

Builder Advanced Tools

Advanced Tools can be found inside the **Form Builder**, **Datalist Builder** as well as in the **Userview Builder**.

New Feature

This is a new feature in Joget Workflow v6.

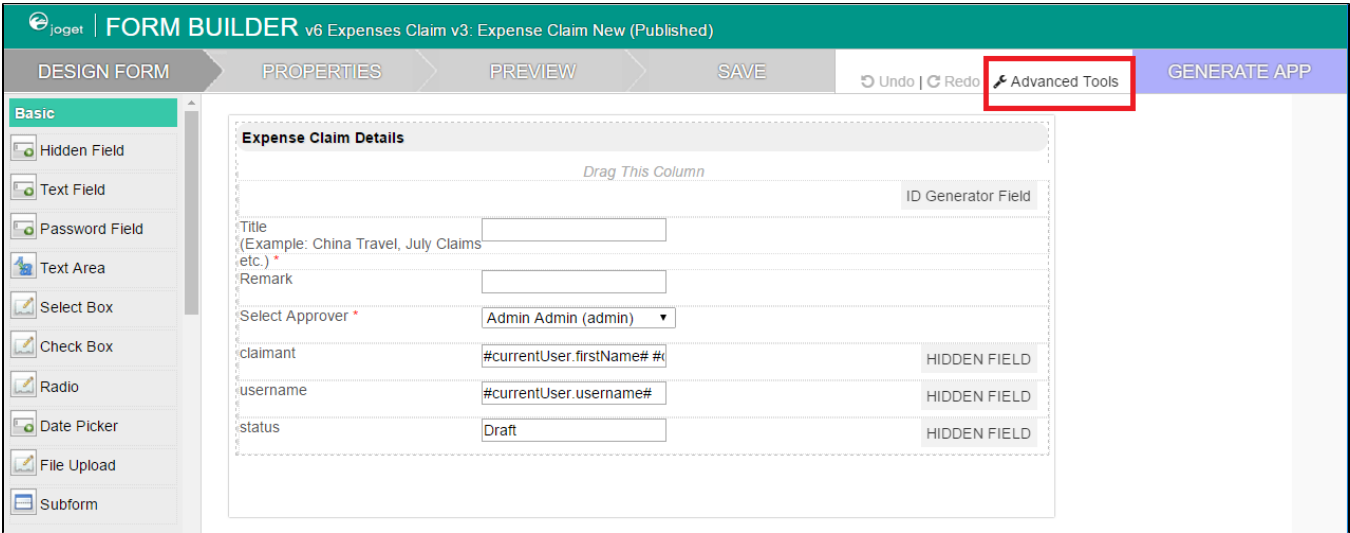


Figure 1: Accessing the Advanced Tools in Form Builder



Figure 2: Advanced Tools - Tabs

In Advanced Tools, you will find 5 tabs in all the builder mentioned and 6 tabs in the Form Builder. In Form Builder, the sixth extra tab is **Table**.

- [Tree Viewer](#)
- [Usages](#)
- [Table](#)
- [i18n Internationalization](#)
- [Diff Checker](#)
- [JSON Definition](#)

