

ISS Approval and Permission Engine Guide

1.0

User Manual

Contents

Introduction	2
Approval Rules	2
Define Approval Rules	2
Approval Rule	2
hasRuleStatus	2
hasCondition	2
hasApprovalGateFlow	3
Condition	3
ConditionAttribute	3
ConditionBusinessType	3
ConditionValue	3
HasConditionType	4
Special Condition	4
System Condition	4
TRUE Condition	4
Approval Gate	4
Usage of Approval Rules	5
Logging	5
SalesDocumentAllowedActions	6
Quote Actions	7
Opportunity Actions	8
Inbox	8
Approval and Routing Annotation	9
Permission	9
Permission Hierarchy	10

Introduction

ISS/CPQ supports Quote Approval based on approval rules concept. This concept offers defining and maintenance of the conditions needed to be checked in rules and rules itself from the system itself. To achieve this certain concepts, the following BusinessTypes are introduced here:

ApprovalRule

Condition

ConditionType

RuleStatus

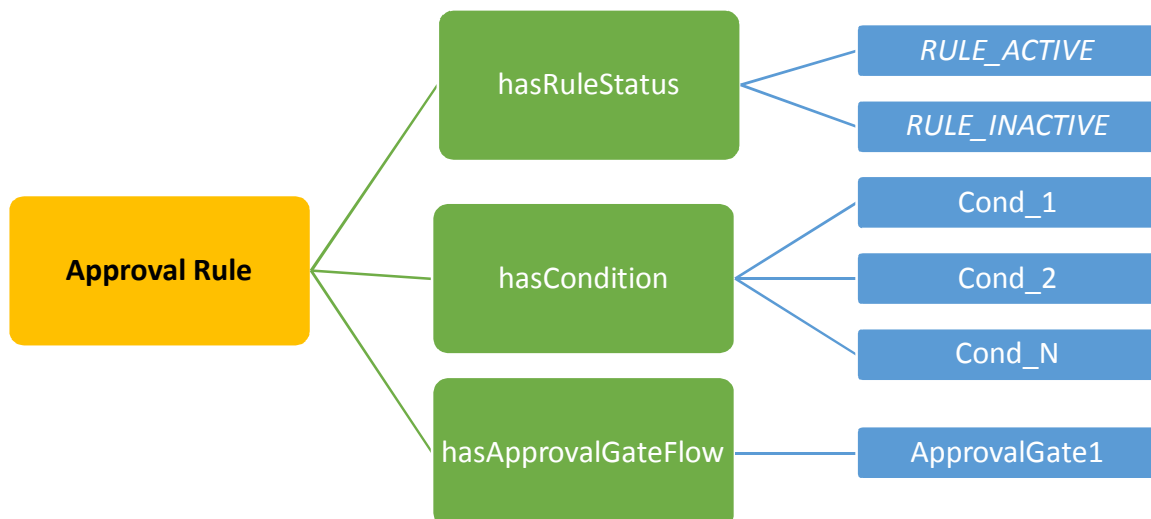
Current ISS approval engine supports both approval and routing. Note that routing can only be done when Quote is in OPEN status.

Approval Rules

Define Approval Rules

Approval Rule

Approval value is the fundamental part of Approval Engine. It is composed of one or more conditions and one mandatory approval gate. It also has its own status active or not.



hasRuleStatus

Available rule statuses are ***RULE_ACTIVE*** and ***RULE_INACTIVE***.

