Scenario: we have requirement like we one Asynchronous plugin and Asynchronous workflows on create message and account entity?

For example creating tasks?

If Asynchronous workflows failed but asynchronous plugin will execute. It meant to say plugin will create the task not workflows 🡪will do after creating the Account entity.

If asynchronous plugin failed but asynchronous workflows will execute. It meant to say workflow will create the task not plugin will do 🡪after creating the Account entity.

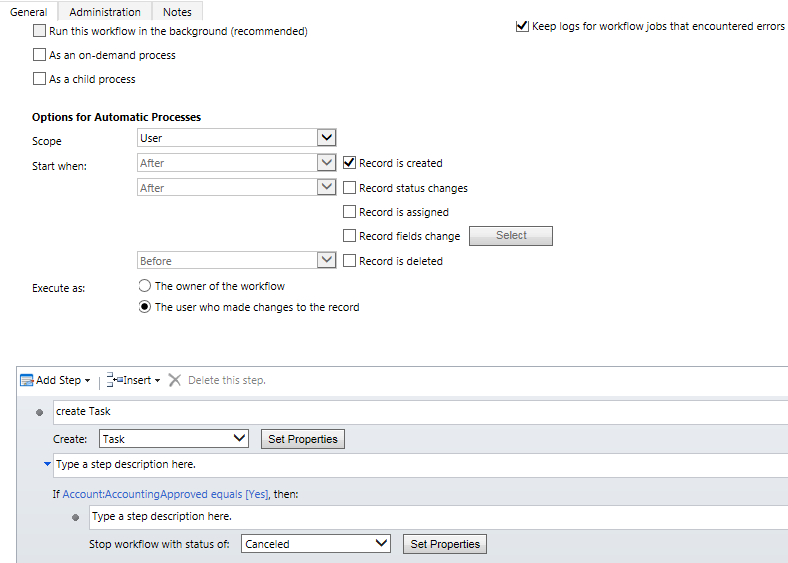
Scenario: we have requirement like we have one synchronous plugin, synchronous workflows and Asynchronous workflows on create message and account entity?

If Synchronous plugin or Synchronous workflows will gets failed, then no tasks will create, it meant to say..... Synchronous workflows, Asynchronous workflows and asynchronous plugin will be there on same entity... will won’t execute. Since it is a roll back option

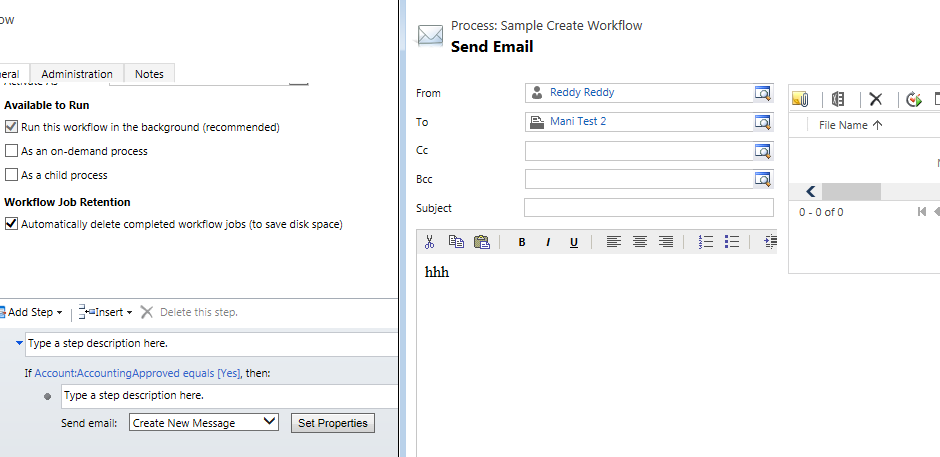
Scenario: we have requirement like we one Asynchronous plugin, synchronous workflows, synchronous plugin and Asynchronous workflows on create message and account entity?

If Asynchronous plugin will gets failed, Synchronous workflow, synchronous plugin, asynchronous workflows will execute...

