



360SMS + Marketing Cloud

360SMS is easily integrated with Marketing Cloud (MC) because of MC's awesome ability to create any Salesforce record which includes the 360SMS "Scheduled SMS" record which is how all programmatic SMS's are created to then send an outgoing SMS. MC has full access to the 360SMS phone numbers and other fields required.

Most customers love the MC Journey builder but do not like MC's SMS capabilities primarily because it's native SMS does not appear against Lead and Contact records. So this is easily solved by simply inserting the Scheduled SMS into their Journey. Below are several screen captures.

Additionally, 360SMS makes it even easier by offering a Native Marketing Cloud plug-in that allows a more simply approach for non-programmers where you choose the various values need to send an SMS such as the From Number and the Template or you can create messages on the fly and even still use Merge Tags from your Salesforce data.

The screenshot displays the Marketing Cloud Journey Builder interface. On the left, the 'ENTRY SOURCES' section lists various data sources, with '360 SMS APP' highlighted under the 'ACTIVITIES' section. The main canvas shows a journey flow starting with an 'ENTRY EVENT' (New Journey - August 23 2022 at 7.32 PM), followed by a '360 SMS APP' activity, a '1 day' wait, and an 'Exit on day 1' activity. A pop-up window provides configuration details for the '360 SMS APP' activity, including fields for 'From' (447897027482), 'To' (MOBILE), 'Channel' (SMS / MMS), 'Templates' (--None--), 'Insert Field' (-- None --), and 'Create Message Body'.

Figure 1 - Use the MC Plug-In for quick SMS or the full Salesforce Data node shown in Figure 2 thru 5 for maximum flexibility

