1
Field Type	Checkbox

Specify New Field Value

Formula Value (Checkbox) =

```
IF( $User.UIThemeDisplayed == "Theme4t", TRUE, FALSE)
```

No syntax errors in merge fields or functions.

Use formula syntax: e.g., Text in double quotes: "hello", Number: 25, Percent as decimal: 0.10, Date expression: Today() + 7

Let's take a look at what this formula actually does. The field 'UIThemeDisplayed' in the user object holds various values depending the user interface that user uses to access Salesforce. For e.g., for Salesforce1 it will carry a value of 'Theme4t'. Refer this documentation for more information about this field. Since the 'Created in Salesforce1' is a checkbox, in this formula we are checking if the user is using Salesforce1 app (UIThemeDisplayed == "Theme4t") and if yes, then setting the value of the checkbox to 'true'. If not, we set it to 'false'. In other words, the only way this field will have a value of 'true' is if this quick action was used by the user in the Salesforce1 app

STEP 3:

Now you can create a report on Contact object and use the 'Created in Salesforce1' check-box to analyze how many contacts where created using Salesforce1 vs other user interfaces

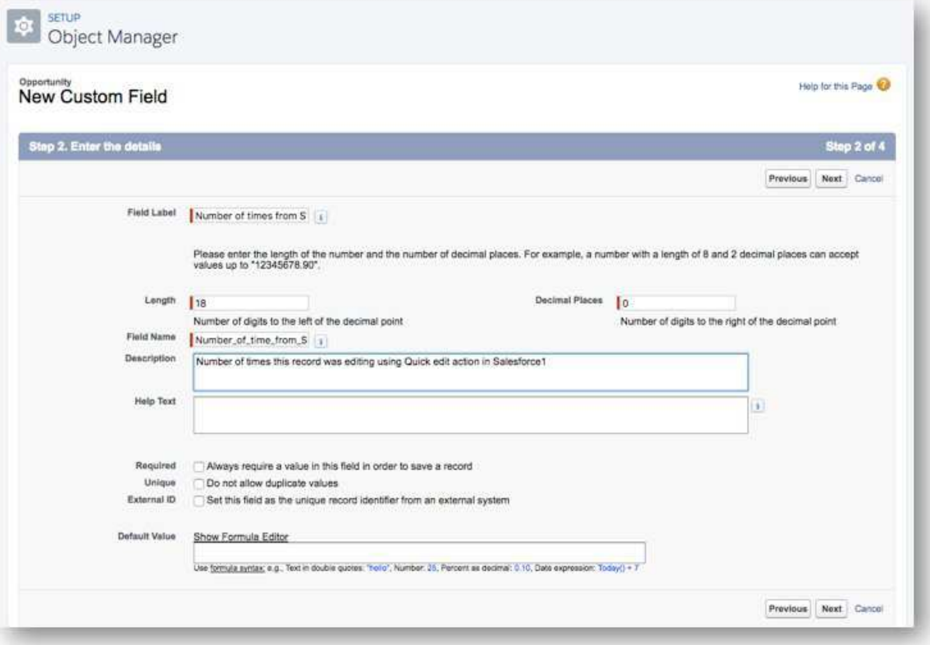
Instead of 'Created in Salesforce1' checkbox, you can also just capture the 'UIThemeDisplayed' field value in a custom field of type 'text'. Then, in your report, you can compare between the user interfaces your users are using to create contacts. For e.g., Lightning Experience vs Classic vs Salesforce1.

Cool, how about record updates?

Let's say you have a quick action to quickly update an opportunity using Salesforce1 app. Unlike 'Create' action, updates happen multiple times on the same record and you may want to measure how often your users are using the quick action in Salesforce1 to update the record. Here is one way achieve this.

STEP 1:

Create a custom field of type 'number' in the opportunities object. Call it 'Number of times from S1'. While at it, make sure you don't add this field to any page layouts, you don't want your users manually updating this field



Field Label: Number of times from S

Please enter the length of the number and the number of decimal places. For example, a number with a length of 8 and 2 decimal places can accept values up to "12345678.90".

