FUJ00171899

Document Title:	EUM DESIGN NOTE: Concurrent Login Changes
Document Reference:	
Release:	
Abstract:	EUM DESIGN NOTE: Concurrent Login Changes
Document Status:	DRAFT
Author & Dept:	Andy Thomas

FUJ00171899

0 Document Control

0.1 Table of Contents

FUJ00171899

0.2 Document History

Only integer versions are authorised for development.

Version No.	Date	Summary of Changes and Reason for Issue	Associated Change CP, CCN or PEAK Reference
0.1	29/11/2017	Initial version	
0.2	04/12/2017	Internal review	
0.3	04/12/2017	Further updates	
0.4	04/12/2017	Added BRD cross reference	
0.5	04/12/2017	Amended for Gareth Seemungal comments.	
0.6	07/12/2017	Clarification when lock button is active Assumption that same HUID can log in multiple times.	
0.7	12/12/2017	Messages are auditable. Lock button ref data needs to be modified so a new application context is not started when it is pressed.	

FUJ00171899

1 Introduction

This document is to capture the high level design for the Enhanced User Management Project. This document should be read in conjunction with the Atos document "Enhanced User Management Horizon Counter Enhancement Business Requirements Specification" (BRS), section 4 titled "Concurrent Login Functional Requirements"

The BRS document describes the change covered by this document thus:

Concurrent login functionality shall enable the user to switch between multiple Horizon counters. This is important to the Horizon user as some transactions are time consuming to complete. Therefore, if they wish to serve a customer on a different counter, they shall have the functionality to switch from the current task (on Counter A) to serve the customer (on counter B) and vice versa.

REQ/CUS/BRS/3488 Note that there are other documents/CP that have been issued that covered additional changes described in the BRS document.

1.1 Target Audience

This document is intended for the following groups to aid them in producing the impacts for these changes

- Fujitsu Design
- Fujitsu Host Developers
- Fujitsu BAL & Counter Developers

The design detailed is not complete and will require further work before coding of the changes commences.

Impacts are not required from Test as these will be covered by a separate CP, however development and integration must impact for the code changes/CIT testing/documentation/baseline changes and release into test process.

1.2 Overview

The design detailed within this document is to support the change to the concurrent login on both the HNG-X and HNG-A counters within certain constraints. No distinction is made between these counter types in the design. As the system currently stands a user (when resolved to a POID) may only login into one counter at a time. The change allows this single user to log in multiple times with in certain constraints defined in this design. The design assumes that either a different HUID or the same HUID can be supplied for these sessions [EUM-CE-CL-70].

A new concept of "data centre awareness of locking" is introduced which essentially means that the data centre has become aware of counters that are in the lock status. Note that this status in the data centre should be held separate from the session status to avoid changes the complex state changes for a session as it goes through the lifecycle of login, logout, forced lockout, recovery etc. The new status only comes into play during the login process or unlock process and is not checked at any other times. It will be set whenever the counter is locked (e.g. manually or via the inactivity timers) and unset when the counter is unlocked. New (auditable) messages to the data centre are required to perform this setting/unsettling.

Throughout this document the term locked/locking will imply this new "data centre awareness of locking".

This design does not alter the restrictions/exemptions for Global users with respect to concurrent login nor where the case where the reference data implies the HUID/POID mappings are not enforced (EUM has not been enabled in the branch). [i.e. this change only affects POID-associated users – pre ENUM user]

Further the interaction with the first login message processes (e.g. resetting of passwords etc) are not to be modified.

FUJ00171899

2 Solution Design

In the current system the user has two options when logging in (counter A) and the associated POID is found to be in use at another counter (counter B). They may

1) Elect to abandon the current login attempt on counter A.
2) Force the other session on counter B to become invalid.

In case (2) no further communications to the data centre will be allowed from counter B and the counter will force a logout when next used. The normal recovery process must then be followed on this counter during the next login.

In the new system these options will still be presented when the session on counter B is not locked but a different message [EUM-CE-CL-10, EUM-CE-CL-40] will be displayed requesting the user to lock counter B manually and retry the login on A. They can then select option (1) and retry the login. This retry is essential to ensure the counter and data centre are in agreement when the user is next logged in.

However if the counter B is in a locked status these options will not be presented and the user will be allowed to continue the login process [EUM-CE-CL-40, EUM-CE-CL-70].

