CM Telecom SMS API

Salesforce Manual

1 Setup 3

1.1 Setting up My Domain 3

1.2 Installation 3

2 Custom Settings 4

3 Object SMS Senders 4

4 Lightning Component 5

5 Process Builder 6

6 Flows 7

7 WebService 8

7.1 Setting up outbound 8

7.2 Setting up inbound 9

8 Extend Component 9

9 Update 2.0 Outbound Messages 11

# Setup

## Setting up My Domain

For Custom Lightning apps the Salesforce ORG needs My Domain setup.

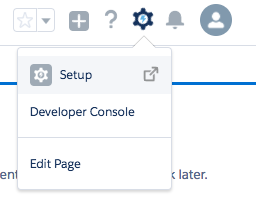


Figure 1 – Setup (top right)

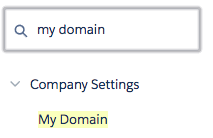


Figure 2 - My Domain in Setup (left sidebar)

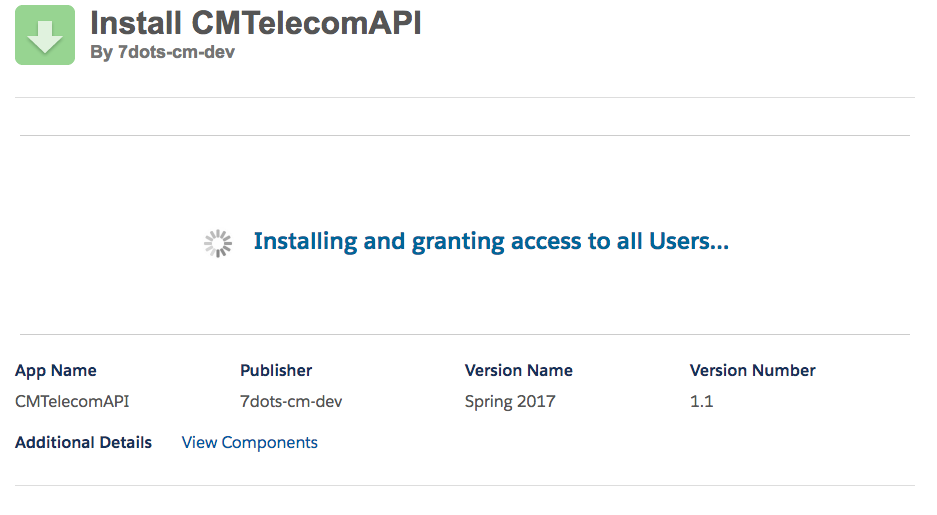
In the My Domain view choose the domain name: **<YOUR\_DOMAIN>-dev-ed.my.salesforce.com**

Once chosen, log In and deploy the domain to all users.

## Installation

After setting up My Domain, use the link below to install the package. Don’t forget to login to the targeted Salesforce ORG before clicking the link.

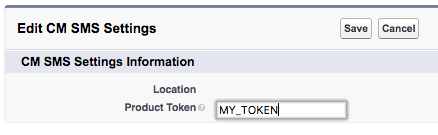
<https://login.salesforce.com/packaging/installPackage.apexp?p0=04t0Y000001Zdcm>



# Custom Settings

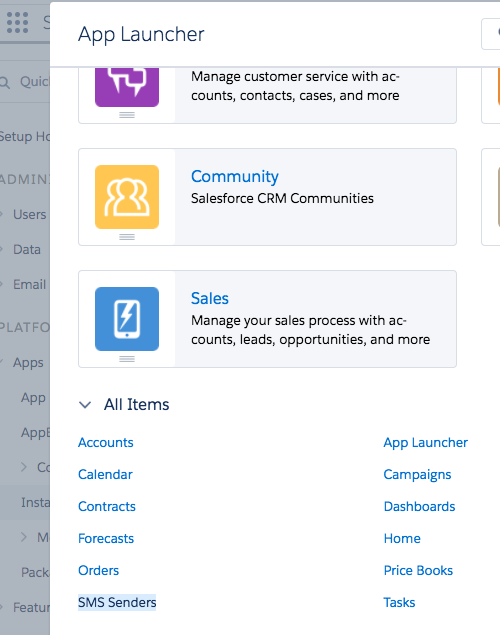
The installation was a success, now it’s time to configure the package. Before using the components the API needs to know the product token. You can find the product token at <https://gateway.cmtelecom.com> under the HTTPS API section. Write it down.