hasCondition

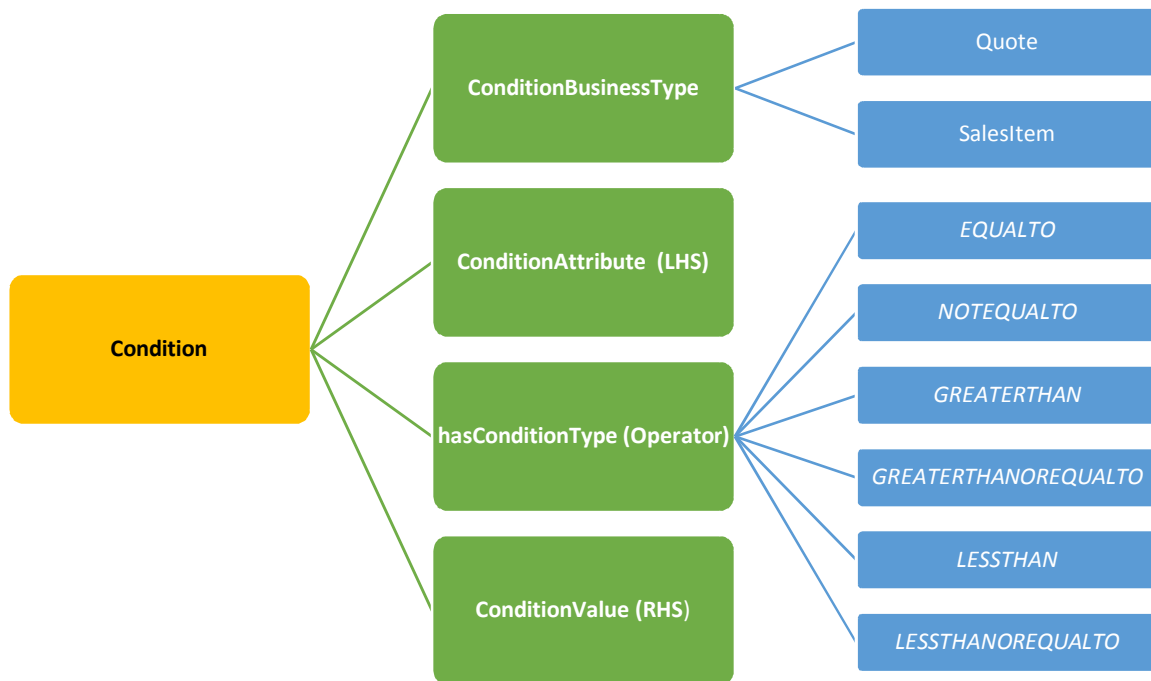
Each approval rule can have multiple conditions. Rule is fired only when *all* the conditions are satisfied.

hasApprovalGateFlow

HasApprovalGateFlow is the outcome of the rule. When the rule is fired, it is the starting point of whole workflow process. Depending on *who* submit the quote, ISS will *recursively* find the approval gate with higher role according to role hierarchy.

Condition

Condition is the most basic and atomic unit of approval rule. A condition can be contained by more than one approval rules.



ConditionAttribute

Condition attribute can be any *Business Data Attribute* and *Business Relation Attribute* on **Quote**, **SalesItem**, **DocumentHeaderPriceltem** and **ItemHeaderPriceltem**. During approval rule finding, the actual value of this attribute is substituted as left operand.

ConditionBusinessType

The domain of condition attribute it belongs to. Currently the two business types supported are **Quote** and **SalesItem**. The condition business type needs to be used either Quote or SalesItem depending on attributes are on Quote level or SalesItem level.

If quote is satisfied all the conditions of the rule, it is assumed that the rule is fired.

If one of the sales Items is satisfied all the conditions of the rule, it is assumed that the rule is fired.

ConditionValue

The value that is used to check in the condition as right operand. User can input any value regardless of the type. Moreover, ISS supports a predefined set of constant values via *ConditionEnvironmentVariables*.

The followings are individuals of condition environment variables:

1. VAR_LOGGED_IN_USER
Represent current login user business object.
2. VAR_LOGGED_IN_USER_ROLE
Represent current login user role business object.

3. VAR_TODAY

Represent today date.

For example, these constant values can be used in conditions such as quote creator must be current logged in user or current logged in user role.

HasConditionType

Condition Type are similar concept to operator in mathematical equation.

Six types of Condition Types are supported as follows:

1. EQUALTO
2. NOTEQUALTO
3. GREATERTHAN
4. GREATERTHANOREQUALTO
5. LESSTHAN
6. LESSTHANOREQUALTO

Among six condition types, depending on the range of conditionAttribute, ISS/CPQ supports the operators as shown in matrix:

Range of Condition Attribute	Supported Operator					
	=	!=	>	>=	<	<=
Decimal	X	X	X	X	X	X
Integer	X	X	X	X	X	X
Boolean	X	X	-	-	-	-
String	X	X	-	-	-	-
Calendar	X	X	-	-	-	-
Business Relation Attribute	X	X	-	-	-	-

Special Condition

Two types of special conditions are provided:

System Condition

System condition is a condition that is applicable to all rules. Common conditions could be defined as system condition.

