

WATS ATML ServicePrepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012Revision
3Page no.
page 1 of 12www.virinco.no
virinco@virinco.no

WATS ATML Service

(Automated Test Markup Language Service)

Rev.No./Date	Author.	Checked.	Approved.
Rev#1: 28.09.12	oef		
Rev#2: 05.11.12	oef		
Rev#3: 13.11.12	oef		

WATS ATML Service

Prepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012

Revision
3

Page no.
page 2 of 12

www.virinco.no
virinco@virinco.no

Table of contents

1	Document purpose	3
2	Abbreviations and Terminology	3
3	Implementation	3
3.1	Report delivery	3
3.2	REST Service endpoint	3
3.3	SOAP Service endpoint.....	3
4	ATML Requirements and restrictions for WATS	4
4.1	TestResults element (root)	4
4.1.1	UUT element	5
4.1.2	TestProgram element	5
4.1.3	TestStation element	5
4.1.4	Personnel element	5
4.1.5	ResultSet element.....	5
4.1.6	TestGroup element	6
4.1.7	Test element.....	7
4.1.8	TestResult element	7
4.2	Test Result data	8
5	Example ATML	9
6	References	12

WATS ATML Service

Prepared by Øystein Fallo	Document number:	Date (dd.mm.yyyy) 13.11.2012	Revision 3	Page no. page 3 of 12	www.virinco.no virinco@virinco.no
------------------------------	------------------	---------------------------------	---------------	--------------------------	--

1 Document purpose

This document describes the functionality of the WATS ATML Service API, and how to use this service. The ATML format is an IEEE (1636.1) standard for Exchanging Test Results and Session Information via the eXtensible Markup Language (XML).

2 Abbreviations and Terminology

ATE	Automatic Test Equipment
ATML	Automatic Test Markup Language
ATS	Automatic Test System
CR/LF	Carriage Return/Line Feed
GUID	Globally Unique Identifier
ISO	International Organization for Standardization
REST	REpresentational State Transfer
SIMICA	Software Interface for Maintenance Information Collection and Analysis
UUID	Universally Unique Identifier
UUT	Unit Under Test
URL	Universal Resource Locator
URN	Uniform Resource Name
W3C	World Wide Web Consortium
WATS	Wide Area Test System
WCF	Windows Communication Foundation
WSDL	Web Service Definition Language
XML	eXtensible Markup Language

3 Implementation

3.1 Report delivery

The atml reception service consists of two different service endpoints. The reports can be delivered using HTTP POST to a REST service, or a SOAP service. The REST service endpoint simplifies the integration with other systems. The SOAP version is implemented for integration with SOAP capable systems.

3.2 REST Service endpoint

The ATML TestResult (IEEE 1336.1) report is sent to the REST Service using HTTP POST method. The ATML Converter will then parse the document and submit it to the database. A SubmitResult xml document is returned as a response to the HTTP POST. The REST Service address is

<http://servername-or-fqdn/wats-dc/atml/Report>

this is the url used for POSTing new reports, and (if enabled) retrieve reports by Report GUID.

The REST version service definition (human readable only) can be found by adding /help to the REST service endpoint.

<http://servername-or-fqdn/wats-dc/atml/help>

3.3 SOAP Service endpoint

A SOAP Service is method centric. To submit a new report using the SOAP Service use on of the following service methods:

Submit	Submit a new atml document (as xml/atml)
SubmitAsStream	Submit a new atml document (as stream/binary data)
SubmitAsString	Submit a new atml document (as text/xml data)

The SOAP Service address is:

<http://servername-or-fqdn/wats-dc/atml.svc>

The service definition (wsdl) for the SOAP version can be retrieved from the service by adding "?wsdl" to the soap endpoint address.

<http://servername-or-fqdn/wats-dc/atml.svc?wsdl>

4 ATML Requirements and restrictions for WATS

The following data is required to deliver a report:

- Header data
 - Serial number
 - Part number
 - Revision
 - Process Code (formerly known as Operation type)
 - Status (Passed/Failed)
 - Start Date/Time
 - Duration (in seconds)
 - Station ID (Name and preferably location and purpose)
 - Operator
 - Sequence/test file name and version
- Step/test data
 - Step/test name
 - Step type (Numeric/string/Boolean)
 - Value (Numeric/string/true/false)
 - Limits
 - Comp Operator (Greater than, Less Than...)
 - Unit (V, A, Ohm)
 - Status (Passed/Failed)

Using ATML 1636.1 as the import format gives a well-defined format based on two standard XML schemas; Common.xsd and TestResult.xsd. These schemas can be found here:

<http://standards.ieee.org/downloads/1636/1636.1-2007/Common.xsd>
<http://standards.ieee.org/downloads/1636/1636.1-2007/TestResults.xsd>

An online schema reference can be browsed at:

<http://grouper.ieee.org/groups/scc20/ATML/Documentation/TestResults.html>

For this version of the import software, only elements needed and supported by WATS will be used.