Tree Viewer

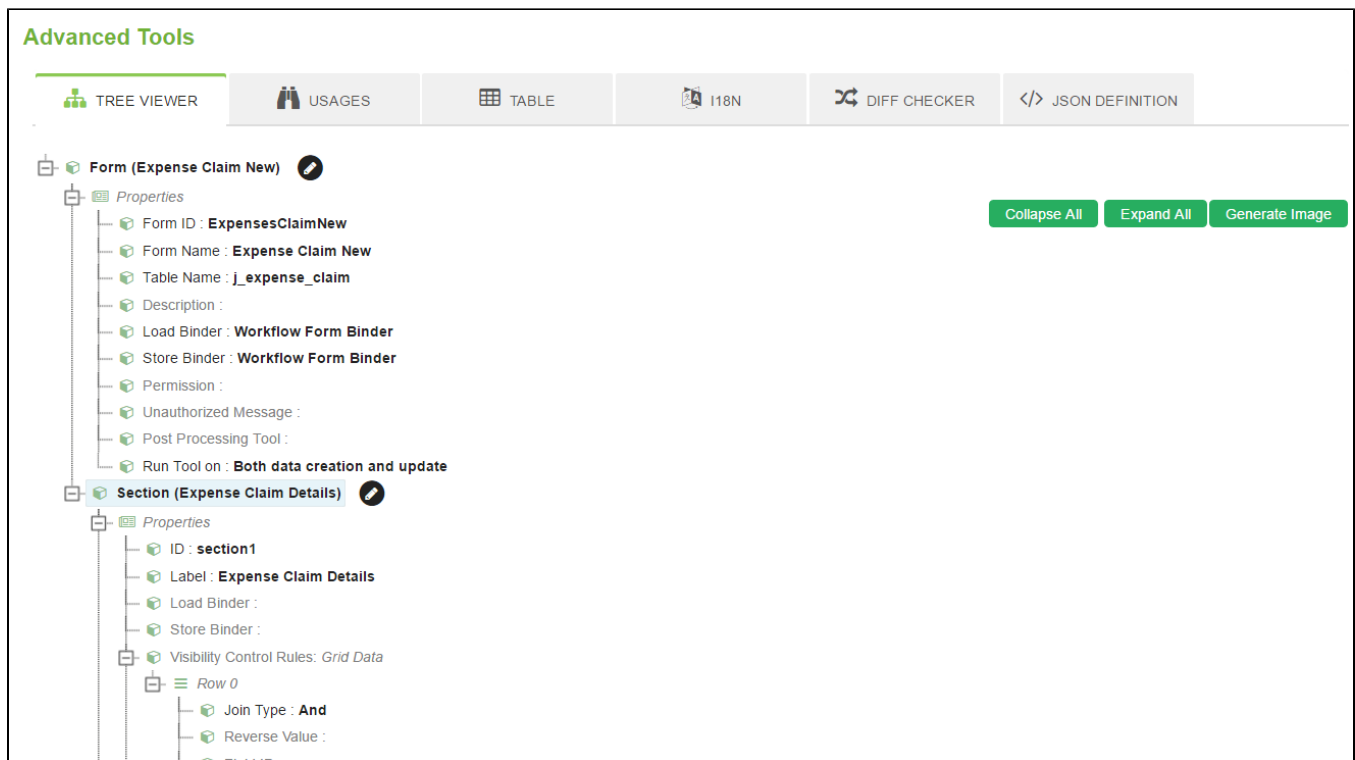


Figure 3: Advanced Tools - Tree Viewer

The tree viewer in the first tab provides the app designer with the overall structure/definition view so that one can easily spots any anomaly and fixes it immediately without too much of hassle and time spent.

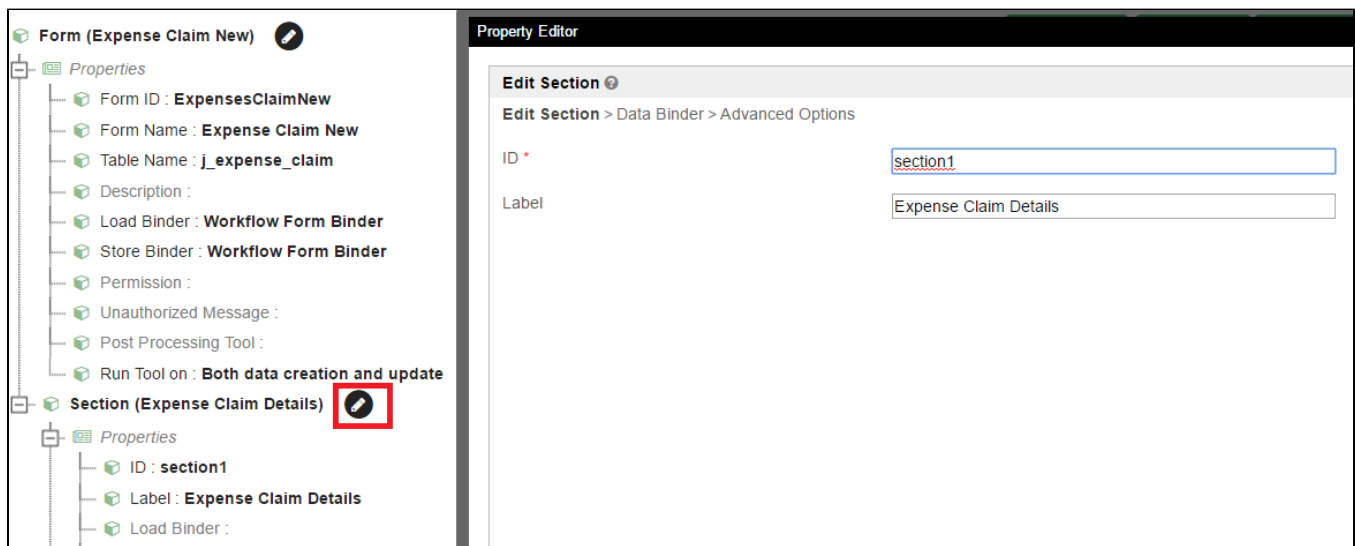


Figure 4: Advanced Tools - Tree Viewer - Editing a Node

Clicking on the edit (pencil) icon on the right of the node allows one to configure/edit its properties right away. Upon saving of the properties, the tree viewer will refresh itself.

