UOML Sample-en

Technical Committee:

OASIS Unstructured Operation Markup Language (UOML-X) TC

This document includes three parts:

1. **Case studies**

Case 1: Use UOML compliant software to exchange documents.

Case 2: Full text search - extract text data from docbase in order to generate a text file called “output.txt”.

Case 3: Image OCR operation **-** Transform image formatted data to a text formatted data.

Case 4: Converts different type of documents to UOML documents.

Case 5: Send documents to fax or cell phone via UOML.

Case 6: Write and make comments to UOML documents.

2. **Detailed usage for all the UOML Instructions.**
3. **Example of UOML API in C.**

1. Case Studies

Case 1: Use UOML compliant software to exchange documents

Steps：

Follow is a scenario on how two people using different word processors to exchange document:

1. Someone generates a document using UOML compliant word processor “A”, and save it as an UOML document. He sends it to another person to read.

2. Receive and read the document:

   The second person receives and opens the document using any available UOML compliant word processors, it is not necessary to use the same software as the first person. He chooses to use word processor “B”.

3. Read ,add and make comments to the document:

UOML standard document resembles a paper document with many digital and intelligent features. So the second person can read, add and make comments to the document on screen just like what we do to a piece of paper using word processor “B”.

4. Save the change and send the document back to the first person.

5. Receive, read, add and make comments again:

The first person receives the reviewed document, and then he can read it with word processor “A”, if needed he can add and make comments to it again.

Case 2: Full text search – extract text data from docbase and generate a text file “output.txt”

Steps：

1. Open docbase, get objectSet handle.
2. Create an empty document called output.txt as the output file.
3. Set the objectSet as the current object.
4.

a) Initialize an incremental variable index to 0.

b) Operate an UOML_GET operation to current objectSet as follow to get the handle of the current index indicated object:

```
<uoml:UOML_GET handle=”current_obj_set_xxx” usage=”GET_SUB”>
    <pos val=”index”/>
</uoml:UOML_GET>
```

If UOML_GET failure, go to step 5, the operation end.

If UOML_GET success, do the follow operation to get this object’s XML expression:

```
<uoml:UOML_GET handle=”current_obj_handle_xxx” usage=”GET__CONTENT”/>
```

If UOML_GET to get XML expression returning value is not UOMLO_TEXT

Go c（directly.

If UOML_GET to get XML expression returning value is UOMLO_TEXT

Get the text data, export to output.txt.

c) ++index, go back to b)

5. End, save the output.txt, close docbase,

Case 3: Image OCR operation – Transform image formatted data to a text formatted data

Steps:

1、 Open docbase, get the objectSet handle.

2、 Set the objectSet as the current object.

3、 Do the follow operation to the current object：

a) Initialize an incremental variable *index* to 0.

b) Operate an UOML_GET operation to current objectSet as follow to get the handle of the current *index* indicated object:

```
<uoml:UOML_GET handle=”current_obj_set_xxx” usage=”GET_SUB”>
    <pos val=”index”/>
</uoml:UOML_GET>
```

If UOML_GET failure, go to step 4, end the operation

If UOML_GET success, do follow operation to get this object’s XML expression:

```
<uoml:UOML_GET handle=”current_obj_handle_xxx” usage=”GET__CONTENT”/>
```

If UOML_GET to get XML expression returning value is not UOMLO_IMAGE

Go to d)

If UOML_GET to get XML expression returning value is UOMLO_IMAGE

Get document’s image data; call a third-party OCR engine to convert image data to text data.

Operate the UOML_DELETE to delete the image data.

```
<uoml:UOML_DELETE handle=”img_obj_handle_xxx”/>
```

c) Put the image converted text data to an UOMLO_TEXT format’s XML structure, using UOML_INSERT to put it to the image data’s original position

```
<uoml:UOML_INSERT handle=”current_obj_set_xxx” pos=”index”/>
```

```
<xobj>
    <text lt_pt="100, 200" br_pt="180, 150" txtEncoding="ASCII" txtData="ocrtxt…"/>
</xobj>
</uoml:UOML_INSERT>
```

d) ++*index*，go back to b（

4、 End, save and close docbase.

