Lesson 5 Configuring Search

ClaimCenter provides a robust search functionality that can find claims, activities, checks, recoveries, and bulk invoices – as well as vendors, through integration with your address book application (such as Guidewire ContactManager). This capability can be extended to search for existing or custom fields that are not currently searchable.

5.1 Percent complete field

Exercise

Succeed Insurance requires a field indicating the percent complete on a claim. This should be a required field. For purpose of upgrading the existing claims, the default value should be 50%.

The field should be searchable in the Claim Search screen. The user inputs a value, and the search engine retrieves claims whose percent field is greater than or equal to that value.

Prerequisites

This lesson assumes that you have successfully completed the **ClaimCenter Configuration Kickstart** course.

ClaimCenter Studio should be running with no components open. ClaimCenter Server should not be running.

Exercise 1: Percent complete field

1. **Add the PercentComplete_Ext field to Claim.etx.**

name	**PercentComplete_Ext**
type	**percentagedec**
nullok	**false**
desc	**Percentage of claim that is complete**
default	**50**

2. **Add the PercentComplete_Ext field to ClaimSearchCritera.etx.**

   Note that the column on the ClaimSearchCriteria entity should ***not*** be required (nullok = **true**).

3. **Add a <Criterion> element to the search-config.xml file.**

   a) Add it to the ClaimSearchCriteria <CriteriaDef> whose targetEntity is **Claim**.

   b) The matchType should be **ge**.

4. **Add a display key for the search label.**
5. **Add the PercentComplete_Ext column as a text input on ClaimSearchOptionalInputSet.pcf.**

   a) Place it directly under the LossType input.

   b) Set the following values:

editable	**true**
id	**Completion**
label	**DisplayKey.get("JSP.ClaimSearch.Claims.PercentComplete_Ext")**
required	**false**
value	**ClaimSearchCriteria.PercentComplete_Ext**
formatType	**percentagePoints**
valueType	**Java.math.BigDecimal**

   c) **Note:** In an actual implementation, you would also create one or more fields on the Claim screens where adjusters could input the Percent Complete value, or a mechanism for calculating it.

6. **Regenerate the Simple Object Access Protocol (SOAP) APIs.**

   a) Open a command window at **C:/Guidewire/ClaimCenter**.

   b) Run the command **gwb genJavaApi**. (This will take about 15 minutes.)

   c) **Note:** You would normally run the command **gwb genDataDict**. To save time in the exercise we will skip this step, which would take an additional five minutes or more.

Test procedure

1. **Start the ClaimCenter server in debug mode.**

2. **Log in to ClaimCenter as Andy Applegate (aapplegate/gw).**

3. **Select Search ➔ Claims ➔ Advanced Search.**

4. **Verify that the Percent Complete field is present.**

5. **Set the Assigned User criterion to Andy Applegate, and click Search.**

   How many claims appear?

6. **Set the Percent Complete criterion to 55 and click Search.**

   Now how many claims appear?

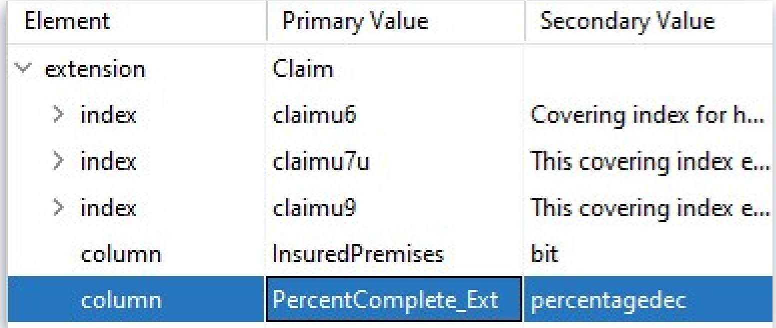
7. **Finally, set the Percent Complete criterion to 49 and click Search.**

   How many claims appear?

