Salesforce - Lead Assessment Engine - documentation

Components

Permission Sets

- Lead_Assessment

	Action	Permission Set Label ↑	Description	License
☐	Del	Clone	Lead Assessment		Salesforce

Apex Classes

- BatchLeadAssessment
- BatchLeadAssessmentScheduler
- LeadAssessmentService
- LeadAssessmentServiceTest
- Utils
- UtilsTest

SObjects

- Lead_Assessment_Delay
- Azure_B2C_Lead_Integration_Fields
- Lead_Assessment_Action__c
- Lead_Assessment_Condition__c
- Lead_Assessment_Log__c
- Lead_Assessment_Query__c
- Lead_Assessment_Rule__c
- Lead_Assessment_Type__c
- Lead
  - **Fields**
    - Cold_Lead_Nurturing__c
    - Sync_To_MC_Cold_Lead_Nurturing__c
    - Lead_Assessment_Rule__c
  - **Record Types**
    - B2C_Housing_Sales

Tabs

- Lead_Assessment_Log__c
- Lead_Assessment_Type__c

Page Layouts

- Lead-B2C Housing Sales
- Lead_Assessment_Action__c-Lead Assessment Action Layout
- Lead_Assessment_Condition__c-Lead Assessment Condition Layout
- Lead_Assessment_Log__c-Lead Assessment Log Layout
- Lead_Assessment_Query__c-Lead Assessment Query Layout
- Lead_Assessment_Rule__c-Lead Assessment Rule Layout
- Lead_Assessment_Type__c-Lead Assessment Type Layout
- Azure_B2C_Lead_Integration_Fields__mdt-Azure B2C Lead Integration Fields Layout
- Lead_Assessment_Delay__mdt-Lead Assessment Delay Layout

LWC Components

- leadAssesmentQueryVerify

Quick Actions

- Lead_Assessment_Rule__c.Verify_Queries

Custom Metadata

- `Azure_B2C_Lead_Integration_Fields.Configuration`
- `Lead_Assessment_Delay.Configuration.Configuration`

Named Credentials

- `Lead_Distributor_B2C_Dev`
- `Lead_Distributor_B2C_Prod`
- `Lead_Distributor_B2C_Test`

Remote Site Settings

- `Lead_Distributor_B2C_Dev`
- `Lead_Distributor_B2C_Prod`
- `Lead_Distributor_B2C_Test`

Instruction

1. Turning on the engine

In order for the engine to run in the background the batch (`BatchLeadAssessment`) must be scheduled

One-time manual step that is needed to run the engine is to execute this code:

```
System.schedule(
            Utils.BATCH_LEAD_ASSESSMENT_SCHEDULER,
            LeadAssessmentService.getNextScheduleExecutionTime(),
            new BatchLeadAssessmentScheduler()
        );
```

After Batch completes, each next execution is delayed by the settings in the CustomMetadata (`Lead_Assessment_Delay.Configuration`)

Minimum delay should be 1 minute in order the Schedule to start

Q Custom Metadata Types

∨ Custom Code

Custom Metadata Types

Didn't find what you're looking for?
Try using Global Search.

SETUP
Custom Metadata Types

All Custom Metadata Types

Custom metadata types enable you to create your own setup objects whose
packaged and installed.

Rather than building apps from data records in custom objects or custom s
Querying custom metadata records doesn't count against SOQL limits.

Action	Label	Installe
Manage Records	⬇ AAkonsult Setting	AAkon
Manage Records	⬇ Avenir Configuration	Aveni
Del | Manage Records	Azure B2C Lead Integration Fields	
Del | Manage Records	CustomNameCredential	
Del | Manage Records	Lead Assessment Delay	
Del | Manage Records	Record as PDF	
Del | Manage Records	Telia Identity Config	
Del | Manage Records	TriggerSetting	

SETUP
Custom Metadata Types

Custom Metadata Type
Lead Assessment Delay

Standard Fields [6] | Custom Fields [3] | Valid

Custom Metadata Type Detail

Edit Delete Manage Lead Assessment Delays

SETUP
Custom Metadata Types

Lead Assessment Delays

View: All ⌄ | Create New View

Action	Label ↑	Lead Assessment Delay Name
Edit	Del	Configuration	Configuration

Lead Assessment Delay

« Back to List: Scheduled Jobs

Lead Assessment Delay Detail   [Edit] [Delete] [Clone]

Label	Configuration
Lead Assessment Delay Name	Configuration
Minutes Delay	15
Hours Delay	0
Days Delay	0
Created By	Ryszard Rogalski, 10.9.2021 15:45

[Edit] [Delete] [Clone]

