Exporting Collections Matters to LegalSuite

25 May 2024

Debt collection often involves recovering debts from a large number of Debtors for a particular Client. Manually capturing these Matters in the LegalSuite program can be an onerous and time-consuming task for the LegalSuite user. In many cases, however, the basic data required to open each file is available in the client’s database or credit control system.

To automate the process of handing over bulk Matters for Debt Collection, LegalSuite has a feature that allows a third party to send data to an attorney which is then imported into the LegalSuite database.

The process involves the third party exporting the debtor information from their system to the LegalSuite API, where it becomes available for a LegalSuite user to view and then import into the LegalSuite program. Once imported, the user can inspect the data, add any additional information that may be required, and start the debt collection process.

This can be of great benefit if a client has numerous delinquent debtors that need to be handed over. It not only avoids the onerous task of taking on each Matter manually, but also avoids data-capturing errors and ensures the debt collection process can be initiated much sooner.

**Exporting the Data**

The data must be sent in an XML format.

The XML data will be validated against an XSD before it is imported into the LegalSuite system. The XSD file can be downloaded from [matters.xsd](https://api.legalsuite.net/export/matters.xsd).

If the XML data does not conform to the rules contained in the XSD, it will be rejected. You should always validate your XML against the XSD before exporting it. You can do this programmatically or use an online tool such as [XML Validator](https://www.liquid-technologies.com/online-xsd-validator).

An example of a valid XML file can be downloaded from [matters.xml](https://api.legalsuite.net/export/matters.xml).

**Notes:**

• The XSD has comments (annotations) to indicate which fields are compulsory (indicated by minOccurs="1") and what data is expected in each field.

• Any elements in the XML must appear in the *sequence* as specified in the XSD, i.e. one after each other in the sequence specified in the XSD.

• All dates must be in the YYYY-DD-MM format.

**The basic structure of the XML**

The XML contains a <matters> element that wraps the entire XML.

The <matters> element has two important *header* elements: <clientCode> and <employeeCode> which as described below.

Inside the <matters> element, there can be multiple <matter> elements which contain the basic details of each delinquent account (a Matter on the LegalSuite system), such as the description, your reference, the amount outstanding etc.

Each <matter> element must have at least one <party> child element.

If the <role> of the <party> element is set to “Defendant”, it should contain the debtor’s personal information, such as their name, address, language and entity (e.g. male, female or business).

You *can* specify the Plaintiff by adding another <party> element with the <role> set to “Plaintiff”, but in most cases, the Plaintiff is the Client. If a Plaintiff is not specified, the Client will be the Plaintiff on each Matter, and typically, it is not necessary to specify the Plaintiff.

Each <party> element can have one or more <contact> child elements which provide the contact information for each Party such as their phone numbers and email address.

The basic structure looks like this:

<matters>

<clientCode>  
<employeeCode>

<matter>

<party>

<contact>

</contact>

</party>

</matter>

<matter>

<party>

<contact>

</contact>

</party>

</matter>

</matters>

**Matters Element**

These two elements in the <matters> element identify the Client and the Employee *on the LegalSuite system*:

1. The **clientCode** element is used to identify *your* company in their Address Book;
2. The **employeeCode** element is the User on *their* system who requested the data and who will be notified that the data has been sent so they can import it into their LegalSuite system.

Example:

<clientCode>FIN1</clientCode>  
<employeeCode>JD</employeeCode>

**Note:** These will be provided to you by the LegalSuite user beforehand.

**Matter Element**

The <matter> elements represent the basic information of each delinquent account.

<matter>

<description>Unpaid account: Mr R Jones</description>  
<reference>1234-2345-0001</reference>  
<claimAmount>5467.89</claimAmount>  
<interestRate>15.75</interestRate>  
<causeOfAction>Account over 90 days</causeOfAction>  
<dateOfDebt>2024-05-01</dateOfDebt>  
<interestFrom>2024-05-01</interestFrom>  
<language>English</language>

. . .

</matter>

**Note:** You can send an unlimited number of <matter> elements.

**Party Element**

The <party> element is a *child* of the <matter> element and represents the Party’s personal information. You can have an unlimited number of <party> elements.

<party>

<role>Defendant</role>  
<type>Male</type>  
<language>English</language>  
<name>Jones</name>  
<firstName>Peter</firstName>  
<title>Mr.</title>  
<identityNumber>3452342342</identityNumber>  
<postalLine1>PO Box 123</postalLine1>  
<postalLine2>Cape Town</postalLine2>  
<postalCode>7100</postalCode>

<contact>

<type>Work</type>  
<value>212-555-1234</value>

</contact>

<contact>

<type>Mobile</type>  
<value>456-345-9012</value>

</contact>

<contact>

<type>Email</type>  
<value>johnjones@gmail.com</value>

</contact>

</party>

**Contact Element**

A <party> element can contain zero or more <contact> elements which provide the contact information for each Party such as their phone numbers and email address.

**Exporting the Data**

Data is exported by sending an HTTP POST request to the following Url:

http://api.legalsuite.net/export/matters?companyCode=ACME01

Every firm that uses LegalSuite, is assigned a unique company code. The **companyCode** in the URL identifies the LegalSuite client who is to receive the data. This will be provided to you by the LegalSuite client, or you can get it from our customer support team.

To access the LegalSuite API, you must register as a *Developer* on our system. Once registered, you will be provided with an *api key,* which must be sent in the header of the POST request to authenticate you and give you the necessary permissions.

Before exporting the XML, you should validate it against the [matters.xsd](https://api.legalsuite.net/export/matters.xsd) otherwise it may be rejected by the LegalSuite API. If it is rejected, you can simply fix the XML and send it again.

Here is an example of a POST request using cUrl in PHP:

$apiUrl = 'https://api.legalsuite.net/export/matters?companyCode=ACME01';

$httpHeader = [

  'Authorization: <your api key>',

  'Content-Type: application/xml'

];

$xmlData = '<matters>...your XML data goes here...</matters>';

$curl = curl\_init();

curl\_setopt\_array($curl, [

CURLOPT\_URL => $apiUrl,

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => '',

CURLOPT\_SSL\_VERIFYPEER => false,

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 30000,

 CURLOPT\_CONNECTTIMEOUT => 30000,

 CURLOPT\_FOLLOWLOCATION => true,

 CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

 CURLOPT\_CUSTOMREQUEST => 'POST',

 CURLOPT\_POSTFIELDS => $xmlData,

CURLOPT\_HTTPHEADER => $httpHeader,

 ]);

 $curlResponse = curl\_exec($curl);

if (curl\_errno($curl)) {

    $response['errors'] = 'Curl Error No: ' . curl\_errno($curl);

    curl\_close($curl);

    return $response;

} else {

$response = json\_decode($curlResponse);

}

 curl\_close($curl);

return json\_encode($response);

**Testing your export**

ACME01 is a test company on our system which you can use to test your code. Please contact us if you would like to send some test data to ACME01 and we will give you a clientCode and employeeCode you can use.

**Conclusion**

If you need any assistance, please contact support at LegalSuite Software and we will be glad to help.

**Note**: This document may be updated from time to time and can be downloaded by clicking [here](https://api.legalsuite.net/export/Exporting%20Collections%20Matters%20to%20LegalSuite.docx).