



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Gplus Adapter for ServiceNow Administrator's Guide

Click-to-call feature

---

## Contents

- [1 Customize the outbound-call scenario](#)
- [2 Enable the click-to-call feature on reference fields](#)

---

Click-to-call makes outbound calls easier.

The click-to-call feature enables to automatically dial a call when clicking a phone number in ServiceNow.

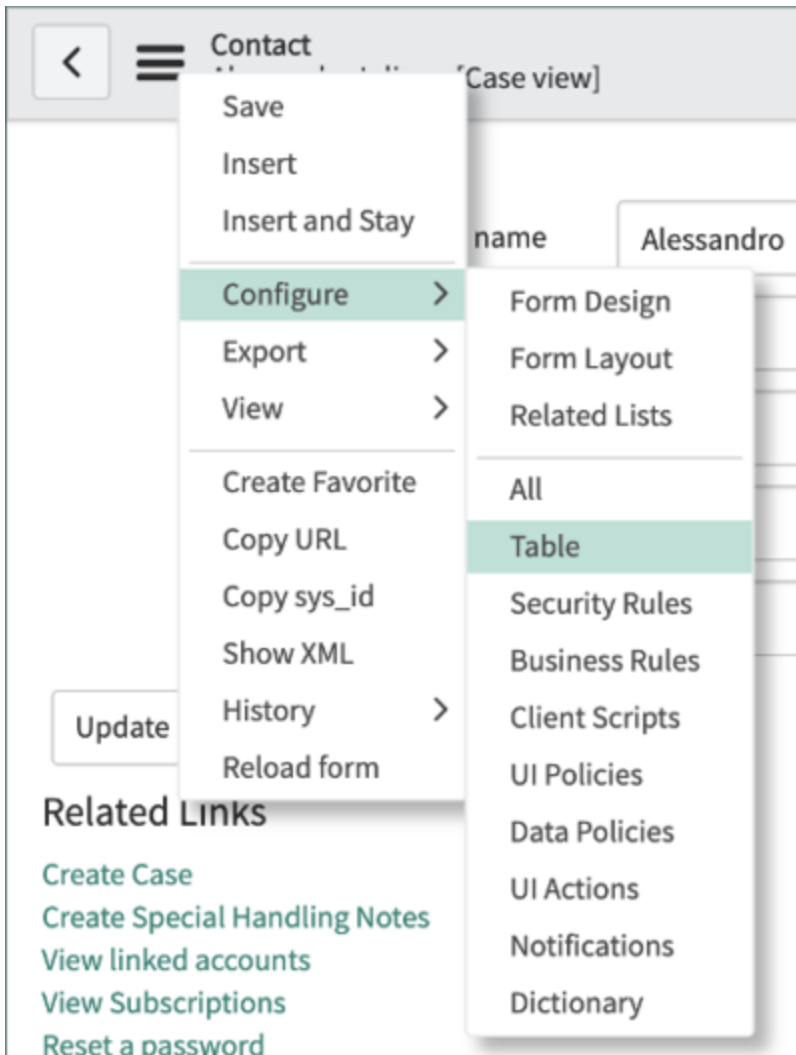
## Customize the outbound-call scenario

Unlike screen pop customization, to customize the outbound-call scenario you must handle an additional file:

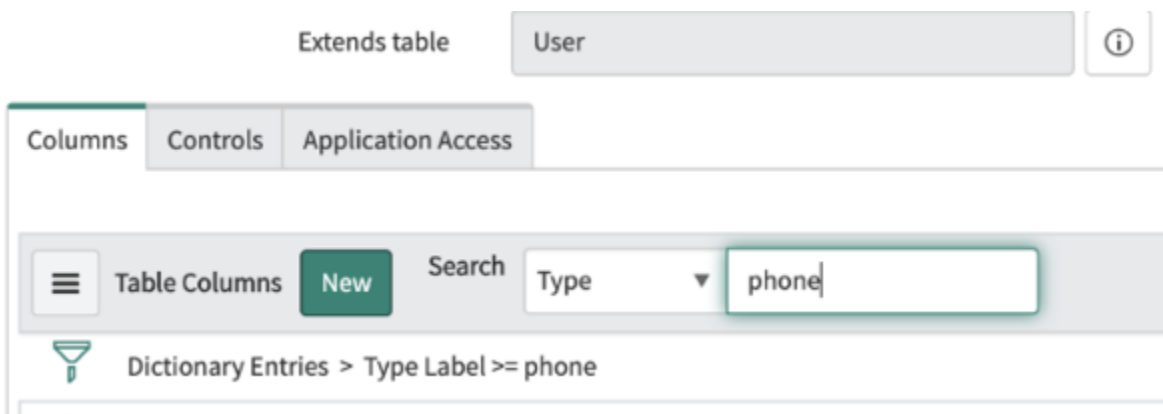
- **x\_sofin\_gplus\_adap\_clicktocall.**

