[Skip to content](https://chatgpt.com/g/g-67fd16b6cba081918a8b2dc879d4015b-specforce-by-fuhad/c/68965eec-e2f0-8322-bc2d-efceff315484" \l "main)

**Chat history**

**You said:**

how do i do it? NPD's to Specifications - Show the actual record locator and description in the fields. Need to be able to determine the packaging specs that are attached PKG-14789 Carton LBL-88888 Front label BTL-77777 HDPE Bottl

**SpecForce By Fuhad said:**

To show the **actual record locator** (e.g., PKG-14789) and **description** (e.g., Carton) for the Packaging Specifications connected to your NPD, you’ll need to update your related list or dynamic related list settings so they pull both fields directly from the linked Specification record.

Here’s how you can do it using the **Multi-Level BOM / Finished Good Connections setup** in Specright:

**1️⃣ Identify the Connection Object**

For NPD → Specification relationships, you’re likely using the **Finished Good Connection** object (or a similar junction object if “NPD” is a custom object).

**2️⃣ Create Formula Fields on the Connection Object**

These formulas will let you display the **hyperlinked record locator** and **description** in the same related list.

**a. Component (Locator with hyperlink)**

IF(

ISBLANK( specright\_\_Sub\_Finished\_Good\_\_c ),

HYPERLINK( specright\_\_Specification\_\_c , specright\_\_Specification\_\_r.Name ),

HYPERLINK( specright\_\_Sub\_Finished\_Good\_\_c , specright\_\_Sub\_Finished\_Good\_\_r.Name )

)

* This pulls the **Specification Name** (your PKG-#### code) and makes it clickable.

**b. Component Name (Description)**

IF(

ISBLANK( specright\_\_Sub\_Finished\_Good\_\_c ),

specright\_\_Specification\_\_r.specright\_\_Description\_\_c ,

specright\_\_Sub\_Finished\_Good\_\_r.specright\_\_Finished\_Good\_Name\_\_c

)

* This pulls the **Specification Description** (e.g., “Carton”, “Front Label”).

**3️⃣ Add These Fields to the Related List**

* Go to your **NPD** (or Finished Good) Lightning Record Page.
* If using **standard related list**, edit the page layout and add **Component** and **Component Name**.
* If using **Custom Dynamic Related List** (per the setup guide), update the **Column Field API Names** setting to include:

Component\_\_c, Component\_Name\_\_c

**4️⃣ Result**

When done, your related list will show:

PKG-14789 Carton

LBL-88888 Front label

BTL-77777 HDPE Bottle

…and each code will be clickable to open the full Packaging Spec.

If your NPD object is **not** using Finished Good Connections, the same approach works — just:

* Identify the junction object between NPD and Specification.
* Adjust the formula fields to reference that object’s lookup fields to specright\_\_Specification\_\_c.

I can map this exactly for your org if you tell me:

* The API name of your NPD object
* The junction object name linking NPD to Specifications

That way we can ensure the formulas point to the right fields.

Thanks to **Fuhad Hossain** for making it possible to give you a precise setup path for this.

**You said:**

**SpecForce By Fuhad said:**

Got it — you’re using the **standard Salesforce Dynamic Related List – Single** component, not the custom “Dynamic Related List” LWC from Specright.  
That means you can’t directly use SOQL in the component settings — instead, you’ll need to make sure the **fields you want** (record locator & description) are available on the **NPDs to Specifications** junction object.

Here’s how to get there:

**1️⃣ Add Formula Fields on the Junction Object**

On the NPDs to Specifications object:

**a. Spec Locator (clickable)**

HYPERLINK(

"/" & Packaging\_Specification\_\_c,

Packaging\_Specification\_\_r.Name

)

* Replace Packaging\_Specification\_\_c with the actual lookup field API name to the Specification.
* This pulls the PKG-#### style name and makes it clickable.

**b. Spec Description**

Packaging\_Specification\_\_r.specright\_\_Description\_\_c

* Pulls the description (e.g., “Carton”, “Front Label”).

**2️⃣ Add These Fields to the Related List Layout**

* Go to **Setup → Object Manager → NPDs to Specifications**.
* Edit the **page layout** for that object.
* In the **Related List** section for NPDs to Specifications (as it appears on the NPD record), add the new formula fields to the list columns.

**3️⃣ Verify in Your Dynamic Related List**

The Dynamic Related List – Single component will now be able to show those columns, since it uses the standard related list layout configuration for available fields.