Case 4: Convert different type of documents to UOML documents

Steps:

1. Please install UOML Writer first if the computer you are using now doesn't have it. UOML Writer is a virtual printer, can be downloading freely from http://www.uoml.org/.

2. Run any third party word processor A. Assume word processor A supports type A document.

3. Using word processor A to open type A document - file.A.

4. Go to word processor A's Print function, from the printer list, choose UOML Writer.

5. A window appears. Write down the converted UOML document name, and the path.

6. Save it. Now file.A has been converted to an UOML document. End.

Case 5: Send documents to fax or cell phone via UOML.

Steps:

1. Get a printable document, it can be any file type; please first convert it to an UOML document via UOML Writer (check case 4 for details).

2. Transform the UOML document to a .bmp file.

a) Open docbase:

```
<uoml:UOML_OPEN path=”/home/admin/storage/1.sep” create=”true” del_exist=”false”/>
```

b) Loop through the docbase, do a serial of UOML_GET instructions to get the handle of the document we need:

```
<uoml:UOML_GET handle=”some_obj_handle_xxx” usage =”GET_SUB”>
        <pos val=”0”/>
</uoml:UOML_GET>
```

c) Do UOML_GET to get the handle of each page from the document.

```
<uoml:UOML_GET handle=”some_obj_xxx” usage =”GET_SUB”>
        <pos val=”0”/>
</uoml:UOML_GET>
```

d) Do UOML_GET to generate a new .bmp file for this page, and get its handle..

```
<uoml:UOML_GET handle=”page_obj_andle_xxx” usage =”GET_PAGE_BMP”>
   <disp_conf format=”BMP” output=”file” end_layer=”1” resolution=”600”
   addr=”/home/admin/output/page.bmp”>
        <clip>
            <subpath data=”s 0,0 1 3000,0 1 3000, 5000 1 0, 5000 1 0,0”/>
        </clip>
   </disp_cong>
</uoml:UOML_GET>
```

3. Send the .bmp file to a third party fax server or a cell phone.

4. User reads the document from fax paper or from a cell phone screen.

5. End.

Case 6: Write and make comments to UOML documents

Steps:

1. Open docbase.

“<uoml:UOML_OPEN path=”/home/admin/storage/1.sep” create=”true” del_exist=”false”/>”);

2. Loop through the docbase, do a serial of UOML_GET operations to get the handles of all the

documents we need:

```
<uoml:UOML_GET handle=”some_obj_handle_xxx” usage=”GET_SUB\”>
  <pos val=\”0\”/>
</uoml:UOML_GET>
```

3. Get the handle for the page which user had made comments and added things on.

4. Create a new layer to this page, and insert a new object set to this layer, get the handle of this layer:

```
<uoml:UOML_INSERT handle=”page_obj_handle_xxxxxx” pos=”0”>
<xobj>
    <layer>
    <vs/>
    </layer>
</xobj>
</uoml:UOML_INSERT>
```

5. Do the UOML_GET to get the handle of the newly created object set.

6. Get the added information from the script board, convert it to UOMLO_PATH object.

7. Insert UOMLO_PATH to the new object set. Here for demonstrate purpose, the path only includes one rectangle. In real case, it might include very complicated path to represent the user’s handwriting:

```
<uoml:UOML_INSERT handle=”obj_set_handle_xxxxxx” pos=”0”>
    <xobj>
        <path>
            <subpath data=”s 0,0 1 3000, 0 1 3000, 5000 1 0, 5000 1 0, 0”/>
        </path>
    </xobj>
</uoml:UOML_INSERT>
```