Follow this procedure to configure the click-to-call feature inside ServiceNow:

- Open the **Contact form** and then the **Contact menu**.
- Select **Configure** and click **Table**.



- Search for the phone fields as shown in the image below.



- Select one field among the proposed **phone-number fields**.

	<u>Mobile phone</u>	<u>Phone Number</u>	(empty)
	<u>Home phone</u>	<u>Phone Number</u>	(empty)
	<u>Business phone</u>	<u>Phone Number</u>	(empty)

- Go to **Default Value** and add the following attribute in the related box:

field\_decoration= x\_sofin\_gplus\_adap\_clicktocall

Attributes

### Important

If the Attributes box is not visible, select **Advanced view** as shown below.

Choice List Specification
Default Value

Displays a list of suggested values in a Choice list. In the Advanced view yo

Choice

### Related Links

[Show Table](#)

[Advanced view](#)

## Enable the click-to-call feature on reference fields

You now have to configure the Adapter click-to-call feature inside ServiceNow for the lookup/reference field. In fact, by default, ServiceNow provides some macros to enable the click-to-call feature on reference fields as well.

The following macros are available:

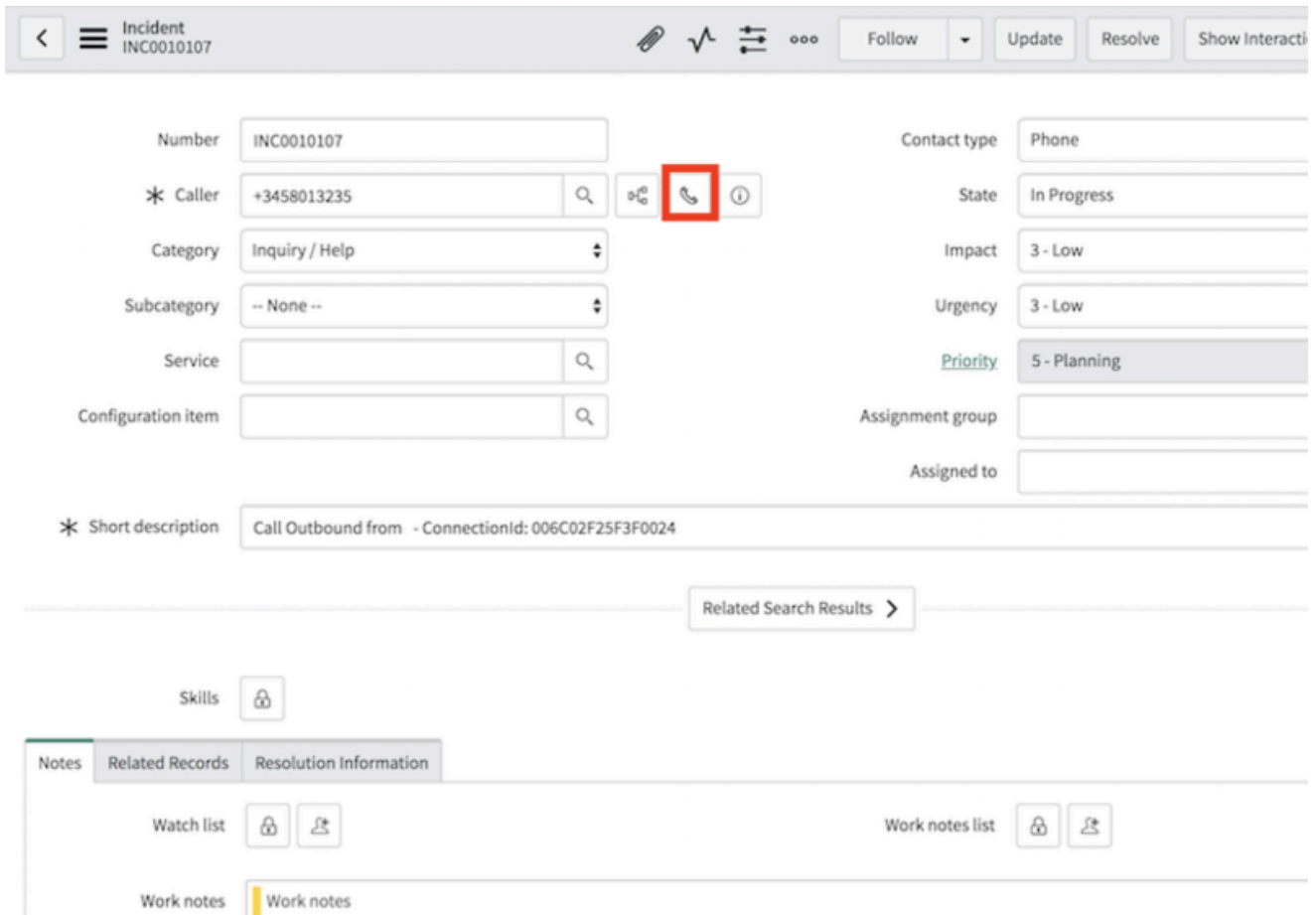
- **show\_phone**

- Requires the OpenFrame plugin activation.
- Configure the Unique Interface (UI) macro using the **ref\_contribution** parameter to display a phone icon next to the **sys\_user** reference field. The icon is only shown if the phone field in the sys\_user record is populated with a phone number.

- **show\_phone\_customer\_service**

- Requires Customer Service Management and OpenFrame plugins activation.
- Configure the Unique Inyerface (UI) macro using the **ref\_contribution** parameter to display a phone icon next to the Contact **customer\_account** reference field in the case form. The icon is only displayed if the phone field in the Contact customer\_account record is populated with a phone number.

It is possible to enable one of these macros on reference fields (such as **Caller** in an **Incident**, or **Contact** in a **Case**). They will display the phone icon next to the field, as shown below, only if the reference field has at least one telephone number.



