



# SUSHI Web Service User Guide

COUNTER-SUSHI Web Service

Version 1.3



---

## Table of Contents

<b>1.0</b>	<b>OVERVIEW</b>	<b>3</b>
1.1	ABOUT SUSHI	3
1.2	ABOUT COUNTER	3
<b>2.0</b>	<b>ACCESSIBLE REPORTS</b>	<b>4</b>
<b>3.0</b>	<b>AUTHENTICATION</b>	<b>4</b>
3.1	IP ADDRESS	5
3.2	REQUESTOR ID	5
3.3	CUSTOMER ID	5
<b>4.0</b>	<b>WSDL AND SERVICE ENDPOINT ADDRESS/URL</b>	<b>5</b>
<b>5.0</b>	<b>MAKING MANUAL REQUESTS TO THE SUSHI WEB SERVER</b>	<b>6</b>
5.1	USING SOAPUI	6
5.2	USING LIQUID XML STUDIO – FREE COMMUNITY EDITION	10
<b>6.0</b>	<b>SCHOLARLY IQ CONTACT INFORMATION</b>	<b>12</b>
<b>APPENDIX A</b>		<b>13</b>
	SAMPLE SOAP REQUEST	13
<b>APPENDIX B</b>		<b>15</b>
	SAMPLE SOAP RESPONSE	15
<b>APPENDIX C</b>		<b>18</b>
	WSDL FOR SCHOLARLY IQ SUSHI WEB SERVICE	18
<b>ADDITIONAL RESOURCES</b>		<b>20</b>

## 1.0 Overview

The Scholarly iQ SUSHI Web Service complies with version 1.6 of the Standardized Usage Statistics Harvesting Initiative (SUSHI) protocol standard (ANSI/NISO Z39.93.2007) for obtaining electronic resource usage data. It follows the Simple Object Access Protocol (SOAP) for message exchange via the Web.

In order to use this service, you must have a SUSHI client that can send and receive SOAP messages conforming to the SUSHI 1.6 standard. In addition, you must be provided access by Scholarly iQ, using a unique authorization key (Requestor ID).

### 1.1 About SUSHI

“The *Standardized Usage Statistics Harvesting Initiative (SUSHI) Protocol* standard ([ANSI/NISO Z39.93-2007](#)) defines an automated request and response model for the harvesting of electronic resource usage data utilizing a Web services framework. It is intended to replace the time-consuming user-mediated collection of usage data reports.

The protocol was designed to be both generalized and extensible, meaning it could be used to retrieve a variety of usage reports. An extension designed specifically to work with COUNTER reports is provided with the standard, as these are expected to be the most frequently retrieved usage reports.

The standard is built on SOAP (Simple Object Access Protocol) for transferring request and response messages. The *GetReport* method is used for transferring *ReportRequest* as the input message and returning *ReportResponses* the output message.

The standard includes a versioned Web Services Description Language (WSDL), to describe the Web service namespace and operations, and a generalized XML schema with the syntax of the SUSHI protocol. Rules for report naming are outlined and complemented by an external reports registry, which provides for the definition of both COUNTER and non-COUNTER reports.”

Source: SUSHI website – <http://www.niso.org/workrooms/sushi>

### 1.2 About COUNTER

“Launched in March 2002, COUNTER (Counting Online Usage of Networked Electronic Resources) is an international initiative serving librarians, publishers and intermediaries by setting standards that facilitate the recording and reporting of online usage statistics in a consistent, credible and compatible way. The first COUNTER *Code of Practice*, covering online journals and databases, was published in 2003. COUNTER's coverage was extended further with the launch of the *Code of Practice* for online books and reference works in 2006. The body of COUNTER compliant usage statistics has steadily grown as more and more vendors have adopted the COUNTER *Codes of Practice*. This has contributed to the new discipline of usage bibliometrics and a great deal of work is underway to try to establish .value metrics. Associated with usage, in which the

COUNTER compliant statistics play an increasingly important role.”

Source: COUNTER website – <http://www.niso.org/workrooms/sushi/#counter>

“NISO and COUNTER have signed a Memorandum of Understanding that assigns the responsibility for maintaining the XML schema version of COUNTER reports to NISO. This will ensure that the SUSHI and COUNTER schemas are always synchronized.”

