33M1-HF1 Topics	Description	Note/TestCase
Fraud Tagging for non-card	Provide the ability to extend Fraud Tagging beyond cards to non-card entities.	No test case available.
Model signature can grow and shrink over versions... should do so without data loss	Currently, the USC and J-OSE add the Zxx segments to the message before it is passed to the OSE for scoring. The segment lengths are deteremined from the model metadata and each segment is allocated the size fo the expected returned data. If the meh record is old, it could be larger than the current segment size and will get truncated.	The fix was verified by testing at champion level, then upgraded to challenger followed by downgrading to champion again. Segments matched and were not truncated.
Support multiple Z00 signature layouts for an entity	CBS model will use Z00 segment (for example) to store account related information. However, the list of variables to store for a credit card account may be very different than the list of variables to store for a debit account. And the 2 lists may be of different sizes. Currently the model publishes a single list of variables for a Znn segment and the OSE and USC share metadata about how much space to allocate in the transaction buffer to how the data. In this case the OSE needs to understand there are several layouts for the Z00 segment and share the max size with the USC.	No test case available.
Rule Domain support	See Fraud Management documentation.	No test case available.
Support data driven alert types in Fraud Tagging	See Fraud Management documentation.	No test case available.
Modify the SOR purge job to support confirmed fraud re-init	A bank customer with an online userid and password is not required to change their online userid when their online account has been compromised and used for fraudulent purposes. Rather the bank customer is asked to change their online password. If in the future the customer's online account is compromised again, then the account will be placed in an alerted state and a bank analyst will contact the customer and ask them to change their password again. Similarly, not all fraud which occurs on a debit card is grounds for re-issuing the debit card. In the case of ATM fraud, the card PIN may be changed and the customer may continue to use their card for future purchases. However, if the physical card is lost, then the card may be reissued with a new card number.	No test case available.
Make estimation use scores 2-4 when deciding a transaction is fraud (for non-card fraud tagging)	Estimation uses RHX_SCORE_1_TAG (and other fields) when deciding whether a transaction is fraud; it ignores values in RHX_SCORE_2_TAG, RHX_SCORE_3_TAG, and RHX_SCORE_4_TAG. To allow tagging data in score fields 2 thru 4 to affect the results of estimation, it should examine them as well RHX_SCORE_1_TAG.	Run and estimation against transactions that have confirmed fraud values set in RHX_SCORE_*_TAG fields other than RHX_SCORE_1_TAG and verify that they are considered fraud appropriately. The values for detecting confirmed fraud are: RHX_SCORE_*_TAG not in (' ','0','1','3') means fraud status IGNORE. RHX_SCORE_*_TAF in ('1','3') means fraud status FRAUD. Other values means fraud status LEGIT.
Lookup list changes are not reflected in lstupdt_user and lstupdt_timestamp	In the SQL statements in RuleService.xml that update FCM_LOOKUP_LIST (for DB2 and Oracle), we do not update lstupdt_user and lstupdt_timestamp. The fix is to pass the user name to the SQL call and fill it and the current timestamp in as part of the update SQL. The column on the screen labelled "Last Uploaded" for lookup lists and content files is always blank. This is because getting the actual last-uploaded timestamp involves opening each content file and that is too resource-intensive for a long list of files or lookup lists, so we don't do it. The fix is to remove this column from the list of content files and to change it to "Last Updated" in the list of lookup lists (the last-updated timestamp of the lookup list definition is readily available in the lookup list definition in FCM_LOOKUP_LIST_DEFINITION). We added function libref to aid in the testing of these changes.	Create a lookup list and note the lstupdt_user and lstupdt_timestamp of the row created in FCM_LOOKUP_LIST_DEFINITION. Update the description of the lookup list from another user ID and check that the row in FCM_LOOKUP_LIST_DEFINITION was updated with new lstupdt_user and lstupdt_timestamp values and that the display of the lookup list shows the new 'Last Updated' value. Delete the lookup list from another user ID and check that the row in FCM_LOOKUP_LIST_DEFINITION was updated with new lstupdt_user and lstupdt_timestamp values. Examine the list of content files to see that the 'Last Uploaded' file is gone but the 'Last Uploaded' information is still in the display of a single content file (after double-clicking on a single content file name).