TRUE Condition

True condition is a condition whose result is always true. For example: It can be used in a situation where modeller needs to make a rule true by default.

Approval Gate

Approval gate has user role who is responsible for current workflow step and next stage (approval gate) if existed.

For the time being, defining approval rules can be done via master data screen.

Usage of Approval Rules

When the quote is in OPEN status, any user can check dynamically what will be the approval flow and who will be the approvers based on the existing quote data and current login user perspective without actually submitting the quote. Anybody who can update the quote can also submit the quote. Once the quote is submitted, the approval flow is fixed and cannot be changed i.e., the approval gates are fixed. If workflow has more than one approval gates to go to (in parallel or in sequence), the workflow process will conclude only when all the approval gates are checked (that is approval gate owner approved the quote).

Logging

For debugging and tracing, turn on debug logging for the following commands:

1. GetApprovalGatesCommand
2. GetApprovalRulesCommand
3. AbstractRuleCommand

Here is a sample of how logging is shown for a system that consists of 5 rules whereas one of the rules is inactive. Satisfied rules with their respective approval gate can also check in the logging.

```
c:\w.GetApprovalRulesCommand - Finding all satisfied approval rules for Quote BusinessObject(uri=http://www.inmindcomputing.com/application/application-implementation.owl#7858a2d8d9c447aca6788d260aae1ce0) with 0 Items.
c:\w.GetApprovalRulesCommand - No. of System Condition ( Quote ) - 0
c:\w.GetApprovalRulesCommand - No. of System Condition ( SalesItem ) - 0
c:\w.GetApprovalRulesCommand - No. of System Condition ( Condition ) - 0
c:\w.GetApprovalRulesCommand - Checking rule ApprovalRepToManager_OnQuote started...
c:\w.GetApprovalRulesCommand - Condition: type=Quote, operand= http://www.inmindcomputing.com/application/application-schema.owl#documentHeaderDiscount(actual value= 50) GREATERTHAN 10 => true
c:\w.GetApprovalRulesCommand - Satisfy all conditions for Quote => false
c:\w.GetApprovalRulesCommand - Checking rule ApprovalRepToManager_OnQuote ended. Result is false
c:\w.GetApprovalRulesCommand - Checking rule ApprovalManagerToVP_OnQuote started...
c:\w.GetApprovalRulesCommand - Condition: type=Quote, operand= http://www.inmindcomputing.com/application/application-schema.owl#documentHeaderDiscount(actual value= 50) GREATERTHAN 40 => true
c:\w.GetApprovalRulesCommand - Satisfy all conditions for Quote => true
c:\w.GetApprovalRulesCommand - Checking rule ApprovalManagerToVP_OnQuote ended. Result is true
c:\w.GetApprovalRulesCommand - Checking rule ApprovalRepToManager_OnLineItem started...
c:\w.GetApprovalRulesCommand - Satisfy all conditions for Quote => false
c:\w.GetApprovalRulesCommand - Checking rule ApprovalRepToManager_OnLineItem ended. Result is false
c:\w.GetApprovalRulesCommand - Rule BusinessObject(uri=http://www.inmindcomputing.com/application/application-implementation.owl#ALWAYSRUNAPPROVAL) is not active
c:\w.GetApprovalRulesCommand - Checking rule ApprovalManagerToVP_OnLineItem started...
c:\w.GetApprovalRulesCommand - Satisfy all conditions for Quote => false
c:\w.GetApprovalRulesCommand - Checking rule ApprovalManagerToVP_OnLineItem ended. Result is false
c:\w.GetApprovalRulesCommand - Time Taken to run: 26
c:\w.GetApprovalGatesCommand - Rule satisfied: BusinessObject(uri=http://www.inmindcomputing.com/application/application-implementation.owl#ApprovalManagerToVP_OnQuote) Approval Gate: Optional.of(BusinessObject(uri=http://www.inmindcomputing.com/application/application-implementation.owl#QuoteApproval_MGR_TO_VP))
```

Below is an example of how the same quote can be seen from two different roles without submission.



