

Paying is not enough

Alexander Wagner
Regensburg, 07.10.2024

Overview



- > Publications in Science
- > Publishing and Libraries
- > Workflow considerations
- > openCost from a Workflow Perspective

Role(s) of Publications

- > Exchange ideas
- > Spread knowledge
- > Further communication

Role(s) of Publications

- > Exchange ideas
- > Spread knowledge
- > Further communication

Newton to Hooke, 1676-02-05

“If I have seen further it is
by standing on ye shoulders of giants.”¹

¹ Original: Bernhard von Chartres, ca. 1120

Role(s) of Publications

- > Exchange ideas
- > Spread knowledge
- > Further communication
- > Claim priority
- > Gain reputation

Newton to Hooke, 1676-02-05

“If I have seen further it is
by standing on ye shoulders of giants.”¹

¹ Original: Bernhard von Chartres, ca. 1120

Role(s) of Publications

- > Exchange ideas
- > Spread knowledge
- > Further communication
- > Claim priority
- > Gain reputation
- > Justify funding
- > Evaluate research

Newton to Hooke, 1676-02-05

“If I have seen further it is
by standing on ye shoulders of giants.”¹

¹ Original: Bernhard von Chartres, ca. 1120

Role(s) of Publications

- > Exchange ideas
- > Spread knowledge
- > Further communication
- > Claim priority
- > Gain reputation
- > Justify funding
- > Evaluate research

- > Make money
- > Mine data

Newton to Hooke, 1676-02-05

“If I have seen further it is
by standing on ye shoulders of giants.”¹

¹ Original: Bernhard von Chartres, ca. 1120



Logos by Wikipedia

A Scientists Point of View

A Scientists Point of View

- > I have to write papers
...for people to read me.
- > I have to ensure to get read
...to get cited.
- > I have to report my papers
...to get funded.
- > I have to get funded
...to do science.

A Scientists Point of View

- > I have to **write** papers
...for people to read me.
- > I have to ensure to **get read**
...to get cited.
- > I have to **report** my papers
...to get funded.
- > I have to **get funded**
...to do science.

I have to key in *the very same data* into

- 1 the **preprint** submission so people read me early
- 2 the **publishers** system to make citations count
- 3 the **funding application** form to finance the publication
- 4 the local **repository** to ensure broad Open Access
- 5 the institutions **CRIS system(s)** to get the credits
- 6 the universities **Bibliography**
- 7 the **web pages** and **media** etc. to spread the word
- 8 the **funders** evaluation system(s) for the next grant

A Scientists Point of View

- > I have to **write** papers
...for people to read me.
- > I have to ensure to **get read**
...to get cited.
- > I have to **report** my papers
...to get funded.
- > I have to **get funded**
...to do science.

I have to key in *the very same data* into

- 1 the **preprint** submission so people read me early
- 2 the **publishers** system to make citations count
- 3 the **funding application** form to finance the publication
- 4 the local **repository** to ensure broad Open Access
- 5 the institutions **CRIS system(s)** to get the credits
- 6 the universities **Bibliography**
- 7 the **web pages** and **media** etc. to spread the word
- 8 the **funders** evaluation system(s) for the next grant

Actually, I wanted to do **science**...

Libraries are Service Facilities

Interaction points:

- 3 the funding application form
- 4 the institutional repository
- 5 the local CRIS system
- 6 the university Bibliography
- 7 the web pages and media etc
- 0 interfaces between systems

Libraries are Service Facilities

Interaction points:

- 3 the funding application form
- 4 the institutional repository
- 5 the local CRIS system
- 6 the university Bibliography
- 7 the web pages and media etc
- 0 interfaces between systems

Reduce manual input!

Reuse data!

Libraries are Service Facilities

Interaction points:

- 3 the funding application form
- 4 the institutional repository
- 5 the local CRIS system
- 6 the university Bibliography
- 7 the web pages and media etc
- 0 interfaces between systems

Reduce manual input!

Reuse data!

*Where is the **sweet spot**?*

*What is a **proper workflow** for the whole publishing process?*

3 The Funding Application Form

What happens, if the funding of publication costs is **not** approved?

