

# Practical examples using Adlib API

**Bert Degenhart Drenth  
Rui Mendes**





# Practical examples using Adlib API

- Commands categories
  - Search
  - Write
  - Select
  - Lock
  - Utilities
- Choosing the correct implementation
  - jQuery plugin
  - Adlib.Data dll
  - Url request



# Search

- Search

- `Wwwopac.ashx?database=<database name>&search=<search statement>&<optional parameters>`

*<database name>*: Name of the database to be queried, defined in adlibweb.xml

*<search statement>*: Consists of field names, operators and values, combined with boolean operators and sort options

*<optional parameters>*: like startfrom & limit, xmltype

- <http://api.adlibsoft.com/site/api/functions/search>



# Search using Url request

Search all records: return all records from collect.inf  
<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&search=all>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
  - <record priref="1" selected="False">
    <administration_name>PDP</administration_name>
    <creator>Pittoni, Giovanni Battista, the younger</creator>
    <creator>Valeriani, Domenico</creator>
    <creator>Valeriani, Giuseppe</creator>
    <creator.date_of_birth />
    <creator.date_of_birth />
    <creator.date_of_birth />
    <creator.date_of_death />
    <creator date of death />
```



# Search using Adlib.Data

```
string url =  
    "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";  
  
// Create a connection to the wwwopac.ashx  
AdlibConnection conn = new AdlibConnection(url);  
  
// Prepare a record set for results  
AdlibRecordSet recordSet =  
    new AdlibRecordSet(conn, "collect.inf");  
  
// Set the type of XML returned  
recordSet.XmlType = XmlType.Unstructured;  
  
// Do the search  
recordSet.Search("all");
```



# Using jQuery plugin

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "collect.inf",
  search: "all",
  xmltype: "unstructured"
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```



# Limiting results

- Parameters startfrom and limit can be used to limit the number of records returned

- Syntax:

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&search=all&startfrom=1&limit=4>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
+ <record priref="1" selected="False">
+ <record priref="2" selected="False">
+ <record priref="3" selected="False">
+ <record priref="5" selected="False">
</recordList>
- <diagnostic>
  <hits>724</hits>
  <xmltype>Undefined</xmltype>
  <first_item>1</first_item>
  <search>All</search>
  <sort />
  <limit>4</limit>
  <hits_on_display>4</hits_on_display>
```



# Limit using Adlib.Data

```
// Prepare a record set for results
AdlibRecordSet recordSet =
    new AdlibRecordSet(conn, "collect.inf");

// Set the type of XML returned
recordSet.XmlType = XmlType.Unstructured;

// Set the startfrom and limit
recordSet.StartFrom = 1;
recordSet.Limit = 4;

// Do the search
recordSet.Search("all");
```





# Limit using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";
```

```
$.adlibdata(url,  
{  
  //arguments  
  database: "collect.inf",  
  search: "all",  
  xmltype: "unstructured",  
  startfrom: 1,  
  limit: 4  
},  
function (adlibJSON) {  
  alert(adlibJSON.diagnostic.hits);  
});
```



# Sorting results

- Optional search keyword *sort* can be used to sort the search result

- Syntax:

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?datab>

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
- <adlibXML>
```

```
- <recordList>
```

```
- <record priref="23" selected="False">
```

```
  <administration_name>PDP</administration_name>
```

```
  <creator>Batoni, Pompeo Girolamo</creator>
```

```
  <creator.date_of_birth />
```

```
  <creator.date_of_death />
```

```
  <creator.history />
```

```
  <creator.qualifier />
```



# Sorting using Adlib.Data

```
// Prepare a record set for results
AdlibRecordSet recordSet =
    new AdlibRecordSet(conn, "collect.inf");

// Set the type of XML returned
recordSet.XmlType = XmlType.Unstructured;

// Set the sort field and order
recordSet.Sort = "creator";
recordSet.Sequence =
    AdlibRecordSet.SortSequence.Ascending;

// Do the search
recordSet.Search("all");
```



# Sorting using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "collect.inf",
  search: "all sort creator ascending",
  xmltype: "unstructured"
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```