Support Backward compatibility of entity field selection in Fraud Tagging	In order to minimize effort in upgrading, this will modify the current fraud tagging to use a coalesce to find transactions with a card number in the FRH_CCMF, FRH_CSMF, FRH_CCCA, and/or FRH_CSCA tables.	No test case available.

Cross site scripting error on login page	The userid is being reflected into the webpage without sanitizing it first. This will leave the webapp page open to XSS attacks.	Used application scanner to verify the exploit was closed.
Cross site scripting error on r_ruleEstimates.do (see report)	The userid was reflected from the incoming HTTP request arguments to the outgoing JSP and subsequent HTML. The JSP code called excapeXML on the userid	Used application scanner to verify the exploit was closed.
Estimation status remains at Stopped even after checking results or canceling it	When an estimation run finished, the SAS program sets the row in FCM_ESTIMATE to job_status=Stopped. When the webapp displays a list of estimations, this status (Stopped) appears until the user requests either the SAS log or the results (report) of the estimation run, at which point, the webapp code scans the log for errors and is supposed to set the status to Completed or Failed. This reset of the status is not occurring as it should. Also when an estimation is canceled, the status does not change to "Canceled". This changing of the status for estimation is not taking effect because in file /com/sas/finserv/creditfraud/rules/creditfraud_rules-services.xml, the name of the method in class RulesServices that makes this update to the status does not appear.	The fix is to rename the method to something that looks more like it is doing an update (previously was 'getEstimate' - not suggestive of an update!) and add that name to /com/sas/finserv/creditfraud/rules/creditfraud_rules-services.xml. For canceling the estimation, the name of the cancelEstimation (cancel*) should be added to reditfraud_rules-services.xml.
Action_close_alert	Customer requirement: The default behavior of the %action_close_alert macro. In the initial design, the invocation of the macro would close any open alerts with a status of either Confirmed Fraud or Verified OK. After reviewing with the customer, it was decided that the macro should only close Confirmed Fraud alerts.	Create an Alert, specifically a Card Alert on a specific card num. Next, from webapp, mark alert verified OK. Create rule that does D14 for alert type and alert value (e.g. Card Alert and card num) Push transaction(s) to cause rule to run Verify alert isn't closed Change status of Alert to Confirmed Fraud Push transaction(s) to cause rule to run Last, verify the Alert is closed.
Job 4019 needs work in defining what is a unique record and what change info is to be captured	The primary key has changed from USERVAR_FIELD_NAME+TENANT_MULTI_ORG_ID to USERVAR_FIELD_NAME+TENANT_MULTI_ORG_ID+BEGIN_BUILD_ID and the type of update for the table has changed from SCD2 to SCD1 so entries in FRX_SCD_COLUMN for this table will be removed (new ddl will also remove the SCD2 columns from the db table). macro fsmrh_frh_user_var_dim.sas has been updated to use the new primary key definition and no longer call macro fsmrh_dimension_mgt.	Data in SOR fsx_user_variable was not being processed since there was already data in the RH for the tenant_multi_org_id + uservar_fiield_name. After making the changes noted in 1 these records were processed and deleted from the SOR trigger table and the RH FRH_USER_VARIABLE_DIM table was updated with over writes for existing records and with inserts for new records.
Reveals DOM Based Cross-Site Scripting problem	A Rule ID value from the parent form is being used to create a default name without sanitizing the value being received and could cause an invalid file name which would generate an error.	The value being passed is sanitized as having numeric only or no value is passed if non-numeric exists and the file is created properly.
UVR segment size truncated in USC USDO screen	In the VDB display of the USDO screen, the user variable segment size is truncated if it is greater than five digits.	The display field was expanded to show the maximum segment size
recovery job abort when retrun code from job 4017 submittal is gt 1	When two running jobs try to submit job 4017 at nearly the same time the return code from the submittal process may not be 0 however the job itself may run successfully for one of the submitalls. The fix is to not abort a job because of the return code but to issue a WARNING in the SAS log. Then subsequent logic will check the FRX_JOB_RUN table and see what the status of job's 4017 latest run status is[a higher job_run_id than what was stored for the submitting job before the submittal code was executed]. If the status indicates	Remove the logic to abort based on call to submit job 4017. Load some records copied from FSX_RULE_PACKAGE back into the SOR table. Schedule a recovery job 3001 and job 4024 to run at the top of the hour. One submitted job 4017 and the other had the file contention error present in its script log. All three jobs ran successfully.