Length: 18
Number of digits to the left of the decimal point

Decimal Places: 0
Number of digits to the right of the decimal point

Field Name: Number_of_time_from_S

Description: Number of times this record was editing using Quick edit action in Salesforce1

Help Text:

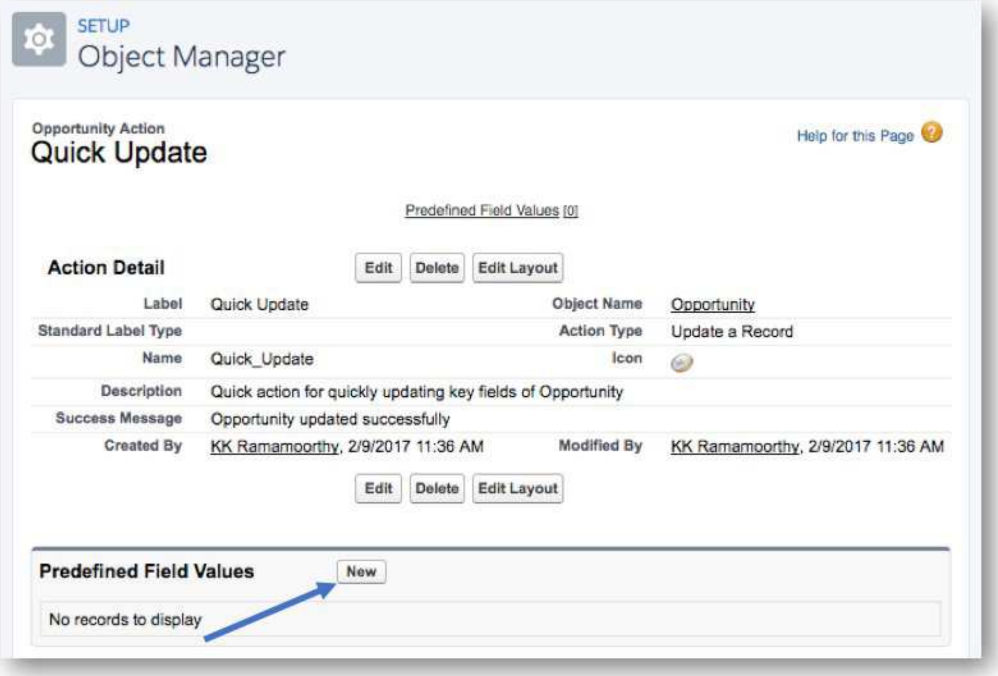
Required: □ Always require a value in this field in order to save a record
Unique: □ Do not allow duplicate values
External ID: □ Set this field as the unique record identifier from an external system

Default Value:
Use formula syntax: e.g., Text in double quotes: "hello", Number: 25, Percent as decimal: 0.10, Date expression: Today() + 7

STEP 2:

Go to the 'opportunity update' quick action and create a new pre-defined field using the custom field 'Number of times from S1' that you created in the above step. For the field value, use the formula below

```
IF( $User.UIThemeDisplayed == "Theme4t",

    IF(ISNULL(Opportunity.Number_of_time_from_S1__c),

        1,

        Opportunity.Number_of_time_from_S1__c + 1

    ),

Opportunity.Number_of_time_from_S1__c

)
```



