

FAQ: Event Publisher in Multi Entity Management

This FAQ document (Frequently Asked Questions) covers questions related to the Events published by the Multi Entity Management Extension APP.

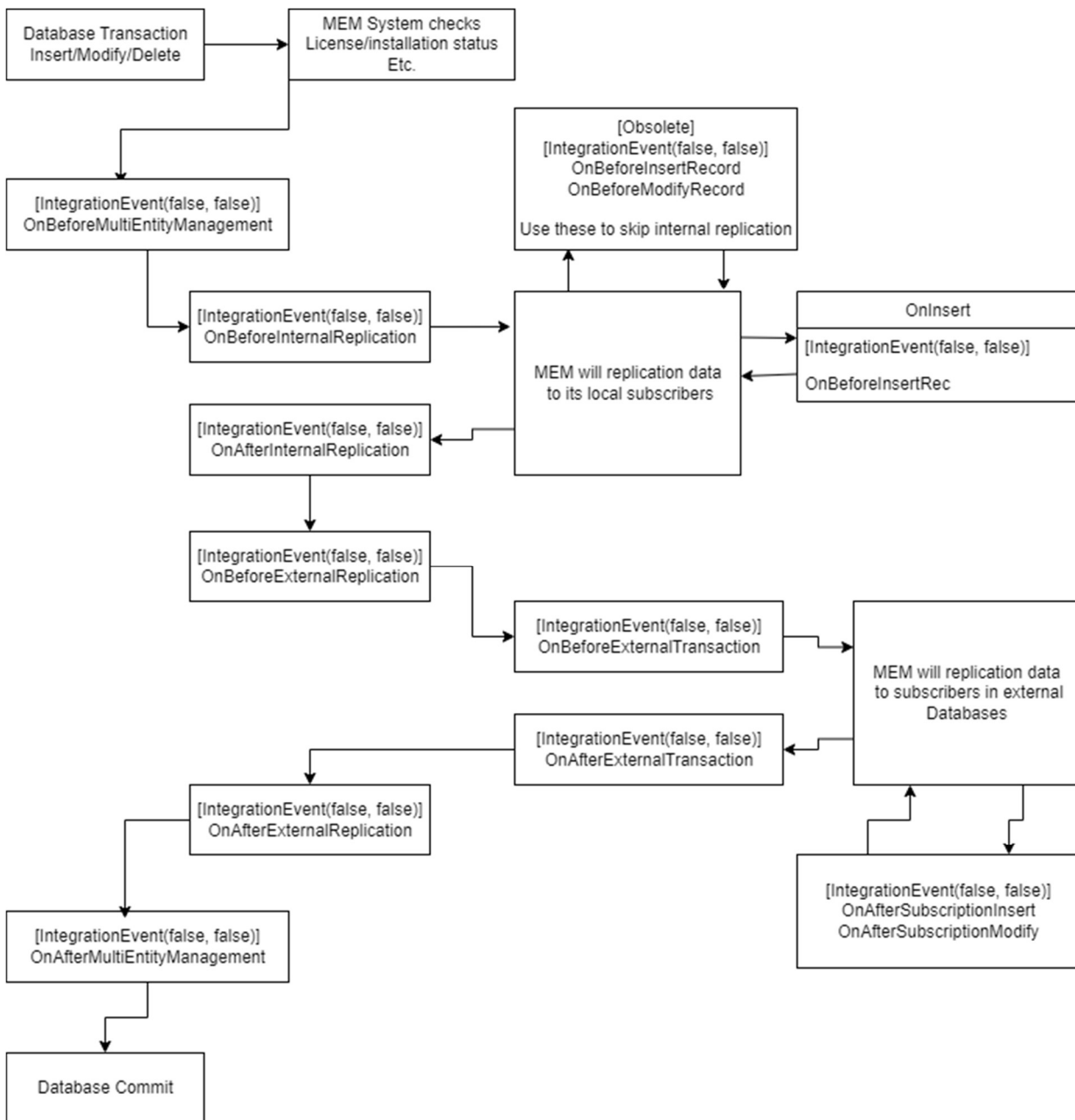
This FAQ apply for Multi Entity Management, version 5.1.0 and later.

Changelog:

- 1.0.1: Descriptions specified in section 1.1 Event Publishers
- 1.2.0: New structure in Event Publishers described in section 1.1 Event Publishers
- 1.3.0: Description of the new event **OnBeforeInsertRec**

1.1 Event Publishers.

All Event Publishers are placed in Codeunit 5397818 "XTEMEMEvent Publisher".