Usages



Figure 5: Advanced Tools - Usages

In the second tab, app designer can inspect on where the current entity is being used. From the screenshot example above, we can tell that the form is being used in another form and also being part of a workflow process. With such feature, app designer will be able to manage the form design well within the scope of its usages.

Table

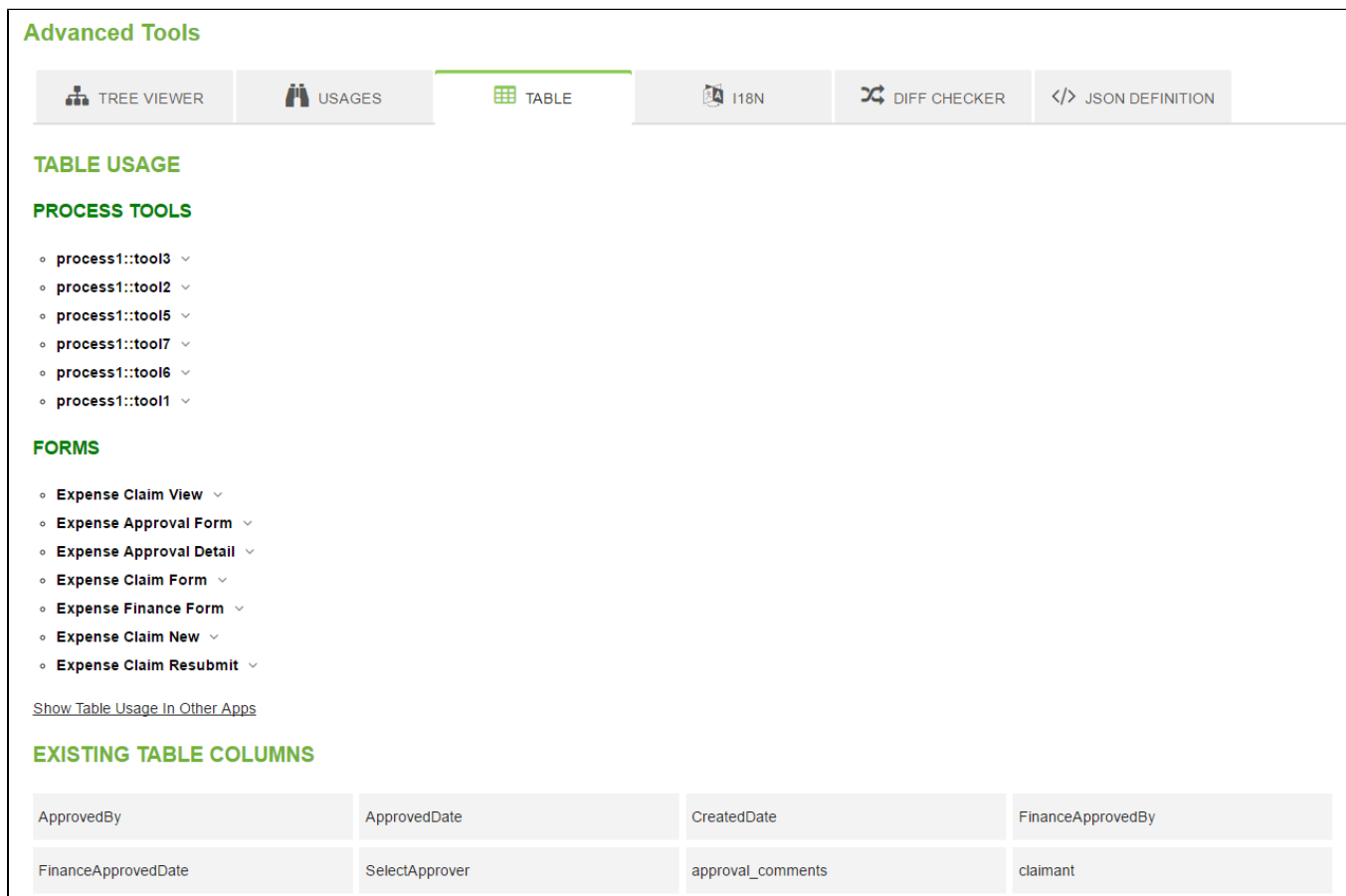


Figure 6: Advanced Tools - Table

The **table** tab is available in the Form Builder. With this feature, app designer can tell on which other entity is sharing/writing into the same database table the current form is currently pointing too. From the screenshot example above, we can see that the table is being accessed by multiple process tools and forms.

We can also click on the **Show Table Usage In Other Apps** to check if there's other apps that are writing to the same table too.