SETUP

Object Manager

Opportunity Action

Quick Update

Help for this Page

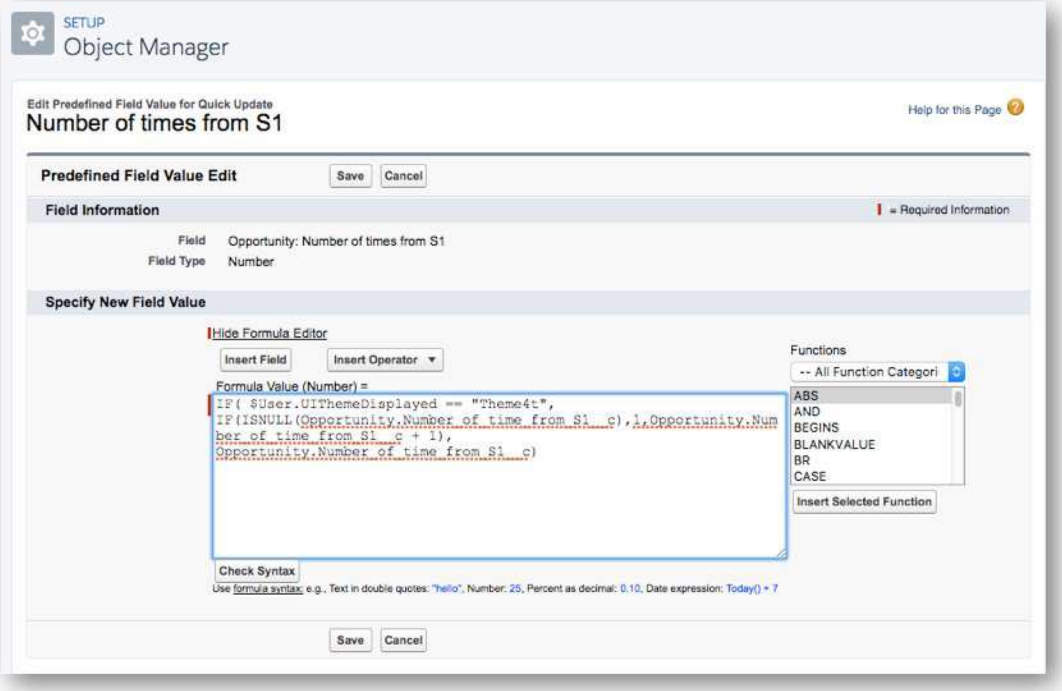
Predefined Field Values [0]

Action Detail

Label	Quick Update	Object Name	Opportunity
Standard Label Type		Action Type	Update a Record
Name	Quick_Update	Icon	
Description	Quick action for quickly updating key fields of Opportunity		
Success Message	Opportunity updated successfully		
Created By	KK Ramamoorthy, 2/9/2017 11:36 AM	Modified By	KK Ramamoorthy, 2/9/2017 11:36 AM

Predefined Field Values

No records to display



Edit Predefined Field Value for Quick Update

Number of times from S1

Predefined Field Value Edit

Field Information

| = Required Information

Field: Opportunity: Number of times from S1

Field Type: Number

Specify New Field Value

Formula Value (Number) =

```
IF( $User.UIThemeDisplayed == "Theme4t",
IF(ISNULL(Opportunity.Number_of_time_from_S1__c),1,Opportunity.Num
ber_of_time_from_S1__c + 1),
Opportunity.Number_of_time_from_S1__c)
```

Use formula syntax: e.g., Text in double quotes: "hello", Number: 25, Percent as decimal: 0.10, Date expression: Today() + 7

STEP 3:

Now you can create a report on Opportunities object and use the 'Number of times from S1' checkbox to analyze how many times an opportunity was updated using Salesforce1 vs other user interfaces.

You can also create a new custom field of type 'Date/Time' and capture the Date and time of updates using formula function Now(). This will enable you to monitor when the user is using Salesforce1 to make the updates

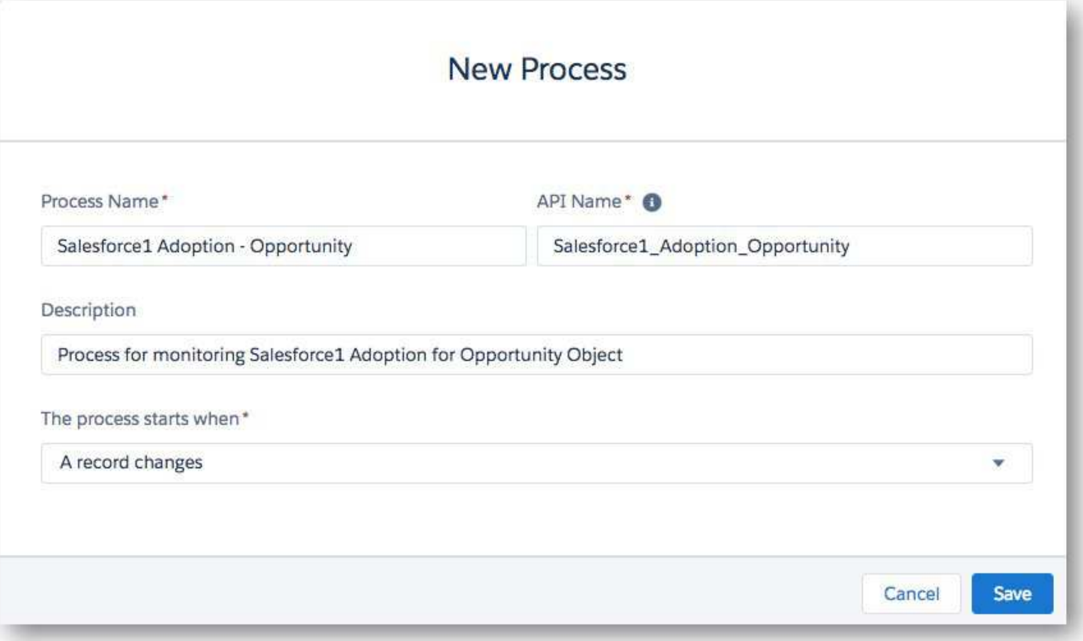
Brilliant, how about standard create and update actions?