---

## Important

If you are not able to see the phone icon next to a telephone-number field, make sure you own administrator's privileges.

Clicking the **phone icon**, a list of the Contact's available phone numbers is displayed.

To allow these macros to work with the Adapter and start a call with a click, you first need to edit them. Follow these steps:

- Search for the **show\_phone** macro and open it.
- Go to the row named **createInteractionData.opened\_for=p.attr('data-caller-id')**.
- Remove the **GlideAjax** section as shown below (adding comments).

```
/*
    var ga = new GlideAjax('global.CTIAjaxUtility');
    ga.addParam('sysparm_name', 'createInteractionWithPropertyCheck');
    ga.addParam('context', JSON.stringify(createInteractionData));
    ga.getXML(function(serverResponse) {
        var result =
serverResponse.responseXML.getElementsByTagName("result");
        var output = JSON.parse(result[0].getAttribute("data"));
        if(output.status == "success"){
            var interaction = {"entity": "interaction", "query"
:"sys_id="+output.fields.interactionSysId, "value":output.fields.number,
"label":"Interaction", "display":false};
            payload.data.data.push(interaction);
        }
        var context = {"payload": payload, "method" :
"openframe_communication" };
        jslog("context with interaction : "+ JSON.stringify(context));
        CustomEvent.fireAll("openframe_request", context);
    });
*/
```

- Replace the previous section with the following snippet:

```
var payload = {"type": "OUTGOING_CALL", "data" : data, "action":"makecall",
"number":p.attr('data-caller-phone'), "attributes": {"ACTIVITY_ID": p.attr('data-task-id'),
"ENTITY_TYPE": p.attr('data-task-table'), "CONTACT_ID":p.attr('data-caller-id')}};
var context = {"payload": payload, "method" : "openframe_communication" };
CustomEvent.fireAll("openframe_request", context);
```

- Click **Save**.

To conclude and enable this, you must associate the macro to **ref\_contributions**: open the Incident table, search for **Caller** and insert **ref\_contributions=show\_phone** or **ref\_contributions=show\_phone\_customer\_service** in the Attribute field.