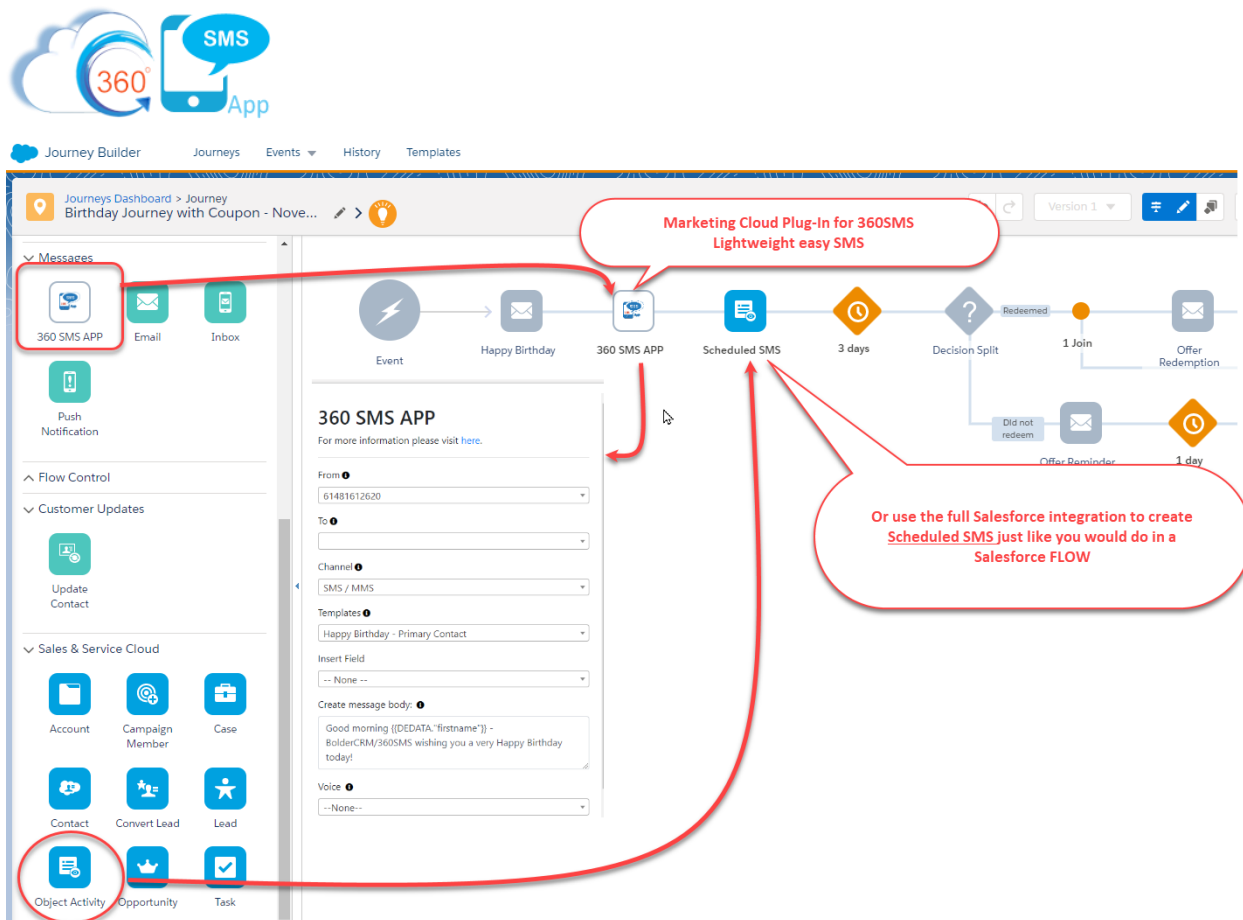


Figure 2 - Or use the full Salesforce integration of MC to create the Scheduled_SMS record just like you would do in a Salesforce Flow

With the Salesforce “Object Activity” you just pick your Salesforce object and create the Scheduled_SMS record setting the values of the key fields such as the PhoneAPI (location of the customers phone value), Sender_Number (outbound number), Channel (WhatsApp, SMS, etc...) and the Template or iText.

Object Activity

SELECT OBJECT SELECT ACTION MAP FIELDS SUMMARY

Review the Object data that was just configured.

ACTIVITY NAME: Scheduled SMS

SALESFORCE OBJECT: Scheduled SMS

CONFIGURE ACTIVITY: Create New

SALES & SERVICE CLOUD	MARKETING CLOUD
Channel	SMS
Contact	Assign to the Person in the Journey
Phone Api	MobilePhone
Related Object Id	Contacts ID
Scheduled Sms Name	Contacts ID
SMS Template	Happy Birthday - Primary Contact
Sender Number	61481612620

Copy to Clipboard

Figure 3 - Marketing Cloud lets you create the Scheduled_SMS with all the parameters you would within Salesforce



Scheduled SMS Record fields explained

For a full and deeper explanation of SMS creation refer to:

<https://boldercrm.com/download.php?f=360-SMS-Sending-SMS-through-Process-Builder.pdf>

Whether creating an SMS in Salesforce or Marketing Cloud the methods are the same. We simply create a **Scheduled_SMS** record. Setting the field **Scheduled_Time** blank will send the SMS immediately while setting the Date/Time will actually schedule it and the 360 Scheduler runs every 1 minute to look for SMS's to send.

First off, **Create a Record** to for the **Scheduled SMS** object then set these fields either via a Process Builder or Flow. These are listed in order of importance, the first 5 being the minimum requirement.