8. Save the docbase.
9. Close the docbase.

2. Usage of UOML Instructions

Follow is the total nine current UOML Instructions:

- □□□□□□□□□□ UOML_OPEN
- □□□□□□□□□□ UOML_CLOSE
- □□□□□□□□□□ UOML_USE
- □□□□□□□□□□ UOML_GET
- □□□□□□□□□□ UOML_SET
- □□□□□□□□□□ UOML_INSERT
- □□□□□□□□□□ UOML_DELETE
- □□□□□□□□□□ UOML_SYSTEM
- □□□□□□□□□□ UOML_RET

For details about each UOML Instruction's properties, please check UOML Specification (current UOML Working Draft 01) Section 3.

UOML_OPEN

Example:

//create or open a document base.

<uoml:UOML_OPEN path="/home/admin/storage/1.sep" create="true" del_exist="false"/>

UOML_CLOSE

Example:

//Close a document base

<uoml:UOML_CLOSE handle="db_handle_xxxxx"/>

UOML_USE

Example:

//set an object as the current object.

<uoml:UOML_use handle="db_handle_xxxxx"/>

UOML_GET

Example 1:

//Get the bitmap expression of one page.

<uoml:UOML_GET handle="page_handle_xxxxx" usage="GET_PAGE_BMP">

<disp_conf format="BMP" output="file" end_layer="1" resolution="600" addr="/home/admin/output/page.bmp">

<clip>

```
            <subpath data=”s 0, 0 1 3000, 0 1 3000, 5000 1 0, 5000 1 0, 0”/>
        </clip>
    </disp_cong>
</uoml:UOML_GET>
```

Example 2:

```
//Get the property value of an object.
<uoml:UOML_GET handle=”some_obj_handle_xxx” usage=”GET_PROP”>
    <property name=”some_property_name_xxx”/>
</uoml:UOML_GET>
```

Example 3:

```
//Get the sub_object handle based on index value.
<uoml:UOML_GET handle=”some_obj_handle_xxx” usage=”GET_SUB”>
    <pos val=”10”/>
</uoml:UOML_GET>
```

Example 4:

```
//Get the XML expression of the object.
<uoml:UOML_GET handle=”some_obj_handle_xxx” usage=”GET_CONTENT” />
```

UOML_SET

Example 1:

```
//Set up object’s properties.
<uoml:UOML_SET handle=” obj_handle_xxxxxx”>
    <property>
        <intVal name = “SomePropName” val=”0”/>
    </property>
</uoml:UOML_SET>
```

Example:

```
//Base on provided XML value to reset the object and all its sub_objects.
<uoml:UOML_SET handle=”some_obj_handle_xxx”>
    <xobj>
```

```
        <uoml:UOMLO_MATRIX fll=”1.0” f12=”00” f21=”0.0” f22=”1.0” f31=””0.0” f32=”1.0”/>
    </xobj>
</uoml:UOML_SET>
```

UOML_INSERT

Example:

```
//Insert an object under another one as its sub-object.
<uoml:UOML_INSERT handle=”parent_obj_handle_xxxxxxx” pos=”3”>
    <xobj>
        <rect lt_pt=”100, 200” br_pt=”180, 150”/>
    </xobj>
</uoml:UOML_INSERT>
```

UOML_DELETE

Example:

```
//Delete an object.
<uoml:UOML_DELETE handle=”handle_xxxxxx”/>
```

UOML_SYSTEM

Example:

```
//Save the document base.
<uoml:UOML_SYSTEM>
    <db_flush handle=”dobbase_handle_xxxxxx” path=”/home/admin/storage/23.sep”/>
</uoml:UOML_SYSTEM>
```

UOML_RET

Example:

```
//Hold returning values of an object.
<uoml:UOML_RET>
    <intVal name=”someRetName” val=”234009”/>
</uoml:UOML_RET>
```

3.Examples of UOML API in C

Following are some examples to implement UOML API through C.