Fair question! The approach detailed above works great for custom quick actions but a user might still use a standard create or update action instead. How do we capture those? Sweat not! the declarative tools strike again! We can use Process Builder to achieve this requirement. If you haven't worked with Process Builder before, now is the time to skill up using this trailhead module. Lets take the example of opportunity object again. The steps below assume that you have already created the custom fields 'Created in S1' and 'Number of times from S1' in the opportunity object

> *NOTE: The approach detailed below will also work for custom actions. So, if you choose to use the process builder approach, you won't need the 'pre-defined fields' approach described above as the process builder approach is all-encompassing. It includes creates and updates from both standard and custom actions in Salesforce1. Your choice!*

STEP 1:

Create a new process builder process. Make sure the process is set to start when 'a record changes'



New Process

Process Name*

Salesforce1 Adoption - Opportunity

API Name*

Salesforce1_Adoption_Opportunity

Description

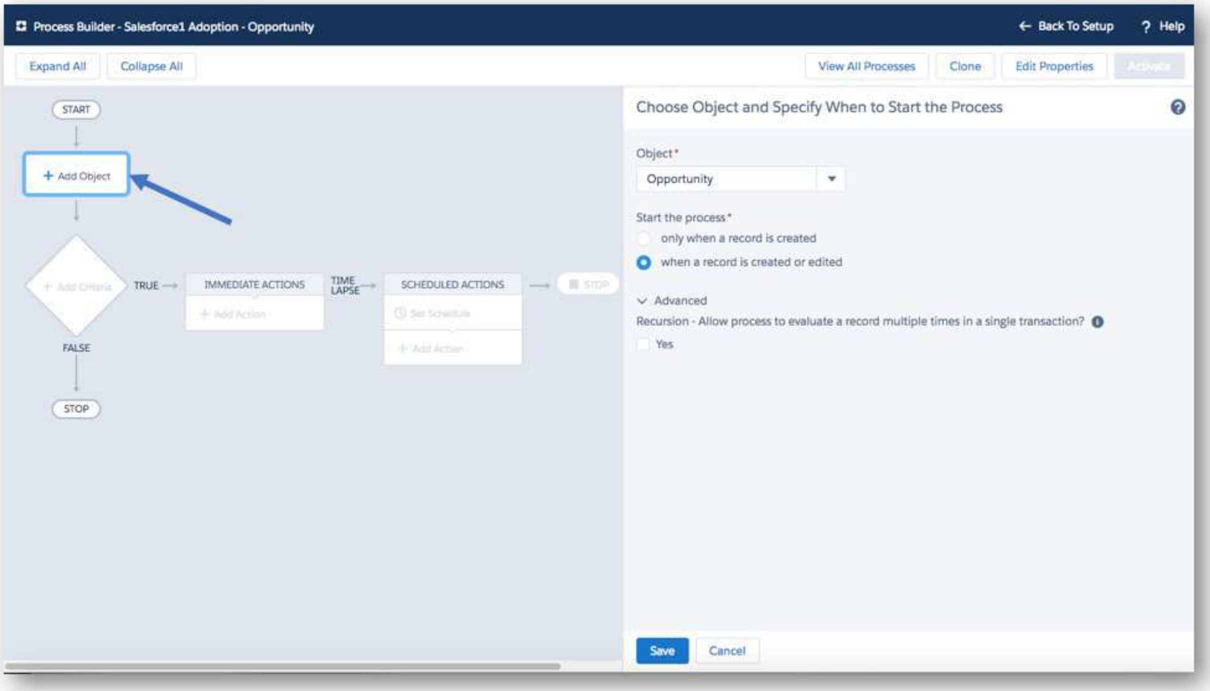
Process for monitoring Salesforce1 Adoption for Opportunity Object

The process starts when*

A record changes

STEP 2:

Click on 'Add Object' and choose 'Opportunity' for the Object and choose to start the process when the record is created or edited. Click on 'Save'.