2. How to use the engine

1. Assign `Lead_Assessment` permission set to the user, configuring the engine
2. Open *Lead Assessment Type* tab

Q Lead Assessment Types   ⊗

Apps

No results

Items

Lead Assessment Types

View All

3. Create new Lead Assessment Type for certain Lead Record Types

New Lead Assessment Type

Information

Lead Assesment Type Name

Configuration

Type		Record Type		
B2C Housing Sales ▼		**Available**		**Chosen**
--None--		Tenant Candidate	▶	B2C Housing Sales
B2B Busniess Premises		Workery+		
B2B Infra		1. YIT Lead	◀	
B2C Housing Rental		2. Business Premise…		
✓ B2C Housing Sales		3. Infra Services		
		4. Municipality Servi…		

Cancel   Save & New   Save

4. Create new *Assessment Rules* for specific *Assessment Type*

5. For each Assessment Rule create
   - *Lead Assessment Condition*
   - *Lead Assessment Action*

Lead Assessment Rule
B2C Housing Condition Rule

Lead Assessment Conditions (3)   New

Lead Assessment Condition: Lead Assesment Condition Name

CONDITION-00003   ▼
CONDITION-00009   ▼
CONDITION-00011   ▼

View All

Lead Assessment Actions (2)   New

Lead Assessment Action Name

Status Qualified   ▼
Lead Cold Nurturing   ▼

View All

Leads (3)

★ Ryszard Monday   ▼   ★ Ryszard Rogalski   ▼
Company: Company:
Lead Status: Qualified   Lead Status: Qualified
Rating: Rating:
javascript:void(0) szard Monday   ▼

Verify queries and action with the use of button - otherwise the Engine may not work properly

⭐ Simple Query

Query	⌄	Rows	Validation
SELECT Id FROM Lead WHERE RecordType.Name IN ('B2C Housing Sales') AND CreatedDate >= LAST_N_DAYS:3 AND Cold_Lead_Nurturing__c = True		0	Query is valid

⭐ Complex Queries

Condition	Query	⌄	Validation
CONDITION-00009	(SELECT Id FROM Account WHERE VIP_Siebel_Id__c != null AND IsPersonAccount = TRUE AND Customer_Class__c IN ('VIP Client', 'Active', 'VIP Active', 'Potential') AND (PersonEmail = [Email] AND PersonMobilePhone = [Mobile_new__c])).size() > 0		Query is valid

≫ Actions

Action	Validation
Status Qualified	Action is valid
Lead Cold Nurturing	Action is valid

3. How to create Lead Assessment Conditions

1. Create new Lead Assessment Condition with 'New' button

⚙ **Lead Assessment Conditions (3)**   New

2. Specify Condition Filter and Condition Value using Salesforce SOQL language rules

New Lead Assessment Condition

Information

* Lead Assessment Rule

 B2C Housing Condition Rule   ✕

Lead Assesment Condition Name

Configuration

* Lead Assessment Query

Search Lead Assessment Queries...   🔍

Condition Value

%Ryszard%

Condition Filter

LIKE   ▼

Cancel   Save & New   **Save**

3. Create/Assign *Lead Assessment Query*

* Lead Assessment Query

Search Lead Assessment Queries...   🔍

＋ New Lead Assessment Query

New Lead Assessment Query

Information

Lead Assesment Query Name

* Condition Name

Query example

Owner

Ryszard Rogalski

Query Piece ⓘ

Name   ✓

Cancel   Save & New   **Save**

*Instructions about creating Lead Assessment Queries are described in the next paragraph

4. Save the configuration and verify it with '*Verify Queries*' button on *Lead Assessment Rule*

4. How to create Lead Assessment Actions

1. Create new Lead Assessment Action with 'New' button

2. Fill out the Lead field name and action value

New Lead Assessment Action

Information

*Lead Assessment Action Name	Lead Assessment Rule
Change lead field value	B2C Housing Condition Rule
Lead Field (API Name)	Lead Field Value
Cold_Lead_Nurturing__c	false

Cancel   Save & New   Save

3. Save the configuration and verify it with 'Verify Queries' button on Lead Assessment Rule

Actions

Action	Validation
Change lead field value	Action is valid
Status Qualified	Action is valid
Lead Cold Nurturing	Action is valid

4. Lead Assessment Queries - specification

Edit QUERY-001

Lead Assesment Query Name	Owner
QUERY-001	Ryszard Rogalski
*Condition Name	Query Piece
Only Cold Leads	Cold_Lead_Nurturing__c

Created By	Last Modified By
Ryszard Rogalski, 16.8.2021 15:37	Ryszard Rogalski, 2.9.2021 9:30