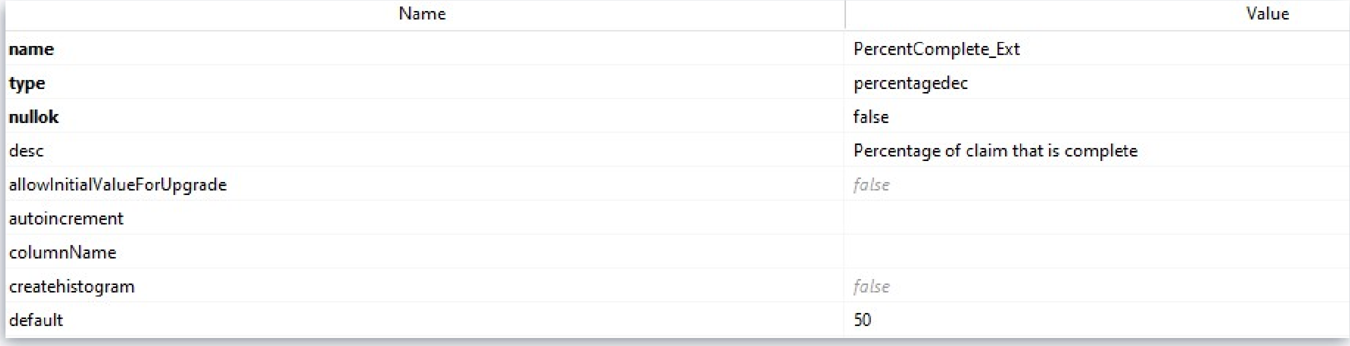
Stop

5.2 Solution

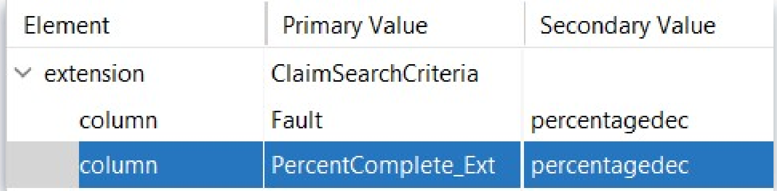
Solution 1: Percent complete field



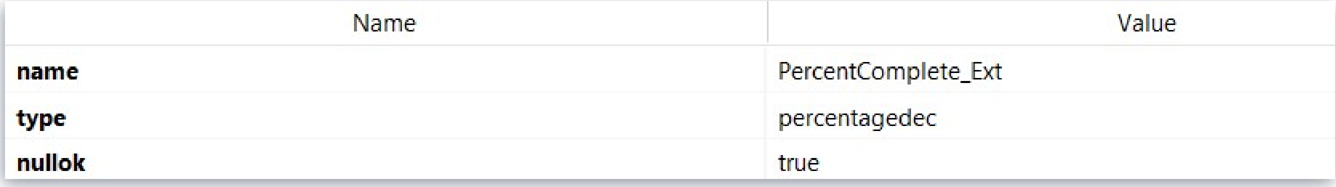
Element	Primary Value	Secondary Value
extension	Claim	
index	claimu6	Covering index for h...
index	claimu7u	This covering index e...
index	claimu9	This covering index e...
column	InsuredPremises	bit
column	PercentComplete_Ext	percentagedec



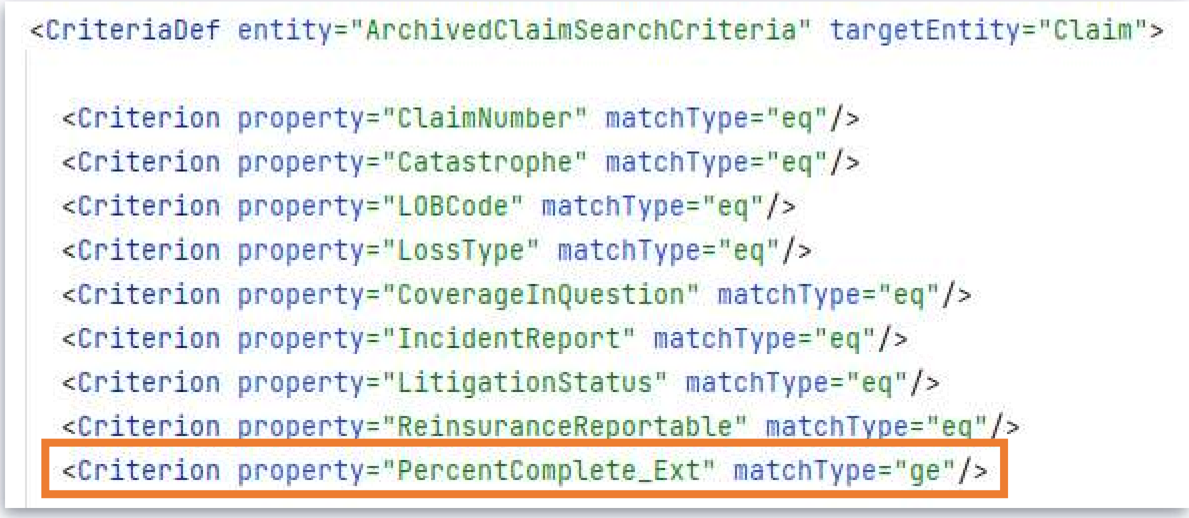
Name	Value
name	PercentComplete_Ext
type	percentagedec
nullok	false
desc	Percentage of claim that is complete
allowInitialValueForUpgrade	*false*
autoincrement	
columnName	
createhistogram	*false*
default	50



