```
import AP.Common.Global;
import AP.FO.BusinessPartner.Global;
import AP.FO.BusinessPartnerRelationship.Global;
import AP.CRM.Global;

var result : BusinessObjectInstanceProcessIntegrationRelevance;
var customer: Customer;
var replicationObjectUUID;
var bupaRelationship: BusinessPartnerRelationship;
var bupaRelationshipCol: collectionof BusinessPartnerRelationship;
var contactpersonRelCol: collectionof BusinessPartnerRelationship;
var categoryCode;

var businessPartner: BusinessPartner;
var customerRole: Customer.Role;
var currBusCharacters: Customer.CurrentBusinessCharacters;


var _customerTypeCode = "159";   //Account
var _BPRelationshipTypeCode = "1405"; //Business Partner relationship
var _businessPartnerTypeCode = "441"; //Business Partner


foreach (replicationObjectUUID in InputData.UUID ) {
        var typeCodeDescription = InputData.ObjectTypeCode.GetDescription();
        // SWITCH: replication object types
        switch(InputData.ObjectTypeCode){

                // Replication Object Type: Customer
```

```
case _customerTypeCode{
    customer = Customer.Retrieve(replicationObjectUUID);
    if(customer.IsSet()){
        currBusCharacters = customer.CurrentBusinessCharacters;
        if(currBusCharacters.IsSet()){
            // Do not replicate the customer if it is a Prospect
            if (!currBusCharacters.ProspectIndicator){
                // Replicate all other customers
                result.UUID.Add(replicationObjectUUID);
            }
        }
    }
}

/*
    Added redundant check for stopping Prospect Replication.
    This is required to handle scenarios where the Prospect is created
    on conversion of Lead to Opportunity.
*/
// Replication Object Type: BusinessPartner
case _businessPartnerTypeCode{
    customer = Customer.Retrieve(replicationObjectUUID);
    if(customer.IsSet()){
        currBusCharacters = customer.CurrentBusinessCharacters;
        if(currBusCharacters.IsSet()){
            // Do not replicate the customer if it is a Prospect
            if (!currBusCharacters.ProspectIndicator){
                // Replicate all other customers
                result.UUID.Add(replicationObjectUUID);
```

```
                    }
                }
            }
            else{
                // Replicate all other business partners
                result.UUID.Add(replicationObjectUUID);
            }
        }

        // Replication Object Type: Business Partner Relationship
        case _BPRelationshipTypeCode{
            // Retrieve the current relationship object
            bupaRelationship =
BusinessPartnerRelationship.Retrieve(replicationObjectUUID);
            if(bupaRelationship.IsSet()){
                /*
                    Stop the relationship replications, if the customer is a Prospect.
                */
                customer =
Customer.Retrieve(bupaRelationship.FirstBusinessPartnerUUID);
                if(customer.IsSet()){
                    currBusCharacters = customer.CurrentBusinessCharacters;
                    if(currBusCharacters.IsSet()){
                        // Replicate the relationships only if the customer is not
a Prospect.
                        if (!currBusCharacters.ProspectIndicator){
                            result.UUID.Add(replicationObjectUUID);

                        }
                    }
                }

            }
        }


        // Replicate all other Replication object types
        default{
            // Replicate other objects
            result.UUID.Add(replicationObjectUUID);
        }
    }
}

result.ObjectTypeCode = InputData.ObjectTypeCode;
return result;
```