Process Builder - Salesforce1 Adoption - Opportunity

Choose Object and Specify When to Start the Process

Object*: Opportunity

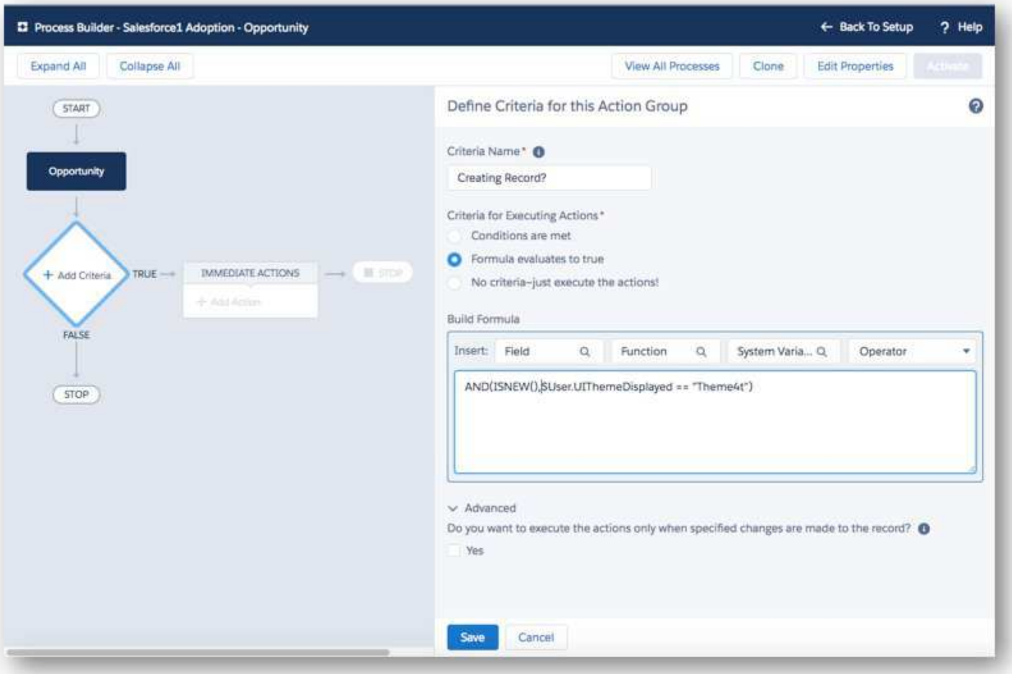
Start the process*
○ only when a record is created
● when a record is created or edited

Advanced
Recursion - Allow process to evaluate a record multiple times in a single transaction?
□ Yes

STEP 3:

Click on 'Add Criteria' and provide the Criteria Name as 'Creating Record?'. Make sure to choose 'Formula evaluates to true' radio button under 'Criteria for Executing Actions' and then use the formula in the formula editor as provided below

```
AND(ISNEW(),$User.UIThemeDisplayed == "Theme4t")
```



Process Builder - Salesforce1 Adoption - Opportunity

Define Criteria for this Action Group

Criteria Name*: Creating Record?

Criteria for Executing Actions*
○ Conditions are met
● Formula evaluates to true
○ No criteria–just execute the actions!