We show a Process Builder approach in Figure 4 and a Flow approach in Figure 5

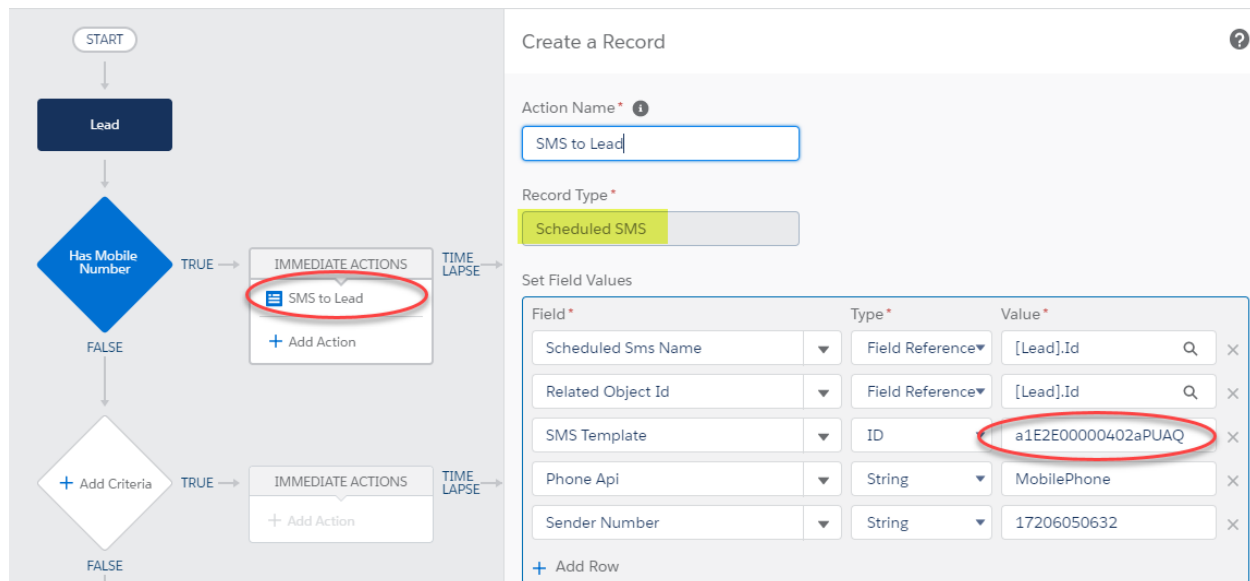


Figure 4 – The most important 5 fields required for sending an SMS – note you can also set the QUESTION field which is the first question of the 360 SMS Survey object (aka iText) which will then trigger an automatic Question/Answer survey. And you can set the Scheduled_Time, Attachments. Look to the flow example for more possibilities and we strongly recommend Flows over Process Builders.



Edit Create Records

Create Salesforce records using values from the flow.

* Label
Scheduled SMS

* API Name
createScheduledSMS

Create a Record of This Object
* Object
Scheduled SMS

Set Field Values for the Scheduled SMS

Field	Value
Name	Task from recTask > Name ID X
OwnerId	Task from recTask > Assigned To ID X
tdc_tsw__Contact__c	fxContactId X
tdc_tsw__Lead__c	fxLeadId X
tdc_tsw__Next_SMS_Send_Date__c	fxScheduledTime X
tdc_tsw__Phone_Api__c	MobilePhone
tdc_tsw__Question__c	vSurveyQuestionId X
tdc_tsw__Related_Object_Id__c	Task from recTask > Name ID X
tdc_tsw__SMS_Template__c	vTemplateId X
tdc_tsw__Scheduled_Time__c	fxScheduledTime X
tdc_tsw__Sender_Number__c	vStickySender X

Cancel Done

Figure 5 - Same basic concept for flows but of course a Flow has considerably more power with it's powerful variables and and additional lookup features.

1. **Scheduled SMS Name:** Must be the record ID field of the triggered object/record and this Record ID's object must match the object of the SMS Template or Survey Question, i.e. if triggering from a Lead the Template's or Survey's object must be LEAD.