Active Event Publisher (Marked as obsolete)

- `[IntegrationEvent(false,false)] - OnBeforeMEMInsertRecord(var RecRef: RecordRef; var Skip: Boolean)`
 - o This Event is triggered by the global event `OnDatabaseInsert` in the MEM code before replication checks and logic. *) - **Obsolete**
- `[IntegrationEvent(false,false)] - OnBeforeMEMModifyRecord(var RecRef: RecordRef; var Skip: Boolean)`
 - o This Event is triggered by the global event `OnDatabaseModify` in the MEM code before replication checks and logic. - **Obsolete**

Active Event Publisher

- [IntegrationEvent(false,false)] - **CrossDatabaseOnAfterInsert**(var RecRef: RecordRef)
 - o This Event is triggered after a record has been inserted in the subscriber database. *)
- [IntegrationEvent(false,false)] - **CrossDatabaseOnAfterModify**(var RecRef: RecordRef)
 - o This Event is triggered after a record has been modified in the subscriber database.
- [IntegrationEvent(false, false)] - **OnBeforeMultiEntityManagement**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; var IsHandled: Boolean)
 - o Right after the database has registered a change (insert/modify/delete) MEM will kick in, and do its default checks. (Is rec temporary, is license approved etc.) This event is then triggered after these default event, but before MEM begins to handle the record.
- [IntegrationEvent(false, false)] - **OnAfterMultiEntityManagement**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction)
 - o This event is triggered after MEM has done its entire flow, but before database commit.
- [IntegrationEvent(false, false)] - **OnBeforeInternalReplication**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; FromCompanyName: Text; var IsHandled: Boolean)
 - o This event is triggered after MEM has acknowledged a record replication is inbound, but before the replication is being performed. (Internal replication only impact companies in same database as the publisher)
- [IntegrationEvent(false, false)] - **OnAfterInternalReplication**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; FromCompanyName: Text)
 - o This event is triggered ones MEM has finished handling the replication for companies in the same database as the publisher. But before database commit.
- [IntegrationEvent(false, false)] - **OnBeforeExternalReplication**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; FromCompanyName: Text; var IsHandled: Boolean)
 - o This event is triggered before MEM will handle its replication flow for external databases.
- [IntegrationEvent(false, false)] - **OnAfterExternalReplication**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; FromCompanyName: Text)
 - o This event is triggered after MEM has finished its replication flow to external database. But before database commit.
- [IntegrationEvent(false, false)] - **OnBeforeExternalTransaction**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; var BigText: BigText; SubscriberCompany: Record "XTEMEMSubscriber Company"; var IsHandled: Boolean)
 - o This event is triggered after MEM has serialized data for external replication, but before the actual transaction have been performed.
- [IntegrationEvent(false, false)] - **OnAfterExternalTransaction**(var RecordRef: RecordRef; Transaction: Enum XTEMEMTransaction; var BigText: BigText; SubscriberCompany: Record "XTEMEMSubscriber Company")
 - o This event is triggered after MEM has replicated data to external companies. But before database commit.
- [IntegrationEvent(false, false)] - **OnBeforeInsertRec**(var RecordRef: RecordRef; ReplicationAllocation: Record XTEMEMReplicationAllocation; var BigText: FromCompany: Text; var IsHandled: Boolean)
 - o This event is similar to **OnBeforeInternalReplication**, but instead of being called before MEM performs any transaction, this event is triggered when MEM starts to insert a record.
 - o Use this event to allow check of the record and skip if necessary. There should be no interactions because the record may not be inserted.

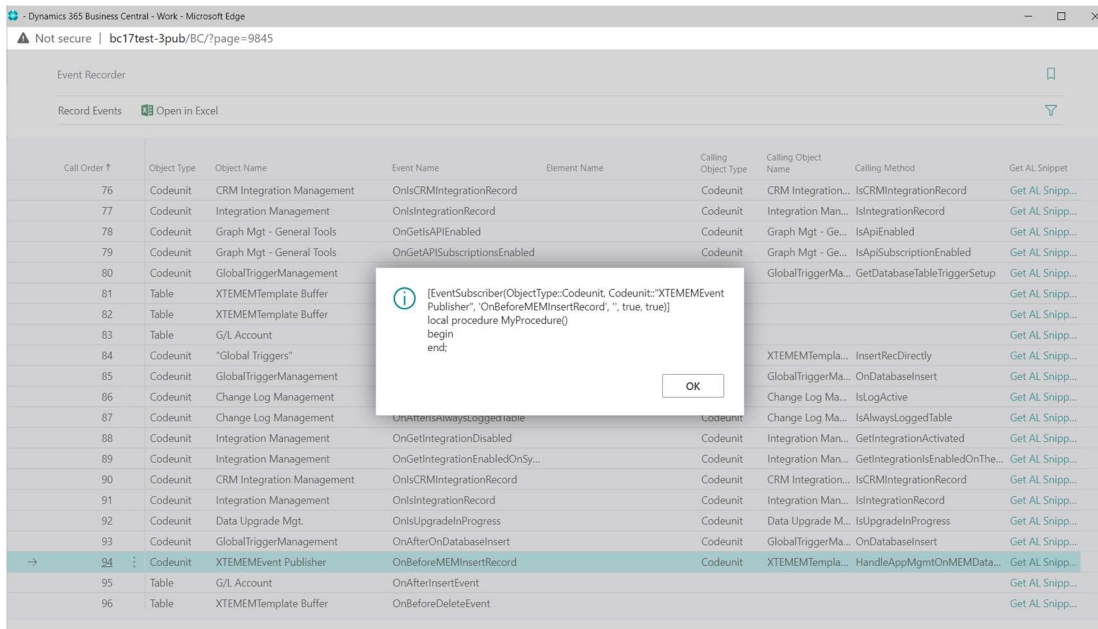
*Note: It is only possible to handle primary key field values unless the record has been created with “delayed insert”.