Any number of counters can exists in the lock status however there can only ever be one counter that is in the unlocked status (for a given POID) on which the user can perform transactions.

Further it is possible that there is a mixture of one unlocked and multiple locked counters. In this case when the user chooses the second option to invalidate the session then the sessions on the non-locked counters will invalidated. The locked sessions will remain active, but in the locked status. It is suggested that the wording on the concurrent login screen is changed to make this clear. This change is via Post Office owned reference data.

The ability to fail active sessions on other counters will be provided for the process of unlocking which will now prompt the user in a similar manner to the screen that appears during the process that detects a concurrent login i.e. ask the user if the active sessions are required to be "FAILED". This brings the processes of logging in for the first time and unlocking a counter closer together in terms of actions that can be performed when unlocked sessions are detected in the data centre.

2.1 Events

The requirements specify events to be created for locking and unlocking a terminal. These events are not new to the design and already raised in the system as identified by CP6678 and require no additional work.

The events are:-

2.1.1.1 Event ID 24 Position Locked

This event records locking of a counter position due to either the user selecting temporary lock or because of a system enforced temporary lock applied due to inactivity. The event is recorded at the counter by the re-login code t is not an immediate audit event, and so will only be recorded when the next auditable message is sent to the BAL.

2.1.1.2 Event ID 25 Position Unlocked

Records a successful re-login by the existing user after a temporary lock. The event is recorded at the counter by the re-login code. It is not an immediate audit event, and so will only be recorded when the next auditable message is sent to the BAL.

2.1.1.3 Event ID 26 Unlock Failed

Records the fact that the attempt to re-login to a counter after a temporary lock has been applied has failed. This can be for one of the following reasons:

1. The user has exceeded the number of attempts that they get to supply valid credentials on the re-login.
2. The user account is locked.
3. The user account is disabled.
4. The user supplies invalid credentials.

FUJ00171899

All these reasons apply to either the existing user attempting to re-login or a new user attempting to login, apart from 3 which only occurs if this is a new user attempting to login, rather than the existing user.

2.1.2 Implications on this Design

It must be understood at what time these events are raised and the implications this has on this design.

As described above these are "deferred" events which mean they will only be sent to the BAL on the *next auditable message*. In the case of both lock (event ID 24) and unlock (event ID 25) this could be some time after the actual event happened. Indeed it may be possible for both events to arrive at the BAL at the same time in the same auditable message even though the counter position has been locked for some time.

This design is NOT proposing to change this behaviour. The events will continue to be raised and recorded as now (these are existing events?! This wasn't clear). The original intent and reasoning for these events still applies i.e. they are not judged to sufficiently important to warrant their own auditable message so are deferred to sometime later. They will be recorded, but at any given time a single snapshot of the events table may not give a 100% accurate view of locked counters which will be given by examining the new lock status of a session.

The data centre (BAL) now requires to be informed immediately of the locking and unlocking of a counter to correctly assess the locked status of the counter. New BAL services will be required to perform this action as described below. Whilst it may seem appropriate to change the events with IDs 24/25 to be logged within these new services they do not need to be, hence the impacts should NOT include modification of when these events are raised, recorded and testing of the events. The events provided a historical record of the locked status, whilst the new data stored in the BRDB give a real time view. These don't need to be consistent at any one point in time, but of course over a longer analysis of the audit trace they will be.

2.2 BRDB Changes

A new column with the name SESSION_LOCKED_TIMESTAMP will be added to the BRDB_BRANCH_USER_SESSIONS to indicate that a user has locked this counter position. This will be of the type TIMESTAMP(4) and NULLs are allowed. The default value will be NULL when a row is created in the BRDB_BRANCH_USER_SESSIONS table. Note the actual value recorded in this table will be of the same format/time base as SESSION_START_TIMESTAMP and SESSION_END_TIMESTAMP in the same table i.e. UTC. See PC0225265 for clarification. The life cycle of this column value is

1) Initial creation of row set to zero
2) When new BAL service "LockCounter" is called the UTC timestamp value in this counter message is used to set the value of SESSION_LOCKED_TIMESTAMP – same way that SESSION_START_TIMESTAMP has its value set [EUM-CE-CL-30, EUM-CE-CL-40].
3) When new BAL service "UnLockCounter" is called the value is set to NULL. This catches the case where the user "unlocks" the counter by entering there username and password. It should also be called when a different user unlocks the counter. Note this action can fail since there may already be an active session on another counter [EUM-CE-CL-50].