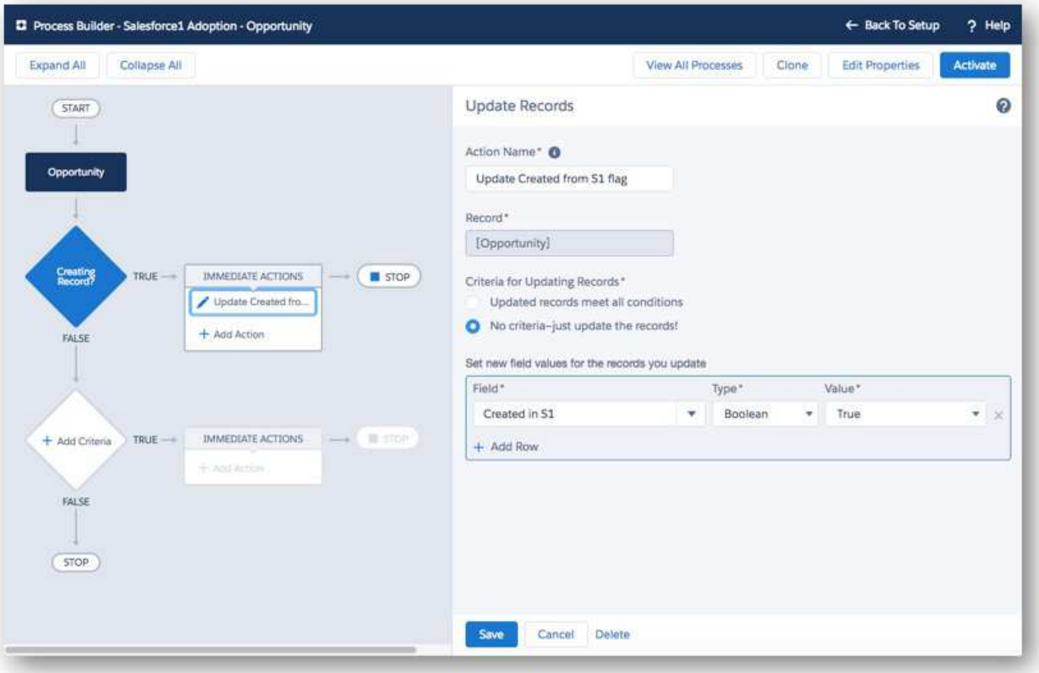
Build Formula

```
AND(ISNEW(),$User.UIThemeDisplayed == "Theme4t")
```

Advanced
Do you want to execute the actions only when specified changes are made to the record?
□ Yes

STEP 4:

Click 'Add Action' under 'Immediate Actions' and choose 'Action Type' as 'Update Records'. Provide the Action Name and choose 'Select the Opportunity record that started your process' under 'Record Type'. Make sure you choose 'No Criteria-just update the records!' under 'Criteria for Updating Records' and assign a value of 'True' to 'Created in S1' field as shown below



Update Records

Action Name* Update Created from S1 flag

Record* [Opportunity]

Criteria for Updating Records*
○ Updated records meet all conditions
● No criteria–just update the records!

Set new field values for the records you update

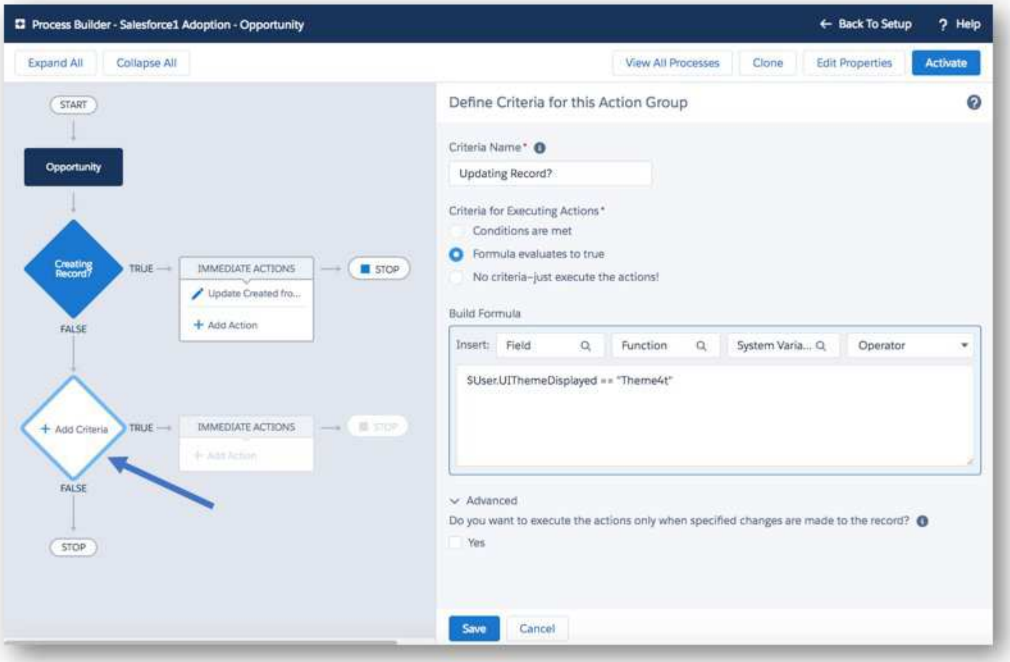
Field*	Type*	Value*
Created in S1	Boolean	True