Synchronous workflow configure for reproduce the above steps:

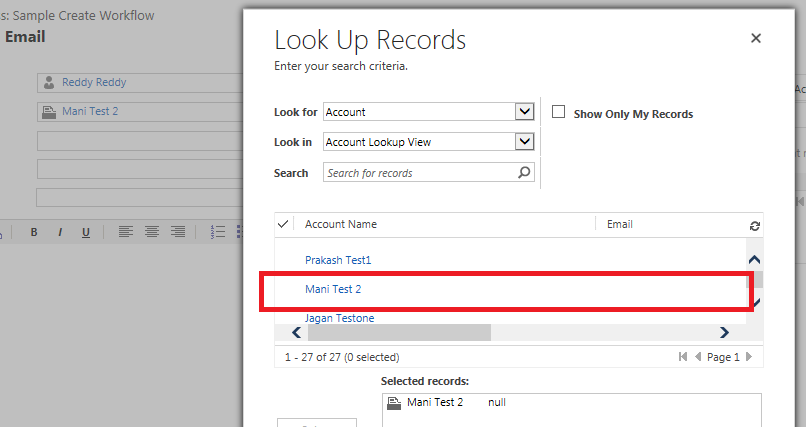


For Asynchronous Workflows Gets Failed:



Click on set Properties it will show the above Send Email Window.

Mani test 2 does not having the email id then these workflows gets failed.



[How to Allow CRM Plugin to Handle Multiple Event Messages](http://stackoverflow.com/questions/30161237/how-to-allow-crm-plugin-to-handle-multiple-event-messages)

I need to create a single plugin to fire on create, update, and delete events of opportunity entities.

Can I do these in a single plugin? If so then how can I do them?

AddItem,

AddListMembers,

AddMember,

AddMembers,

AddPrincipalToQueue,

AddPrivileges,

AddProductToKit,

AddRecurrence,

AddToQueue,

AddUserToRecordTeam,

Assign,

AssignUserRoles,

Associate,

BackgroundSend,

Book,

Cancel,

CheckIncoming,

CheckPromote,

Clone,

Close,

CopyDynamicListToStatic,

CopySystemForm,

Create,

CreateException,

CreateInstance,

Delete,

DeleteOpenInstances,

DeliverIncoming,

DeliverPromote,

DetachFromQueue,

Disassociate,

Execute,

ExecuteById,

Export,

ExportAll,

ExportCompressed,

ExportCompressedAll,

GenerateSocialProfile,

GrantAccess,

Handle,

Import,

ImportAll,

ImportCompressedAll,

ImportCompressedWithProgress,

ImportWithProgress,

LockInvoicePricing,

LockSalesOrderPricing,

Lose,

Merge,

ModifyAccess,

PickFromQueue,

Publish,

PublishAll,

QualifyLead,

Recalculate,

ReleaseToQueue,

RemoveFromQueue,

RemoveItem,

RemoveMember,

RemoveMembers,

RemovePrivilege,

RemoveProductFromKit,

RemoveRelated,

RemoveUserFromRecordTeam,

RemoveUserRoles,

ReplacePrivileges,

Reschedule,

Retrieve,

RetrieveExchangeRate,

RetrieveFilteredForms,

RetrieveMultiple,

RetrievePersonalWall,

RetrievePrincipalAccess,

RetrieveRecordWall,

RetrieveSharedPrincipalsAndAccess,

RetrieveUnpublished,

RetrieveUnpublishedMultiple,

RetrieveUserQueues,

RevokeAccess,

Route,

RouteTo,

Send,

SendFromTemplate,

SetRelated,

SetState,

SetStateDynamicEntity,

TriggerServiceEndpointCheck,

UnlockInvoicePricing,

UnlockSalesOrderPricing,

Update,

ValidateRecurrenceRule,

Win

Yes you can use same plugin to execute it on create, update and delete

Write your plugin as follows...

public void Execute(IServiceProviderserviceProvider)

{

IPluginExecutionContext context = (IPluginExecutionContext)serviceProvider.GetService(typeof(IPluginExecutionContext));

IOrganizationServiceFactoryserviceFactory = (IOrganizationServiceFactory)serviceProvider.GetService(typeof(IOrganizationServiceFactory));

IOrganizationService service = serviceFactory.CreateOrganizationService(context.UserId);

if (context.MessageName == "Create")

{

//write the logic what you want this plugin to do on Create

}

if (context.MessageName == "Update")

{

//write the logic what you want this plugin to do on Update

}

if (context.MessageName == "Delete")

{

//write the logic what you want this plugin to do on Delete

}

}

Register your plugin on Create, Update, Delete for Opportunity Entity using Plugin registration tool