# Complex searches

- Boolean operators and parentheses can be used to form complex search statements
- Parameter xmltype can be used to change the way the result xml is structured: xmltype=grouped or xmltype=structured (default)
- Syntax:  
[http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&search=creator='hals,frans' or \(creator='Heemskerck, Maerten van' and title=portrait\)&xmltype=grouped](http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&search=creator='hals,frans' or (creator='Heemskerck, Maerten van' and title=portrait)&xmltype=grouped)



# Complex searches

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
  - <record priref="11" selected="False">
    <administration_name>PDP</administration_name>
    <object_number>150</object_number>
    <priref>11</priref>
  - <Production>
    <creator>Hals, Frans</creator>
    <creator.date_of_birth>1583</creator.date_of_birth>
    <creator.date_of_death />
    <creator.history />
    <creator.qualifier />
    <creator.role>painter</creator.role>
    <production.notes />
    <production.place />
  </Production>
  - <Reproduction>
    <reproduction.creator />
```



# Using Adlib.Data

```
// Create a connection to the wwwopac.ashx
AdlibConnection conn = new AdlibConnection(url);

// Prepare a record set for results
AdlibRecordSet recordSet =
    new AdlibRecordSet(conn, "collect.inf");

// Set the type of XML returned
recordSet.XmlType = XmlType.Grouped;

// Do the search
recordSet.Search("creator='hals, frans' or (creator='Heemskerck, Maerten van' and title=portrait)");
```



# Using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "collect.inf",
  search: "creator='hals, frans' or (creator='Heemskerck, Maerten
          van` and title=portrait)",
  xmltype: "grouped"
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```





# Insert, update and delete

- Write
  - Inserting, updating and deleting records is allowed through the Adlib API.
  - Include *writeAllowed* element in adlibweb.xml
    - `<databaseConfiguration database="collect.inf">`  
`<database>collect</database>`  
`<writeAllowed>True</writeAllowed>`
- <http://api.adlibsoft.com/site/api/write>



# Insert using Url request

- `Wwwopac.ashx?database=<database name>&command=insertrecord&xmltype=<xmltype>&data=<xml data>`

*<database name>*: Name of the database into which we insert a record, defined in `adlibweb.xml`

*<xmltype>*: Form of xml used in the *xml data*. This can be structured or grouped

*<xml data>*: the record in Adlib xml format; `preref` field MUST be 0



# Insert using Url request

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=externalobjects&xmltype=grouped&command=insertrecord&data=...>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
  - <record priref="10000002">
    <priref>10000002</priref>
    <title>Insert record using Url request</title>
  </record>
</recordList>
```



# Insert using Adlib.Data

```
// Create a connection to the wwwopac.ashx  
conn = new AdlibConnection(url);
```

```
// Create a new adlib record  
record = new AdlibRecord(conn, "externalobjects");
```

```
// Set the priref field  
record["priref"] = "0";
```

```
// Set the title field  
record["title"] = "A new title"
```

```
// Insert the record in the database  
record.Insert();
```



# Using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";
```

```
var xmldata = '<adlibXML><recordList><record>' +  
              '<preref>0</preref><title>test title</title>' +  
              '</record></recordList></adlibXML>';
```

```
$.adlibdata(url,  
{  
  //arguments  
  database: "externalobjects",  
  command: "insertrecord",  
  data: xmldata,  
  xmltype: "grouped"  
},  
function (adlibJSON) {  
  alert(adlibJSON.diagnostic.hits);  
});
```



# Delete using Url request

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=externalobjects&command=deleterecord&priref=1000002>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
  <recordList />
  - <diagnostic>
    <hits>1</hits>
    <xmltype />
    <message>Record 1000003 has been deleted from database collect</message>
  </diagnostic>
</adlibXML>
```



# Delete using Adlib.Data

```
// Create a connection to the wwwopac.ashx
conn = new AdlibConnection(url);

// Create a new adlib record
record = new AdlibRecord(conn, "externalobjects");
record.Search(10000005);
if (record != null)
{
    // Delete the record
    record.Delete();
}
```



# Delete using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "externalobjects",
  command: "deleterecord",
  priref: 10000005
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```





# Update using Url request

- Command=*updaterecord*
- Rest if the syntax is the same as the *insert* command; only difference is that the record in *<xml data>* should already exist in the database
- <http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&xmltype=grouped&command=updaterecord&data=...>