Source: SUSHI website – <http://www.niso.org/workrooms/sushi>

## 2.0 Accessible Reports

The following reports may be obtained via the Scholarly iQ SUSHI Web Service. These reports conform to the COUNTER Code of Practice for Journals and Databases Release.

Report ID	Report Name	Description
JR1	Journal Report 1	Number of successful full-text article requests by month and journal
JR1a	Journal Report 1a	Number of successful full-text article requests from an archive by month and journal
JR3	Journal Report 3	Number of Successful Item Requests and Turnaways by Month, Journal and Page-Type
JR4	Journal Report 4	Total Searches Run by Month and Service
JR5	Journal Report 5	Number of Successful Full-Text Article Requests by Year and Journal
DB1	Database Report 1	Total searches and sessions by month and database
DB3	Database Report 3	Total searches and sessions by month and service
CR1	Consortium Report 1	Number of successful full-text journal article or book chapter requests by month
CR2	Consortium Report 2	Total searches by month and database
BR1	Book Report 1	Number of Successful Title Requests by Month and Title
BR2	Book Report 2	Number of Successful Section Requests by Month and Title

## 3.0 Authentication



---

To successfully access Scholarly iQ's SUSH Web Service, the following credentials will be required:

- IP Address
- A unique authorization key (Requester ID)
- Customer ID

The authorization process entails the validation of the authorization key provided in the SUSHI request sent to Scholarly iQ's servers. In addition, the process will verify the requestor's IP addresses and the reports the requestor is able to access.

### **3.1 IP Address**

The IP address of the requesting SUSHI client must be registered with Scholarly iQ to be granted access. Either the platform or Scholarly iQ should be notified prior to initiating SUSHI requests.

### **3.2 Requestor ID**

The authorization key is a 128-bit integer (16 bytes) GUID that is generated upon registering with Scholarly iQ's SUSHI service. This can be provided by the platform (i.e. publisher) or Scholarly iQ.

### **3.3 Customer ID**

The Customer ID is an identifier that is platform specific and dictates which specific account (i.e. institution or consortium) is requesting usage statistics during harvesting. Scholarly iQ or the platform can provide the CustomerID(s) required.

Feel free to contact our SUSHI Web Service Administrator listed in Section 6 down below, send an email to [support@scholarlyiq.com](mailto:support@scholarlyiq.com), and/or visit our web site at [www.scholarlyiq.com](http://www.scholarlyiq.com) for more information.

## **4.0 WSDL and Service Endpoint Address/URL**

The WSDL may be retrieved from:  
[http://sushi.scholarlyiq.com/docs/counter\\_sushi3\\_0.wsdl](http://sushi.scholarlyiq.com/docs/counter_sushi3_0.wsdl)

For your convenience, the WSDL is printed in this guide (Appendix C).

The web service endpoint address for the Scholarly iQ SUSHI Web Service is:  
<http://sushi.scholarlyiq.com/sushiservice.asmx>.

## 5.0 Making Manual Requests to the SUSHI Web Server

Making a call to the Scholarly iQ SUSHI Web Service manually will require the following:

A third party tool that is capable of making SOAP calls. Listed below are just a few examples:

- SoapUI (<http://www.soapui.org/>)
- Liquid XML (<http://www.liquid-technologies.com/xmlstudio/xmlstudio.aspx>)
- Serials Solutions Open-Source SUSHI client (<http://code.google.com/p/sushicounterclient>)
  - This tool will convert an XML response from a SUSHI server to .CSV format
  - This tool currently supports JR1, DB1 and DB3
- University of Pennsylvania Library “Penn Libraries” Sushi Web Client (<https://labs.library.upenn.edu/SushiWebClient/SushiCall>)