In the following sections the xml fragments assume default namespace is <http://www.ieee.org/ATML/2007/TestResults> and namespace prefix c: resolves to <http://www.ieee.org/ATML/2006/Common>

4.1 TestResults element (root)

The TestResults element must specify the uuid attribute. The uuid attribute is a GUID uniquely identifying a test report.

The following elements are required for successfully importing a report into WATS:

- UUT, contains Partnumber, Revision, Serialnumber and optionally subunit information.
- TestProgram, contains Sequence Name and Version
- TestStation, contains Test station name, and preferably location and purpose.
- Personnel, contains operator id/name
- ResultSet, contains process code and the test results

```
<TestResults xmlns="http://www.ieee.org/ATML/2007/TestResults"
```

WATS ATML Service

Prepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012

Revision
3

Page no.
page 5 of 12

www.virinco.no
virinco@virinco.no

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:c="http://www.ieee.org/ATML/2006/Common"
uid="7C67F609-BB24-4EEE-B523-79F73531CA09" >
<!-- Contents -->
</TestResults>

```

4.1.1 UUT element

The UUT element must specify a partnumber, a revision number and a serial number.

```

<UUT UutType="hardware">
  <c:Definition>
    <c:Identification designator="PartNumber">
      <c:Version>A.1</c:Version>
      <c:ModelName>A101503</c:ModelName>
    </c:Identification>
  </c:Definition>
  <c:SerialNumber>123901000084</c:SerialNumber>
</UUT>

```

4.1.1.1 Subunit information

Subunit information can be included in the extension node (xml freeform node) of the UUT element. The Subunit information consists of a SubUnits root element with any number of SubUnit childnodes. The SubUnits element is not specified in any schema, and must use empty namespace (xmlns="")

```

<c:Extension>
  <SubUnits xmlns="">
    <SubUnit Type="Controller Card" PN="A110021" SN="123625000016" Rev="A.2" />
    <SubUnit Type="Power Card" PN="A110022" SN="1232280000541" Rev="B.3" />
  </SubUnits>
</c:Extension>

```

All attributes (Type,PN,SN and Rev) are required in the SubUnit element.

4.1.2 TestProgram element

Contains sequence file name and version

```

<TestProgram>
  <c:Definition version="1.1.0" name="A101503_A.1_1.1.0.seq" >
    <c:Identification />
  </c:Definition>
</TestProgram>

```

The version attribute is required, and must contain at least 3 parts (in dot-decimal notation)

The c:Identification element is required, but can be empty.

4.1.3 TestStation element

Contains identity information about the test station

```

<TestStation>
  <c:Definition name="SITE21-ATE005">
    <c:Description>Location="Venture, Shanghai, China",Purpose=Automated Burn-In</c:Description>
    <c:Identification />
  </c:Definition>
</TestStation>

```

The c:Description element is optional, but it is but highly recommended to specify both Location and Purpose for the station. Location and purpose is specified as key/value pairs separated by comma. If the value contains commas, it must be quoted as shown in the example fragment above.

The c:Identification element is required, but can be empty.

4.1.4 Personnel element

The personnel element specifies the operator identity. Only the ID attribute is logged I WATS.

```

<Personnel>
  <SystemOperator ID="ATE0per315" name="ATE Operator 315" />
</Personnel>

```

4.1.5 ResultSet element

The ResultSet element is the same xsd type as TestGroup, but it is used slightly differently.

