

Rhodes Group

Documentation For Review

Doc Review Version: 6

04-14-2016

TABLE OF CONTENTS

CHAPTER 1 : Revised Insurance Tables and Order Module Rule

Insurance Based Test Exclusion Rule	2
Insurance Definitions	5
FO_INS	6
FO_INS_Tests	11
FO_InsAdds	14
FO_INS_Policy_Messages	15

CHAPTER 1 : Revised Insurance Tables and Order Module Rule

This document contains

- » revised insurance table descriptions for
 - » [FO_INS on page 6](#)
 - » [FO_INS_Tests on page 11](#)
 - » [FO_InsAdds on page 14](#)
 - » [FO_INS_Policy_Messages on page 15](#)
- » the Order Module Rule -- [Insurance Based Test Exclusion Rule on the next page](#)

There are several questions and concerns included in the Insurance Based Test Exclusion rule doc. In particular, I have concerns about the database construct which seems to provide an opportunity for problems with the rule definitions.



Note: These tables are revised based on the review comments received 03-28-16. There are additional **questions**.

Conventions used in these listings [any can be changed or removed]:

Yellow highlighted items are questions, or things that need more explanation.

Blue highlighted items are the changes that were made based on reviewer comments.

Purple highlighted items are reminders for future cleanup of the terms/writing/definitions. Just makes them easier to find in the doc.

Insurance Based Test Exclusion Rule

This rule set provides a way to mark a test code as orderable or not based on the primary insurance code and policy masks. These rules are configured in the `FO_INS_Tests` table. The rules are executed at the time a test is ordered on the order screen.

For example, Glucose can be excluded if it belongs to a particular insurance plan, or insurance code, and the policy begins with the characters "LH". The policy component of the rule may be omitted if it's not required.



Note: These rules only apply if the insurance code is a primary insurance code as defined by insurance rank in the `FO_Ins_Tests` table.

Per notes in the `FO_Ins_Tests` table field definitions, do we need to write better / more extensive guidelines for defining rules?

Concerned about the database table constructs – particularly the `InsRank` in the `FO_Ins_Tests` table. It could be in conflict with the rank defined in the `FO_Ins` table and create conflict between how the system operates and the defined rules. Also could create a situation where rule(s) are defined but irrelevant by other definitions or settings. Seems there is a need for clearer guidelines for defining rules. And some words for troubleshooting problems when rules are defined but don't appear to execute properly.

Can't find these two fields in my db schema, [supposedly in the `FO_Ins_Tests` table?]

- `TestCode-Priority`

- `TestCode`

To activate this rule set, you must:

1. assign the `evalinstest` permission [ABN Insurance Module] to a user group. See Assigning Permissions for more information.
2. assign the `insordrestrict` permission [ABN Insurance Module] to a user group. See Assigning Permissions for more information.
3. `FO_Tests.InsCodeForTestInsRules`: assign insurance codes that apply to a test for every test for which this rule set should apply. This field defines a list of insurance codes for which this test should be checked for rules. If all insurance codes apply to a test code and should be evaluated for these rules, enter [ALL].
4. For every insurance for which you want this rule set to apply, you must enter information, a complete record, into the `FO_INS_Tests` table. Each record in the table represents one rule.
 - » `InsCode`: Insurance code for this rule.
 - » `RuleSequence`: Orders the sequence of rule processing for this insurance code. Number the rules so that general rules are evaluated first, then specific rules. Rules can be contradictory so be sure that the sequence is correct. All rules will be evaluated before a final decision is made.
 - » `AllowTestOrder`: Indicates if the test code is allowed for this order and insurance.

- » **InsPolicyStartsWith**: Policy number starting characters. Rule applies only if policy number starts with these characters. Skip this entry if rule applies to all policy numbers for this insurance.
- » **InsRank**: Should always be 1. (does FP prevent entry of something different or just ignore the rule?, does FP message user if InsRank does not match the InsRank in the FO_Ins table – ie how the insurance can be used in the application?)
- » **Payable_Test**: Represents the test codes that are payable for this rule.
- » **NotPayable_Test**: Represents the test codes that are NOT payable for this rule.
- » **AllTestsPayable**: Determines if NO tests are payable and, if set to 1, overrides the Payable_Test and NotPayable_Test fields.
- » **NoTestsPayable**: Determines if ALL tests are payable and, if set to 1, overrides the Payable_Test and NotPayable_Test fields.
- » **TestCode-Priority**: Rule should be evaluated on this (these) test code and priority condition (s) only. (<<<< field not in my db schema, and where is the priority code set for the test code?)
- » **TestCode**: Rule should be evaluated on these test code(s) only. (<<<< not in my db schema)
- » **UserMessage**: The message displayed when this rule is the final rule and it applies. Often is a way to inform the user that this insurance does not allow this test to be ordered.