Cancel   Save & New   Save

1. **Simple queries**
   Simple queries are queries that are describing a specific field in *Query Piece* field
   Implementation requires filling out the proper Lead object field API Name
2. **Complex queries**
   Complex query allows user to create any query related to the Lead record
   In order to connect complex query with a lead field [field] notation is needed
   Example:

Edit QUERY-007

Lead Assesment Query Name
QUERY-007

Owner
🐻 Ryszard Rogalski

* Condition Name

Is Existing Customer VIP by Email And Phone

Query Piece ⓘ

```
SELECT Id FROM Account WHERE
VIP_Siebel_Id__c != null AND
IsPersonAccount = TRUE AND
Customer_Class__c IN ('VIP Client', 'Active', 'VIP Active',
'Potential') AND (PersonEmail = [Email] AND
PersonMobilePhone = [Mobile_new__c])
```

Created By
🐻 Ryszard Rogalski, 31.8.2021 11:23

Last Modified By
🐻 Ryszard Rogalski, 2.9.2021 10:22

[Cancel] [Save & New] [Save]

[Field] notation allows the engine to treat these values as a Merge field using Lead's fields

```
PersonEmail = [Email] AND PersonMobilePhone = [Mobile_new__c])
```

3. **Verification**

After creating *queries* and *conditions*, remember to verify them with 'Verify Queries' button.

'Verify Queries' button invokes the SOQL queries directly to the database, so validation is not executed by the engine, but by the Salesforce itself.

5. Lead Assessment Engine actions

In order to execute Lead Assessment Actions, the following conditions must be satisfied:
- Lead must be qualified by at least **one** *Lead Assessment Rule*
(Lead is qualified by *Lead Assessment Rule* whenever **all** *Lead Assessment Conditions* are satisfied under this specific *Lead Assessment Rule*)

6. Lead Assessment Endpoint request

Currently, the engine is not only executing *Lead Assessment Actions* configured under the specific *Lead Assessment Rule*. It is also executing the callout request to the Azure.
Actions specified after *Lead Assessment Qualification* by the *Lead Assessment Rules* are written under the `executePostAssessmentLogic` method inside `LeadAssessmentService` class.

It is responsible for executing *Lead Assessment Actions* under specific *Lead Assessment Rule,* which is qualifying the lead and executing `synchronizeWithLeadDistributor` method.

7. Engine's synchronization with Lead Distributor

As describer in pargraph no. 6, after Lead is qualified by *Lead Assessment Conditions* under a specific *Lead Assessment Rule* - it's data is synchronized with a Lead Distributor.

Configuration of the Lead Distributor callout is configured in the custom metadata

```
Azure_B2C_Lead_Integration_Fields.Configuration
```

Azure B2C Lead Integration Fields

Azure B2C Lead Integration Fields Detail

[Edit] [Delete] [Clone]

Label	Configuration
Azure B2C Lead Integration Fields Name	Configuration
firstName	FirstName
lastName	LastName
phone	Mobile_new__c
email	Email
leadIDPortal	Id
localities	City_Of_Interest__c
districts	District__c
areaMarketingNames	Marketing_Name__c
projectNames	Project_Name__c
leadOrigin	Lead_Origin__c
privacyAgreementAccepted	Privacy_Agreement_Accepted__c
marketingEmailAllowed	!HasOptedOutSolicit__c
projectBusinessSegment	Project_Business_Segment__c
contactWithPhone	!DoNotCall
Created By	Ryszard Rogalski, 9.9.2021 10:54

[Edit] [Delete] [Clone]

Whenever a change is required in the endpoint callout, there is a modification needed in the Custom Metadata, as well as in the `LeadAssessmentService` class - `LeadAzureIntegrationWrapper` class.