2. **Related Object Id:** Set the Related Object Id to the record ID of the record where the SMS History should show up and even more importantly this governs the Object/Record that an Incoming SMS response will come into.
 - a. This field has nothing to do with the Template/Survey object, it only governs where the SMS History appears after it's sent. Note that if using the **Scheduled Time** field to schedule the SMS instead of sending it immediately, this field DOES NOT make it so that the Scheduled SMS shows in the related list of the record. You must explicitly also set that objects ID value as well, e.g. `Scheduled_SMS.ContactId`
 - b. Even though it seems redundant this adds value especially for triggers like Event Reminders where you set the Scheduled SMS Name = `Event.Id` while setting the Related Object Id = `Event`. **Whold** because the Salesforce Event object does not allow related lists like the SMS History. Other use cases would be triggering from an Opportunity but wanting the SMS to appear in the Contact so that the reply comes back to the Contact record. However, in this use case you are better off to keep the Related Object Id = `Opportunity.Id` and use the techniques documented in the section below named "[Error! Reference source not found.](#)"
3. **Phone Api:** Supply the phone field **API Name** to send the message TO. This can be a string such as "MobilePhone" or "Mobile_Phone__c" (custom field) or the actual phone value referenced from the record, i.e. `Lead.MobileNumber` but it can also be pulled from the Incoming SMS `SENDER_NUMBER`, set with a formula, or for SMS to employees you might use the `User.MobileNumber`.
 - a. Especially when using the **Scheduled Time** field to schedule the SMS, we strongly recommend using the Field Name for the PhoneAPI value rather than the Field Reference because the App will dynamically get the phone value at the time it fires, whereas if you explicitly set the value via Field Reference but the MobilePhone field is blank at the time, your SMS won't fire even if you set the value. Or if the MobilePhone field changes in between the time that the message is scheduled versus when it Sends, then the old value would be used.
4. **Sender Number:** This is the number that you are sending FROM. This can also be a referenced field such as `Lead.Owner.Phone` (so as to send from different sales people) or a formula such as different geographies using different numbers, such as `Lead.Sticky_Sender__c` (more below in the section titled: [Error! Reference source not found.](#)). If you only have one outbound number in your org, the field is optional and need not be supplied.
 - a. The sender number must always be in format `CountryCode+Number` with absolutely no formatting or special characters, e.g. `17206050632`. When dynamically setting the SenderNumber such as from `Contact.Owner.Phone` or from the Incoming `SMS_History.To_Number` use this valuable formula below to strip out all the characters.



```
/******  
Here's an example of getting the Sender Number from the  
Contact.Owner.Phone field (USER object).  
*****/  
'1' &  
SUBSTITUTE(  
SUBSTITUTE(  
SUBSTITUTE(  
SUBSTITUTE(  
SUBSTITUTE(  
SUBSTITUTE( Owner.Phone ,  
"(", "" ),  
")", "" ),  
" ", "" ),  
"-", "" ),  
"+", "" ),  
".", "" )
```

- a. We recommend using the special 360SMS custom field named **Sticky_Sender__c** for the Sender Number value. This is a custom formula field that the **Send SMS** buttons and **Conversation View** recognize to override the users Default SMS Number with the formula value. The idea is that the number “Sticks” to the record, so the customer always receives SMS from the same number.
 - i. The formula field allows one to define their own business logic for record-based sender numbers so the customer always receives SMS from the same number.
 - ii. This is useful for geography-based solutions (USA vs. UK numbers) or when each Record.Owner has their own sender number.
 - iii. Area Code matching – many orgs want the SMS Number to match the customers Area Code so it looks like a local number. They use an Area Code formula plus a sticky_sender formula that matches area code to a list of matching SMS Numbers.
 - iv. Sticky_Sender is especially useful for Batch Texting so a marketing user can batch text many customers across many numbers.

Create Message

Send To (Selected Contact: 3)

John Smith
Peyton Manning
Joe Montana

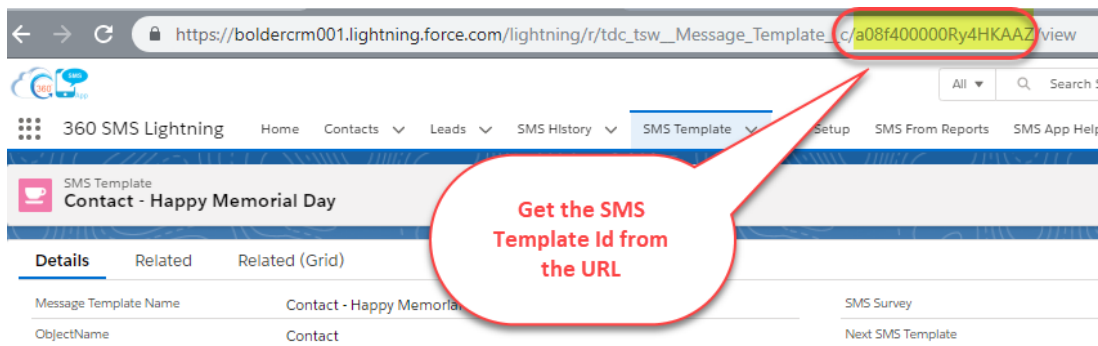
Sender Phone

Sticky Sender

Send To

Mobile Phone

5. **The Message Text:** There are three ways to set the body of the message that will be sent
 - a. **SMS Template:** Set the ID of the Template to be used. This can be obtained from the URL of the template.
 - i. You may also use a reference field such as Contact.SMS_Template (if you’ve created a SMS Template Id on your Contact) – see the section *Create a Master Send SMS Handler*