This example below shows how a cascade of rules can be used to customize insurance contract specific rules.

Ins Code	Rule Seq.	Allow Test Order	Ins Policy Starts With	INS Rank	Payable Test	Not Payable Test	All Tests Payable	No Tests Payable	User Message
<i>MCR</i>	1	0	NULL	1	NULL	NULL	0	1	This insurance cannot be used to order any tests
<i>MCR</i>	2	1	12	1	NA K CL	NULL	0	0	NULL
<i>MCR</i>	3	0	13	1	NULL	CL-S	0	0	This Test Cannot be paid for Policies starting with 12

The example above represents three records in the FO_INS_Tests table for Insurance Code "MCR". Each record is sequentially ordered using the RuleSequence field which determines the logical flow of rule evaluation.

The first record represents the default rule for the tests in this insurance code. The default rule is defined to make all tests not allowed, because only certain tests are allowed to be ordered. The user message field contains the message that users will see if this rule applies as the final decision.

The second and third rules are configured to allow specific tests to be ordered under specific conditions.

The second rule allows the test codes "NA", "K" and "CL" to be ordered if the insurance policy starts with "12". There is no user message defined for display if this rule applies as the final decision.

The third rule does not allow for the test code "CL" to be ordered if paired with the priority code "S" and the insurance policy starts with "12". The user message field contains the message that users will see if this rule applies as the final decision.

See FO_Tests for more information about field definitions and allowed values. (Not included in Review Demo 6)

See FO_INS_Tests on page 11 for more information about field definitions and allowed values.

Insurance Definitions

For review, revised table descriptions and field name notes.

- » [FO_INS on the next page](#)
- » [FO_INS_Tests on page 11](#)
- » [FO_InsAdds on page 14](#)
- » [FO_INS_Policy_Messages on page 15](#)

FO_INS

Field	Type	Notes
Ins_ID	int	ID field, a counter. Unique
InsCode	varchar (50)	Enter a unique insurance code.
Ins_Name	varchar (50)	Name for this insurance.
Policy_Mask	varchar (50)	Mask character value(s) for the policy number entered. Multidimensional pipe delimited value, with leading and trailing pipes. Enter allowed value combinations separated by pipes. Allowed values: * = any character # = numeric character @ = Text character IE: @@##### ##### Allows for the definition of multiple edit masks for each insurance code.
Group_Mask	varchar (50)	Edit mask for the group number entered on the demographic screen. Enter an edit mask using allowed values. Allowed values: # = numeric character @ = Text character
Type	varchar (5)	Informational Site-specific use, not used by FAST
DisplayKey	varchar (80)	Informational Site-specific use, not used by FAST
UseABN	int	Indicates if ABN and medical necessity rules will apply when this insurance is selected for a patient on the demographic screen. Enter 0, 1, or skip this entry 0 or skip this entry = no medical necessity rules apply 1 = medical necessity rules apply

Field	Type	Notes
UseMSP	int	Indicates if MSP rules apply. Enter 0, 1, or skip this entry 0 or skip this entry = no MSP rules apply 1 = MSP rules apply
UseReferral	int	Indicates if this insurance code requires a referral ID. Enter 0, 1, or skip this entry 0 or skip this entry = no referral required 1 = referral is required Causes the system to activate the referral screen in an order.
ReviewCarrier	varchar (50)	Carrier code this insurance code is associated with for medical necessity rules. Reflected in the FO_CPTReview table.
ReviewCPTs	varchar (8000)	Defines CPT codes checked for medical necessity rules for this insurance code. Multidimensional pipe delimited value, with leading and trailing pipes. Enter valid CPT codes separated by pipes. The system uses this list to speed medical necessity rules by only checking the CPT codes in the list for cross reference in the FO_CPTReview table.
RequiredFields	varchar (300)	Defines list of insurance related fields that are required fields, if present. Enter single characters from the allowed values. Allowed values: P = policy number, also makes the relationship field mandatory G = insurance group number I = ICD / Diagnosis IE: PG makes the policy, relationship and group fields required fields
UserName	varchar (50)	Name of user who last edited the record. Informational
CreateDate	datetime	Date record was created. Informational
FINSystemChargeTo	varchar (50)	Default financial system charge to. Enter a financial system charge to.