The left diagram is the role hierarchy. The approval flow is defined as if quote discount is greater than 40, first approval gate will be sales Manager and then second approval gate will be sales VP.

The fig (1) shows if the quote is submitted by Sales Rep, it will go through all the steps as sales Rep's role is inferior role.

The fig (2) shows if the quote is submitted by Sales Manager, it will skip the sales Manager role as approval role is the same or inferior role of current login user and will directly goes to sales VP.

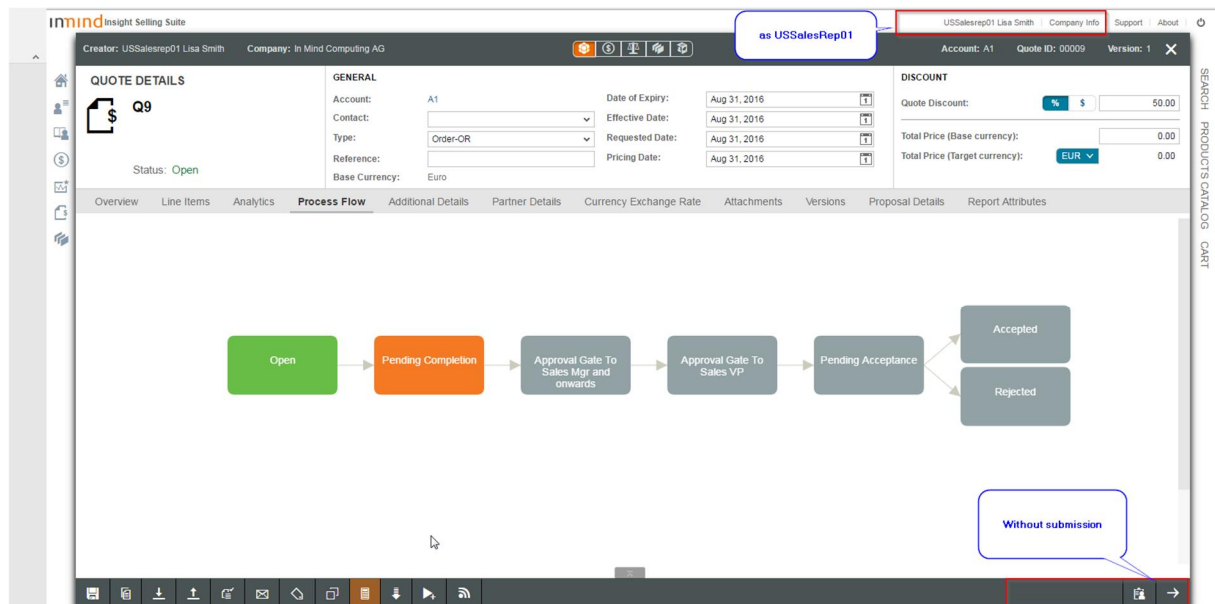


FIGURE 1 FROM SALES REP'S PERSPECTIVE

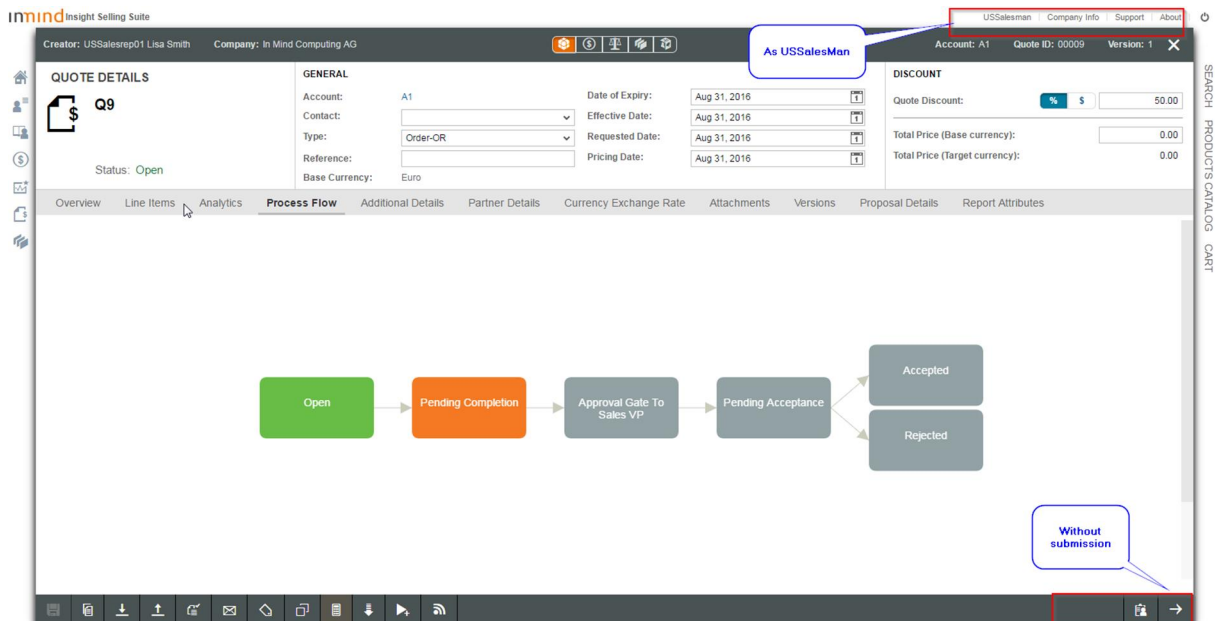


FIGURE 2 FROM SALES MAN'S PERSPECTIVE

SalesDocumentAllowedActions

SalesDocumentAllowedActions are determined programmatically by ISS?CPQ. There is no restriction such as only creator can submit quote. Below are the matrixes for quote and opportunity actions per salesDocumentStatus.

Explanation

APPR refers *Approved*

REJ refers *REJECTD*

Creator is the person who create the quote or opportunity.

Submitter is the person who submit the quote.

Router is the person who responsible to input necessary data.

Approver is the person whose task is either approve or reject the quote.

Quote Actions

QUOTE

Who You Are	Status	Actions							
		SUBMIT	ROUTE	DONE	APPR	REJ	ACCEPTED	REJECTED	WITHDRAW
Creator	OPEN	X	X						
Routers	OPEN		X	X					
Submitter	OPEN								
Approvers	OPEN								
Other Users	OPEN	X	X						

Who You Are	Status	Actions							
