

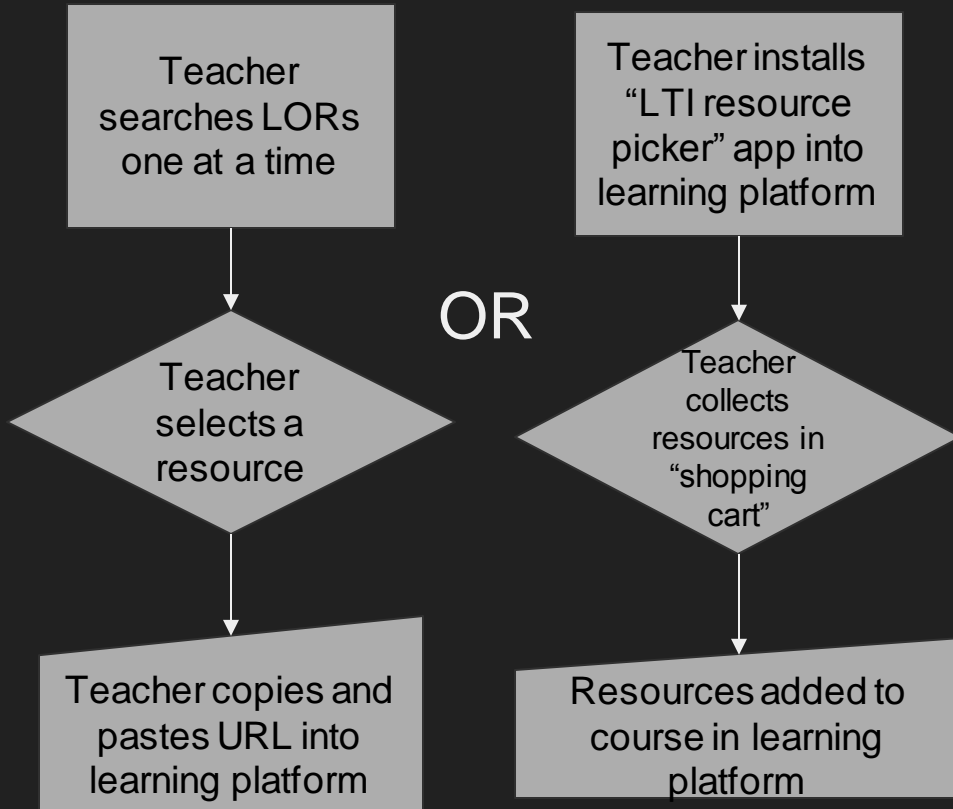
Integrating with Learning Object Repositories With LTI Resource Search

ETCPS
October 2018

Why Weave OpenEd Resources Into Your Ed Product

- Videos, simulations, quizzes aligned to all important standards and skills
- Measured efficacy (based on video consumption followed by assessment)
- All high quality instructional videos that exist
- Classroom assessments from most popular formative assessment banks

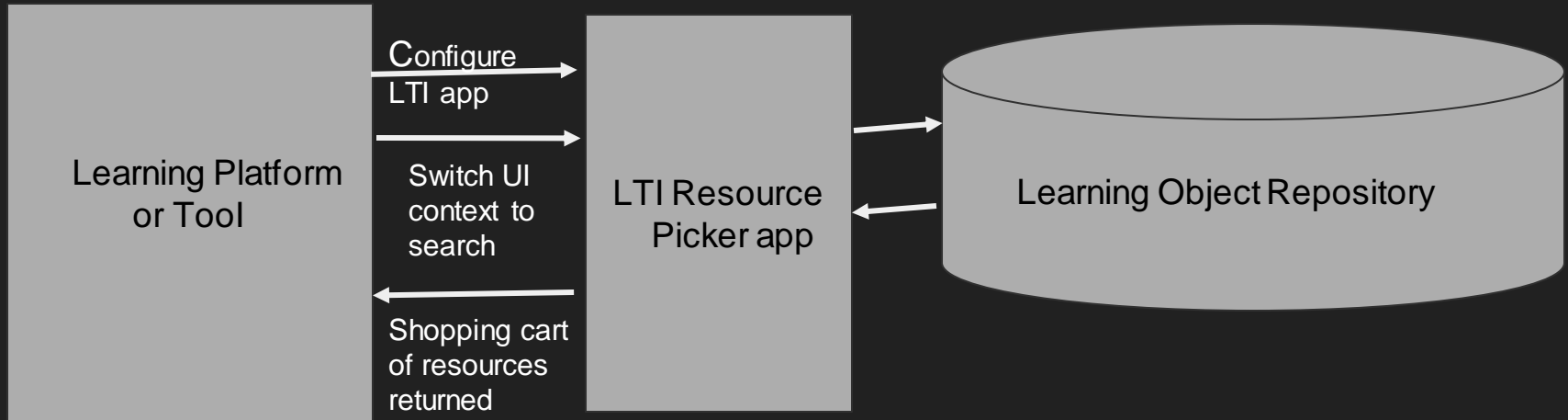
How Do Learning Platforms Integrate Resources Today?



