Push Notifications in Acumatica

Vladimir Perov
Technical Account Manager
Acumatica
vperov@acumatica.com

Table of Contents

General Information

- Feature was first introduced in 2017R2
- This demo is done on 2018R1 Update 6, but all scenarios and code examples are backward compatible

Enabling Push Notifications

- When Installing from wizard – everything is configured automatically
- When doing web install it is required to configure MSMQ on server (MSMQ is required whether you are going to use it as a notification destination or not)

Notifications overview

Notification Example

```
{
    'Inserted': [{
        'CaseID': '000115’,
        'Subject': 'Case 01 - Create Case’,
        'Description': null,
        'Status' : 'New'}],
    'Deleted': [],
    'Query': 'Push-Cases',
    'CompanyId': 'Company’,
    'Id': '2c9008fa-4120-494a-85fe-73ca039422ff',
    'TimeStamp': 63633030349128761O
}
```

Notification fields

Inserted - new rows in query.

Deleted - rows that were present in query but were removed. Comparing deleted and inserted sets will show updated fields.

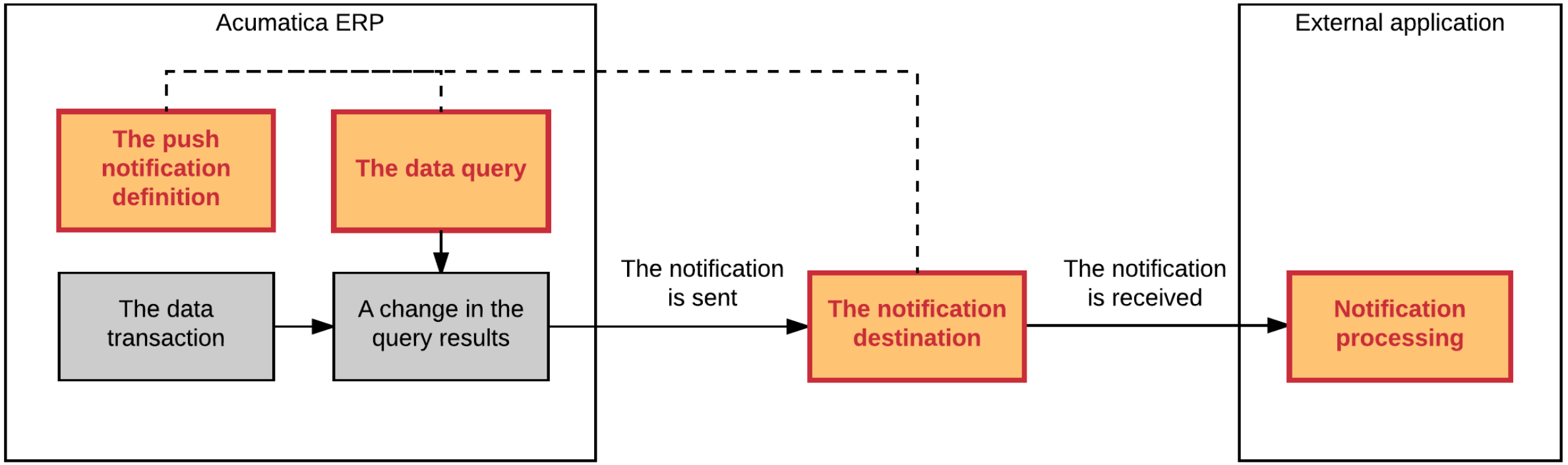
Query - name of source definition (class name for Built-In definitions or Generic Inquiry name for GI definition).

CompanyId - login name of company.

Id - transaction identifier generated on DB level. We guarantee at least one delivery, so ‘Id’ and ‘Query’ fields can be used for deduplication.

TimeStamp - value that is guaranteed to increase with every transaction. Can be used to define order.

Sending a notification



Recommendations for the Data Queries

- Use as simple a data query as possible.
- Do not use aggregation and grouping in the query.
- Do not use joins of multiple detail tables (like Sales Order – Shipments – Shipment lines, i.e. several many-to-many relationships).
- Inner joins in queries may work a bit slower than left joins.
- For a query defined by using a generic inquiry, do not use a formula on the **Results Grid** tab of the Generic Inquiry (SM.20.80.00) form.

How to configure a notification

Configuring a notification for GI

Demo

Built-in notification definitions

```
public interface IInCodeNotificationDefinition
{
	Tuple<BqlCommand, PXDataValue[]> GetSourceSelect();

	Type[] GetRestrictedFields();
}
```

Verifying built-in definition

It is possible to check data returned by built-in definition using PushNotifications endpoint:

http(s)://<Acumatica ERP instance URL>/PushNotifications/<full class name of the built-in definition>

For example:

http://localhost/2018R106/PushNotifications/PushNotificationsDemo.TestInCodeDefinition

This works only for classes that were added to a push notification definition on SM302000 screen.

Built-in notification definitions

Demo

Initial data synchronization

- GI exposed via OData
  - Set “Expose via OData” to true on Generic Inquiry definition screen
  - Access OData feed: http(s)://<Acumatica ERP instance URL>/OData/<GI Name>

  Example: http://localhost/2018R106/OData/AP-Bills%20and%20Adjustments
- Built-in definition endpoint
  - Make sure that definition is used in an active push notification
  - Access it via http(s)://<Acumatica ERP instance URL>/PushNotifications/<full class name of the built-in definition>

  Example: http://localhost/2018R106/PushNotifications/PushNotificationsDemo.TestInCodeDefinition

Built-in notification destinations

Webhooks

A webhook is an HTTP address to which Acumatica ERP sends POST requests with notification information.

Specifics:

- Webhook should return successful response (20x responses)
- It is possible to add custom headers (Authentication, API keys, etc.), but avoid adding headers that are already sent by default
- Notifications that were not sent can be found on Process Push Notifications (SM502000) screen (stored for 2 days only)

Configuring a notification with webhooks

Demo

SignalR

SignalR is Microsoft technology that enables real-time web functionality. It works as an abstraction layer over several types of transports, such as WebSockets, Server-Sent Events and long polling.

Specifics:

- Client libraries are available for .NET/Java/JavaScript/Objective-C etc.
- Less reliable than other destinations provided out of the box. If no clients are listening when notification is triggered, then it is not sent at all.

Configuring a notification with SignalR

Demo

MSMQ

Acumatica can send a notification to a local or remote private Microsoft message queue.

Specifics:

- The message queue is the most reliable destination type protected from network failures.
- Notifications that were not sent can be found on Process Push Notifications (SM502000) screen (stored for 2 days only)

Configuring a notification with MSMQ

Demo

Additional resources

Acumatica Help: https://help.acumatica.com

ASP.NET SignalR: http://signalr.net/

Q & A

Thank You!

https://adn.Acumatica.com