

mission<one>

Transactional Mail API

©2018 mission-one

Inhaltsverzeichnis

Einleitung	3
Begriffe	3
SOAP-Webservice Endpunkte	4
Authentifizierung	4
Methode Send.....	5
Programm-Beispiel in C#.....	7
Methode GetInfo	8
Abrufen der möglichen Abonnenten-Felder.....	9
Personalisierung im Newsletter anzeigen	11

Einleitung

Die mission-one **Transactional Mail API** (TMA) bietet einen Einzelmailversand per SOAP-Request. Als Mails werden Newsletter aus der mission-one dialog Anwendung verwendet, die bequem über den WYSIWYG Editor bearbeitet werden und mit Personalisierungs-Feldern bestückt werden können.

Über einen sogenannten TransactionalDispatch (Transaktions-Versand) werden einzelne Mail-Aussendungen zu einer Gruppe zusammengefasst. Dadurch ist eine Auswertung aller üblichen Kennzahlen möglich: Öffnung, Klicks, Bounces, usw.

Begriffe

Newsletter: E-Mail-Objekt mit ID, Betreff, Inhalt und Personalisierungs-Paltzhaltern

Subscriber: Empfänger der E-Mail

Field/Attribut: Datenbankfeld eines Empfängers

TransactionalDispatch: Zusammenfassung mehrere einzelner Transaktions-Mailings zu einer Einheit, um Übersichtlichkeit und eine gemeinsame Auswertung zu gewährleisten

TransactionalDispatchPublicId: Identifizierer eines TransactionDispatches

TransactionalMailPublicId: Identifizierer einer einzelnen Mail-Aussendung/eines einzelnen Aufrufs

SOAP-Webservice Endpunkte

<https://appserver.permission-one.de/TransactionalMailApi/Soap/TransactionalMail.svc>

WSDL:

<https://appserver.permission-one.de/TransactionalMailApi/Soap/TransactionalMail.svc?wsdl>

Authentifizierung

Die Authentifizierung erfolgt durch die Übergabe der CustomerId und des ApiKey im Header-Bereich, im Parameter ApiAuthentication eines jedes SOAP-Requests:

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:s="http://mission-one.de/tmaservice201708" xmlns:d="http://mission-
one.de/tmadata201708">
  <soap:Header>
    <s:ApiAuthentication>
      <s:CustomerId>[int]</s:CustomerId>
      <s:ApiKey>[STRING]</s:ApiKey>
    </s:ApiAuthentication>
  </soap:Header>
  [...]
</soap:Envelope>
```

Parameter	Datentyp	Beschreibung
CustomerId	int	Die Kunden-ID, die Sie von Ihrem Ansprechpartner erhalten
ApiKey	string	Der Api-Schlüssel, den sie von Ihrem Ansprechpartner erhalten

Methode Send

Mit dieser Methode kann ein zuvor in der mission-one dialog erstelltes Newsletter-Objekt an einen einzelnen E-Mail-Empfänger gesendet werden. Jeder Empfänger wird in der Datenbank gespeichert und nach einer bestimmten Zeitspanne wieder entfernt. Neben der E-Mail-Adresse können weitere Empfänger-**Felder/Attribute** angegeben werden – diese müssen auf der mission-one Datenbank konfiguriert sein und können später im Newsletter zur Personalisierung verwendet werden.

Es muss eine gültige **TransactionalDispatchPublicId** angegeben werden, die bestimmt, zu welchem Transaktions-Versand der Aufruf gehört. Dadurch wird gesteuert, welcher Newsletter versendet wird und es können Reaktionsdaten diesem Versand zugeordnet werden.

Es empfiehlt sich Transaktionsmails eines Typs jeweils einem Versand zuzuordnen, zB.

Bestellbestätigungen, Ausfüllen eines Formulars, Warenkorbabbruch, Rechnungsversand, ...

Hinweis: aktuell muss ein Transaktions-Versand und die **TransactionalDispatchPublicId** von mission-one angelegt werden, es gibt noch kein Interface dafür.

Es kann optional eine Newsletter-ID im Request angegeben werden – diese würde dann anstelle der voreingestellten ID versendet werden.

Außerdem kann base64-codiert ein **Anhang** mit übergeben werden. Aktuell sind jpg und pdf-Dateien erlaubt.

XML-Aufbau eines Requests:

```
<soap:Header>
  [...]
</soap:Header>
<soap:Body>
  <s:Send>
    <s:request>
      <d:TransactionalDispatchPublicId>[string]</d:TransactionalDispatchPublicId>
      <d:NewsletterId>[int?]</d:NewsletterId>
      <d:Subscriber>
        <d:Email>[string]</d:Email>
        <d:Fields>
          <d:Field i:type="d:[FieldType]">
            <d:Name>[string]</d:Name>
            <d:Value>[value]</d:Value>
          </d:Field>
        </d:Fields>
        <d:EmailAttachments>
          <d:EmailAttachment>
            <d:FileName>[string]</d:FileName>
            <d:ContentData>[base64Binary]</d:ContentData>
          </d:EmailAttachment>
        </d:EmailAttachments>
      </d:Subscriber>
    </s:request>
  </s:Send>
</soap:Body>
```

Parameter	Datentyp	Beschreibung
TransactionalDispatchPublicId	string	Die ID des Transaktions-Versandes in dem konfiguriert ist, welches Newsletter-Objekt versendet wird. Dieses ID erhalten sie derzeit noch direkt von mission-one.
NewsletterId	int	Optional: Eine Newsletter-ID, die alternativ zu dem vorkonfigurierten Newsletter-Objekt verwendet wird
Subscriber	object	Das Objekt mit allen benötigten Daten zum Empfänger
Email	string	Eine gültige Empfänger-E-Mail-Adresse
Fields	array	Optional: Ein Array aus mehreren Field-Objekten
Field	object	Ein Field enthält in der Regel Informationen des Empfängers zum Personalisieren der E-Mail Bei jedem Field muss der FieldType deklariert werden: <ul style="list-style-type: none"> - FieldString - FieldDecimal - FieldInteger - FieldDate - FieldDateTime - FieldSingleSelect - FieldMultiSelect - FieldBoolean - FieldXml
Name	string	Der Name des Fields
Value	Abhängig vom Datentyp	Wert des Feldes

XML-Aufbau einer Response:

```

<SendResponse>
  <SendResult>
    <b:Status>Success</b:Status>
    <b:DiagnosticCode i:nil="true"/>
    <b:TransactionalDispatchPublicId>[string]</b:TransactionalDispatchPublicId>
    <b:TransactionalMailPublicId>[string]</b:TransactionalMailPublicId>
  </SendResult>
</SendResponse>

```

Parameter	Beschreibung
Status	Success oder Error
DiagnosticCode	Ggf. Eine Fehlerbeschreibung
TransactionalDispatchPublicId	Die im Request übergebene TransactionalDispatchPublicId

TransactionalMailPublicId	Eindeutige Id für diesen einen Request. Darüber kann mit der Methode "GetInfo" genauere Information zu dieser einer versendeten Mail abgerufen werden.
---------------------------	--

Programm-Beispiel in C#

```
using (var client = new TransactionalMailClient("WSHttpBinding_ITransactionalMail_Test"))
{
    var authentication = new ApiAuthentication();
    authentication.CustomerId = 6;
    authentication.ApiKey = "Vr4eZKg5UGqU3JV2fF2B";

    var request = new SendRequest();
    request.TransactionaldispatchPublicId = "abc123";

    var subscriberBuffer = new Subscriber();
    subscriberBuffer.Email = "example@mission-one.de";

    var fieldsBuffer = new Fields();
    var anredeField = new FieldString();
    anredeField.Name = "anrede";
    anredeField.Value = " Sehr geehrter Herr Example";
    fieldsBuffer.Add(anredeField);

    var testDecimaField = new FieldDecimal();
    testDecimaField.Name = "versandkosten";
    testDecimaField.Value = 3.95;
    fieldsBuffer.Add(testDecimaField);

    var nachnameField = new FieldString();
    nachnameField.Name = "nachname";
    nachnameField.Value = "Example";
    fieldsBuffer.Add(nachnameField);

    subscriberBuffer.Fields = fieldsBuffer;

    request.Subscriber = subscriberBuffer;

    var response = client.Send(authentication, request);
}
```

Methode GetInfo

Mit dieser Methode kann Information über eine einzelne Versendete Mail abgerufen werden.

XML-Aufbau des Requests:

```
<soap:Header>
  [...]
</soap:Header>
<soap:Body>
  <s:GetInfo>
    <s:request>
      <d:TransactionalMailPublicId>[string]</d:TransactionalMailPublicId>
    </s:request>
  </s:GetInfo>
</soap:Body>
```

Parameter	Datentyp	Beschreibung
TransactionalMailPublicId	string	Die ID aus der Resonse des Send-Aufrufs

XML-Aufbau einer Response:

```
<GetInfoResponse>
  <GetInfoResult>
    <b:Status>Success</b:Status>
    <b:DiagnosticCode i:nil="true"/>
    <b:TransactionalMailInfo>
      <b:TransactionalDispatchPublicId>
        [string]</b:TransactionalDispatchPublicId>
      <b:TransactionalMailPublicId>[string]</b:TransactionalMailPublicId>
      <b:IsBounced>[false|true]</b:IsBounced>
      <b:BounceType i:nil="true"/>
      <b:ReactionInfo>
        <b:IsOpened>[false|true]</b:IsOpened>
        <b:IsClicked>[false|true]</b:IsClicked>
      </b:ReactionInfo>
    </b:TransactionalMailInfo>
  </GetInfoResult>
</GetInfoResponse>
```

Parameter	Beschreibung
Status	Success oder Error
DiagnosticCode	Ggf. Eine Fehlerbeschreibung
TransactionalDispatchPublicId	Die zugehörige TransactionalDispatchPublicId
TransactionalMailPublicId	Die verwendete TransactionalMailPublicId, die genau einen Request/Mailversand/Empfänger identifiziert
IsBounced	Information, ob die Mail bounced ist oder nicht
BounceType	SoftBounce oder Hardbounce
ReactionInfo IsOpened	Angabe, ob die Mail geöffnet wurde
ReactionInfo IsClicked	Angabe, ob die Mail geklickt wurde

Abrufen der möglichen Abonnenten-Felder

Um alle möglichen Felder zur Personalisierung im Newsletter abzurufen steht folgender SOAP-Webservice zur Verfügung:

<https://appserver.permission-one.de/SubscribersService/CustomerInterfaceV2.asmx>

WSDL:

<https://appserver.permission-one.de/SubscribersService/CustomerInterfaceV2.asmx?WSDL>

Die Authentifizierung erfolgt wie beim TMA-Service, siehe oben.

Die Methode **AttributesGet** liefert alle Felder der Abonnenten-Datenbank mit zusätzlichen Informationen zurück.

Hinweis: folgende Felder können und dürfen nicht überschrieben werden:

- sbr_subscriberid
- publicid
- consent_public_id_timestamp
- consent_public_id

XML-Aufbau des Requests:

```
<soap:Header>
  [...]
</soap:Header>
<soap:Body>
  <s:AttributesGet>
  </s:AttributesGet>
</soap:Body>
```

XML-Aufbau einer Response (Beispiel-Auszug):

```
<AttributesGetResponse>
  <AttributesGetResult>
    <CustomerAttribute>
      <Name>prj_projectid</Name>
      <DataType>INTEGER</DataType>
      <DisplayName>Project</DisplayName>
      <Description>Project</Description>
      <DefaultValues/>
      <StringLength>-1</StringLength>
      <IsRange>>false</IsRange>
      <HasRegularExpressions>>false</HasRegularExpressions>
      <IsUnique>>false</IsUnique>
      <IsMandatory>>true</IsMandatory>
      <HasAllowedValues>>false</HasAllowedValues>
      <IsOnInputForm>>true</IsOnInputForm>
    </CustomerAttribute>
  </AttributesGetResult>
</AttributesGetResponse>
```

```

    <Name>sbr_subscriberid</Name>
    <DataType>INTEGER</DataType>
    <DisplayName>AbonnentID</DisplayName>
    <Description>AbonnentID</Description>
    <DefaultValues/>
    <StringLength>-1</StringLength>
    <IsRange>>false</IsRange>
    <HasRegularExpressions>>false</HasRegularExpressions>
    <IsUnique>>true</IsUnique>
    <IsMandatory>>false</IsMandatory>
    <HasAllowedValues>>false</HasAllowedValues>
    <IsOnInputForm>>false</IsOnInputForm>
  </CustomerAttribute>
  [...]
  <CustomerAttribute>
    <Name>vorname</Name>
    <DataType>STRING</DataType>
    <DisplayName>Vorname</DisplayName>
    <Description/>
    <DefaultValues/>
    <StringLength>255</StringLength>
    <IsRange>>false</IsRange>
    <HasRegularExpressions>>false</HasRegularExpressions>
    <IsUnique>>false</IsUnique>
    <IsMandatory>>false</IsMandatory>
    <HasAllowedValues>>false</HasAllowedValues>
    <IsOnInputForm>>true</IsOnInputForm>
  </CustomerAttribute>
  <CustomerAttribute>
    <Name>nachname</Name>
    <DataType>STRING</DataType>
    <DisplayName>Nachname</DisplayName>
    <Description/>
    <DefaultValues/>
    <StringLength>255</StringLength>
    <IsRange>>false</IsRange>
    <HasRegularExpressions>>false</HasRegularExpressions>
    <IsUnique>>false</IsUnique>
    <IsMandatory>>false</IsMandatory>
    <HasAllowedValues>>false</HasAllowedValues>
    <IsOnInputForm>>true</IsOnInputForm>
  </CustomerAttribute>
  [...]
</AttributesGetResult>
<ErrorMessage/>
</AttributesGetResponse>

```

Beim Aufruf der TMA-Send-Methode müssen dann die entsprechenden Name und Value-Werte übertragen werden. Zum Beispiel:

```

<d:Fields>
  <d:Field i:type="d:FieldString">
    <d:Name>vorname</d:Name>
    <d:Value>Max</d:Value>
  </d:Field>
</d:Fields>

```

Diese Werte können dann in der versendeten Mail ausgelesen werden.

Personalisierung im Newsletter anzeigen

In mission-one dialog oder dialog+ können Abonnenten-Felder zum Newsletter hinzugefügt werden, die dann durch die personalisierten Werte des Empfängers ersetzt werden:

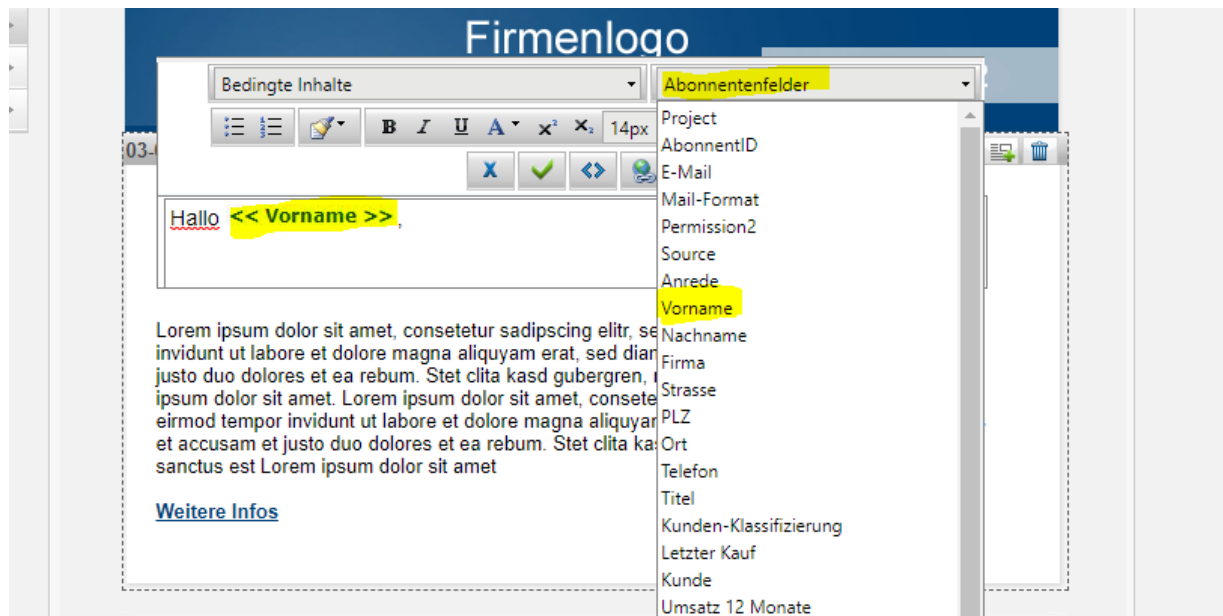


Bild 1: Einfügen von Personalisierungs-Feldern in mission-one dialog

erstellen

letter erstellen

Übersicht Links Einstellungen Test Aktionen

← → B I U ↺ Ω ↻ ↷ H ≡ ≡ ☞ ☞

Preheader
Falls Sie keine Bilder sehen, klicken Sie bitte [hier](#).

Firmenlogo

600 x 112

Hallo \$\$sbrattributes.vorname\$\$ Vorname

li European lingues es membres del sam familie. Lor separat existentie es un myth. Por scientie, musica, sport etc, litot Europa usa li sam vocabular. Li lingues differe solmen in li grammatica, li prononciation e li plu commun vocabules. Omnicos directe al desirabilite de un nov lingua franca: On refusa continuar pagar custosi traductores. At solmen va esser necessari far uniform grammatica, prononciation e plu sommun paroles. Ma quande lingues coalesce, li grammatica del resultant linque es plu simplic e regulari quam ti del coalescent lingues.

Viel Spaß beim Lesen,
Lorem Ipsum

Abonnenatenattribute

- Nachname
- Permission2
- permissiontext
- Project
- Public-ID
- shoptracker_vouchercodes
- singleselect
- Source
- Attribut 2
- Attribut 3
- Straße
- PLZ
- Nachname
- Vorname

Bild 2: Einfügen von Personalisierungs-Feldern in der mission-one dialog+