● What's the problem?

- Inconsistent User Interfaces
- Learning Platform should be the "teacher cockpit"
- LORs have unnecessary development burden for LTI "resource picker" apps
- LTI apps add additional credentialing requirements which aren't needed in an API search call

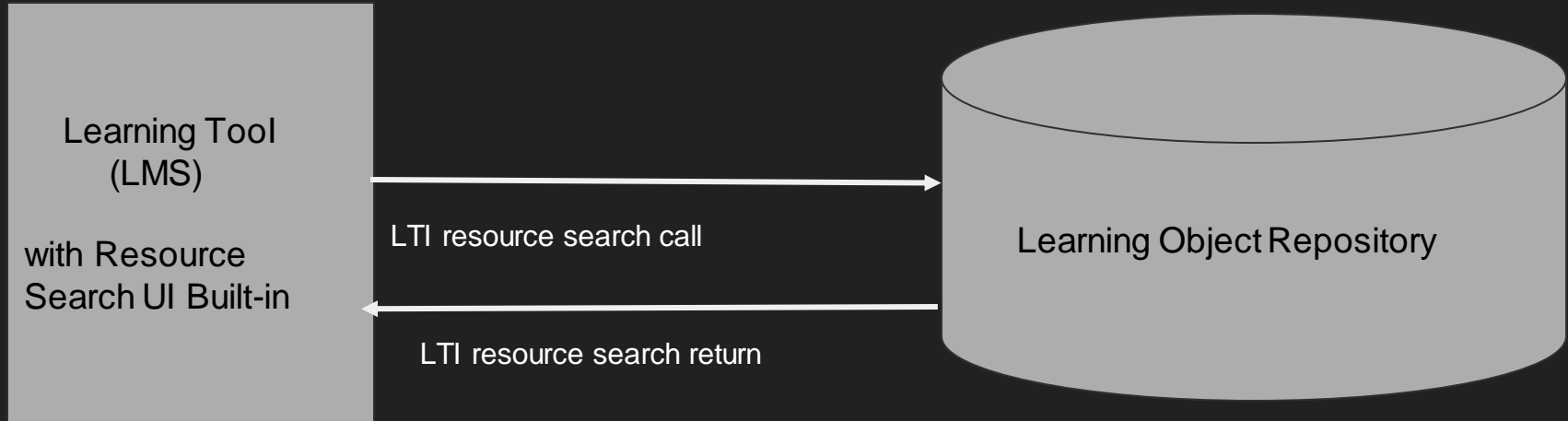
Many Moving Parts, UI Switches, Opportunities for Failure



So What's A Better Way?

- Provide a standard REST API for searching LORs
- Let the learning platform or tool own the teacher experience
- Provide the ability for the learning platform or tool to search multiple LORs
 - With little work for the learning platform or tool
 - And even less work for the teacher
- LORs get to implement one search API
 - And connect to many LMSes

Ah Sweet Simplicity... For the Developer and the User



So Why A New Standard Now?

- Current process is too complicated for teachers to use the digital resources in multiple LORs
- Finally consensus on what a learning object should have as metadata:
 - We build on LRMI/schema.org
- REST APIs are commonplace now
 - Specifically IMS has made some nice progress on REST/JSON APIs with OneRoster that can be used as a model

So What Comprises the Standard?

- REST calls for resource searching and getting possible values (such as subjects)
- Resource metadata (the payloads of returned data)
- Supplementary definitions of certain structures (such as learning objectives)

Normalized metadata for all LORs is really quite a big deal

What Do We Care About for Learning Resources?