Rules editor can send a rule request approval mail to all users.	Rules Studio needs to isolate rules by business unit, so only the Rules Writers/Editor in the same Business Unit should be displayed in the user drop down list for the "Request Approval" window.	In the "Request Approval" window, only the Rules Writers/Editor in the same Business Unit is displayed for the drop down list.
User who has 'Console Administrator' role only can deploy rules.	A user with 'Console Administrator' role but not 'Rules Administrator' role could deploy rules.	If a user only has 'Console Administrator' role assigned to them, they no longer have the ability to deploy rules.
Multi Score system: Remove the high water mark score on the analyst workstation	Currently the analyst workstation shows the high water mark score on the alert screen. Recommended that a better value to display would be alert creation date	alert screen no longer displays the high water mark, but instead will show the alert creation date.
No response when retriving transaction alerts from alert transaction grid	For data-driven alert types, alert type labels and images are defined in fcm_alert_type table, and retrieved into a separate resource bundle object. Then a jsp page will use the resource bundle when it needs to display an alert label and image. The definition of the resource bundle object was missing in the jsp page that returns CF alerts associated to a transaction. This led to jsp render exception.	1. Log in to SFM as an analyst 2. Service a card alert (with transactions), check the CF checkbox of a transaction, then statusing the card as Confirmed Fraud. 3. Now open the same card alert, switch it to view the customer from linked entity link. 4. Go to the same transaction in the transaction grid, it should have a CF flag displays. 5. Click on the CF flag to retrieve CF alerts (including the card alert) from this transaction. A spinning image appears, and waiting for the results
Estimation fire count and detail row list are inconsistent if %ACTION_FIRE is used	Trasactions fired only by %ACTION_FIRE do not appear in the list of detail rows although the fired value is correct.	The number of rows displayed when you click on Transactions:…fired in the estimation summary results will match the number shown even when the rules being estimated used %ACTION_FIRE to fire rules.

Import of queue or variable rules into tenant installation creates invalid rule types	When you import a variable rule or queue rule into a tenant installation, the rule types are retained. This causes problems because we allow only Auth rules in tenant installations. The variable rules are only mildly upsetting - they do not display as rule type Variable (since we've suppressed the display of rule type), but they still execute in front of all the Auth rules, so their execution order is counter-intuitive. Queue rules cause a more serious problem since they make the AGS attempt to start up an AGE, which fails immediately (multiorg is wrong plus possibly other problems - I have not investigated further since this is not a supported configuration). The fix is for import to change the rule type from Variable or Queue to Auth and put a warning message out for the rule (as we do when we assume a multiorg for a rule with an unrecognized multiorg).	Import a queue and a variable rule into a multi-tenant systme. They should be converted to Auth rules and a warning message should be generated in the rule.
%SYSOUT NOT GETTING RESOLVED	During installation, the %SYSOUT does not get resolved when adding DDNAME OUTRULES to the SASOSE JCL.	The correct value is being passed to the SYSOUT parameter.
After any user variable operation, Edit Segment button appears even if it should be View Segment	In the Rules tab task List Variables, if doing any operation on a variable (new, edit, view), after you return to the list of variables, the Edit segment button appears instead of the View segment button even when the segment is "read only".	Performing new, edit, etc. on user variables in a segment created by another business unit and the Edit segment button should not be displayed.
BPI changes to prevent the RFC1323 zero window problem	The problem occurred due to the window scaling algorthm in TCP when the rfc1323 feature is enabled. RFC1323 was intended for transmission of very large files or for use in communcation networks. It is well known that this feature causes adverse throughput for request/response applications like Fraud Management. If you want to do some more reading on the problem, search for "zero window" or "silly window syndrome".	No test case available.