2.2.1 Document changes

Following document(s) need changing.

DEV/APP/LLD/0199: Schema Definition For Branch Database, Standby Branch Database And Branch Support System.

Host ERWin schema diagram.

2.3 BAL Changes

The BAL changes are limited to three areas.

1) New LockCounter service message (GenericBRDBModification/LockCounter) [EUM-CE-CL-30].
2) New UnLockCounter service message (UnLockCounter) [EUM-CE-CL-50]

FUJ00171899

3) Changes to the concurrent login checks [EUM-CE-CL-10].

The first change will be implemented by using the GenericBRDBModification services since this service is only required to change a status field SESSION_LOCKED_TIMESTAMP in the BRDB_BRANCH_USER_SESSIONS to update the value to the current UTC time.

The SQL created for LockCounter should use the header.TimeSent[Timestamp] field from the request to set the value on LockCounter and set to NULL when UnLockCounter is called. The header.TokenId[String] and header.FadHash[Int] should be used to identify the row to modify which should also be checked to be in the correct "ACTIVE" SESSION_STATUS. The number of rows modified is returned to the counter and this should always be 1.

The second message must be handled by a new service. It will need to do the same concurrent login checks as done as described below in detectUserAlreadyLoggedOn() method. There is a potential for this service to fail if active sessions are found. The service has an option that will set the SESSION_LOCKED_TIMESTAMP (using the header.TimeSent[Timestamp] field from the request) and at the same time mark all ACTIVE sessions that are not locked as "FAILED". This is similar behaviour as concurrent login. During a normal unlock process this service may get called twice – once to unlock (and this fails due to other ACTIVE sessions) and a second time to perform the unlock action and make ACTIVE sessions as failed. These cannot be combined since the user must choose to mark ACTIVE sessions as FAILED.

These message are auditable.

Due to the introduction of new services the osr_monitor.xml will require updating. Impacts for the BAL must consider this update and the resultant baseline production.

2.3.1 Changes to Concurrent Login Checks

Within the BAL the checks performed in the code SessionDAOImpl::detectUserAlreadyLoggedOn() need to be modified to allow logins where all the other counters that a user is logged into are in the "Locked" status [EUM-CE-CL-20, EUM-CE-CL-40].

Note that this change MUST only be performed on the users in branches that have a POID association. For branch users that not have not been migrated over then the checks should remain as is. CIT will need to confirm this via regression testing.

The SQL query PoidSessionAlreadyExists needs to be modified to return the value of the SESSION_LOCKED_TIMESTAMP column on matching rows. We must continue to return ALL the sessions that match that have a SESSION_STATUS of "ACTIVE". This is because the detectUserAlreadyLoggedOn() method performs clean-up of the sessions during the login process. For example ACTIVE session for the Counter that the user is logging into will be marked as FAILED irrespective if they are in a LOCKED status.

Within the detectUserAlreadyLoggedOn() method there is a while loop that detects the concurrent session.

This while loop takes an ***ordered*** result set. The first row is assumed to be an active session on some other counter rather than the current one and a Boolean foundExistingActiveUserSession is set to true, other sessions are then marked as invalid. The check needs to exclude counters that are in a locked status (as indicated as a NON-NULL SESSION_LOCKED_TIMESTAMP) to ensure these sessions remain in an ACTIVE status.

2.3.2 Changes to Concurrent Login Service

If a user chooses to continue the login process despite there been sessions in the unlocked status then the BAL service ConcurrentLogonServiceHandlerImpl will be called. This will need to be modified to ensure only sessions in the ACTIVE state that are unlocked are marked as FAILED. The changes is believed to be limited to the SQL query PoidSessionAlreadyExists which needs to ignore any sessions that have a NON-NULL SESSION_LOCKED_TIMESTAMP [EUM-CE-CL-20].

2.3.3 Document changes

DEV/APP/LLD/0020: HNGX Low Level Design for Logon Use Case

DEV/APP/SPG/0017: HNG-X Counter Business Application Support Guide

DES/APP/HLD/0060: UCR Document for the Branch Administration Service Barrel

Impactors to determine other documentation.

FUJ00171899

2.4 Counter Changes

A modification is required to enable the manual lock button during a transaction. Currently the system has the following restrictions regarding when the button is enabled which must be (partly shown in red) removed.