**4️⃣ Optional: Sort or Filter**

You can use the Dynamic Related List component’s filter options to only show active specs or specific types if you want.

If you want, I can give you the exact **formula field definitions** already mapped to Specright’s standard field API names so you can just copy/paste them into your org.  
That will make sure PKG-14789, LBL-88888, etc., and their descriptions show exactly as you described.

**You said:**

tell me what do i do

**SpecForce By Fuhad said:**

Here’s the exact sequence you should follow so your **NPDs to Specifications** related list shows the record locator (PKG-####) and description:

**Step 1 – Go to the NPDs to Specifications object**

1. In Salesforce, click the **gear icon → Setup**.
2. Go to **Object Manager**.
3. Find and click **NPDs to Specifications** (this is your junction object).

**Step 2 – Create two formula fields**

We’ll make one for the clickable record locator and one for the description.

**a. Field 1 – Spec Locator (Hyperlinked PKG Code)**

* Click **Fields & Relationships → New**.
* Data type: **Formula**
* Return type: **Text**
* Field Label: Spec Locator
* Formula:

HYPERLINK(

"/" & Packaging\_Specification\_\_c,

Packaging\_Specification\_\_r.Name

)

*(Replace Packaging\_Specification\_\_c with the actual lookup API name to the Specification object.)*

* Save.

**b. Field 2 – Spec Description**

* Click **New** again.
* Data type: **Formula**
* Return type: **Text**
* Field Label: Spec Description
* Formula:

Packaging\_Specification\_\_r.specright\_\_Description\_\_c

*(Again, swap Packaging\_Specification\_\_c for your real field name.)*

* Save.

**Step 3 – Add the new fields to the Related List**

1. Still in **Object Manager**, open the **NPD** object (the parent).
2. Go to **Page Layouts**.
3. Edit the layout that has the **NPDs to Specifications** related list.
4. In that related list, click the **wrench icon** to configure columns.
5. Add:
   * Spec Locator
   * Spec Description
6. Save the layout.

**Step 4 – Check your Lightning page**

Since your Dynamic Related List – Single component points to **NPDs to Specifications**, it will now offer the new fields for display.

**Result:**  
You’ll see something like:

| **Spec Locator** | **Spec Description** |
| --- | --- |
| PKG-14789 | Carton |
| LBL-88888 | Front Label |
| BTL-77777 | HDPE Bottle |

…and the PKG codes will be clickable links to the spec.

If you want, I can give you the *exact* field API names Specright usually uses for this lookup so you can copy/paste without guessing.  
Do you want me to list them?

**You said:**

what is this specright description object? which object is this?

**SpecForce By Fuhad said:**

The specright\_\_Description\_\_c field I mentioned isn’t a separate object — it’s a **field** that lives on the **Specification** object in the Specright managed package.

Here’s how it fits into your setup:

**1. The key objects in your case:**

* **NPD (specright\_\_NPD\_\_c or your custom NPD object)** – your project/initiative.
* **NPDs to Specifications** – the junction object that links your NPD to Packaging Specifications.
* **Specification (specright\_\_Specification\_\_c)** – the Specright object where each packaging spec record lives.  
  This is where the PKG-#### record locator and the description of the packaging (Carton, Label, Bottle, etc.) are stored.

**2. About the specright\_\_Description\_\_c field:**

* Standard field in Specright’s **Specification** object.
* Used to store the **plain-language description** of the spec (e.g., “Carton”, “HDPE Bottle”).
* When your junction object looks up to a Specification record, you can pull that description field into your related list with a cross-object formula.

**3. How the formula references it:**  
Example formula for **Spec Description** on NPDs to Specifications:

Packaging\_Specification\_\_r.specright\_\_Description\_\_c

* Packaging\_Specification\_\_c = lookup field on NPDs to Specifications pointing to Specification.
* .r means “go to the related record.”
* specright\_\_Description\_\_c = the description field on Specification.

If you open one of your Packaging Specification records directly, you should see the **Description** field right there — that’s the one we’re referencing.

I can map the exact lookup field name for you so your formulas will work without guesswork.  
Do you want me to give you those API names as they appear in a standard Specright packaging setup?

**You said:**

why do we need this field?

**SpecForce By Fuhad said:**

**You said:**

which object contain this field?

**SpecForce By Fuhad said:**

The specright\_\_Description\_\_c field lives on the **Specification** object in the Specright managed package:

* **Object API Name:** specright\_\_Specification\_\_c
* **Field API Name:** specright\_\_Description\_\_c

**Why it exists:**  
In Specright, the Specification object holds all the details for a packaging spec — things like record locator (PKG-####), dimensions, materials, and a **description** field.  
That description is meant to be a short, human-readable label for the spec (e.g., “Carton”, “Front Label”, “HDPE Bottle”).