At the bottom of it, **existing table columns** are listed down for app designer's reference.

i18n Internationalization

Advanced Tools

TREE VIEWER

USAGES

TABLE

I18N

DIFF CHECKER

JSON DEFINITION

Expense Claim Details	<div>zh_CN Save</div> <div>详情</div>	<div>Choose Locale</div>
Ref	<div>参考</div>	
Remark	<div>备注</div>	
Select Approver	<div>审批</div>	
Title (Example: China Travel, July Claims etc.)	<div>题目</div>	

Figure 7: Advanced Tools - I18n Internationalization

Labels defined in the builder will be automatically listed down here for internationalization purpose. We can define in as many languages as we want to.

Diff Checker

Advanced Tools



TREE VIEWER



USAGES



TABLE



I18N



DIFF CHECKER



JSON DEFINITION

Merge & Update

Your changes will be automatically merged into the following JSON definition when saved. You can also click the "Merge & Update" button to see the merged changes directly.

```
{
  elements: [
    0: {
      elements: [
        0: {
          elements: [
            1: {
              "className": "org.joget.apps.form.lib.TextField",
              "properties": {
                "id": "code",
                "label": "Code",
                "value": "",
                "maxLength": "",
                "validator": {
                  "className": "",
                  "properties": {}
                },
                "encryption": "",
                "placeholder": "",
                "size": "",
                "readonly": "",
                "readonlyLabel": "",
                "workflowVariable": "",
                "style": "",
                "storeNumeric": ""
              },
            },
            5: {
              "className": "org.joget.apps.form.lib.TextField",
              "properties": {
                "id": "Priority",
                "label": "Priority",
                "value": "",
                "maxLength": "",
                "validator": {
```

There are new changes saved during your editing as follows:

```
{
  elements: [
    0: {
      elements: [
        0: {
          elements: [
            1: {
              "className": "org.joget.apps.form.lib.TextField",
              "properties": {
                "id": "code",
                "label": "Code",
                "value": "",
                "maxLength": "",
                "validator": {
                  "className": "",
                  "properties": {}
                },
                "encryption": "",
                "placeholder": "",
                "size": "",
                "readonly": "",
                "readonlyLabel": "",
                "workflowVariable": "",
                "style": "",
                "storeNumeric": ""
              },
            },
            5: {
              "className": "org.joget.apps.form.lib.TextField",
              "properties": {
                "id": "Priority",
                "label": "Priority",
                "value": "",
                "maxLength": "",
                "validator": {
```

Figure 8: Advanced Tools - Diff Checker

The **Diff Checker** capability makes collaborative works possible without losing the modifications done by other teammates who are working on the same entity at the same time. In this section, one can tell which are the changes have been made since the initial first load of his/her builder against the copy in the server. Click on **Merge & Update** button will ensure that all modifications are merged.



Upon pulling the latest changes from the server, clicking on **Merge & Update** will merge the changes, but will **NOT** automatically save the form, to allow users to continue editing.

Take note that the Diff Checker does **NOT** pull changes from the server in real-time. To see new changes, click into the Diff Checker tab again.

JSON Definition

Advanced Tools

TREE VIEWER

USAGES

TABLE

I18N

DIFF CHECKER

</> JSON DEFINITION

Update

```
1 {
2   "className": "org.joget.apps.form.model.Form",
3   "properties": {
4     "id": "ExpensesClaimNew",
5     "loadBinder": {
6       "className": "org.joget.apps.form.lib.WorkflowFormBinder",
7       "properties": {}
8     },
9     "tableName": "j_expense_claim",
10    "description": "",
11    "postProcessorRunOn": "both",
12    "name": "Expense Claim New",
13    "storeBinder": {
14      "className": "org.joget.apps.form.lib.WorkflowFormBinder",
15      "properties": {}
16    },
17    "postProcessor": {
18      "className": "",
19      "properties": {}
20    },
21    "permission": {
22      "className": "",
23      "properties": {}
24    },
25    "noPermissionMessage": ""
26  },
27  "elements": [
28    {
29      "elements": [
30        {
31          "elements": [
32            {
33              "className": "org.joget.apps.form.lib.IdGeneratorField",
34              "properties": {
35                "id": "ref",
36                "workflowVariable": "",
37                "hidden": "true",
38                "label": "Ref",
39                "format": "????",
40                "envVariable": "counter"
41              }
42            },
43            {
44              "className": "org.joget.apps.form.lib.TextField",
45              "properties": {
46                "readonlyLabel": "",
47                "placeholder": "",
48                "validator": {
49                  "className": "org.joget.apps.form.lib.DuplicateValueValidator",
50                  "properties": {
```

Figure 9: Advanced Tools - JSON Definition

We can inspect the actual definition of the entity that we are working on here. If required, we can amend the definition within the text editor itself and click on the **Update** button to commit.