Element	Primary Value	Secondary Value
extension	ClaimSearchCriteria	
column	Fault	percentagedec
column	PercentComplete_Ext	percentagedec

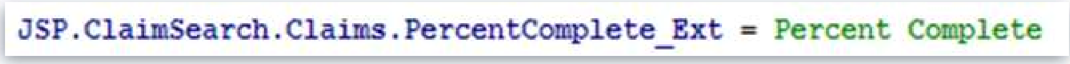


Name	Value
name	PercentComplete_Ext
type	percentagedec
nullok	true

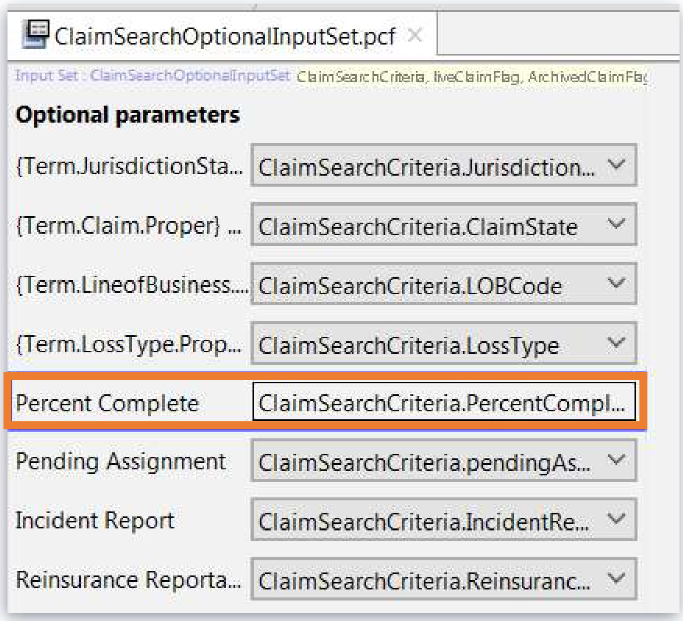


```
<CriteriaDef entity="ArchivedClaimSearchCriteria" targetEntity="Claim">

  <Criterion property="ClaimNumber" matchType="eq"/>
  <Criterion property="Catastrophe" matchType="eq"/>
  <Criterion property="LOBCode" matchType="eq"/>
  <Criterion property="LossType" matchType="eq"/>
  <Criterion property="CoverageInQuestion" matchType="eq"/>
  <Criterion property="IncidentReport" matchType="eq"/>
  <Criterion property="LitigationStatus" matchType="eq"/>
  <Criterion property="ReinsuranceReportable" matchType="eq"/>
  <Criterion property="PercentComplete_Ext" matchType="ge"/>
```