3 The Funding Application Form

What happens, if the funding of publication costs is **not** approved?

The author(s) will

- 1 ...be grateful for our service
- 2 ...not publish
- 3 ...use another venue
- 4 ...happily fill in another form

“Oh, indeed! You're right! n***** is too expensive! I'll publish in some other journal, they'll probably accept it. Just a few more referees and another half a year. Time well spent.
Oh, again funding is not approved?! Well...”

3 The Funding Application Form

What happens, if the funding of publication costs is **not** approved?

The author(s) will

- 1 ...be grateful for our service
- 2 ...not publish
- 3 ...use another venue
- 4 ...happily fill in another form

“Oh, indeed! You’re right! n***** is too expensive! I’ll publish in some other journal, they’ll probably accept it. Just a few more referees and another half a year. Time well spent.
Oh, again funding is not approved?! Well...”

- 1 ...not be amused
- 2 ...find a work around
- 3 ...handle it themselves
- 4 ...waste a lot of time

It is payed anyway,
but **way more expensive**.

We create “APCs in the wild”...

4 The Institutional Repository

- > Mints and supplies **persistent identifiers**
- > Ensures permanent **full text** access
- > Supports many publication types
- > Complies to **funder requirements**
- > Feeds to (some) **funder reports**
- > Uses **standardized** formats / interfaces
- > Allows for **easy data-reuse**
- > submission \approx "application form"?

4 The Institutional Repository

- > Mints and supplies **persistent identifiers**
- > Ensures permanent **full text** access
- > Supports many publication types
- > Complies to **funder requirements**
- > Feeds to (some) **funder reports**
- > Uses **standardized** formats / interfaces
- > Allows for **easy data-reuse**
- > submission \approx "application form"?

Already holds required data

- > bibliographic description
- > administrative data
- > legal data, licences
- > full text files
- > technical data

4 The Institutional Repository

- > Mints and supplies **persistent identifiers**
- > Ensures permanent **full text** access
- > Supports many publication types
- > Complies to **funder requirements**
- > Feeds to (some) **funder reports**
- > Uses **standardized** formats / interfaces
- > Allows for **easy data-reuse**
- > submission \approx "application form"?

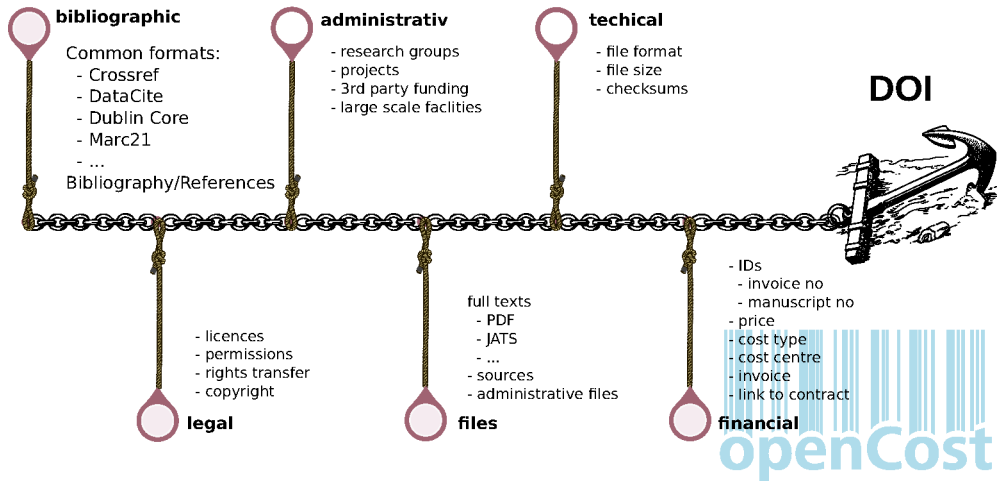
Already holds required data

- > bibliographic description
- > administrative data
- > legal data, licences
- > full text files
- > technical data

Why not...

...just add financial data?

Give me a repository and a persistent ID and...