```

<ResultSet ID="0" startDateTime="2012-09-26T20:34:38.91" [endDateTime="2012-09-26T20:34:39.397"]>
  <Parameters>
    <Parameter ID="ProcessCode">
      <Data>
        <c:Datum xsi:type="c:string">

```

WATS ATML Service

Prepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012Revision
3Page no.
page 6 of 12
www.virinco.no
virinco@virinco.no

```

    <c:Value>10</c:Value>
  </c:Datum>
</Data>
</Parameter>
</Parameters>
<Outcome value="Failed" />
<TestGroup {attribute definition: see 4.1.6} >
  <!-- Root step contents-->
</TestGroup>
</ResultSet>

```

ResultSet must be specified, and can only appear once in the TestResults.

The ID attribute is required, but it is not used in WATS. It must be specified to fulfill schema validation rules.

The startDateTime attribute is required.

The endDateTime attribute is optional. If specified, it will be used to calculate the test duration.

A parameter with ID=ProcessCode is required. The value (string) of this parameter must match either the Code or the GUID of an existing Process marked as TestOperation in WATS. During export Process GUID will be exposed as a parameter with ID=ProcessGUID.

The Outcome element is required, and contains the report test result value (acceptable values are Passed, Failed or Aborted). The WATS Report result "Terminated" is not supported in ATML.

The schema allows multiple Actions (common base type for TestGroup, Test and SessionAction) in the ResultSet, but the import to WATS will only allow one root step. If there is more than one TestGroup in the import file, only the first will be imported. Any other Action elements will be disregarded in the ResultSet element.

4.1.6 TestGroup element

The TestGroup element is a container element for other TestGroups, Tests or SessionAction. When imported to WATS, it is known as a sequence.

```

<TestGroup ID="2" name="Call subsequence 1" startDateTime="2012-09-26T20:34:39.023" [endDateTime="2012-09-26T20:34:39.397"]
[operatingMode="Main"] [userDefinedType="SequenceCall"]>
  <Parameters>
    <Parameter ID="Sequence">
      <Data>
        <c:Collection>
          <c:Item name="File">
            <c:Datum xsi:type="c:string">
              <c:Value>C:\Users\administrator\Documents\Test Sequence1.seq</c:Value>
            </c:Datum>
          </c:Item>
          <c:Item name="Name">
            <c:Datum xsi:type="c:string">
              <c:Value>Subsequence 1</c:Value>
            </c:Datum>
          </c:Item>
          <c:Item name="Version">
            <c:Datum xsi:type="c:string">
              <c:Value>1.0.0.0</c:Value>
            </c:Datum>
          </c:Item>
        </c:Collection>
      </Data>
    </Parameter>
  </Parameters>
  <Outcome value="Passed" />
  [ <Test ... </Test> {0..n} ]
  [ <TestGroup ... </TestGroup> {0..n} ]
</TestGroup>

```

The ID attribute is required, and must be unique throughout the testgroup/test hierarchy.

The name attribute is required and contains the sequence step's name (not necessarily the same as sequence name).

The startDateTime attribute is required and contains the start date/time for the sequence

The endDateTime attribute is optional. If specified, it will be used to calculate the sequence step's total time.

WATS ATML Service

Prepared by Øystein Fallo	Document number:	Date (dd.mm.yyyy) 13.11.2012	Revision 3	Page no. page 7 of 12	www.virinco.no virinco@virinco.no
------------------------------	------------------	---------------------------------	---------------	--------------------------	--

The operatingMode attribute is optional can and can contain one of the following values: Setup, Main or Cleanup. This attribute is mapped to WATS step group. If unspecified or unparsable it will be imported as "Main".

The userDefinedType attribute is optional. This attribute is mapped to WATS step type. If unspecified it will be imported as "SequenceCall".

The Outcome element is required, and contains the sequence result value (acceptable values are Passed, Failed or Aborted). The WATS Step results "Done", "Error", "Skipped" and "Terminated" are not supported in ATML.

A Parameter with ID=Sequence can be used to specify sequence information. It is recommended to specify sequence information whenever possible, and at least on the root step. If Sequence information is not provided for the root step, the less detailed information from the TestProgram element will be used.

4.1.7 Test element

The Test element contains information about a specific test step