+ Add Row

STEP 5:

Click on 'Add Criteria' and provide the Criteria Name as 'Updating Record?'. Make sure to choose 'Formula evaluates to true' radio button under 'Criteria for Executing Actions' and then use the formula in the formula editor as provided below

```
$User.UIThemeDisplayed == "Theme4t"
```



Define Criteria for this Action Group

Criteria Name* Updating Record?

Criteria for Executing Actions*

○ Conditions are met

● Formula evaluates to true

○ No criteria—just execute the actions!

Build Formula

```
$User.UIThemeDisplayed == "Theme4t"
```

Advanced

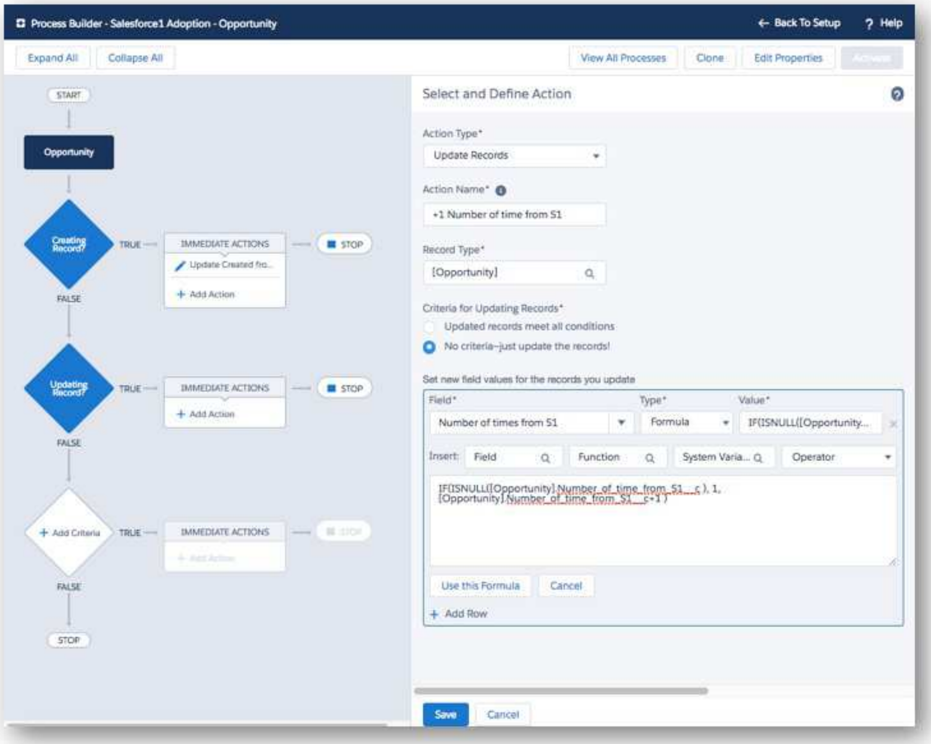
Do you want to execute the actions only when specified changes are made to the record?

□ Yes

STEP 6:

Click 'Add Action' under 'Immediate Actions' and choose 'Action Type' as 'Update Records'. Provide the Action Name and choose 'Select the Opportunity record that started your process' under 'Record Type'. Make sure you choose 'No Criteria-just update the records!' under 'Criteria for Updating Records'. Then assign the formula to the 'Number of times from S1' field as below

```
IF(ISNULL([Opportunity].Number_of_time_from_S1__c ),

    1,

    [Opportunity].Number_of_time_from_S1__c+1

)
```



Select and Define Action

Action Type*: Update Records

Action Name*: +1 Number of time from S1

Record Type*: [Opportunity]

Criteria for Updating Records*

○ Updated records meet all conditions

● No criteria–just update the records!