In Salesforce go to setup once again and search for ‘custom settings’, open the page. You’ll be able to see a new Custom Setting entry called ‘CM SMS Setup, click on ‘manage’ next to the name. A new view opens where you can define all needed values, for this package however, we need one. Click on new at the top. Fill in the product token you wrote down and save the entry.



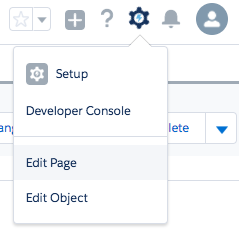
# Object SMS Senders

The package contains a custom object named ‘SMS Senders’. In this object you can define multiple users with their phone numbers. This list of users is accessible in the lightning component for quick messaging. The object is found by clicking on the 3 by 3 dots at the top left (under the Salesfoce logo) and choosing ‘SMS Senders’.

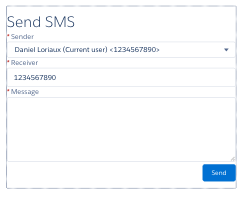


# Lightning Component

The SMS component is setup generic so that it works on any object. You can add the component by clicking on the setup wheel and choosing ‘Edit page’. (Or ‘Edit Object’ and adding a new Lightning Record Page).



In the App Builder view you can drag the custom component ‘CMTelecomComponent’ from the leftbar to the canvas. Once placed, you can define two values for the view. The phone and origin field. The phone field is used to target the API name of the field, so that it knows which field to read and populate the ‘receivers’ field with. You can find out the API names in the object settings. The origin field is used for each event that is created per message, to indicate which process started the message.





Once added to the page, it is ready to be used.

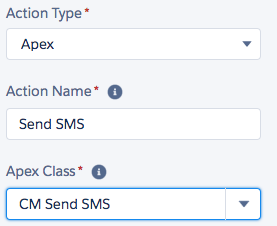
# Process Builder

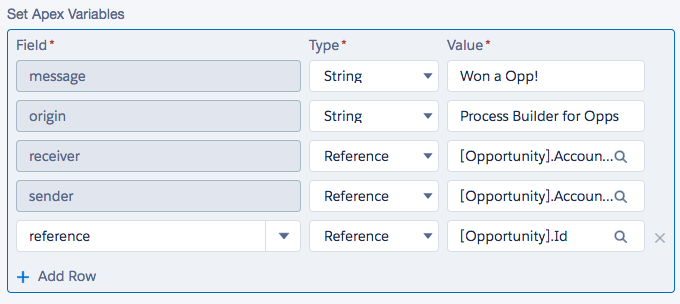
You’re able to automate SMS with Process Builder using the API that came with the package. First step is to create a process for any object you like. Go to setup and search for ‘Process Builder’. Create or edit a process.

As immediate action, select type ‘Apex’ and the class ‘CM Send SMS’. This will show some variables that needs to be setup before saving.

* Message: the message to be send
* Origin: where the message came from
* Receiver: the receiver of the message
* Sender: the sender of the message
* Reference: (optional) the record id that initializes the process, this is needed to save events related to that record.

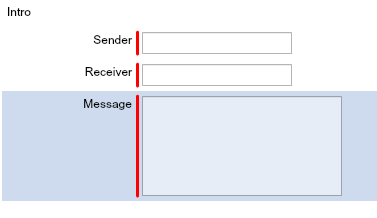
Once saved, the message will be send when the required criteria is met.