5 The Local CRIS System / 6 The Bibliography

Does it

- ? ...mint and supply persistent ids
- ? ...ensure full text access
- ? ...comply to funder requirements
- ? ...support all relevant publication types
- ? ...supply standardized formats / interfaces

5 The Local CRIS System

/

6 The Bibliography

Does it

- ? ...mint and supply persistent ids
- ? ...ensure full text access
- ? ...comply to funder requirements
- ? ...support all relevant publication types
- ? ...supply standardized formats / interfaces

If **no**,
add integration with the [repository](#).

Probably, center the workflow on the repository.

5 The Local CRIS System / 6 The Bibliography

Does it

- ? ...mint and supply persistent ids
- ? ...ensure full text access
- ? ...comply to funder requirements
- ? ...support all relevant publication types
- ? ...supply standardized formats / interfaces

If **no**,
add integration with the [repository](#).

Probably, center the workflow on the repository.

If **yes**,
it holds holds required data.

Why not...

...just add financial data?

NaN The ILS

- > Allows for **easy budgeting**
- > Already knows how to handle costs
- > Support all relevant publication types

(all you need are diligent cataloguers)

NaN The ILS

- > Allows for **easy budgeting**
- > Already knows how to handle costs
- > Support all relevant publication types

(all you need are diligent cataloguers)

Does not

- > mint and supply persistent IDs
- > ensure full text access
- > comply to funder requirements
- > allow for easy enhancements
- > handle dependent works well

NaN The ILS

- > Allows for **easy budgeting**
- > Already knows how to handle costs
- > Support all relevant publication types

(all you need are diligent cataloguers)

Does not

- > mint and supply persistent IDs
- > ensure full text access
- > comply to funder requirements
- > allow for easy enhancements
- > handle dependent works well

Administrative Tool

Require **interfaces** to the Repository and/or CRIS system **to serve scientists**.

The Sweet Spot(s)

1 The Repository

2 The CRIS system / The Bibliography

(Most likely requires integration with the repository.)

3 The ILS

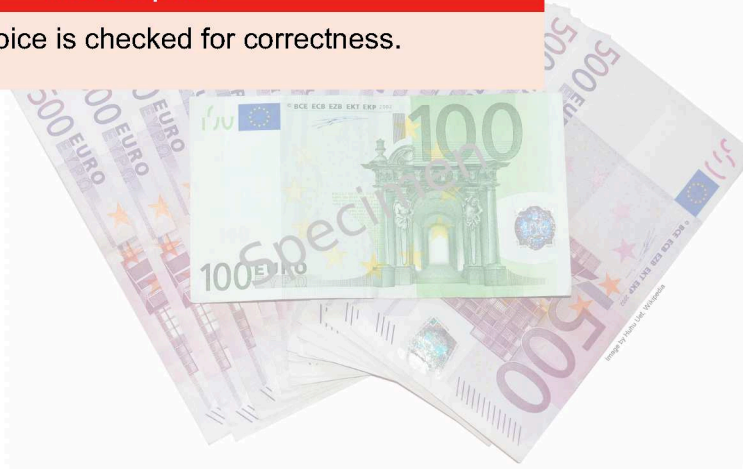
(Always requires interfaces to feed other systems.)

4 The "Funding Application Form"

Workflow

Assumption

Every invoice is checked for correctness.



Workflow

Assumption

Every invoice is checked for correctness.

Define,

- 1 a central point to handle all publication-related invoices

(Think buying books, just way more expensive...)

- 2 if and how (internal) cost splitting works

(Allow the library to charge different accounts and cost centres.)

- 3 the most suitable system from a scientists point of view

(Get out most for *them*.)

- 4 Quality of Service

(A journal is called ...*Express* or ...*Letters* for a reason.)

Workflow

Assumption

Every invoice is checked for correctness.

Define,

- 1 a central point to handle all publication-related invoices

(Think buying books, just way more expensive...)

- 2 if and how (internal) cost splitting works

(Allow the library to charge different accounts and cost centres.)

- 3 the most suitable system from a scientists point of view

(Get out most for *them*.)

- 4 Quality of Service

(A journal is called ...*Express* or ...*Letters* for a reason.)

...like for any book...

Once an invoice arrives...