Field	Type	Notes
BillPatient	bit	Bit flag to determine if the system assumes that a patient may be billed. Enter 0 or 1 0 = do not intend to bill patient 1 = bill patient If the patient should be billed, a transaction should be sent to the FIN system.
EditDate	datetime	Date record was last edited. Informational
AutoVerify	varchar (50)	Indicates to the system that when this insurance is picked as primary on the patient demographic screen, the system may auto verify the patient. Enter 0, 1, or skip this entry 0 or skip this entry = do not auto verify patient 1 = auto verify patient User must have permission autoverifyins1 for system to auto verify.
Ins_Rank	varchar (50)	Determines the ranking of the insurance when a search for insurances is performed on the patient demographic screen. Skip this entry = any spot Enter an allowed value. Allowed values: PRIMARY = for primary spot only SECONDARY = for secondary and tertiary spot If SECONDARY, the system only allows the user to use it in the secondary or tertiary spots of the patient demographic screen.
Ins_User_Assignment	varchar (255)	Defines the user group(s) who can search for this insurance code. Multidimensional pipe delimited value, with leading and trailing pipes. Enter valid user group codes separated by pipes.
ClientsPerfLabAssignment	varchar (500)	Enter [ALL] Do NOT change without Rhodesapproval. This is not the same as the 'C-' portion of performing lab rules definition.

Field	Type	Notes
MedNecessityFormCodes	varchar (255)	<p>Defines a list of medical necessity (ABN) related forms available when the ABN button on the order screen is clicked.</p> <p>Skip this field to default to ABNENGLISH ABNSPANISH </p> <p>Multidimensional pipe delimited value, with leading and trailing pipes.</p> <p>Enter allowed values separated by pipes.</p> <p>Allowed values:</p> <p>ABNENGLISH</p> <p>ABNSPANISH</p> <p>BLANKABNENGLISH</p> <p>BLANKABNSPANISH</p> <p>GENERICENGLISH</p> <p>GENERICSPANISH</p> <p>BLANKGENERICENGLISH</p> <p>BLANKGENERICSPANISH</p> <p>To use custom forms, enter 'FC =' followed by FO_MedicalNecessity_Form_Definition.FormCode</p>
LabPopup	varchar (80)	Text message presented to a lab user when this insurance code is selected on the patient demographic screen.
ClientPopup	varchar (80)	Text message that is presented to a client user when this insurance code is selected on the patient demographic screen.
NotPrimaryIfPolicyStartsWith	varchar (50)	<p>Defines policy number starting character(s) that cause a warning to be displayed.</p> <p>The warning is displayed if an insurance code is selected and the policy number entered starts with this (these) character(s).</p> <p>Enter the policy number starting character(s).</p>
DiagCodingMethod	varchar (50)	<p>Enter valid diagnosis coding method, 9 or 10, for this insurance.</p> <p>Value is referenced for Medical Necessity Rules.</p>
BillFeeSchedule	varchar (80)	<p>Default fee schedule code for this insurance when designated as the primary insurance.</p> <p>Enter a valid fee schedule code.</p>
BillingOption	varchar (80)	<p>Default billing option code for the default fee schedule code for this insurance when designated as the primary insurance.</p> <p>Enter a valid billing option code.</p>

Field	Type	Notes
HRAFormCode	varchar (50)	Default form code to use for health forms for this insurance. Enter a valid form code. >>> NEED to Check Definition <<<
HRAFormFrequencyInDays	bigint	Number of days between HRA form collections. Enter an integer as the number of days. >>> NEED to Check Definition <<<

Definitions of valid MedNecessityFormCodes for the Medicare ABN form:

ABNENGLISH = English form

BLANKABNENGLISH = Blank English form

ABNSPANISH = Spanish form

BLANKABNSPANISH = Blank Spanish form

GENERICENGLISH = Generic English form

BLANKGENERICENGLISH = Blank Generic English form

GENERICSPANISH = Generic Spanish form

BLANKGENERICSPANISH = Blank Generic Spanish form

FO_INS_Tests

This table contains the rules for test restrictions based on insurance related data.

Each record in this table represents a rule. Each rule, or instruction, is assigned to an insurance code and is executed for each test ordered at the time a user enters the test on the order screen.

The decision for processing these rules is the question "Is this test payable by this insurance and should the user be allowed to order the test?"

The system may process multiple rules for each test or condition. It is important that the rules are prioritized appropriately using the rule sequence so exception rules are processed before generic rules.