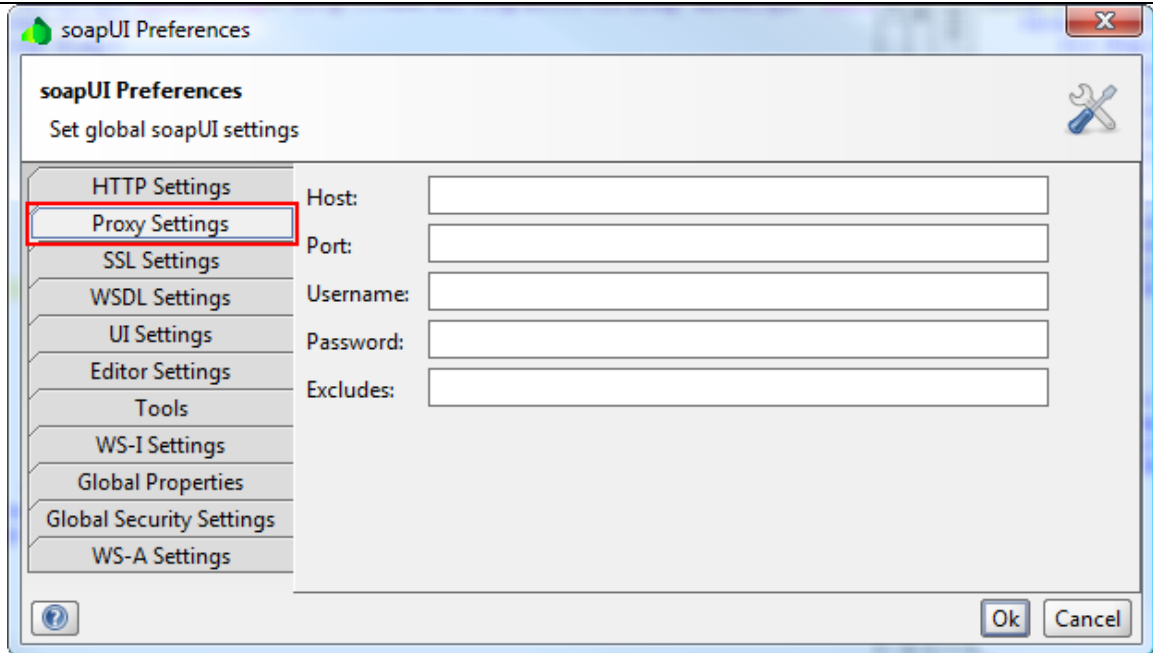
Scholarly iQ provides two URL options for making the SUSHI request:

- <http://sushi.scholarlyiq.com/sushiservice.wsdl>
- <http://sushi.scholarlyiq.com/sushiservice.asmx?wsdl>

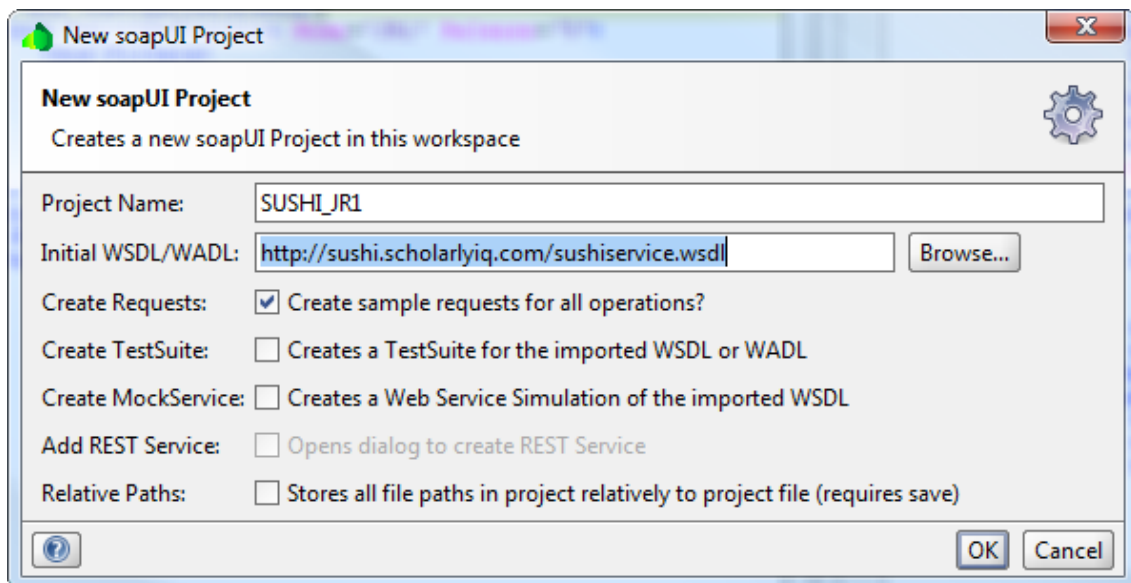
Scholarly iQ recommends a single request should be limited to a date range of one month and no more than 1 calendar year for improved response times.