- Check the invoice for correctness
- Create a record (may be preliminary bibliographic data)
- Add cost data to the record
- If a funding application is required: fill it out for your scientists and hand it in
- Add publishers full text (you are about to buy it...)
- Add digital invoice as (private) full text
- Hand the invoice to financial controlling
- Release the record to Open Access
(according the licence just bought)



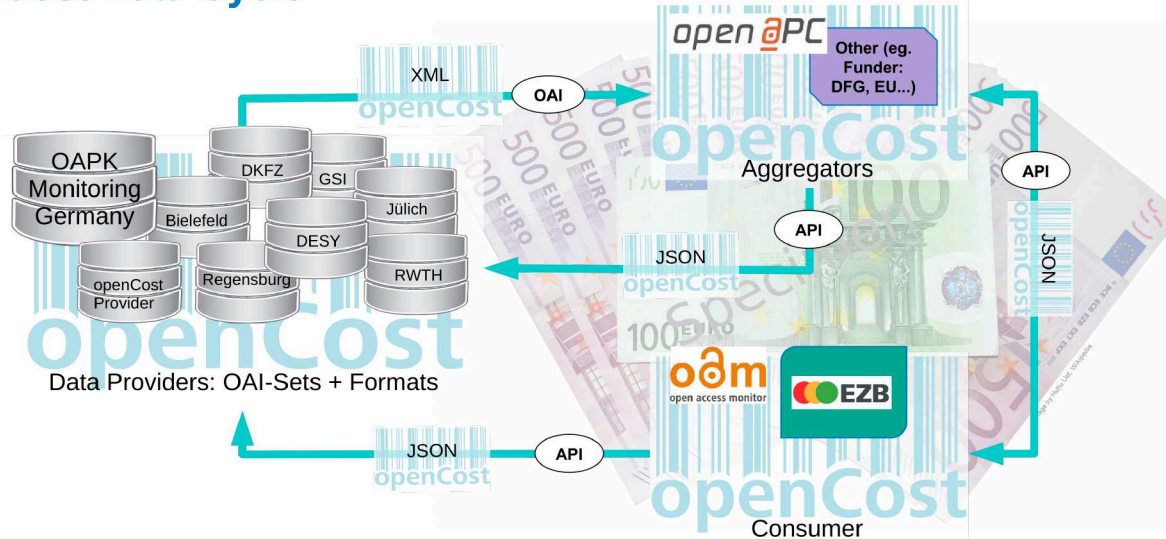
Once an invoice arrives...

- Check the invoice for correctness
- Create a record (may be preliminary bibliographic data)
- Add cost data to the record
- If a funding application is required: fill it out for your scientists and hand it in
- Add publishers full text (you are about to buy it...)
- Add digital invoice as (private) full text
- Hand the invoice to financial controlling
- Release the record to Open Access
(according the licence just bought)

Ensure
data is passed on
to other systems

(CRIS, web, funders, cost monitoring...)

Cost Data Cycle



Why OAI-PMH?

This interface is ancient and uncool.

“We use REST and JSON these days, not HTTP and XML...”

Why OAI-PMH?

This interface is ancient and uncool.

"We use REST and JSON these days, not HTTP and XML..."

- > Widely established
- > Numerous software libraries
- > Stood the test of time
- > Proven to **scale well**
- > Format-agnostic

Why OAI-PMH?

This interface is ancient and uncool.

"We use REST and JSON these days, not HTTP and XML..."

- > Widely established
- > Numerous software libraries
- > Stood the test of time
- > Proven to **scale well**
- > Format-agnostic

Who wants the data?

Why OAI-PMH?

This interface is ancient and uncool.

"We use REST and JSON these days, not HTTP and XML..."

- > Widely established
- > Numerous software libraries
- > Stood the test of time
- > Proven to **scale well**
- > Format-agnostic

Who wants the data?

Some aggregator.

Why OAI-PMH?

This interface is ancient and uncool.

"We use REST and JSON these days, not HTTP and XML..."

- > Widely established
- > Numerous software libraries
- > Stood the test of time
- > Proven to **scale well**
- > Format-agnostic

Who wants the data?

Some aggregator.
Probably, some I am not yet aware of...

Why OAI-PMH?

This interface is ancient and uncool.