- b. **[iText/Survey] Question:** As an alternative to setting an SMS Template, you may set the **Question** field. The QuestionId is usually the first question of a 360 SMS **Survey**. This then triggers the survey question and responses are automatically triggered from there on. We highly recommend utilizing the Survey object instead of Templates because most outbound SMS will likely generate a response which can be automatically handled by the 360 SMS Survey/ChatBot tool.
- i. It is an important, purposeful and brilliant design element that we pass in a **Question Id** of the survey rather than the Survey Id. Even though 99% of the time you will use the QuestionId of the first question we have seen numerous use cases where it is useful to start a customer in the middle question of some particular survey such as when the survey is serving two purposes and the top of the survey asks questions that a particular customer has already answered or we have the information but we don't want to create a whole new survey so we strategically send them a different question in the survey. **BRILLIANT!!!**

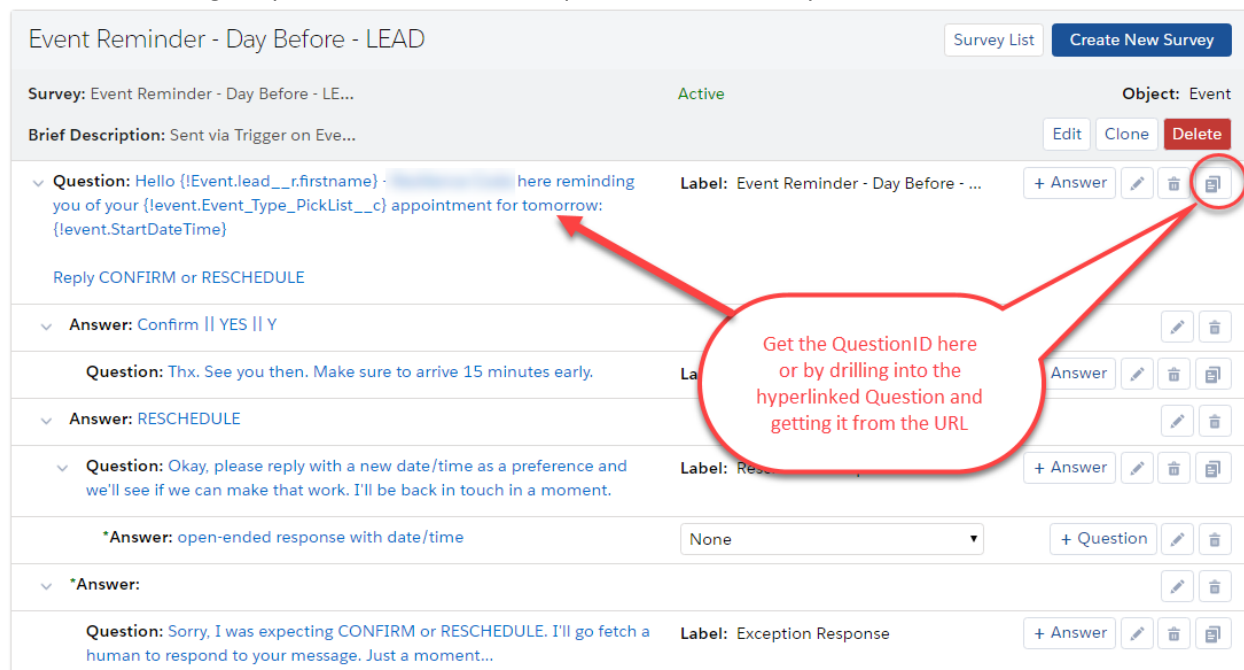


Figure 6 - Obtain the question Id from the first question of a Survey



- c. **Message Text:** You may create your own message text with a formula or field reference although we rarely recommend this as then you have no way to modify your message text other than modifying your Process Builder. Still it's a powerful and distinct option.
- 6. **Owner:** As of version 1.196.54 and above you can set the Scheduled_SMS.Owner and the resulting SMS_History.Owner will carry forward. **THIS IS SUPER IMPORTANT** because ownership of the Outgoing message controls who will receive the notification for the incoming reply. See below for a more detailed discussion of ownership implications and techniques.