The lock button is disabled before login (to remain) and disabled during transactions (now to be enabled).

The clerk must complete or abandon that particular transaction before locking (to be removed). This does not mean finish dealing with the customer, just that particular transaction (online banking, etu, post mail item, etc) and the key being they need to go back to the menu before they can lock (to be removed).

Also the lock/resume/suspend button is only enabled at times that a lock can be performed without interrupting current processing on the counter. For example the user cannot “lock” the computer whilst there is any “transient”* message box on the screen. A “transient” message means a “message which can go away without user intervention”, such as printing or print in progress or a current network interaction. [EUM-CE-CL-35].

The automatic locking of a counter will still occur after the current inactivity timeouts kick in.

The counter requires the following changes.

1) Each time the counter is locked – either explicitly or by inactivity timeout – then the new BAL service LockCounter must be called [EUM-CE-CL-30, EUM-CE-CL-90]. If this call fails due to network unavailability then in the case of a user initiated lock session the user must be informed and requested if the lock action should continue. In the case of a lock due to an inactivity timeout then the session must be locked and the failure is silently ignored. The latter scenario cannot "fail". In both cases the data centre will be unware of the locking of the counter so must treat it as a non-locked counter. Subsequent login attempts from other counters would treat this as a non-locked session[1]. It is believed that this new service call must be made from RelogonBLO::executeBusinessLogic() before the lock screen is displayed. A message should be displayed if this lock fails and it’s a user initiated lock request.
2) Each time the counter is unlocked then the new BAL service UnlockCounter must be called. If this service call fails then the user should be informed, given options and the lock will remain in place. The UnlockCounter service may fail in two different ways.
    a. If there is already a counter in the ACTIVE status that is not locked [EUM-CE-CL-50]. The user should be prompted (via new messages) if they wish to terminate the ACTIVE sessions in the same way as concurrent login. This design does not allow this session to be remotely locked.
    b. Remote communication fails to the data centre. Again the user is informed but not give any options. Failure in this case is acceptable since if communication with the data centre is not operational then no transactions can be performed so remaining in the locked status is acceptable. It is believed that this new service should be called from RelogonBLO::existingUserRelogon() before the LoginUIA is destroyed.

    After the counter is unlocked transactions will be allowed to continue [EUM-CE-CL-60].
3) When logging into a counter and concurrent login has been detected a new message should be displayed [EUM-CE-CL-10] to inform the user that they must either abandon these sessions or lock them and retry the login.
4) The lock button must be enabled during transactions [EUM-CE-CL-30]. Note this introduces an issue where the current lock button does not immediately lock the counter – rather an option box is displayed allowing the three options of suspend/resume and lock. Which of these buttons can be used is dependent on the following. This will not change.

    If the mode is not serve customer

    OR

    This user does not have Transactions permissions

    OR

    This user is in the Default Stock Unit

[1] There is an inconsistency here since the user will be able to "see" the counter is locked – but an attempt at another counter would inform them it is not locked. Training will have to be used to overcome this inconsistency. If the lock was to fail then the counter could remain in a non-locked state for some considerable time which is deemed a security issue.

FUJ00171899

Then

Only the Lock option is enabled (suspend and resume disabled).

Else if the basket is not empty

Then

Only the Suspend and Lock Options are enabled (Resume disabled)

Else

Suspend, Resume and Lock are enabled

Initial prototyping has shown that if the screen is locked during a transaction then the initial page of the transaction is reshown on unlock. This must be changed so the screen displayed when locked is the one reshown after unlock. Other options on the screen behaviour will be considered and should be indicated as alternatives in the impacting for example popping up a dialog box with the options. Note the button action should be modified as well to ensure that it's not defined to start a new UI application.

2.1.1 New/Modified Messages

The following new/modified messages are required

1) A new message for failed unlocking of a counter due to network unavailability.
2) A new message when during the unlocking of a counter other counters in an unlocked status are found. This is similar to the concurrent login messages.
3) A new message informing the user that the lock failed due to network unavailability. Only displayed with manual locking not automatic locking of counter.
4) A new message when attempting to login and there are non-locked sessions [EUM-CE-CL-10]

2.1.2 Documentation changes

The following documents will require changing

1) DEV/GEN/MAN/0006 to allow the button to be enabled during transactions.
2) DES/APP/HLD/0047 to ensure that Suspend & Resume options are disabled during transactions.