`LeadAzureIntegrationWrapper` is required since Azure endpoint fields may be arrays, and then Developer should specify this value in the `LeadAzureIntegrationWrapper` class and initialize it as a list.

```
public Boolean isColdLead {get;set;}
public LeadAzureIntegrationWrapper(){
    this.isColdLead = false;
    this.channel = Utils.COLD_LEAD_NURTURING_AZURE;
    this.projectBusinessSegment = 'Housing';
    this.countryInterface = Utils.COUNTRY_CODE_FI; //ISO 639-1 two-letter country code
    this.initializeLists();
}

public void initializeLists(){
    this.localities = new List<String>();
    this.districts = new List<String>();
    this.areaMarketingNames = new List<String>();
    this.projectNames = new List<String>();
}
```
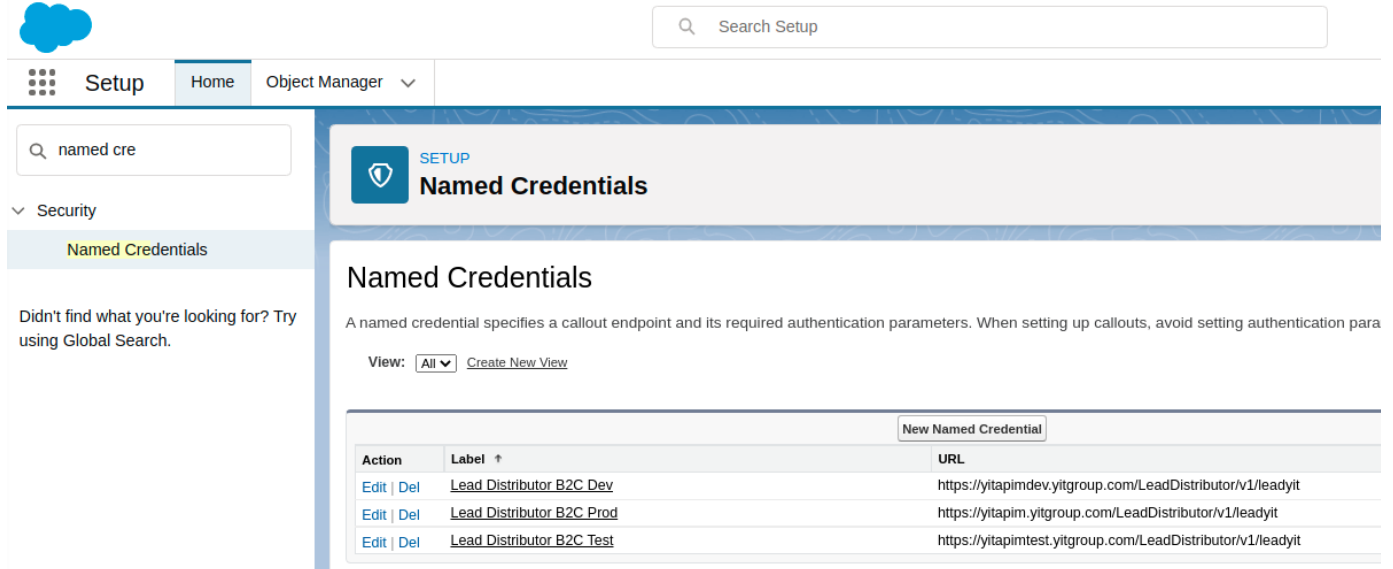
8. Azure Lead Distributior authorization - Salesforce Named Credentials

Salesforce stores Azure Lead Distributor keys under the Named Credentials

If modification of keys/urls are needed it shoud be done under this Named Credentials

SETUP
Named Credentials

Label	Lead Distributor B2C Dev
Name	Lead_Distributor_B2C_Dev
URL	https://yitapimdev.yitgroup.com/LeadDistributor/v1/leadyit

▼ Authentication

|	
Certificate	
Identity Type	Named Principal
Authentication Protocol	Password Authentication
Username	API Key
Password	••••••••

▼ Callout Options

|	
Generate Authorization Header	☐
Allow Merge Fields in HTTP Header	☑
Allow Merge Fields in HTTP Body	☐
Outbound Network Connection	

Named Credential choice in the engine is based on the Sandbox Type in the `getLeadDistributorEndpoint` method of `LeadAssessmentService` class

```
public static String getLeadDistributorEndpoint() {
    String orgType = UTILS.getOrgType();
    if(orgType == Utils.DEVELOPER){
        return Utils.LEAD_DISTRIBUTOR_DEV_CREDENTIAL;
    }else if(orgType == UTILS.DEVELOPER_PRO){
        return Utils.LEAD_DISTRIBUTOR_DEV_CREDENTIAL;
    }else if(orgType == UTILS.PARTIAL_COPY){
        return Utils.LEAD_DISTRIBUTOR_UAT_CREDENTIAL;
    }else if(orgType == UTILS.FULL_COPY){
        return Utils.LEAD_DISTRIBUTOR_UAT_CREDENTIAL;
    }else if(orgType == UTILS.PRODUCTION){
        return Utils.LEAD_DISTRIBUTOR_PROD_CREDENTIAL;
    }else{
        return null;
    }
}
```

For the *Salesforce Production* - Lead Distributor B2C *Prod* named Credential will be used.

For the *Salesforce Sandbox FullCopy/PartialCopy* - Lead Distributor B2C *Test* named Credential will be used.

For the *Salesforce Sandbox Developer/DeveloperPro* - Lead Distributor B2C *Dev* named Credential will be used.

8. Salesforce's deployment package/files

workbench.developerforce.com

Tests needed to be run during the deployment on Production:

- `LeadAssessmentServiceTest`
- `UtilsTest`

retrieve_09S5r0...0000qLQ6EAM.zip

package.xml