



IMS App Note: Simple LTI Search v1

29 March 2016

Introduction:

Institutions have often looked for a way to easily and quickly search repositories for specific resources. Safari Montage has implemented a way to solve this problem using IMS Learning Tools Interoperability® (LTI)®.

Although Simple LTI Search is not a formal part of the Learning Tools Interoperability Standard, this application note provides information on how this functionality can be achieved using LTI and a description of how Safari Montage utilizes LTI to solve a common problem for institutions.

Our sincere thanks to David Gappa of Safari Montage for sharing this information.

SAFARI MONTAGE™



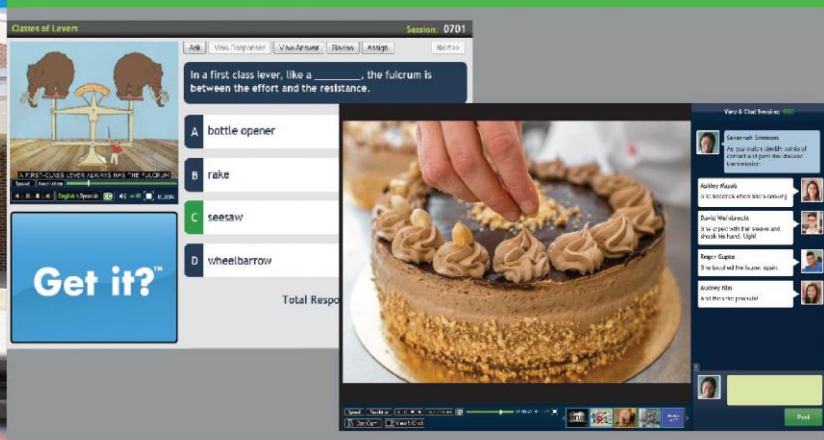
VIDEO STREAMING LIBRARY



LEARNING OBJECT REPOSITORY



IPTV & LIVE MEDIA STREAMING



DIGITAL LESSON TOOLS

SIMPLE LTI SEARCH

A Paper By: David Gappa • SAFARI Montage • March 2015- 2016

The goal is to develop an easy way to search an external system for objects matching specific search criteria. Instead of utilizing the IMS Global Thin Common Cartridge® (Common Cartridge) standard, an external call is used to allow for the most up to date results and eliminate the need to package a static group of objects for distribution.

Several open standards were reviewed to determine an appropriate approach to obtain this functionality. IMS Global standards were the initial choice since LTI® (Learning Tools Interoperability®) is an easily implemented standard and is widely used. We are currently using the standard more for Learning Object Interoperability instead of using tools but the standard is flexible enough to handle it. We looked at OpenSearch for the search portion of the requirement along with proprietary specifications and other standards.

Technical Specifications

The approach was to utilize open standards if possible and utilize IMS Global standards specifically. The LTI standard was used as the query mechanism to retrieve search results. Utilizing LTI to establish a trust relationship between the two vendors and identifying the user allows for a seamless integration for the end user. It also allows for the external vendor to revoke access if the customer doesn't renew their subscription. LTI is a great simple tool to allow access to another vendors system and allows the ability to extend functionality through custom parameters. Approaching LTI in a slightly different angle than the initial intention gives the ability to use it as a means of communication from server to server instead of the traditional user to server approach. This allows us to build a simple search mechanism that matches the look and feel of our product as a whole. The phases of this are: search, view, storage, and play.

Search

The call is the normal LTI call from the server to the external server with a custom parameter called "custom_safarimontage_search". The custom parameter takes a query string as an argument. The results are returned in JSON.