Impactors to determine other documentation.

3 New Releases

The following components will require a new release. Impacts should detail the package names, this list is given here to enable integration to understand the extent of these updates with respect of modified packages.

No new packages are required.

1) All the BAL baselines will require updating – including the package released to the BMX.
2) All the CBA counter baselines both HNG-A (including MSI) and HNG-X will require updating.

4 Requirements Traceability

Requirements mention "stock unit can be locked" – this has been clarified to mean "`the transaction type`"

Tag/Id	Name	Priority	Description	Acceptance criteria	Satisfied By

FUJ00171899

EUM-CE-CL-10	Existing Unlocked Session Check	M	On initial login, the Horizon data centre shall check if the user's associated POID has an existing Unlocked session within the Branch	On Login, the Data Centre will send a message through to the Horizon Counter if the HUID associated to the POID has an existing Unlocked session. Message: "Please return and lock the previous session or remotely terminate it." **TBC**	Section 2 and 2.4.1
EUM-CE-CL-15	Existing Unlocked Session User Initiated System Lock	S	On login (e.g. to Counter B), if Horizon identifies that the User is logged on at another counter in an "Unlocked" state (e.g. Counter A), Horizon shall present the User with an option to lock the previous session. Thus, negating the need for the User to physically lock counter A. Subject to constraints as per EUM-CE-CL-30.	On Login, the Data Centre will send a message through to the Horizon Counter if the HUID associated to the POID has an existing Unlocked session. Message 1 (Req EUM-CE-CL-10): "Please return and lock the previous session or remotely terminate it." Message 2 "Please confirm you wish to lock the previous session remotely – Yes / No" (Text tbc) Linked to Requirement EUM-CE-CL-30	Section 2.4. The remote session cannot be locked. The only action is for its sessions to be marked as FAILED.
EUM-CE-CL-20	HUID Branch Check	M	On initial login, the Horizon data centre shall check if the users HUID associated to the POID is being used in another Branch.	On Login, the Data Centre will send a message through to the Horizon Counter if the HUID associated to the POID is in use in another Branch. Message: **TBC**	Section 2.3.1 and 2.3.2
EUM-CE-CL-30	Locking a Horizon Counter	M	The user shall pause a transaction session by locking the session, the Horizon data centre can check the stock units can be paused or must be completed.	On Locking, the Data Centre shall send a message through to the Horizon Counter if the stock units can be paused or must be completed. Messages: - Locked message - Must be completed message. **TBC**	Section 2.4, No stock unit check – locks can only occur provided conditions in Section 2.4 are met.
EUM-CE-CL-35	Locking a Horizon Counter whilst "Transaction in progress"	S	Horizon shall be enhanced to enable a User to lock a counter session whilst transactions are in progress, then login (and serve) at a different Counter.	Locking Counter A (and serving on Counter B) shall not impede the progress of any transactions in progress or within the basket of Counter A.	It is not possible to lock a counter that is processing data. The user cannot "lock" the computer whilst there is any "transient"* message box on the screen. A "transient" message means a "message which can go away without user intervention", such as printing or print in progress or a current network interaction. The lock button will be enabled during a transaction
EUM-CE-CL-40	Initial Login to Another Horizon Counter	M	On Initial login to Another counter the user shall input the HUID and associated password into the	The Data Centre will: - Check the HUID associated to the POID credentials - Check the HUID is in use	Section 2 2.2 and 2.3.1