# Update using Adlib.Data

```
// Create a connection to the wwwopac.ashx  
conn = new AdlibConnection(url);  
record = new AdlibRecord(conn, "collect.inf");
```

```
// Find record 38  
record.Search(38);  
if (record != null)  
{  
    // Change the title  
    record["title"] = "Portrait of the painter";
```

```
// Update the record  
record.Update();  
}
```



# Update using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

var xmldata = '<adlibXML><recordList><record>' +
              '<priref>10000005</priref>'+
              '<title>Portrait of the painter</title>' +
              '</record></recordList></adlibXML>';

$.adlibdata(url,
{
  //arguments
  database: "externalobjects",
  command: "updaterecord",
  data: xmldata,
  xmltype: "grouped"
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```



# Select & Deselect records

- Select
  - The *Selectrecord* command adds a record to a selection of records  
The *Deselectrecord* command removes a record from a selection of records
  - A selection can only be made if in the current session a search yielded a result set, and you want to select a record in that result set
  - <http://api.adlibsoft.com/site/api/functions/selectrecord>



# Select using Url request

- First create a resultset by doing a search±  
<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&search=all>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
+ <record priref="1" selected="False">
- <record priref="2" selected="False">
  <administration_name>PDP</administration_name>
  <creator>Dyck, Anthonie van</creator>
  <creator.date_of_birth />
  <creator.date_of_death />
  <creator.history />
  <creator.qualifier />
  <creator.role>painter</creator.role>
  <object_number>2043</object_number>
  <priref>2</priref>
```



# Select using Url request

- Then select record number 2 from the previous result:  
<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&command=selectrecord&priref=2>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
+ <record priref="1" selected="False">
- <record priref="2" selected="True">
  <administration_name>PDP</administration_name>
  <creator>Dyck, Anthonie van</creator>
  <creator.date_of_birth />
  <creator.date_of_death />
  <creator.history />
  <creator.qualifier />
  <creator.role>painter</creator.role>
  <object_number>2043</object_number>
  <priref>2</priref>
```



# Deselect using Url request

- Then deselect record number 2 from the previous result:  
<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&command=deselectrecord&preref=2>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
- <recordList>
+ <record preref="1" selected="False">
- <record preref="2" selected="False">
  <administration_name>PDP</administration_name>
  <creator>Dyck, Anthonie van</creator>
  <creator.date_of_birth />
  <creator.date_of_death />
  <creator.history />
  <creator.qualifier />
  <creator.role>painter</creator.role>
  <object_number>2043</object_number>
  <preref>2</preref>
```



# Select using Adlib.Data

```
// Prepare a record set for results
```

```
recordSet = new AdlibRecordSet(conn, "collect.inf");
```

```
// Search all records
```

```
recordSet.Search("all");
```

```
// Select record 2
```

```
recordSet.SelectRecord(2);
```

```
// Deselect record 2
```

```
recordSet.DeSelectRecord(2);
```





# Select using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{ database: "collect.inf", search: "all" },
function (adlibJSON) {
    alert(adlibJSON.diagnostic.hits);
    $.adlibdata(url,
    { database: "collect.inf", command: "selectrecord", priref: 2 },
    function (adlibJSON) {
        alert(adlibJSON.diagnostic.hits);
        $.adlibdata(url,
        { database: "collect.inf", command: "deselectrecord", priref: 2 },
        function (adlibJSON) {
            alert(adlibJSON.diagnostic.hits);
        });
    });
});
```



# Locking & unlocking records

- Lock
  - The *Lockrecord* command locks a record for editing, given its priref.
  - The *Unlockrecord* command removes a lock from a record.
  - <http://api.adlibsoft.com/site/api/lock>



# Locking using Url request

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&command=lockrecord&priref=38>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
  <recordList />
- <diagnostic>
  <hits>1</hits>
  <xmltype />
  <message>Record 38 locked for user NETWORK SERVICE</message>
</diagnostic>
</adlibXML>
```



# Locking using Adlib.Data

```
// Create a connection to the wwwopac.ashx  
AdlibApi api = new AdlibApi(url);
```