### 5.1 Using SoapUI

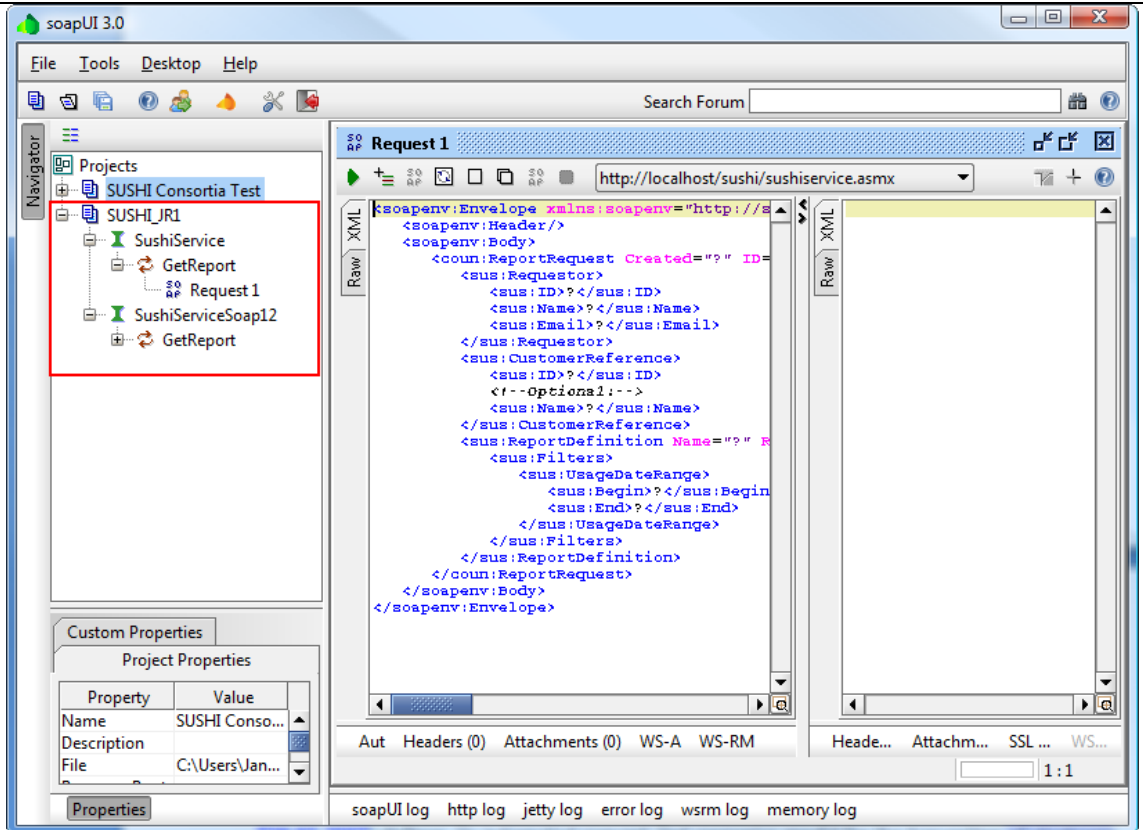
Download SoapUI. You only need the “free” version which should work on any operating system that has Java installed. If you are operating behind a proxy service, you will need to tell SoapUI by going to File → Preferences → Proxy Settings (or ctrl+alt-p) and entering the required information.



Start the program. Click on File → "New WSDL Project." Give the project a name. For example, if you want to request a JR1 report, you may want to call the project "SUSHI\_JR1". In this tutorial, it will be called "SUSHI\_JR1." For "Initial WSDL/WADL" use the following Url: <http://sushi.scholarlyiq.com/sushiservice.wsdl>



Since the WSDL provides enough information to fully describe the service, a sample soap request will be automatically generated for you by SoapUI. Once you press "OK", a project will be created on the left in the "Navigator" window.