		SUBMIT	ROUTE	DONE	APPR	REJ	ACCEPTED	REJECTED	WITHDRAW
Creator	COMPLETED								
Routers	COMPLETED								
Submitter	COMPLETED								X
Approvers	COMPLETED				X	X			
Other Users	COMPLETED								

Who You Are	Status	Actions							
		SUBMIT	ROUTE	DONE	APPR	REJ	ACCEPT	REJECTED	WITHDRAW
Creator	APPROVED						X	X	
Routers	APPROVED								
Submitter	APPROVED						X	X	X
Approvers	APPROVED						X	X	
Other Users	APPROVED						X	X	

Who You Are	Status	Actions							
		SUBMIT	ROUTE	DONE	APPR	REJ	ACCEPT	REJECTED	WITHDRAW
Creator	ACCEPTED								
Routers	ACCEPTED								
Submitter	ACCEPTED								X
Approvers	ACCEPTED								
Other Users	ACCEPTED								

Who You Are	Status	Actions							
		SUBMIT	ROUTE	DONE	APPR	REJ	ACCEPT	REJECTED	WITHDRAW
Creator	REJECTED								
Routers	REJECTED								

Submitter	REJECTED								X
Approvers	REJECTED								
Other Users	REJECTED								

Opportunity Actions

Opportunity

Who You Are	Status	Actions				
		SUBMIT	WITHDRAW	QUOTE	WON	LOST
Creator	OPEN	X		X		X
Owner	OPEN	X		X		X

Who You Are	Status	Actions				
		SUBMIT	WITHDRAW	QUOTE	WON	LOST
Creator	COMPLETED		X	X		X
Owner	COMPLETED		X	X		X

Who You Are	Status	Actions				
		SUBMIT	WITHDRAW	QUOTE	WON	LOST
Creator	APPROVED		X	X	X	X
Owner	APPROVED		X	X	X	X

Who You Are	Status	Actions				
		SUBMIT	WITHDRAW	QUOTE	WON	LOST
Creator	LOST		X			
Owner	LOST		X			

Who You Are	Status	Actions				
		SUBMIT	WITHDRAW	QUOTE	WON	LOST
Creator	QUOTED		X		X	X
Owner	QUOTED		X		X	X

However, from UI Profile the actions can be hidden for a User Role and also sales document status.