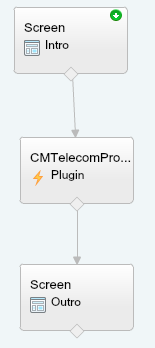


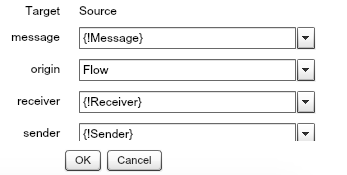


# Flows

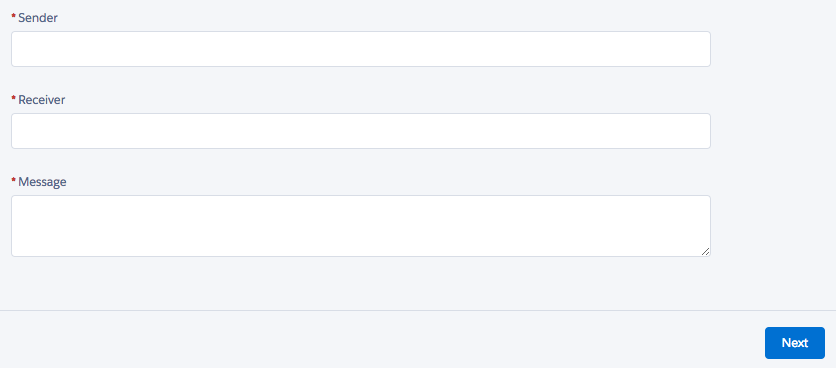
With flows you are able to send messages with the API that came with the package. Just as Process Builder you can select the apex class (in flows it’s named ‘CMTelecomProcessPlugin’) and feed values to it that were defined earlier in the flow. You can find Flows by going to Setup and search for ‘Flows’. Create or edit a new flow and add the plugin to the canvas. Select the items to map all the needed variables.







Once created you’re able to add the flow to any page, just like you would with a lightning component. Once added the flow is rendered and ready to use for the end user.



# WebService

To be able to receive webhook messages from the CM portal, it is necessary to enable public sites and give public rights to the Gateway API class. Go to Setup and search for ‘Sites’. Create a site domain if prompted. Once created click on new site at the bottom. For the homepage you can select ‘UnderConstruction’ as the site will on enable the web service. Fill in all required fields and click on save.

Once created click on ‘Public Access Settings’, next ‘Apex Class Access’. Enable the following classes and save:

* CMSMSAPI.CMTelecomWebServiceOutboundMsg
* CMSMSAPI.CMTelecomWebServiceInboundMsg