Expand SUSHI\_JR1 → SushiService → getReport. Double-click on “Request 1.” A window should appear that contains two sub windows. The one on the left is the SOAP request while the one on the right is the response. In the left window, the project generated a template SOAP request.

To get the request to work with the Scholarly iQ SUSHI web service, you’ll need to replace some of the question marks (“?”) with relevant data.

At the very least, the following elements should be filled in:

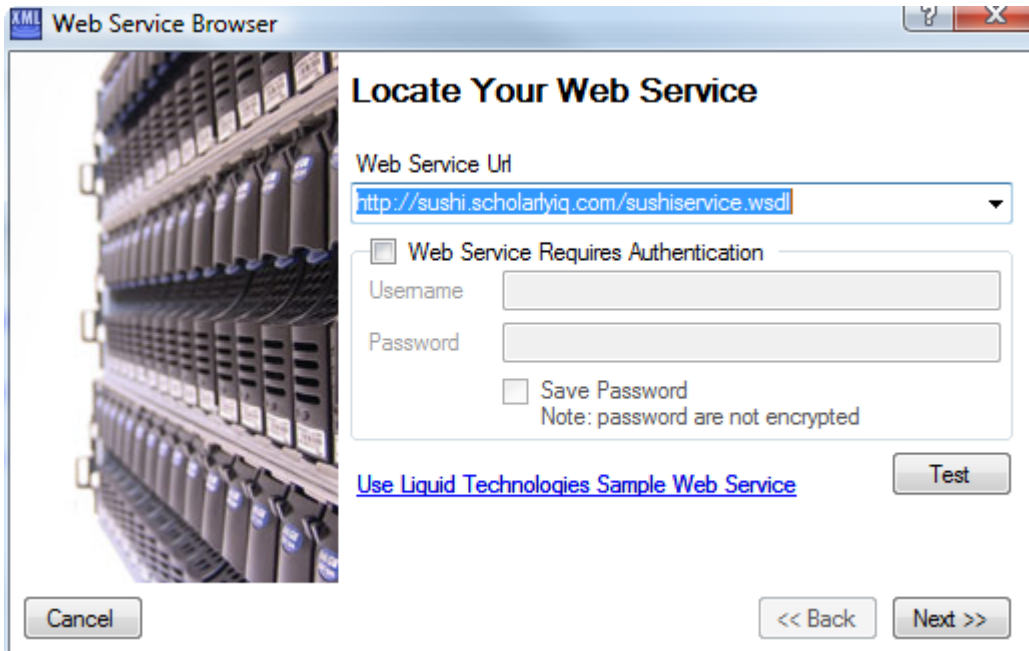
1. The ID in the Requestor element should have the unique identifier (GUID) that was supplied to you by Scholarly IQ. Note that this field is “case” sensitive and all the alphabetic characters should be in UPPERCASE.
2. The ID in the “CustomerReference” element will be the Customer Reference ID that was issued to you by email from Scholarly IQ.
3. The Name attribute in “ReportDefinition” must reference a valid counter report (JR1, DB1, CR1, etc.). the Scholarly IQ Web Service currently supports JR1, JR1a, JR3, JR4, CR1, CR2, DB1 and DB3 reports (release 3). Support for other reports will be made available in the future.
4. The Release attribute in ReportDefinition must be a valid release for the specified counter report (1, 2, etc.). Scholarly IQ currently supports Release 3.
5. The Begin and End elements of “UsageDateRange” must have a valid date range of the form YYYY-MM-DD. Statistics for a given month typically become available within the first two weeks of the subsequent month.

After supplying the values, press the “play” button  on the upper left corner of the request window. After a few seconds, you should receive the response in the right response window. Although home-grown clients are necessary to produce the information, SoapUI is a great tool to supply the XML output which can then be imported into another system for processing.

## 5.2 Using Liquid XML Studio – Free Community Edition


Using Liquid XML Studio is similar to the functionality of SoapUI in the sense that it provides the same request and response to and from the Scholarly iQ SUSHI web service. To use Liquid XML Studio, visit their website and download from here: <http://www.liquid-technologies.com/xmlstudio/xmlstudio.aspx>.