Sample Launch URL

`https://[server name]/SAFARI/api/digitallibrary/search.php`

Sample custom_safarimontage_search Parameter

`keywords=exploration&per_page_count=5&page_number=0&from_grade=4&to_grade=6&sort_by=rank&sort_direction=asc`

The acceptable parameters of the query string are listed here:

Parameter Description	Example	Options {default value}
page number	page_number=1	integer value {default=0}
per page count	per_page_count=5	integer value {default=10}

keywords	keywords=Calvin Coolidge	string values {default=""}
from grade	from_grade=4	integer value -1 through 13 {default=-1}
to grade	to_grade=13	integer value -1 through 13 {default=13}
standard code	std_code=4	string values {default=""}
sort by	sort_by=rank	rank,title,grades,publisher,creat eddatetime {default = rank}
sort direction	sort_direction=desc	asc,desc {default = desc}

```

{
  "results":{
    "count":100,
    "1":{
      "title":"Safety Smart Science with Bill Nye the Science Guy: Germs and Your Health",
      "abstract":"Bill Nye the Science Guy explains the science behind germs and overall health. Students will learn the difference between communicable and non-communicable diseases, how germs cause sickness and how to prevent the spread of germs.",
      "from grade":"4",
      "to grade":"4",
      "url":"http://10.10.1.158/?a=226666&d=28951AA",
      "lti":"http://10.10.1.158/?a=226666&d=28951AA",
      "standards":[
        {
          "label":"MI.5.1",
          "GUID":"3CCD383C-C7A5-11DB-ABAF-F681ADECDFD11"
        }
      ],
      "thumbnail":"http://10.10.1.158/SAFARI/thumbnails/b9d32ead018b33496.jpg"
    },
    ...
  }
}

```

The results are returned as JSON as shown below:

The LTI field signifies that the link is an LTI launch URL, and should be used as such, while the URL field is for no security. At least one is required.

View

Once we get the results as JSON, we parse the data and format the HTML to match our look and feel and create links to a LTI launch based on the URL of the object.

Storage

The user has the ability to “import” the LTI link into the system by clicking an icon. This will then store the LTI link and associate it to the external vendor’s key/secret (same as the search LTI launch). Once the object is stored, we will then create a thumbnail of the screenshot of the URL accessed to give feedback to the user. Subsequent calls to the external search engine will result in identification based on the unique ID and if matching an already saved LTI link, we will display the thumbnail and internal functionality instead of the more generic simple search result.

Play

On the search results or once stored in the system. The object is treated just like any other LTI object within the system. The user can click play and the system will create a LTI launch to the external object and display.

Expansion

Since the calling mechanism is a LTI Post with a custom parameter for the search, this protocol is expandable based on the ability to add more parameters to the URL or adding an additional custom parameter for larger data requests. For example, adding “&subjects=math|science” as a filter. The return JSON can be easily expanded to include more fields, URLs to thumbnails, or anything else needed to display and reference each object. For example, the return JSON for an object could contain object IDs to related titles or could include state standards data.

Enhancements

To make this protocol clean and expandable without utilizing a ton of post variables or get variables, the use of JWT (JSON Web Tokens, <http://jwt.io>) might be a clean way to do this. So for example, instead of using a query string in the custom_safarimontage_search post variable, this could be straight JSON data or encoded JWT using the secret used for the LTI signing. If LTI changes to using JWTs, this would still work since it would

```
{ "alg": "HS256", "type": "JWT" }

{
  "iss": "schoola.safarimontage.com",
  "iat": 1457630027,
  "exp": 1457716427,
  "keywords": "Calvin Coolidge",
  "page number": "1",
  "per page count": "25",
  "from grade": "4",
  "to grade": "8",
  "standard code": "3CCD383C-C7A5-11DB-ABAF-F681ADECDFD11",
  "sort by": "rank",
  "sort direction": "desc"
}
```

just be double signed.

The first part of the sample is the header, the second part is the payload. Each piece is encoded per JWT

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG91IiwiaWF0IjoiOnRydWV9.TJVA95OrM7E2cBab30RMhrHDcEfxjYoYZgeFONfh7HgQ
```

and a signature is added, these are separated by a dot.

This string is then used as the value in the post parameter of `custom_safarimontage_search`. The field names should be changed to match IMS Global standard names for those fields and the vocabulary for each field should match the standard used in other standards.

The search results can now be done the same way using JWT. The payload can be quite large but since it's JSON, it maps to the JWT nicely and since the return is now signed, it can be verified as being genuine preventing any interception of data. All communications should be done using SSL but if not, the JWT can include encryption using JWE (JSON Web Encryption).