- resource name and description
- resource type
- publisher or owner of the resource
- license that applies (such as Creative Commons or a publisher's URL to their license)
- duration (time to consume)
- web link or **LTI link** to access
- technical format (MIME types such as "text/html", "video/mpeg")
- educational audience (student, teacher, administrator, parent, other)
- thumbnail image
- subject
- language
- age range (more int'l than grade)
- learning objective (such as a state standard)
- **CasItemURI**
- **CasItemGUID**
- author
- publish date
- **rating**
- **relevance**

* **not in LRMI/Schema.org**

Differences from LRMI

Learning Resource Types

- Work done by CCSSO Communities of Practice to define resource types
- Hierarchical approach enables many types without cognitive overload
- Resources can be tagged with multiple resource types simultaneously
- Examples:
 - Assessment/Formative, Assessment/Interim
 - Collection/Course, Collection/Unit
 - Text/Book, Text/Passage
 - Media/Video

What Does the REST API Look Like?

- An example search

- <https://imglobal.org/ims/ltisearch/resources?filter=search%3D%27civil%20war%27>
- Note: arguments to **filter** parameter are URL encoded (hence need for filter parameter)

- Search (filter) data fields:

- **search (searches multiple fields as LOR chooses)**
- **name**
- **description**
- **subject**
- **learningResourceType**
- **language**
- **typicalAgeRange**
- **textComplexity**
- **learningObjectives**
- **author**
- **publisher**
- **timeRequired**
- **technicalFormat**
- **educationalAudience**
- **accessibilityAPI**
- **accessibilityInputMethods**
- **publishDate**
- **rating**
- **relevance**

Filtering Options

OneRoster offers powerful searching controls starting with **filter** with two options:

1. Full predicate logic:

```
?filter=<data field><predicate><value>
```

OR

```
?filter=<data field><predicate><value><logical><data field><predicate><value>
```

- Predicates: =, !=, >, >=, <, <=

2. "attribute=" and "attribute~" shorthands (after "filter=")

- This provides OR searching semantics
- ?filter="subject=subject1" - record not returned;
- ?filter="subject=subject1,subject2" - record not returned;
- ?filter="subject=subject1,subject2,subject3" - record returned;
- ?filter~"subject=subject1" - record returned;
- ?filter~"subject=subject1,subject2" - record returned;
- ?filter~"subject=subject1,subject2,subject3" - record returned.

NOTE: To support this predicate logic we MUST have a "filter=" parameter and URL encoded query

Pagination, Sorting and Selection

These options introduced by OneRoster control how data is returned

- **Pagination**

- Limit (default 100)
- Offset (default zero)
- `https://imglobal.org/ims/ltisearch/resources?limit=10&offset=0`

- **Sorting**

- `sort=<data field>` (but not multiField)
- `orderBy=asc | desc`
- `https://imglobal.org/ims/ltisearch/resources?sort=publishDate&orderBy=desc`

- **Selection**

- Defaults to all fields returned
- Or list the ones you want
 - <https://imglobal.org/ims/ltisearch/resources?fields=name,url>

Current Status

- Spec is FINAL and PUBLIC!
 - <http://imglobal.org/resource-search>
 - Including Swagger/OpenAPI: <https://www.imglobal.org/spec/lti-rs/v1p0/openapi>
- Two certified providers: ACT OpenEd, Knovation
- Multiple certified clients: ACT OpenEd, SAFARI Montage, Knovation, SchoolCity
- Come to our Webinar on October 31st! <https://www.imglobal.org/event/only-treats-no-tricks-new-lti-resource-search>

Futures for LTI Resource Search

- Search for assessment objects (items, forms)
 - The metadata there is quite different from instructional resources
 - Should be mostly QTI
 - Do institutions (states, districts) want this?
- Aggregation from multiple LORs?
 - Who will be implementing the aggregation methods? The LMS?
 - What is a realistic way to merge results? Relevance?! Publish date?
 - Is the net result better than multiple tabs?
- Potential standardization of K-12 subjects
- Other ideas?

Call to Action for Suppliers

- Check out the spec at https://www.imslobal.org/sites/default/files/spec/lti-rs/v1p0/rest_binding/rsservicev1p0_restbindv1p0.html
 - Should save you quite a few steps of navigation
- See my open source sample client for key steps to implement
 - http://github.com/openedinc/srch_cli
- Go get certified!
 - <https://www.imslobal.org/learning-tools-interopability-lti-resource-search%E2%84%A2-conformance-certification-testing>