Once installed, you can begin to make a call to the Scholarly iQ SUSHI Web Service by opening the Web Service Browser window: Tools → Browse Web Service



- Enter the Scholarly iQ SUSHI web service Url <http://sushi.scholarlyiq.com/sushiservice.wsdl> in the Web service Url field. Press the “Test” button to verify that you are able to connect successfully to the web service. If you do not get a successful test, the reason might be that your IP has not been granted permission to use the service. If this happens, please contact Scholarly iQ’s Support Group.
- Press “Ok” and you will be presented with the request elements to begin your call to the Scholarly iQ SUSHI web service.

- At the very least, the following elements should be filled in:
  - The ID in the Requestor element should have the unique identifier (GUID) that was supplied to you by Scholarly iQ. Note that this field is “case” sensitive and all the alphabetic characters should be in UPPERCASE.
  - The ID in the “CustomerReference” element will be the Customer Reference ID that was issued to you by email from Scholarly iQ.
  - The Name attribute in “ReportDefinition” must reference a valid counter report (JR1, DB1, CR1, etc.). The Scholarly iQ Web Service currently support the following reports: JR1, JR1a, JR3, JR4, JR5, CR1, CR2, DB1 and DB3 (release 3). Support for BR1 and BR2 book reports (release 1) are also available for harvesting. Support for other reports will be made available in the future.
  - The Release attribute in *ReportDefinition* must be a valid release for the specified counter report (1, 2, etc.). Scholarly iQ currently supports Release 3.
  - The Begin and End elements of “*UsageDateRange*” must have a valid date range of the form YYYY-MM-DD. Statistics for a given month typically become available within the first two weeks of the subsequent month.

After filling in the values, press the “Execute” button  on the screen above the request window. After a few seconds, you should receive the response. Although home-grown clients are necessary to produce the information, Liquid XML Studio is a great tool to supply the XML output which can then be imported into another system for processing.

Whether you use SoapUI, Liquid XML Studio or another 3<sup>rd</sup> party client, the same results should be achieved.

## 6.0 Scholarly iQ Contact Information

SUSHI Support Email(s)	<a href="mailto:support@scholarlyiq.com">support@scholarlyiq.com</a> OR <a href="mailto:john.milligan@scholarlyiq.com">john.milligan@scholarlyiq.com</a>
SUSHI Contact Phone	(210) 846-3049
SUSHI Administrator	John Milligan
Scholarly iQ Web Site	<a href="http://www.scholarlyiq.com">http://www.scholarlyiq.com</a>

## Appendix A

### Sample SOAP Request

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tnsb:ReportRequest
xmlns:tns="http://www.niso.org/schemas/counter"
xmlns:tnsa="http://www.niso.org/schemas/sushi"
xmlns:tnsb="http://www.niso.org/schemas/sushi/counter"
xmlns:tnsc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" Created="2009-
10-22T11:51:26.02Z" ID="A000000-AAAA-00AA-AAAA-AA0A000000AA">
      <tnsa:Requestor>
        <tnsa:ID>A000000-AAAA-00AA-AAAA-AA0A000000AA</tnsa:ID>
        <tnsa:Name>Scholarly iQ</tnsa:Name>
        <tnsa:Email>sushisupport@scholarlyiq.com</tnsa:Email>
      </tnsa:Requestor>
      <tnsa:CustomerReference>
        <tnsa:ID>Site/InstitutionID</tnsa:ID>
        <!--Optional -->
        <tnsa:Name>Scholarly iQ</tnsa:Name>
      </tnsa:CustomerReference>
      <tnsa:ReportDefinition Name="JR1" Release="3">
        <tnsa:Filters>
          <tnsa:UsageDateRange>
            <tnsa:Begin>2009-09-01</tnsa:Begin>
            <tnsa:End>2009-09-30</tnsa:End>
          </tnsa:UsageDateRange>
        </tnsa:Filters>
      </tnsa:ReportDefinition>
    </tnsb:ReportRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

#### Notes:

1. The ID attribute in the <Requestor> element should have the unique identifier (GUID) that was supplied to you by Scholarly iQ.
2. The Name and Email attributes in the <Requestor> element will automatically be replaced by the Name and Email attributes in the SOAP Response with data from your account. These attributes in the requestor element may be left blank.