Field	Type	Notes
InsTests_ID	int	ID field, a counter. Unique
InsCode	varchar (50)	Insurance code for this rule. Enter a valid insurance code.
RuleSequence	int	Used to order the sequence of rule processing for this insurance code. Enter an integer. Remember that exception rules need to be processed before generic rules.
AllowTestOrder	int	Determines if test can be ordered for this insurance. Enter 0 or 1 0 = do not allow test order 1 = allow test order (does it default to 0 if skip this entry?)
InsPolicyStartsWith	varchar (50)	Policy number starting characters. Rule applies only if policy number starts with these characters. Enter a string of characters. Skip this entry if rule applies to all policy numbers for this insurance. Limits the tests a user can order for a specific insurance to only policy numbers containing these characters. Used in conjunction with the AllowTestOrder field. (if AllowTestOrder=0 and this field has 'ABC', does that mean that tests can be ordered for any policy number that does not include 'ABC' -- so a rule could be made like this and another rule with AllowTestOrder=1 and no value in this field, thus only excluding tests for policies that contain 'ABC'?)

Field	Type	Notes
INSRank	int	<p>Insurance rank.</p> <p>Always enter 1</p> <p>1 = primary insurance, this rule will apply</p> <p>This rule will NOT apply if this entry is skipped or any other value is entered.</p> <p>(why wouldn't this just be referential to the InsRank in FO_Ins.? Why enter a rule that doesn't apply? This construct seems to allow an admin to enter a value contradictory to how the insurance is used in for an order. IE enter a 2 here so no rules apply and the insurance is primary for an order. Actually so many ways to be contradictory to the insurance usage on an order.) Do we need to come up with better guidelines for writing rules?</p>
Payable_Test	varchar (4000)	<p>Defines the list of test codes that will be paid by this insurance.</p> <p>Multidimensional pipe delimited value, with leading and trailing pipes.</p> <p>Enter valid test codes separated by pipes.</p> <p>Skip this entry ignore this component for the rule decision.</p>
NotPayable_Test	varchar (4000)	<p>Defines the list of test codes that will NOT be paid by this insurance.</p> <p>Multidimensional pipe delimited value, with leading and trailing pipes.</p> <p>Enter valid test codes separated by pipes.</p> <p>Skip this entry ignore this component for the rule decision.</p>
AllTestsPayable	int	<p>Determines if ALL tests are payable.</p> <p>Enter 0 or 1</p> <p>0 = no</p> <p>1 = yes, all tests are payable</p> <p>Overrides the Payable_Test and NotPayable_Test list fields if set to 1.</p>
NoTestsPayable	int	<p>Determines if NO tests are payable.</p> <p>Enter 0 or 1</p> <p>0 = no, all or some tests are payable</p> <p>1 = yes, all tests are NOT payable</p> <p>Overrides the Payable_Test and NotPayable_Test list fields if set to 1.</p>
UserMessage	varchar (80)	Text message displayed if this rule is the decision point.

Need to map out the table relationships.]

FO_InsAdds

This table contains the address for each insurance code.

Each insurance code has one address.

Field	Type	Notes
InsAdd_ID	int	ID field, a counter. Unique
InsCode	varchar(50)	Insurance code. Enter a valid insurance code.
Ins_Name	varchar(50)	Insurance name for address.
Add1	varchar(80)	First address line.
Add2	varchar(80)	Second address line.
City	varchar(50)	City
St	varchar(50)	State
ZipCode	varchar(50)	Zip code
Tel	varchar(50)	Phone number
DisplayKey	varchar(100)	Informational
UserName	varchar(50)	Name of user who created the record. Informational
CreateDate	datetime	Date record was created. Informational
EditDate	datetime	Date record was last edited. Informational

FO_INS_Policy_Messages

This table contains definitions of insurance and policy specific messages.

The `insevalpolmsg` permission [Registration Module] must be assigned to enable this functionality. The message will display as the user exits the policy number field in the application if the insurance code and content of the policy number matches the definitions.

There can be more than one message per insurance.

If multiple values need to be compared with an "OR" statement for one insurance code, then more than one record should be created for the same insurance code. Each record will be evaluated separately and if the condition matches then that message will be displayed. When more than one message applies, the messages to display will be combined together into one message prompt.

Field	Type	Notes
Ins_PolMsg_ID	bigint	ID field, a counter. Unique
InsCode	varchar (50)	Insurance code. Enter a valid insurance code.
PolicyContainsText	varchar (50)	String(s) of text to be search for in the policy number. This is a 'policy number contains' search. Multidimensional pipe delimited value, with leading and trailing pipes. Enter the search strings, separated by pipes. For this message to display, every search string must be present. IE: an 455 Policy AN123455 will show message Policy AN454123 will not show message
UserMessage	varchar (255)	Enter the message to display if all pipe delimited strings from PolicyContainsText field above are present in the policy number.
CreateDate	datetime	Date record was created. Informational
CreateUser	varchar (50)	User who created record. Informational
EditDate	datetime	Date record was last edited. Informational

Field	Type	Notes
EditUser	varchar (50)	User who last edited record. Informational