Inbox

ISS inbox would not be showing both Quote and Opportunity except in the following final statuses: ACCPETED, REJECTED, SENTTOERP and WON.

Also a current logged in user will find all Quote and Opportunity in his/her inbox if he/she is a creator or submitter or owner of that document and the document is not in a final status.

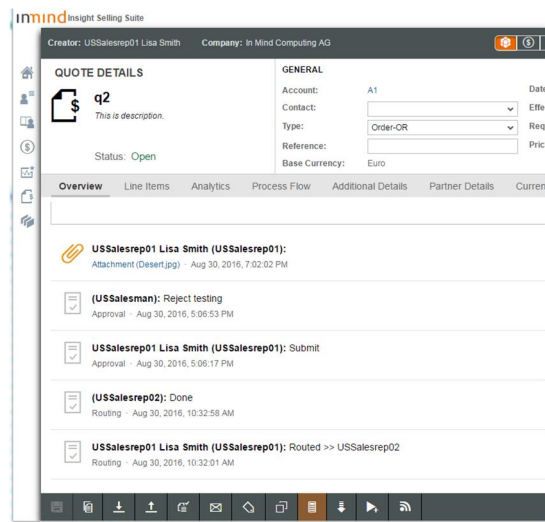
In addition to above the current logged will also see those document that need approval if the user is an approver and the Sales Document Status is COMPLETED.

For routing users, they will see the document if the Quote is routed to them and the Sales Document Status is OPEN.

In case the routed user has send the quote back (new action send) he/she can find the quote is sent items of the inbox, so long the Quote status is OPEN.

Approval and Routing Annotation

All approval and routing information such as who is the submitter, who is the router, whom the quote is routed to, who approved or rejected is also tracked by annotations on the quote.



Permission

Permission can be divided into two part: namely static permissions and dynamic restrictions. Static permissions are permissions that are loaded once the user logs into the system and are next applied on each REST call that user makes on ISS/CPQ. Dynamic restrictions are on the granted permissions. So for example if a user has a DeleteQuotePermission he or she can be restricted in performing this operation by using a restriction. Which is evaluated at run time.

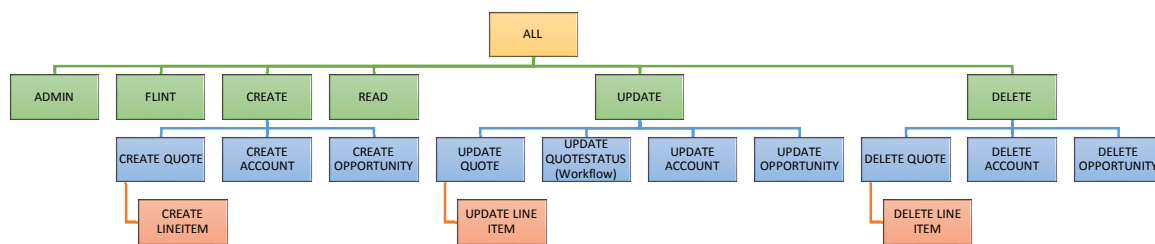
Static permission can be defined on Role level. Restrictions (Restriction Rules) can be defined either on a User level or on User Role level. Each restriction is defined on a specific permission. Each condition has business type on which the restriction will work. For 1608 release the supported business types for restrictions are Quote and Sales Item.