3. The ID attribute in the <CustomerReference> element should have the Site/Institution ID that was supplied to you by Scholarly iQ or the Consortium Member ID.
4. The Name field in the <CustomerReference> element is an optional field and may be customized by the client (e.g. Client Name).
5. The <ReportDefinition> Name field should have the name of the report that the client would like to process. Available reports are: JR1, JR1a, JR3, JR4, DB1, DB3, CR1, CR2, BR1 and BR2. The report types that are available for the client's account can be obtained from Scholarly iQ.
6. The Release attribute in <ReportDefinition> must be a valid release number for the specified counter report (1, 2, etc.). Scholarly iQ currently supports Release 3.
7. The Begin and End fields of <UsageDateRange> must have a valid date range of the form YYYY-MM-DD. Statistics for a given month typically become available within the first two weeks of the subsequent month.

## Appendix B

### Sample SOAP Response

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <rr:ReportResponse Created="2009-11-01T12:35:24.24Z"
ID="A0000000-AA00-00AA-A000-AA0A000000AA"
xmlns:rr="http://www.niso.org/schemas/sushi/counter">
      <Requestor xmlns="http://www.niso.org/schemas/sushi">
        <ID>A0000000-AA00-00AA-A000-AA0A000000AA</ID>
        <Name>Scholarly IQ</Name>
        <Email>john.milligan@scholarlyiq.com</Email>
      </Requestor>
      <CustomerReference xmlns="http://www.niso.org/schemas/sushi">
        <ID>A265547</ID>
        <Name>Kochi Univ of Technology</Name>
      </CustomerReference>
      <ReportDefinition Name="JR1" Release="3"
xmlns="http://www.niso.org/schemas/sushi">
        <Filters>
          <UsageDateRange>
            <Begin>2009-09-01</Begin>
            <End>2009-09-30</End>
          </UsageDateRange>
        </Filters>
      </ReportDefinition>
      <Report xmlns="http://www.niso.org/schemas/sushi/counter">
        <Report ID="JR1:3" Version="3" Name="JR1" Title="Journal
Report 1" xmlns="http://www.niso.org/schemas/counter">
          <Vendor>
            <ID>A0000000-AA00-00AA-A000-AA0A000000AA</ID>
            <Contact>
              <Contact>Scholarly IQ Support</Contact>
              <E-mail>sushisupport@scholarlyiq.com</E-mail>
            </Contact>
            <WebSiteUrl>http://www.scholarlyiq.com</WebSiteUrl>
          </Vendor>
          <Customer>
            <ID>Kochi Univ of Technology</ID>
            <WebSiteUrl/>
            <ReportItems>
              <ItemIdentifier>
                <Type>Online_ISSN</Type>
                <Value>1089-7550</Value>
              </ItemIdentifier>
            </ReportItems>
          </Customer>
        </Report>
      </Report>
    </rr:ReportResponse>
  </soap:Body>
</soap:Envelope>
```

```
<ItemPlatform>Scitation</ItemPlatform>
<ItemPublisher>AIP</ItemPublisher>
<ItemName>Journal of Applied Physics</ItemName>
<ItemDataType>Journal</ItemDataType>
<ItemPerformance>
  <Period>
    <Begin>2009-09-01</Begin>
    <End>2009-09-30</End>
  </Period>
  <Category>Requests</Category>
  <Instance>
    <MetricType>ft_html</MetricType>
    <Count>0</Count>
  </Instance>
  <Instance>
    <MetricType>ft_pdf</MetricType>
    <Count>20</Count>
  </Instance>
  <Instance>
    <MetricType>ft_ps</MetricType>
    <Count>0</Count>
  </Instance>
  <Instance>
    <MetricType>ft_total</MetricType>
    <Count>20</Count>
  </Instance>
</ItemPerformance>
</ReportItems>
<ReportItems>
  <ItemIdentifier>
    <Type>Online_ISSN</Type>
    <Value>1089-490X</Value>
  </ItemIdentifier>
  <ItemPlatform>Scitation</ItemPlatform>
  <ItemPublisher>APS</ItemPublisher>
  <ItemName>Physical Review C</ItemName>
  <ItemDataType>Journal</ItemDataType>
  <ItemPerformance>
    <Period>
      <Begin>2009-09-01</Begin>
      <End>2009-09-30</End>
    </Period>
    <Category>Requests</Category>
    <Instance>
      <MetricType>ft_html</MetricType>
      <Count>0</Count>
    </Instance>
    <Instance>
      <MetricType>ft_pdf</MetricType>
      <Count>0</Count>
    </Instance>
```