FUJ00171899

			Horizon Counter screen.	in the same branch - Check if another counter transaction session is locked by the User - Allow the user to transact on the "second" counter. Messages: **TBC**	
EUM-CE-CL-50	Unlocking a Locked transaction session Horizon Counter By Password	M	The user shall unlock a (Locked) transaction session by entering their Horizon Password associated to the HUID that locked the transaction session.	On Unlocking the (Locked) transaction session, the System will check if the password matches the HUID that locked the transaction setting and if it does, shall unlock the session for the user. Only 1 "Active" unlocked session shall be available at a time. Messages: - Unlocked message - Incorrect password message. **TBC**	Section 2.3
EUM-CE-CL-60	Completing an Unlocked transaction session on a Horizon counter	M	The user can complete and unlock a (Locked) transaction and end the transaction session.	On completing the Unlocking the (Locked) transaction session, the Data Centre will capture the data event that the unlocked session has been completed and send a confirmation message back to the horizon counter. Messages: - Confirmation of completion message - Failed completion message. **TBC**	Section 2.4
EUM-CE-CL-70	HUID to POID association	M	A single HUID associated to a single POID can be used to complete a concurrent Login.	A single HUID can lock / unlock multiple sessions. Multiple HUIDs linked to a single POID should not be able to login concurrently.	Section 2
EUM-CE-CL-80	New Event for Locking	M	A new event for the Locking of a transaction suspension shall be created to enable Audit & reporting.	Horizon will be able to report on all events for the Locking of a transaction suspension.	Section 2.1. No new event will be created.
EUM-CE-CL-85	New Event for Unlocking	M	A new event for the unlocking of a transaction suspension shall be created to enable Audit & reporting (ARQ purposes).	Horizon will be able to report on all events for the Locking of a transaction suspension.	Section 2.1. No new event will be created. This should read "… all events for the ***un***locking of a transaction suspension"
EUM-CE-CL-90	Forced Horizon Timeout	C	The forced time out of Horizon = X mins Note: Value to be confirmed by Fujitsu.	Horizon will close the transaction if the counter is not used within 74 mins	Section 2.4
EUM-CE-CL-100	Forced Horizon Timeout Message	M	In a forced timeout scenario, the system shall present a message to the Horizon User to advise their session is about to / has expired. Message text TBC. Note: Fujitsu to confirm if this is "As	The message is successfully presented to the Horizon User at timeout / after timeout. If the screen is in a locked status, the message is rendered on top of the screen lock.	It is believed that this is the current behaviour of the counter. No new functionality will be required. **Counter impactors to confirm.**

FUJ00171899

			Is" functionality.		
EUM-CE-CL-110	Forced Horizon Timeout Message at Active Counter	C	In a forced timeout scenario, the system shall present a message to the Horizon User on their currently active terminal, to advise their session is about to / has expired. Message text TBC. Note: Fujitsu to confirm if this is "As Is" functionality.	The message is successfully presented to the Horizon User at timeout / after timeout.	It is believed that this is the current behaviour of the counter. No new functionality will be required. **Counter impactors to confirm.**
EUM-CE-CL-120	Background Reports run in a "Inactive" Counter status	M	If a counter is progressing Back Office reporting activity e.g. Report Printing, the activity can complete even if a User locks the session and moves to another counter. Fujitsu to advise if constraints exist on specific back office transactions / activities.	Start a back office report on Counter A, lock the session and move to Counter B. The activity on Counter A shall complete as expected.	It is not possible to lock a counter that is processing data. The user cannot "lock" the computer whilst there is any "transient"* message box on the screen. A "transient" message means a "message which can go away without user intervention", such as printing or print in progress or a current network interaction.

5 Assumptions

Some use cases are not considered in the requirements and assumptions have been made on how the user should interact with them. Additionally some requirements are omitted from the design due to constraints in the system, these are highlighted in the table above red implies unable to implement, yellow implies partially/alternative implemented.

Note this design brings the use cases of concurrent login and unlocking a counter much closer together since they now may both invalidate (FAIL) session on other counters.

1) Only one HUID can be logged in at the same time irrespective of the status locked/unlocked of other counters. Assume as this keeps auditing of users activity via their HUID simple. As of 07/12/2017 Assumed that any HUID may be used not just the current one.
2) Locking counter when no data centre connection is available (either explicitly or by timeout) will result in the data centre NOT considering the counter as locked for any concurrent login checks. In the cases of a user initiated lock the user will have a choice to continue to lock, retry or abandon. In the case of an automatic inactivity timeout the lock will proceed and the failure ignored. Only if the counter is locked in this manner will the data centre be unaware of the lock.
3) Unlocking a counter may encounter the case where another counter is already in the unlocked status [EUM-CE-CL-15]. This case will be treated concurrent login and the user will have the option to terminate any other active sessions.
4) The use cases where a counter can be locked (i.e. mid transaction or operation) will remain as currently defined. See comment to EUM-CE-CL-120
5) When a login is attempted and a mixture locked/unlocked counters are found then only the unlocked counter will have their sessions modified to FAILED if continuation of the login is requested by the user.
6) Timing of events registration at data centre remains unchanged. See 2.1.2