```
<Test ID="4" name="Numeric Limit Test" startDateTime="2012-09-26T20:34:39.127" [endDateTime="2012-09-26T20:34:39.397"]
[operatingMode="Main"] [userDefinedType="SequenceCall"]>
  <Outcome value="Passed" />
  [ <TestResult> ... </TestResult> {0..n} ]
</Test>
```

The ID attribute is required, and must be unique throughout the testgroup/test hierarchy.

The name attribute is required and contains the test step's name.

The startDateTime attribute is required and contains the start date/time for the sequence

The endDateTime attribute is optional. If specified, it will be used to calculate the sequence step's total time.

The operatingMode attribute is optional can and can contain one of the following values: Setup, Main or Cleanup. This attribute is mapped to WATS step group. If unspecified or unparsable it will be imported as "Main".

The userDefinedType attribute is optional. This attribute is mapped to WATS step type. If unspecified the contents of TestResult will be used to select one of the following types: Action, NumericLimitTest, PassFailTest, StringValueTest, MultipleNumericLimitTest, MultiplePassFailTest, MultipleStringValueTest.

The Outcome element is required, and contains the test (step) result value (acceptable values are Passed, Failed or Aborted). The WATS Step results "Done", "Error", "Skipped" and "Terminated" are not supported in ATML.

The Test element can contain any number of TestResults. If no TestResult elements are found, the Test is considered an "Action" Step. If multiple TestResult elements exists, the step is considered a multiple-measurement step.

4.1.8 TestResult element

TestResult contains the actual measurement(s) of a step, and its limits.