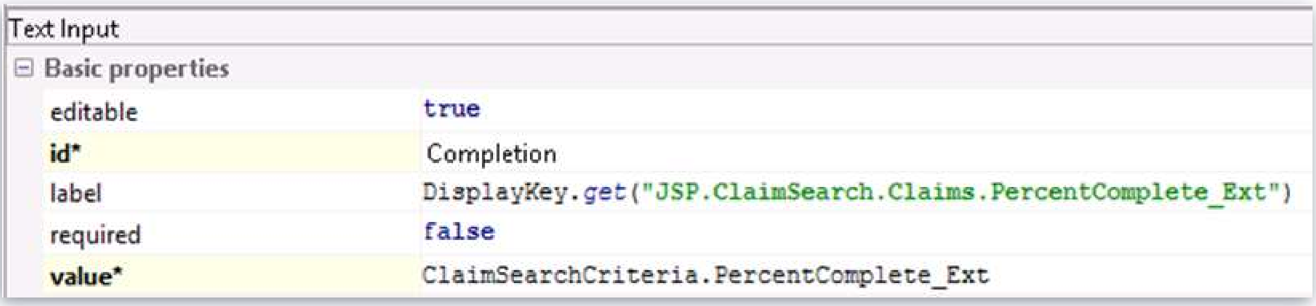
```
JSP.ClaimSearch.Claims.PercentComplete_Ext = Percent Complete
```



ClaimSearchOptionalInputSet.pcf
Input Set : ClaimSearchOptionalInputSet ClaimSearchCriteria, liveClaimFlag, ArchivedClaimFla

Optional parameters

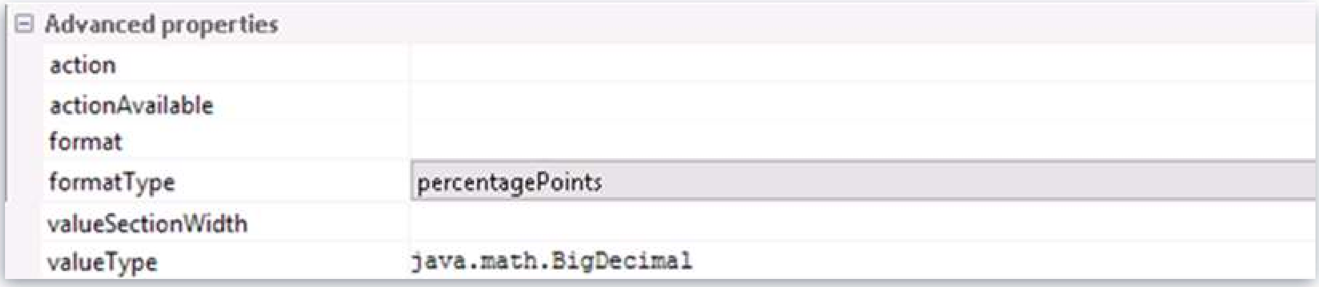
{Term.JurisdictionSta...	ClaimSearchCriteria.Jurisdiction...
{Term.Claim.Proper} ...	ClaimSearchCriteria.ClaimState
{Term.LineofBusiness....	ClaimSearchCriteria.LOBCode
{Term.LossType.Prop...	ClaimSearchCriteria.LossType
Percent Complete	ClaimSearchCriteria.PercentCompl...
Pending Assignment	ClaimSearchCriteria.pendingAs...
Incident Report	ClaimSearchCriteria.IncidentRe...
Reinsurance Reporta...	ClaimSearchCriteria.Reinsuranc...



Text Input

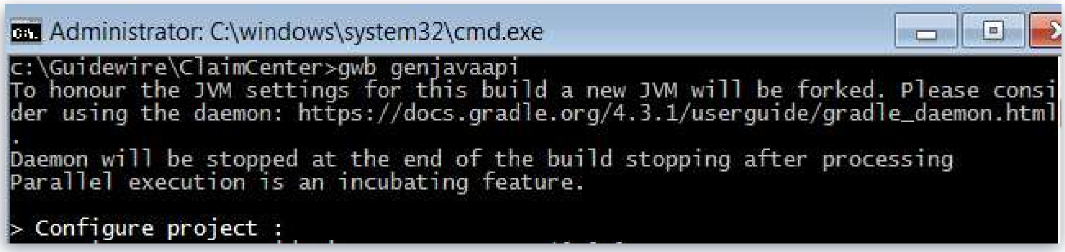
⊟ Basic properties

editable	true
id*	Completion
label	DisplayKey.get("JSP.ClaimSearch.Claims.PercentComplete_Ext")
required	false
value*	ClaimSearchCriteria.PercentComplete_Ext