## Setting up outbound

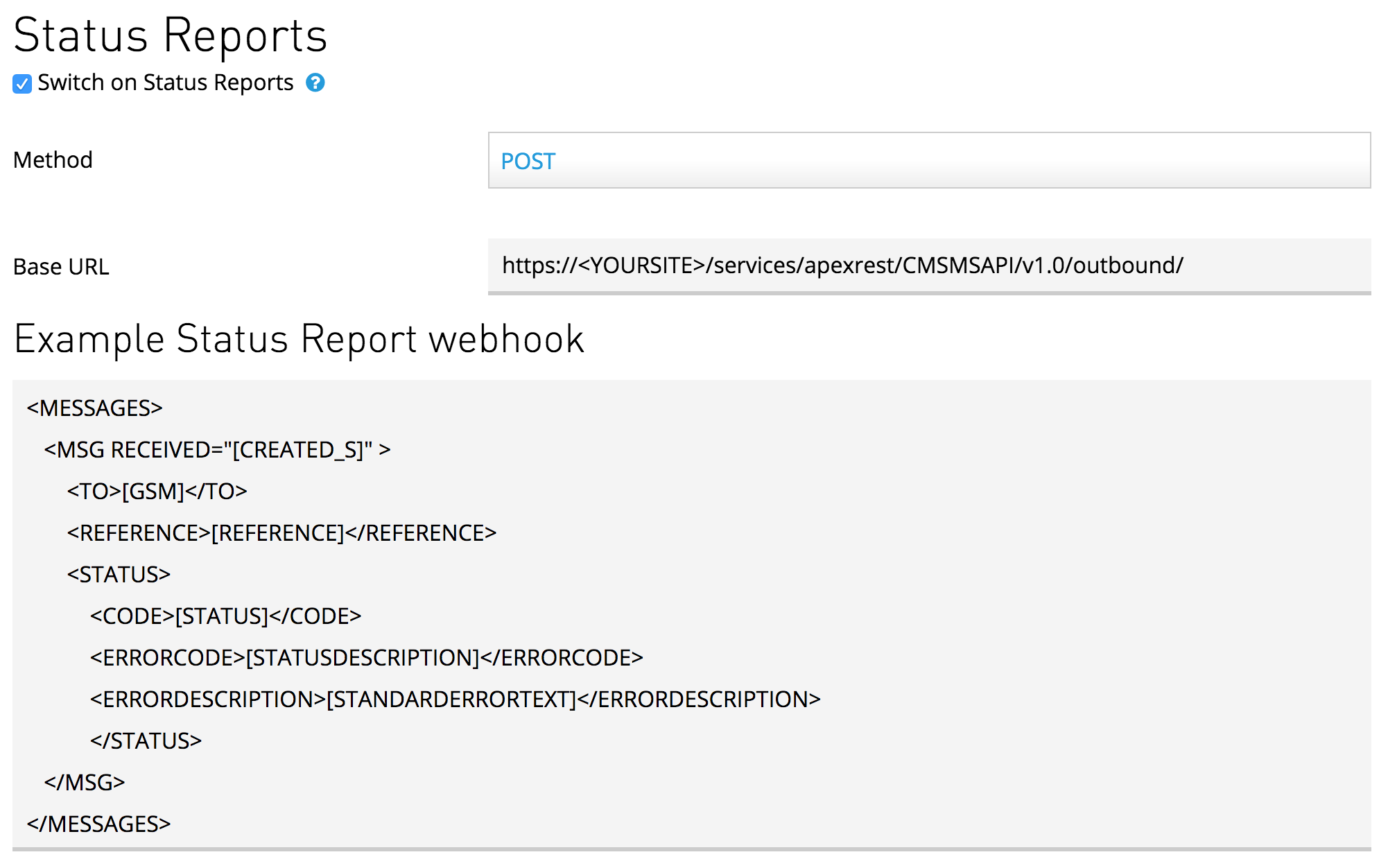
Login to the CM website and setup the webservice at the Messaging Gateway page (<https://gateway.cmtelecom.com>) and Status Report section. Your Base Url will be:

**<Your\_Site>/services/apexrest/CMSMSAPI/v1.0/outbound/**

**for example:**

https://company-uat-developer-edition.eu11.force.com/services/apexrest/CMSMSAPI/v1.0/outbound/

And make sure that the properties: **REFERENCE** and **STATUS** are send with the webhook as well. Use **POST** as method. See image below for the setup.