"We use REST and JSON these days, not HTTP and XML..."

- > Widely established
- > Numerous software libraries
- > Stood the test of time
- > Proven to **scale well**
- > Format-agnostic
- > **Aggregator-agnostic**

Who wants the data?

Some aggregator.
Probably, some I am not yet aware of...

Why OAI-PMH?

This interface is ancient and uncool.

"We use REST and JSON these days, not HTTP and XML..."

- > Widely established
- > Numerous software libraries
- > Stood the test of time
- > Proven to **scale well**
- > Format-agnostic
- > **Aggregator-agnostic**

Who wants the data?

Some aggregator.

Probably, some I am not yet aware of...

Minimized technical workload at the data providers.

No need to write individual code for each service.

Data Updates

- > Corrections (e. g. German classics: 3.000 € → 3000 €)
- > Add missing records (e. g. taming APCs)
- > Deleting records

Data Updates

- > Corrections (e. g. German classics: 3.000 € → 3000 €)
- > Add missing records (e. g. taming APCs)
- > Deleting records
- > Subsequent payments (e. g. DEAL)
- > Refunding (e. g. with prepayments)

Data Updates

- > Corrections (e. g. German classics: 3.000 € → 3000 €)
- > Add missing records (e. g. taming APCs)
- > Deleting records
- > Subsequent payments (e. g. DEAL)
- > Refunding (e. g. with prepayments)

OAI ensures to transfer only

- > changed records
- > new records
- > deleted records

It can easily be forced to (re-)retrieve **all** records.

Data Updates

- > Corrections (e. g. German classics: 3.000 € → 3000 €)
- > Add missing records (e. g. taming APCs)
- > Deleting records
- > Subsequent payments (e. g. DEAL)
- > Refunding (e. g. with prepayments)

OAI ensures to transfer only

- > changed records
- > new records
- > deleted records

It can easily be forced to (re-)retrieve **all** records.

For aggregators...

...updating of records is seamless and efficient.

Data Updates

- > Corrections (e. g. German classics: 3.000 € → 3000 €)
- > Add missing records (e. g. taming APCs)
- > Deleting records
- > Subsequent payments (e. g. DEAL)
- > Refunding (e. g. with prepayments)

OAI ensures to transfer only

- > changed records
- > new records
- > deleted records

It can easily be forced to (re-)retrieve **all** records.

For aggregators...

...updating of records is seamless and efficient.
Data can be transferred **frequently** and **regularly**.

Why is openCost defined in XML?

- > embeddable
- > standardized
- > UTF-8 safe
- > allows validation (`opencost.xsd`)
- > used in many processes
- > OAI-PMH is defined on XML

Why is openCost defined in XML?

- > embeddable
- > standardized
- > UTF-8 safe
- > allows validation ([opencost.xsd](#))
- > used in many processes
- > OAI-PMH is defined on XML

openCost uses *XML light*

- > type/value constructs
- > no attributes

```
<institution>
  <id>
    <type>ror</type>
    <value>
      https://ror.org/01js2sh04
    </value>
  </id>
</institution>
```

```
"institution": {
  "id": {
    "type": "ror",
    "value":
      "https://ror.org/01js2sh04"
  }
}
```


Why is openCost defined in XML?

- > embeddable
- > standardized
- > UTF-8 safe
- > allows validation ([opencost.xsd](#))
- > used in many processes
- > OAI-PMH is defined on XML

openCost uses *XML light*

- > type/value constructs
- > no attributes

```
<institution>
  <id>
    <type>ror</type>
    <value>
      https://ror.org/01js2sh04
    </value>
  </id>
</institution>
```

```
"institution": {
  "id": {
    "type": "ror",
    "value":
      "https://ror.org/01js2sh04"
  }
}
```

openCost-XML allows

easy, unambiguous serialization in other formats

Thank you!

Contact

Deutsches Elektronen-
Synchrotron DESY

www.desy.de

Alexander Wagner
0000-0001-9846-5516
Central Library
alexander.wagner@desy.de
+49-40-8998-1758
[10.3204/PUBDB-2024-06081](https://doi.org/10.3204/PUBDB-2024-06081)



Typeset by lua^AT_EX