For 1608 release, all available permissions are defined in Application-schema as follows:

1. CREATEPERMISSION
2. CREATEQUOTEPERMISSION
3. CREATELINEITEMPERMISSION
4. CREATEACCOUNTPERMISSION
5. CREATEOPPORTUNITYPERMISSION
6. UPDATEPERMISSION
7. UPDATEQUOTEPERMISSION
8. UPDATEQUOTESTATUSPERMISSION
9. UPDATELINEITEMPERMISSION
10. UPDATEACCOUNTPERMISSION
11. UPDATEOPPORTUNITYPERMISSION

12. READPERMISSION
13. DELETEPERMISSION
14. DELETEQUOTEPERMISSION
15. DELETEDLINEITEMPERMISSION
16. DELETEACCOUNTPERMISSION
17. DELETEOPPORTUNITYPERMISSION
18. FLINTPERMISSION
19. ADMINPERMISSION
20. ALLPERMISSION

Permission Hierarchy



If a user gets a Higher level permission for example CREATEPERMISSION, he/she gets the permission to Create: Account, Quote etc, however if he possesses only CREATELINEITEMPERMISSION he/she can only create that object and not objects such as Account or Quote.

Below is how customizing looks like for a role with granted permission (CREATE, UPDATE, DELETE and READ) and applied restrictions (for delete quote restriction --- only creator can delete for the quote he or she created).

```
<!-- http://www.inmindcomputing.com/application/application-implementation.owl#ROLE_SALES_REP_US -->
```

```
<owl:NamedIndividual rdf:about="&ai;ROLE_SALES_REP_US">
```

```
<rdf:type rdf:resource="&as;Role"/>
```

```
<rdfs:label rdf:datatype="&xsd:string">SALES REP US</rdfs:label>
```

```
<as:roleAuthority rdf:datatype="&xsd:int">8</as:roleAuthority>
```

```
<ps:objectName rdf:datatype="&xsd:string">ROLE_SALES_REP_US</ps:objectName>
```

```
<as:hasPermission rdf:resource="&as;CREATEPERMISSION"/>
```

```
<as:hasPermission rdf:resource="&as;UPDATEPERMISSION"/>
```

```
<as:hasPermission rdf:resource="&as;DELETEPERMISSION"/>
```

```
<as:hasPermission rdf:resource="&as;READPERMISSION"/>
```

```
<as:hasRestrictionRule rdf:resource="&ai; CreatorRestrictionOnQuote"/>
```

Granted
Permission/s

Applied
Restriction/s

```

</owl:NamedIndividual>

<!-- http://www.inmindcomputing.com/application/application-implementation.owl#CreatorRestrictionOnQuote-->
<owl:NamedIndividual rdf:about="&ai; CreatorRestrictionOnQuote">
    <rdf:type rdf:resource="&as;RestrictionRule"/>
    <ps:objectName rdf:datatype="&xsd:string">Creator Restriction on Delete</ps:objectName>
    <as:hasCondition rdf:resource="&ai;ConditionCreator"/>
    <as:hasRuleStatus rdf:resource="&as;RULE_ACTIVE"/>
    <as:hasPermission rdf:resource="&as;DELETEPERMISSION"/>
</owl:NamedIndividual>

<!-- http://www.inmindcomputing.com/application/application-implementation.owl#ConditionCreator -->
<owl:NamedIndividual rdf:about="&ai;ConditionCreator">
    <rdf:type rdf:resource="&as;Condition"/>
    <ps:objectName rdf:datatype="&xsd:string">Current Login User is Creator</ps:objectName>
    <as:conditionAttribute rdf:datatype="&xsd:string">http://www.inmindcomputing.com/application/application-
schema.owl#containsCreator</as:conditionAttribute>
    <as:conditionBusinessType>http://www.inmindcomputing.com/application/application-
schema.owl#Quote</as:conditionBusinessType>
    <as:conditionValue rdf:datatype="&xsd:string">http://www.inmindcomputing.com/application/application-
schema.owl#VAR_LOGGED_IN_USER</as:conditionValue>
    <as:hasConditionType rdf:resource="&as;EQUALTO"/>
</owl:NamedIndividual>

```

Permissions are checked in each REST/Service call. However, restrictions are evaluated for Specific Operations only. For 1608 release these being:

Quote

1. Quote deletion
2. Quote update
3. Quote status update (workflow/approval actions)

SalesItem

4. SalesItem deletion

The restriction concept can further be applied in other cases and need be with prior requirement analysis.