This document has a deeper technical discussion on [SMS History Ownership](#)

- 7. **Scheduled Time:** The scheduled time field can left blank in which case it's IMMEDIATELY sent but **WE DO NOT RECOMMEND EVER SENDING IT IMMEDIATELY**, please consider adding at least 1 or 2 minutes to NOW() because it buys you just that extra 1 or 2 minutes to delete the Scheduled SMS if you made a mistake and we believe having everything Scheduled creates a more consistent SF User experience and you can troubleshoot 1000% easier.
 - a. If using this feature, you MUST also set the Lookup field of the object whose Related List it should appear under, such as Contact or Lead. Setting the **Related Object ID** field is not the same as setting ContactId or LeadId, that field has a different purpose noted below.
 - b. Usually you will use a formula such as these common ones below. Remember, Salesforce date math is in DAYS so you might Google your desired syntax for minutes and hours.
 - i. NOW() + 1 /* Same time as today + 1 day, i.e. tomorrow */
 - ii. NOW() + (1/24) /* 1 hour from now */
 - iii. NOW() + (10/60/24) /* 10 minutes from now */
 - iv. Specifying a specific time of day can be tricky because Salesforce stores Time values in Greenwich Mean Time (GMT) which you must compensate for. Below are two common example formulas that compute a specific Date plus Time.
 - v. Ask about the 360 SMS [Time Zone Anything™](#) product for dynamic Time Zone and Daylight Savings Time handling.

```
/* Schedule Birthday wish for this yr 16:00 GMT = 8:00am Mountain Time */
/* DateTime has to be in format: YYYY-MM-DD HH:MM:SS */
DATETIMEVALUE( TEXT(YEAR(TODAY())) & "-" &
                TEXT(MONTH([Contact].Birthdate)) & "-" &
                TEXT(DAY([Contact].Birthdate)) & " 16:00:00"
                )
```

- c. More on Scheduled SMS in these two documents:
 - i. [Triggered Scheduled SMS w/ Process Builder](#)



ii. Triggered Scheduled SMS – Event Reminders

```

/*****
SF date math is tricky when wanting a specific time of day such as 8:00am EST. SF wants the
field in format: YYYY-MM-DD HH:MM:SS, plus we have to compensate that SF returns everything in
Greenwich Mean Time so you have to adjust for your time zone.

1. Convert ActivityDateTime (shown in picklist as label TIME) into a text string
2. We just need the YYYY-MM-DD portion of it, so take left 10 characters
3. Now add the time component back adjusting for time zone Eastern Time = GMT minus 5 hrs.
4. Lastly convert it back into a datetimevalue
*****/

DATETIMEVALUE(LEFT(TEXT(ActivityDateTime),10) & " 13:00:00")

/* 13:00PM GMT (1:00 PM in UK) = 8:00am EST (GMT -5)*/

```

8. **Attachment/Document Id:** You may set an optional Document.Id value of a picture or other supported document stored in the Salesforce Document object. Refer to the dedicated section below titled: [Error! Reference source not found.](#)
9. **Channel:** You may specify a channel such as **SMS** or **WhatsApp**. Not using the parameter or setting it to null will send the message as SMS.



About the Author

Steve Roch, CEO of Bolder CRM is an SMS Industry expert having worked or consulted with the top three SMS Apps on the Salesforce AppExchange and also having built the popular Salesforce app [ActionGrid™](#), acquired by Conga in April-2016 and now renamed [Conga Grid™](#). Bolder CRM is the exclusive distributor of 360 SMS in the United States, Canada and the United Kingdom.

Learn more about Steve and Bolder CRM at <https://boldercrm.com/360SMS> and <https://www.linkedin.com/in/steveroch/>

Call/Text: [+1 720.605.0632](tel:+17206050632) or [+44 1234 480 564](tel:+441234480564) or [+61 488 845 944](tel:+61488845944)

WhatsApp: [+1 303.800.3258](tel:+13038003258) or [click here](#) initiate a WhatsApp convo with Steve

Email: steve@boldercrm.com