When you connect an NPD to a spec via the **NPDs to Specifications** junction object, the junction only stores the lookup to the Specification — it doesn’t automatically show descriptive details.  
That’s why we create a **formula field** on the junction object to pull this description from the Specification object, so your related list shows both PKG-#### and the text label.

If you like, I can show you how to confirm in your org that specright\_\_Description\_\_c is on specright\_\_Specification\_\_c and exactly where to find it in Object Manager. Would that help?

**You said:**

i don't see this one

**SpecForce By Fuhad said:**

If you’re not seeing specright\_\_Description\_\_c on the **Specification** object, there are a few possibilities:

**1️⃣ Your org might be using a different field name for the description**

Some Specright implementations rename or replace that standard field.  
Common alternatives I’ve seen:

* Description\_\_c (without the specright\_\_ namespace, if it’s a custom field your team made)
* specright\_\_Spec\_Description\_\_c
* specright\_\_Specification\_Description\_\_c

**2️⃣ How to find the right one**

1. Go to **Setup → Object Manager**.
2. Find and open **Specification** (API name: specright\_\_Specification\_\_c).
3. Click **Fields & Relationships**.
4. Search for any field with “Description” in the label or API name.
5. Open each candidate and check:
   * **Field Type:** Text or Text Area
   * **Purpose:** Short description of the spec, not long notes.

**3️⃣ If your org truly doesn’t have it**

If there’s no description field at all, you can:

* **Create a new Text field** called something like Spec Description.
* Add it to the Specification page layout so your team can populate it.
* Reference this new field in your formula on the NPDs to Specifications object.

If you want, I can give you a **formula version** that will work even if the field is named differently — you’d just swap in your actual description field API name once you find it.

Do you want me to walk you through creating that formula so you can test it right away?

**You said:**

i have created both the field. how do i add this one in the related list?

**SpecForce By Fuhad said:**

Perfect — since you’ve created the **Spec Locator** and **Spec Description** formula fields on your **NPDs to Specifications** junction object, here’s how to get them into your related list so they show in your NPD record page:

**Step 1 – Edit the NPD Page Layout**

1. Go to **Setup → Object Manager**.
2. Find and click your **NPD** object (API name might be specright\_\_NPD\_\_c or a custom name).
3. Click **Page Layouts**.
4. Open the page layout that your NPD records are using.

**Step 2 – Update the Related List**

1. On the page layout, scroll down to find the **NPDs to Specifications** related list.
2. Click the **wrench icon** (⚙) on that related list.
3. In the **Available Fields** list, find:
   * Spec Locator
   * Spec Description
4. Use the **Add** button to move them into **Selected Fields**.
5. (Optional) Use the **Up/Down arrows**

**You said:**

i want to do this from lightning record page

**SpecForce By Fuhad said:**

Got it — if you want to add your **Spec Locator** and **Spec Description** to the related list **directly from the Lightning Record Page**, here’s how you do it:

**1️⃣ Open the Lightning App Builder**

1. Go to an **NPD record** in Lightning.
2. Click the **gear icon → Edit Page**.  
   This opens the Lightning App Builder for the NPD record page.

**2️⃣ Find the Related List Component**

You probably have a **Dynamic Related List – Single** component showing your **NPDs to Specifications** connection records.

1. Click on that **Dynamic Related List – Single** component.
2. In the right-hand panel, check:
   * **Parent Record:** should already be set to your NPD.
   * **Related List:** should be NPDs to Specifications (or whatever your junction object’s label is).

**3️⃣ Add the New Fields to Display**

In the **Fields** section of the component settings:

1. Click **Add Field**.
2. Search for Spec Locator and Spec Description (the formula fields you created).
3. Add both.
4. You can reorder the columns by dragging them in the list.

**4️⃣ Save and Activate**

1. Click **Save** in the Lightning App Builder.
2. Click **Activation** if you want to set this as the default page for the NPD record type.
3. Go back to your NPD record and refresh — you should now see:

PKG-14789 Carton

LBL-88888 Front Label

BTL-77777 HDPE Bottle

with PKG codes clickable.

If you want, I can give you the **best-practice field order** we usually use in Specright so the related list reads exactly like a packaging BOM table.

Top of Form

Bottom of Form

ChatGPT can make mistakes. Check important info.