Implementation:

UOML_call: the interface to send a request to UOML.

UOML_parse: the interface to parse the return value from UOML**.**

Send an UOML_OPEN call

String strRet = uoml_call(“<uoml:UOML_OPEN path=\”/home/admin/storage/1.sep\” create=”true” del_exist=”false”/>”)

Send an UOML_CLOSE call

String strRet = uoml_call(“<uoml:UOML_CLOSE handle=”\”db_handle_xxxxx\”/>)”);

Send an UOML_SYSTEM Call – save the docbase

String str_ret = uoml_call(“<uoml:UOML_SYSTEM>

<db_flush handle=\”docbase_handle_xxxxx\” path\”/home/admin/storage/23.sep\”/>

</uoml:UOML_SYSTEM>”);

Send an UOML_USE call – set the current object

strRet = uoml_call(“<uoml:UOML_USE handle=\”obj_handle_xxxxxx\”/>”);

Send an UOML_GET call

Example 1:

//Get the bitmap expression of one page.

String strRet = uoml_call(“<uoml:UOML_GET handle=”\”page_handle_xxxxx\” usage = \”GET_PAGE_BMP\”/>

<disp_conf format=\”\BMP\” output=”\file\” end_layer=\”1\” resolution=\”600\” addr=\”/home/admin/output/page.bmp\”>

<clip>

<subpath data=\”s 0, 0 1 3000, 0 1 3000, 5000 1 0, 5000 1 0,0\”/>

</clip>

</disp_conf>

</uoml:UOML_GET>”);

Example 2:

//Get one of the property value of an object.

String strRet = uoml_call(“<uoml:UOML_GET handle=”\”some_obj_handle_xxxxx\” usage = \”GET_PROP\”/>

<property name=\”some_property_name_xxx\”/>

</uoml:UOML_GET>”);

Example 3:

//Get a sub_object handle based on position value.

String strRet = uoml_call(“<uoml:UOML_GET handle=”\”page_handle_xxxxx\” usage = \”GET_SUB\”/>)”);

<pos val=\”10\”/>

</uoml:UOML_GET>”);

Example 4:

//Get the XML expression of an object.

String strRet = uoml_call(“<uoml:UOML_GET handle=”\”page_handle_xxxxx\” usage = \”GET_CONTENT\”/>)”);

Send an UOML_INSERT call

String strRet = uoml_call(“<uoml:UOML_INSERT handle=”\”parent_handle_xxxxx\” pos = \”3\”/>

<xobj>

<ret lt_pt=\”100,200\” br_pt=\”180, 150\”/>

</xobj>

</uoml:UOML_GET> “);

Send an UOML_DELETE call

String strRet = uoml_call(“<uoml:UOML_DELETE handle=”\”handle_xxxxx\”/>”);

Send an UOML_SET call

Example 1:

//Set up a property value for an object.

String str_ret = uoml_call(“<uoml:UOML_SET handle=\”obj_handle_xxxxxx\”>

<property>

<intVal name=\”SomePropName\” val=\”0\”/>

</property>

</uoml:UOML_SET>”);

Example 2:

```
//Reset the content of an object based on the given XML expression.
String str_ret = uoml_call("<uoml:UOML_SET handle=\"obj_handle_xxxxxx\">
        <xobj>
            <matrix f11=\"1.0\" f12=\"0.0\" f21=\"0.0\" f22\"1.0\"f31=\"0.0\" f32\"1.0"/>
        </xobj>
</uoml:UOML_SET>");
```

Use UOML_PARSE

Example 1:

```
//Get the returning value after using UOML instructions.
String strHandle= uoml_parse(strRet,"handle");
```

Example 2:

```
//Check the UOML instruction operation is success or not.
Bool success = uoml_parse(strRet,"SUCCESS");
```

Example 3:

```
//Get the error code after using UOML instruction.
String err_info = uoml_parse(strRet,"ERR_INFO");
```