## Setting up inbound

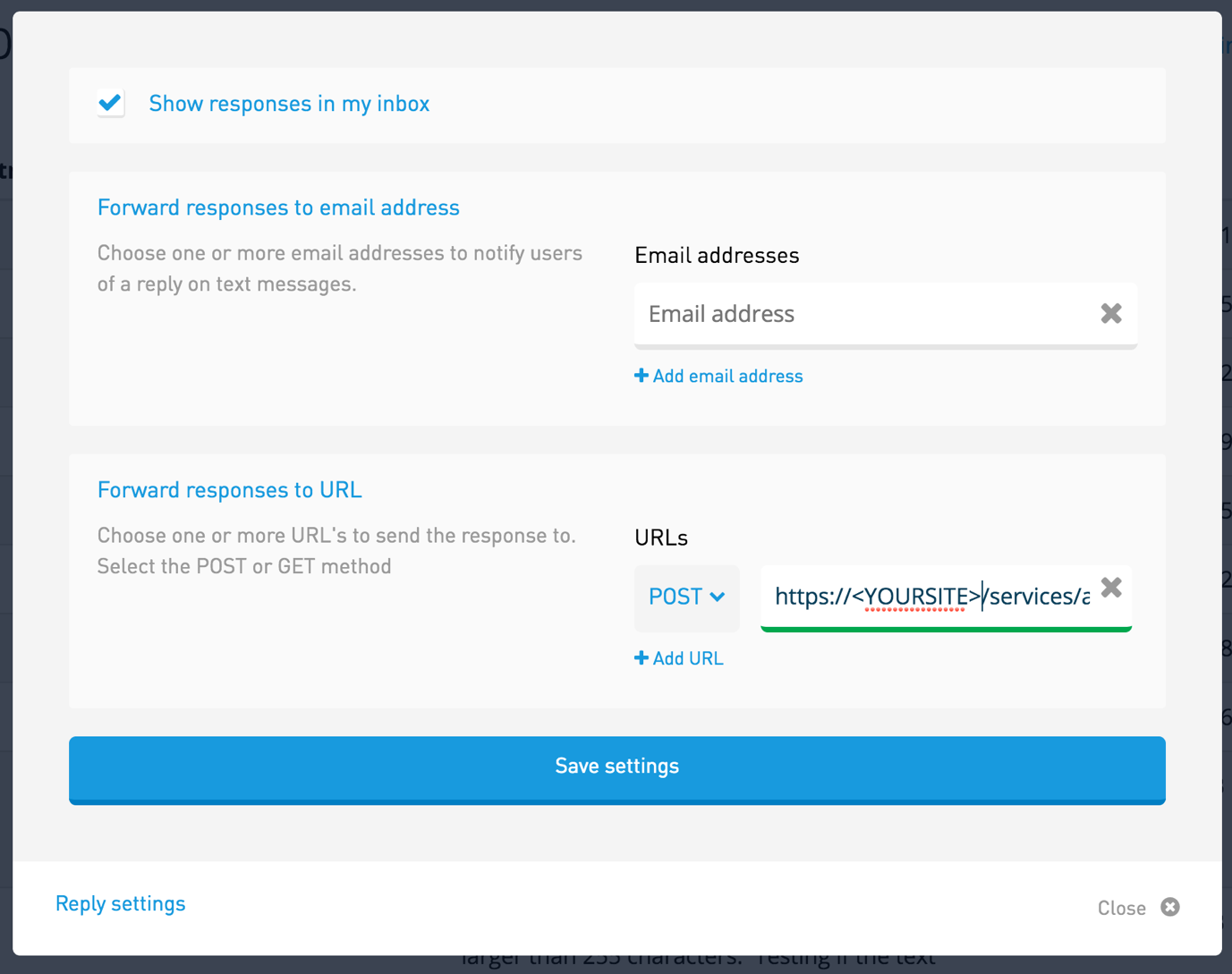
Login to the CM website and go to the “Received messages” page (<https://received.cmtelecom.com>). Once there, click on the top right link called “Reply Settings” next to the cogwheel. A popup will open. In section “Forward responses to URL” enter your Salesforce URL.

**<Your\_Site>/services/apexrest/CMSMSAPI/v1.0/inbound/**

**for example:**

https://company-uat-developer-edition.eu11.force.com/services/apexrest/CMSMSAPI/v1.0/inbound/

**Use POST as method**. See image below for the setup.



# Extend Component

To extend the component and feed it with dynamic values you only need to create a new Lightning Component with the attribute “extends” matching the component “CMSMSAPI:CMTelecomComponent”. See the below example for a override created for a quick action.

<aura:component extends="CMSMSAPI:CMTelecomComponent">

<aura:set attribute="field" value="CMSMSAPI\_\_Phone\_\_c" />

<aura:set attribute="origin" value="Send SMS Quick Action" />

</aura:component>

# Update 2.0 Outbound Messages

As of update 2.0 it is now possible to receive outbound messages with the CMTelecomAPI. Some crucial changes:

* Added webservice /v1.0*/outbound/* with POST support.
* Added webservice v1.0/inbound/ with POST support.
* Deprecated webservice /v1.0/gateway/. It only supports GET, to use POST use /v1.0*/outbound/* instead.
* All events created by the API conform the naming format: <sender number or name>: <message> instead of the origin
* Origin is added to Event description