

- c) What challenges did you face while creating the form pattern, if any?
- d) Were you able to verify the addition of the form to the policy of your LOB? How did you do it?



13.5 Solutions

13.5.1 View form patterns

1. Log in as superuser su/gw
2. Go to Administration → Business Settings → Policy Form Patterns
3. Find the Businessowners policy line form pattern with the Form Number BP0001

Answer the following questions:

- a) What is the name of the form?

Businessowners Coverage Form

- b) Is the Policy Change tab available for this form? How do you know?

Yes, because it is one of the transactions listed in the Transaction Types tab.

- c) When can a form be inferred during a job? Where do you specify that information?

Forms can be inferred at quote and bind time. Whether to infer a form at quote time or both quote and bind time is specified on the Basics tab by the Boolean question "Any data related to this form collected after quote?"

- d) Where do you specify form inference conditions?

The Inference Tab provides some common inference conditions.

13.5.2 Configure a form pattern for personal auto policies

1. Log in as superuser su/gw
2. Start by configuring a form pattern in Administration → Business Settings → Policy Form Patterns
3. Specify the following details:
 - Form name: Notice of Oregon No-Fault Laws XX
 - Form code: PA_OR_XX_Ext
 - Form number: PA OR 00 XX

- Edition: 0100
- When to infer: When the policy is quoted
- Endorsement number: Does not need to be issued
- Availability: Only available in Oregon

Example:

New Policy Form Pattern [Up to Policy Form Pattern](#)

Basics

Products

Transaction Types

Jurisdictions

Inference

Code *

PA_OR_01_Ext

Number *

PA OR 00 01

Edition *

0100

Name *

Notice of Oregon No-Fault Laws 01

Any data related to this form collected after quote? *

☐ Yes
☒ No

Assign an endorsement number to this form? *

☐ Yes
☒ No

Priority *

-1

Integration Fields

Reference Code

Basics

Products

Transaction Types

Add

Remove

⚙

☐ Product

☐ Personal Auto

Basics

Products

Transaction Types

Add

Remove

⚙

☐ Policy Transaction Type

☐ Submission

☐ Renewal

☐ Rewrite

[Basics](#)
[Products](#)
[Transaction Types](#)
[Jurisdictions](#)
[Inference](#)

Rows are sorted by order of evaluation
Start Dates are inclusive, End Dates are exclusive

Availability Table Add Duplicate ▾ Remove

<input type="checkbox"/> Available? *	Start Date *	End Date	Jurisdiction
<input type="checkbox"/> <input checked="" type="radio"/> Yes <input type="radio"/> No	01/01/2000	MM/dd/yyyy	Oregon ▾

Jurisdictional Replacements

Group Code

This form is the jurisdiction-specific version of:

[Basics](#)
[Products](#)
[Transaction Types](#)
[Jurisdictions](#)
[Inference](#)

Forms are only inferred when Product, Transaction Types, and Jurisdiction criteria also match.

Policy line

Form inference conditions

13.6 References



Review

What is a form and how do you add it to a policy?

Forms are documents detailing specific aspects of the policy. Forms contain legal information about a policy and are part of the policy contract.



Review

A list of inferred forms is stored in the database for each policy period. The actual form content is created in the document production system by merging the policy data with the form template associated with the form on the policy.

PolicyCenter generates forms based on details that the user has specified. The process of determining what forms should be generated is called **form inference**.



Find

For more information about form multiplicity, refer to PolicyCenter documentation. Some business cases may require multiple instances of the same form. For example, a personal auto policy with multiple vehicles may require a single form for each vehicle. PolicyCenter supports form multiplicity, but it does require configuration.



Best Practice

Permissions and forms

To access the form administration user interface, users must log in as a user with the appropriate permissions, which are:

- `formpatcreate`: permission to create new form patterns
- `formpatview`: permission to access the forms editor and view existing form patterns
- `formpatedit`: permission to edit existing form patterns
- `formpatdelete`: permission to delete existing form patterns



Stop