```
<Instance>
  <MetricType>ft_ps</MetricType>
  <Count>0</Count>
</Instance>
<Instance>
  <MetricType>ft_total</MetricType>
  <Count>0</Count>
</Instance>
</ItemPerformance>
</ReportItems>
<ReportItems>
  <ItemIdentifier>
    <Type>Online_ISSN</Type>
    <Value>1550-2368</Value>
  </ItemIdentifier>
  <ItemPlatform>Scitation</ItemPlatform>
  <ItemPublisher>APS</ItemPublisher>
  <ItemName>Physical Review D</ItemName>
  <ItemDataType>Journal</ItemDataType>
  <ItemPerformance>
    <Period>
      <Begin>2009-09-01</Begin>
      <End>2009-09-30</End>
    </Period>
    <Category>Requests</Category>
    <Instance>
      <MetricType>ft_html</MetricType>
      <Count>0</Count>
    </Instance>
    <Instance>
      <MetricType>ft_pdf</MetricType>
      <Count>0</Count>
    </Instance>
    <Instance>
      <MetricType>ft_ps</MetricType>
      <Count>0</Count>
    </Instance>
    <Instance>
      <MetricType>ft_total</MetricType>
      <Count>0</Count>
    </Instance>
  </ItemPerformance>
</ReportItems>
</Customer>
</Report>
</Report>
</rr:ReportResponse>
</soap:Body>
</soap:Envelope>
```

## Appendix C

### *WSDL for Scholarly iQ SUSHI Web Service*

```
<definitions xmlns:tns="SushiService"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:sc="http://www.niso.org/schemas/sushi/counter"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  name="SushiService"
  targetNamespace="SushiService"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" />

  <types>
    <xsd:schema>
      <xsd:import schemaLocation="counter_sushi3_0.xsd"
        namespace="http://www.niso.org/schemas/sushi/counter" />
    </xsd:schema>
  </types>

  <message name="GetReportIn">
    <wsdl:documentation
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" />
    <part name="messagePart"
      element="sc:ReportRequest" />
  </message>

  <message name="GetReportOut">
    <wsdl:documentation
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" />
    <part name="messagePart"
      element="sc:ReportResponse" />
  </message>

  <portType name="SushiServiceInterface">
    <wsdl:documentation
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" />

    <operation name="GetReport">
      <wsdl:documentation
        xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" />
      <input message="tns:GetReportIn" />
      <output message="tns:GetReportOut" />
    </operation>
  </portType>
</definitions>
```

```
</portType>

<binding name="SushiService"
  type="tns:SushiServiceInterface">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetReport">
    <soap:operation soapAction="SushiService:GetReportIn"
      style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>

<binding name="SushiServiceSoap12"
  type="tns:SushiServiceInterface">
  <soap12:binding
transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetReport">
    <soap12:operation soapAction="SushiService:GetReportIn"
      style="document"/>
    <input>
      <soap12:body use="literal"/>
    </input>
    <output>
      <soap12:body use="literal"/>
    </output>
  </operation>
</binding>
<service name="SushiService">
  <port name="SushiServicePort"
    binding="tns:SushiService">
    <soap:address
location="http://sushi.scholarlyiq.com/sushiservice.asmx"/>
  </port>
</service>
</definitions>
```

## Additional Resources

NISO's SUSHI Workroom: <http://www.niso.org/workrooms/sushi>

IT & Digital Development, University of Pennsylvania Library:  
[http://docs.google.com/View?docid=d2dhjwd\\_140d923m7fh](http://docs.google.com/View?docid=d2dhjwd_140d923m7fh)

SoapUI Testing Tool: <http://www.soapui.org/>

Liquid XML Studio: <http://www.liquid-technologies.com/xmlstudio/xmlstudio.aspx>