```
// Lock record 38  
api.LockRecord("collect.inf", 38);
```



# Locking using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "collect.inf",
  command: "lockrecord",
  priref: 38
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```



# Unlocking using Url request

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?database=collect.inf&command=unlockrecord&priref=38>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
  <recordList />
- <diagnostic>
  <hits>1</hits>
  <xmltype />
  <message>Record 38 unlocked</message>
</diagnostic>
</adlibXML>
```



# Unlocking using Adlib.Data

```
// Create a connection to the wwwopac.ashx
```

```
AdlibApi api = new AdlibApi(url);
```

```
// Unlock record 38
```

```
api.UnlockRecord("collect.inf", 38
```



# Unlocking using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "collect.inf",
  command: "unlockrecord",
  priref: 38
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```





# Utilities

- Utilities
  - Several functions to get meta data about the database or version information
  - <http://api.adlibsoft.com/site/api/utilities>



# Version using Url request

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?command=getversion>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <adlibXML>
  - <recordList>
    - <record priref="0">
      <priref>0</priref>
      <version>AdlibInternetServer, Version=3.6.0.39581, Culture=neutral,
        PublicKeyToken=null</version>
    </record>
  </recordList>
- <diagnostic>
  <hits>1</hits>
  <xmltype>Unstructured</xmltype>
</diagnostic>
</adlibXML>
```



# Version using Adlib.Data

```
// Create a connection to the wwwopac.ashx  
api = new AdlibApi(url);
```

```
// Get the version string  
string version = api.GetVersion();
```

```
// Returns:  
// AdlibInternetServer, Version=3.6.0.39581,  
// Culture=neutral, PublicKeyToken=null
```



# Version using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url, { command: "getversion" },
function (adlibJSON) {
    alert(adlibJSON.diagnostic.hits);
});
```



# Getting metadata

- The *Getmetadata* command returns a list of fields and their properties (in AdlibXML format) from a database.
- <http://api.adlibsoft.com/site/api/functions/getmetadata>



# Metadata using Url request

<http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx?command=getmetadata&database=collect.inf>

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
- <adlibXML>  
- <recordList>  
  - <record>  
    <tag>IN</tag>  
    + <fieldName>  
    + <displayName>  
      <length>60</length>  
      <type>Text</type>  
      <repeated>True</repeated>  
      <isLinked>False</isLinked>  
    </record>  
  - <record>  
    <tag>AG</tag>  
    - <fieldName>  
      <value lang="0">administration_name</value>  
      <value lang="1">afdeling</value>
```



# Metadata using Adlib.Data

```
// Create a connection to the wwwopac.ashx
```

```
AdlibApi api = new AdlibApi(url);
```

```
// Load the meta data into the document
```

```
XmlDocument document =  
    api.GetMetaData("collect.inf");
```



# Metadata using jQuery

```
var url = "http://test.adlibsoft.com/adlibapi/api/wwwopac.ashx";

$.adlibdata(url,
{
  //arguments
  database: "collect.inf",
  command: "getmetadata"
},
function (adlibJSON) {
  alert(adlibJSON.diagnostic.hits);
});
```





# Choosing the correct implementation

- jQuery plugin
  - Benefits
    - Improved web experience with AJAX requests. Response of the interface to the actions of the user is much faster
    - Works with JSON data, lighter *on the wire* than XML
    - Convenient way to off-load processing onto client code
    - Can be used in combination with Adlib.Data dll or Url requests.
  - Drawbacks
    - Works only with HTTP GET methods. Write commands are possible but not desirable.
    - Easy to intercept data between the browser and wwwopac.ashx. Having writeAllowed set to true in wwwopac.ashx in this scenario should be analysed in terms of security flaws. Having a specific configured wwwopac.ashx to deal with AJAX requests could be an option.



# Choosing the correct implementation

- Adlib.Data dll
  - Benefits
    - Easy to use
    - Hides the complexity of generating posting and getting urls
    - Gets XML data from wwwopac.ashx
    - Good option to use XSLT to generate final HTML
    - wwwopac.ashx can be exposed only internally
  - Drawbacks
    - Platform dependent (Windows dll)



# Choosing the correct implementation

- URL Request
  - Benefits
    - Platform independent
    - Gets XML data from `wwwopac.ashx`
    - Good option to use XSLT to generate final HTML
    - `wwwopac.ashx` can be exposed only internally
  - Drawbacks
    - Code can be more complex and difficult to maintain