Set new field values for the records you update

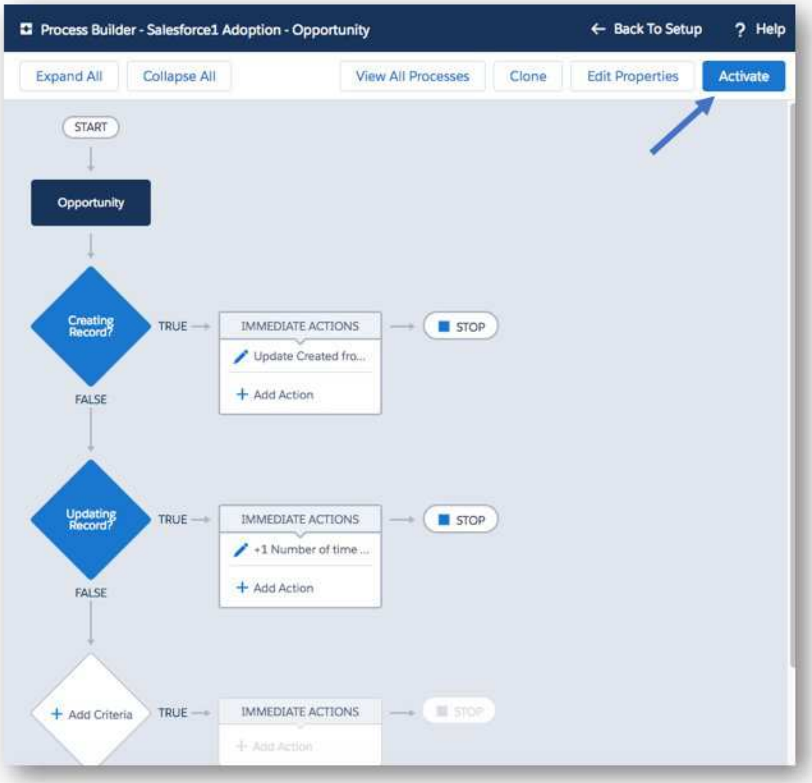
Field*	Type*	Value*
Number of times from S1	Formula	IF(ISNULL([Opportunity...

Insert: Field | Function | System Varia... | Operator

```
IF(ISNULL([Opportunity].Number_of_time_from_S1__c ), 1,
[Opportunity].Number_of_time_from_S1__c+1 )
```

STEP 7:

Finally, activate the process by choosing the 'Activate' button. Your final process should look like shown below



Now, you have created a process that gets triggered whenever an opportunity is created or updated irrespective of the user interface used by the user. However, the 'Created in S1' and 'Number of time from S1' are only updated if the opportunity was created or updated using Salesforce1. Unlike the 'pre-defined fields' trick in custom actions, this process builder process works even if the user creates or updates the opportunity from standard action buttons in Salesforce1.

Instead of updating custom fields in each object, you can create a new custom object and create records in it with the record IDs of the record that started the process. This way, you only need one custom object that holds information about all records that are created and updated using Salesforce1. Then you can perform some nifty analysis but usually that will require some custom coding and hence beyond the scope of this document

All good, but how about record views in Salesforce1?

Sure, the techniques discussed above provide a great way to measure adoption and engagement for record creates and updates but one of the main benefits of Salesforce1 is to provide those micro-moments for the user to quickly view critical business data, while on the move. To monitor the usage for such requirements, you will need more complex event logging and it can be achieved through 'Event Monitoring'. You can find more information about Event monitoring here. You can also find a trailhead module on Event Monitoring here. Since Event Monitoring is a topic in itself, we will cover how to use it to monitor Salesforce1 adoption in a part 2 of this document

Additional Resources

- Getting Started with Salesforce1
- Trailhead: Salesforce1 Mobile Basics
- Trailhead: Salesforce1 Rollout
- Salesforce Accelerators
- Salesforce1 Mobile App Adoption Accelerator

This document is maintained by the Mobile Success Specialists at Salesforce. For comments, questions and edits, please reach out to KK Ramamoorthy through Salesforce Success Community (https://success.salesforce.com/profile?u=0053000000AmNpiAAF) or post a message to the Salesforce1 Mobile App Success chatter group (https://success.salesforce.com/_ui/core/chatter/groups/GroupProfilePage?g=0F9300000001qep)