```
<TestResult ID="1" name="Measurement 1">
  <Outcome value="Passed" />
  <TestData>
    <c:Datum xsi:type="c:double" standardUnit="mA" value="142.12941" />
  </TestData>
  <TestLimits>
    <Limits>
      <c:LimitPair operator="AND">
        <c:Limit comparator="GE">
          <c:Datum xsi:type="c:double" value="120" />
        </c:Limit>
        <c:Limit comparator="LE">
          <c:Datum xsi:type="c:double" value="150" />
        </c:Limit>
      </c:LimitPair>
    </Limits>
  </TestLimits>
</TestResult>
```

The ID attribute is required, and must be unique within the Test (step).

WATS ATML Service

Prepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012

Revision
3

Page no.
page 8 of 12

www.virinco.no
virinco@virinco.no

The name attribute is required if the step is a multiple measurement step. It specifies the measurement name.

The Outcome element is required, and contains the measurement's result value (acceptable values are Passed, Failed or Aborted). The WATS Step results "Done", "Error", "Skipped" and "Terminated" are not supported in ATML.

The TestData element is required for numericlimit and stringvalue step types. If the TestData element is missing, the measurement will be imported as a Boolean measurement.

4.2 Test Result data

The following data is required to deliver a report:

- Header data
 - Serial number
 - Part number
 - Revision
 - Process Code (formerly known as Operation type)
 - Status (Passed/Failed)
 - Start Date/Time
 - Duration (in seconds)
 - Station ID (Name and preferably location and purpose)
 - Operator
 - Sequence/test file name and version
- Step/test data
 - Step/test name
 - Step type (Numeric/string/Boolean)
 - Value (Numeric/string/true/false)
 - Limits
 - Comp Operator (Greater than, Less Than...)
 - Unit (V, A, Ohm)
 - Status (Passed/Failed)

Using ATML 1636.1 as the import format gives a well defined format based on two standard XML schemas; Common.xsd and TestResult.xsd. These schemas can be found here:

<http://standards.ieee.org/downloads/1636/1636.1-2007/Common.xsd>

<http://standards.ieee.org/downloads/1636/1636.1-2007/TestResults.xsd>

An online schema reference can be browsed at:

<http://grouper.ieee.org/groups/scc20/ATML/Documentation/TestResults.html>

For this version of the import software, only elements needed and supported by WATS will be used.

WATS ATML Service

Prepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012Revision
3Page no.
page 9 of 12
www.virinco.no
virinco@virinco.no

5 Example ATML

Example of a complete ATML containing a UUT test result:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--ATML Example for delivery of test results to WATS. uuid should be unique for each test-->
<TestResults xmlns="http://www.ieee.org/ATML/2007/TestResults" xmlns:c="http://www.ieee.org/ATML/2006/Common"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" uuid="bf2bee3e-ddb9-4a85-81c2-f4674117213e">
  <TestProgram>
    <!--Definition of test program (sequence file)-->
    <c:Definition name="Test program name" version="3.2.3.3" >
      <!-- identification, not used by WATS - but required by schema -->
      <c:Identification />
    </c:Definition>
  </TestProgram>
  <Personnel>
    <!--Test operator-->
    <SystemOperator ID = "administrator"/>
  </Personnel>
  <TestStation>
    <c:Definition name="TestStationName" >
      <!--Test station name#Purpose#Location-->
      <c:Description>Location="China, Dongguan",Purpose="Final Function Tester"</c:Description>
      <c:Identification/>
    </c:Definition>
    <!-- Optional if manufacturer has a unique GUID for test station -->
    <c:SerialNumber>
      {389A81AB-7F7E-468e-AE25-A472D00D315B}
    </c:SerialNumber>
  </TestStation>
  <UUT UutType="hardware">
    <!--Define partnumber and revision-->
    <c:Definition>
      <c:Identification designator="Partnumber">
        <c:ModelName>PartnumberX</c:ModelName>
        <c:Version>2.3</c:Version>
        <c:IdentificationNumbers>
          <!--Sub parts-->
          <c:IdentificationNumber qualifier="Controller" type="Part" number="9827349234"/>
          <c:IdentificationNumber qualifier="Secondary board" type="Part" number="162543A"/>
        </c:IdentificationNumbers>
      </c:Identification>
    </c:Definition>
    <!--Serial number of tested UUT-->
    <c:SerialNumber>
      123456789
    </c:SerialNumber>
  </UUT>
  <ResultSet ID="8" startDateTime="2009-02-10T15:46:21.360" endDateTime = "2009-02-10T15:46:22.107">
    <Parameters>
      <!--Operation type: POBA Test, Final Function Test, Calibration etc...-->
      <Parameter ID="ProcessCode">
        <Data>
          <c:Datum xsi:type="c:string">
            <!-- A vldid Process Code or Process GUID must be specified -->
            <c:Value>10</c:Value>
          </c:Datum>
        </Data>
      </Parameter>
    </Parameters>
    <!--Overall test result-->
    <Outcome value="Failed" />
    <!--Root step of test, must be present-->
    <TestGroup name="MainSequence" ID="8" startDateTime = "2009-02-10T15:46:21.360" endDateTime = "2009-02-10T15:46:22.107" >
      <Parameters>
        <Parameter ID="Sequence">
          <Data>
            <c:Collection>
              <c:Item name="File">
                <c:Datum xsi:type="c:string">
                  <c:Value>C:\Users\administrator\Documents\Test Sequence1.seq</c:Value>
                </c:Datum>
              </c:Item>
              <c:Item name="Name">
                <c:Datum xsi:type="c:string">
                  <c:Value>Subsequence 1</c:Value>
                </c:Datum>
              </c:Item>
              <c:Item name="Version">
                <c:Datum xsi:type="c:string">
                  <c:Value>1.0.0.0</c:Value>
                </c:Datum>
              </c:Item>
            </c:Collection>
          </Data>
        </Parameter>
      </Parameters>
      <Outcome value="Failed" />
      <!--Place test results below. Recursive, supported TestResult ID is:
      Numeric, -->
      <Test ID = "9" name = "Main test 1" startDateTime = "2009-02-10T15:46:21.731" endDateTime = "2009-02-10T15:46:21.755" >
        <Outcome value = "Failed" />
        <TestResult ID = "Numeric">
          <TestData>
            <c:Datum value = "4.995016385476" xsi:type="c:double" standardUnit="volts"/>
          </TestData>
          <TestLimits>
            <Limits>
              <c:LimitPair operator = "AND">
                <c:Limit comparator = "GE">
                  <c:Datum value = "9" xsi:type="c:double"/>
                </c:Limit>
                <c:Limit comparator = "LE">
                  <c:Datum value = "11" xsi:type="c:double"/>
                </c:Limit>
              </c:LimitPair>
            </Limits>
          </TestLimits>
        </TestResult>
      </Test>
    </TestGroup>
  </ResultSet>
</TestResults>
```

WATS ATML Service

Prepared by
Øystein Fallo

Document number:

Date (dd.mm.yyyy)
13.11.2012

Revision
3

Page no.
page 10 of 12

www.virinco.no
virinco@virinco.no

```

        </c:Limit>
      </c:LimitPair>
    </Limits>
  </TestLimits>
</TestResult>
</Test>
<TestGroup name="SubTestSeq 1" ID="10" startDateTime = "2009-02-10T15:46:21.767" endDateTime = "2009-02-10T15:46:21.891" >
  <Outcome value="Failed" />
  <Test ID = "11" name = "Num test 1.1" startDateTime = "2009-02-10T15:46:21.825" endDateTime = "2009-02-10T15:46:21.826" >
    <Outcome value = "Failed" />
    <TestResult ID = "Numeric">
      <TestData>
        <c:Datum value = "8.989020304584" xsi:type="c:double"/>
      </TestData>
      <TestLimits>
        <Limits>
          <c:LimitPair operator = "AND">
            <c:Limit comparator = "GE">
              <c:Datum value = "9" xsi:type="c:double"/>
            </c:Limit>
            <c:Limit comparator = "LE">
              <c:Datum value = "11" xsi:type="c:double"/>
            </c:Limit>
          </c:LimitPair>
        </Limits>
      </TestLimits>
    </TestResult>
  </Test>
  <Test ID = "12" name = "String test 1.2" startDateTime = "2009-02-10T15:46:21.833" endDateTime = "2009-02-10T15:46:21.833" >
    <Outcome value = "Passed" />
    <TestResult ID = "String">
      <TestData>
        <c:Datum xsi:type="c:string" >
          <c:Value>
            stringlimit
          </c:Value>
        </c:Datum>
      </TestData>
      <TestLimits>
        <Limits>
          <c:Expected comparator = "EQ">
            <c:Datum xsi:type="c:string">
              <c:Value>
                stringlimit
              </c:Value>
            </c:Datum>
          </c:Expected>
        </Limits>
      </TestLimits>
    </TestResult>
  </Test>
  <Test ID = "13" name = "Boolea test 1.3" startDateTime = "2009-02-10T15:46:21.838" endDateTime = "2009-02-10T15:46:21.838" >
    <Outcome value = "Failed" />
  </Test>
  <Test ID = "14" name = "MultiNumttest 1.4" startDateTime = "2009-02-10T15:46:21.843" endDateTime = "2009-02-10T15:46:21.885"
  >
    <Outcome value = "Failed" />
    <TestResult ID = "Measurement 0">
      <TestData>
        <c:Datum value = "0" xsi:type="c:double"/>
      </TestData>
      <TestLimits>
        <Limits>
          <c:LimitPair operator = "AND">
            <c:Limit comparator = "GE">
              <c:Datum value = "9" xsi:type="c:double"/>
            </c:Limit>
            <c:Limit comparator = "LE">
              <c:Datum value = "11" xsi:type="c:double"/>
            </c:Limit>
          </c:LimitPair>
        </Limits>
      </TestLimits>
    </TestResult>
    <TestResult ID = "Measurement 1">
      <TestData>
        <c:Datum value = "0" xsi:type="c:double" standardUnit="volts"/>
      </TestData>
      <TestLimits>
        <Limits>
          <c:LimitPair operator = "AND">
            <c:Limit comparator = "GE">
              <c:Datum value = "9" xsi:type="c:double"/>
            </c:Limit>
            <c:Limit comparator = "LE">
              <c:Datum value = "11" xsi:type="c:double"/>
            </c:Limit>
          </c:LimitPair>
        </Limits>
      </TestLimits>
    </TestResult>
    <TestResult ID = "Measurement 2">
      <TestData>
        <c:Datum value = "0" xsi:type="c:double" nonStandardUnit="mV"/>
      </TestData>
      <TestLimits>
        <Limits>
          <c:LimitPair operator = "AND">
            <c:Limit comparator = "GE">
              <c:Datum value = "9" xsi:type="c:double"/>
            </c:Limit>
            <c:Limit comparator = "LE">
              <c:Datum value = "11" xsi:type="c:double"/>
            </c:Limit>
          </c:LimitPair>
        </Limits>
      </TestLimits>
    </TestResult>
  </Test>

```

WATS ATML Service

Prepared by Øystein Fallo	Document number:	Date (dd.mm.yyyy) 13.11.2012	Revision 3	Page no. page 11 of 12	www.virinco.no virinco@virinco.no
------------------------------	------------------	---------------------------------	---------------	---------------------------	--

```
</c:Limit>
</c:LimitPair>
</Limits>
</TestLimits>
<Outcome value = "Failed" />
</TestResult>
</Test>
</TestGroup>
</TestGroup>
</ResultSet>
</TestResults>
```

WATS ATML Service

Prepared by Øystein Fallo	Document number:	Date (dd.mm.yyyy) 13.11.2012	Revision 3	Page no. page 12 of 12	www.virinco.no virinco@virinco.no
------------------------------	------------------	---------------------------------	---------------	---------------------------	--

6 References

Name	Url
ATML Online reference	http://grouper.ieee.org/groups/scc20/ATML/Documentation/TestResults.html