1.2 Subscribing on Multi Entity Management Events.

Please use the Event Recorder to get an overview over where the events are triggered.

[Events Discoverability - Business Central | Microsoft Docs](#)

Example of the Event Recorder in Business Central 17.



The screenshot shows the 'Event Recorder' interface in Microsoft Edge. It displays a table of recorded events. A modal dialog box is open over the table, showing the following text:

```
[EventSubscriber(ObjectType:Codeunit, Codeunit::'XTEMEMEvent
Publisher', 'OnBeforeMEMInsertRecord', '', true, true)]
local procedure MyProcedure()
begin
end;
```

The table below represents the data visible in the Event Recorder:

Call Order #	Object Type	Object Name	Event Name	Element Name	Calling Object Type	Calling Object Name	Calling Method	Get AL Snippet
76	Codeunit	CRM Integration Management	OnIsCRMIntegrationRecord		Codeunit	CRM Integration...	IsCRMIntegrationRecord	Get AL Snipp...
77	Codeunit	Integration Management	OnIsIntegrationRecord		Codeunit	Integration Man...	IsIntegrationRecord	Get AL Snipp...
78	Codeunit	Graph Mgt - General Tools	OnGetIsAPIEnabled		Codeunit	Graph Mgt - Ge...	IsAPIEnabled	Get AL Snipp...
79	Codeunit	Graph Mgt - General Tools	OnGetAPISubscriptionsEnabled		Codeunit	Graph Mgt - Ge...	IsAPISubscriptionEnabled	Get AL Snipp...
80	Codeunit	GlobalTriggerManagement				GlobalTriggerMa...	GetDatabaseTableTriggerSetup	Get AL Snipp...
81	Table	XTEMEMTemplate Buffer						Get AL Snipp...
82	Table	XTEMEMTemplate Buffer						Get AL Snipp...
83	Table	G/L Account						Get AL Snipp...
84	Codeunit	"Global Triggers"				XTEMEMTempla...	InsertRecDirectly	Get AL Snipp...
85	Codeunit	GlobalTriggerManagement				GlobalTriggerMa...	OnDatabaseInsert	Get AL Snipp...
86	Codeunit	Change Log Management				Change Log Ma...	IsLogActive	Get AL Snipp...
87	Codeunit	Change Log Management	OnAlwaysLoggedTable		Codeunit	Change Log Ma...	IsAlwaysLoggedTable	Get AL Snipp...
88	Codeunit	Integration Management	OnGetIntegrationDisabled		Codeunit	Integration Man...	GetIntegrationActivated	Get AL Snipp...
89	Codeunit	Integration Management	OnGetIntegrationEnabledOnSys...		Codeunit	Integration Man...	GetIntegrationEnabledOnThe...	Get AL Snipp...
90	Codeunit	CRM Integration Management	OnIsCRMIntegrationRecord		Codeunit	CRM Integration...	IsCRMIntegrationRecord	Get AL Snipp...
91	Codeunit	Integration Management	OnIsIntegrationRecord		Codeunit	Integration Man...	IsIntegrationRecord	Get AL Snipp...
92	Codeunit	Data Upgrade Mgt.	OnIsUpgradeInProgress		Codeunit	Data Upgrade M...	IsUpgradeInProgress	Get AL Snipp...
93	Codeunit	GlobalTriggerManagement	OnAfterOnDatabaseInsert		Codeunit	GlobalTriggerMa...	OnDatabaseInsert	Get AL Snipp...
94	Codeunit	XTEMEMEvent Publisher	OnBeforeMEMInsertRecord		Codeunit	XTEMEMTempla...	HandleAppMgmtOnMEMData...	Get AL Snipp...
95	Table	G/L Account	OnAfterInsertEvent					Get AL Snipp...
96	Table	XTEMEMTemplate Buffer	OnBeforeDeleteEvent					Get AL Snipp...

1.3 Need of Event hooks.

Any XtensionIT Partner can suggest a placement of an Event Hook to activate an Event Publisher.

Please send your suggestion to info@xtensionit.com with description of placement and which Publisher to activate.