⊟ Advanced properties

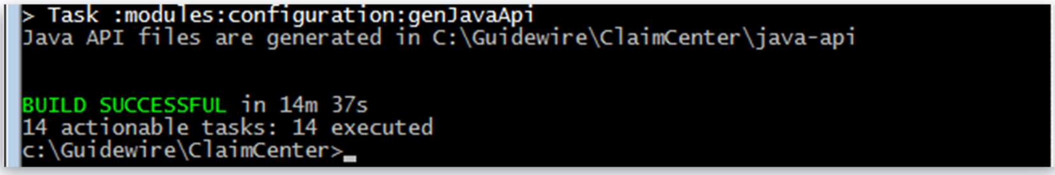
action	
actionAvailable	
format	
formatType	percentagePoints
valueSectionWidth	
valueType	java.math.BigDecimal



Administrator: C:\windows\system32\cmd.exe

```
c:\Guidewire\ClaimCenter>gwb genjavaapi
To honour the JVM settings for this build a new JVM will be forked. Please consi
der using the daemon: https://docs.gradle.org/4.3.1/userguide/gradle_daemon.html
.
Daemon will be stopped at the end of the build stopping after processing
Parallel execution is an incubating feature.

> Configure project :
```



```
> Task :modules:configuration:genJavaApi
Java API files are generated in C:\Guidewire\ClaimCenter\java-api


BUILD SUCCESSFUL in 14m 37s
14 actionable tasks: 14 executed
c:\Guidewire\ClaimCenter>_
```

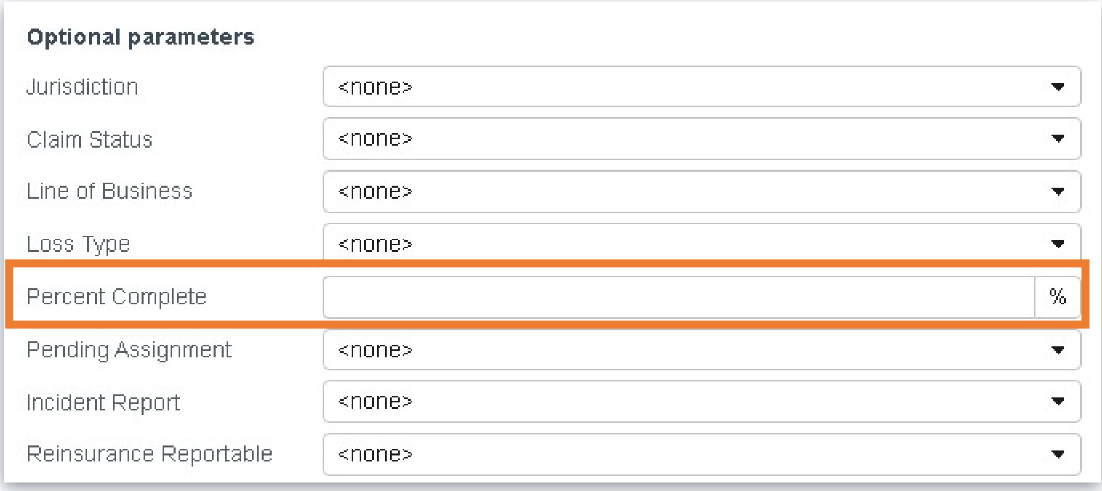
Test procedure

1. **Start the ClaimCenter server in debug mode.**

2. **Log in to ClaimCenter as Andy Applegate (aapplegate/gw).**

3. **Select Search ➔ Claims ➔ Advanced Search.**

4. **Verify that the Percent Complete field is present.**



Optional parameters

Jurisdiction	<none>
Claim Status	<none>
Line of Business	<none>
Loss Type	<none>
Percent Complete	%
Pending Assignment	<none>
Incident Report	<none>
Reinsurance Reportable	<none>

5. **Set the Assigned User criterion to Andy Applegate, and click Search.**

   How many claims appear?

   6 (may vary depending on how many claims you have created))

6. **Set the Percent Complete criterion to 55 and click Search.**

   Now how many claims appear?

   0

7. **Finally, set the Percent Complete criterion to 49 and click Search.**

   How many claims appear?

   6 (or whatever you saw for question #5)