SLSP

Managing mass corrections in SLSP: approaches and experiences

17.09.2025, IGeLU conference in Siena
Raphaël Rey & Nikol Stepan

Nikol Stepan

Senior Data and System Specialist /
Business Analyst

Raphaël Rey

Team Leader System Operations

Agenda

1. Mass Corrections in a Consortium
   a) Examples
   b) Organizational Context
2. Tools
   a) Alma integrated tools
   b) External Database
   c) Alma API Wrapper
3. Conclusion
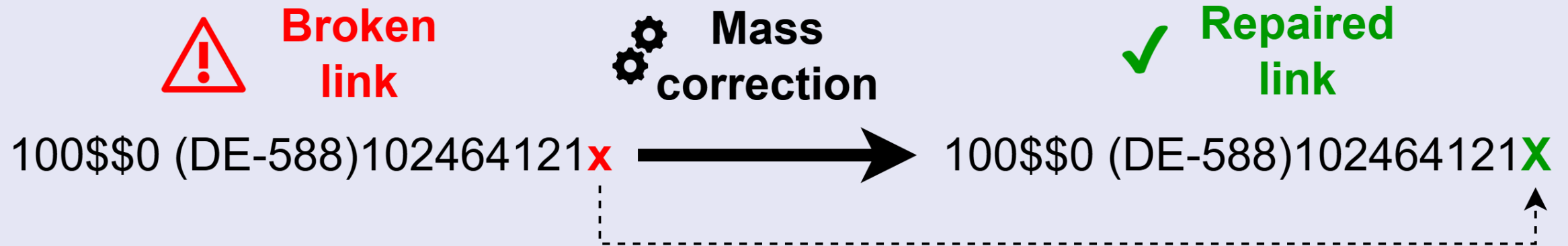
1. Mass Corrections in a Consortium
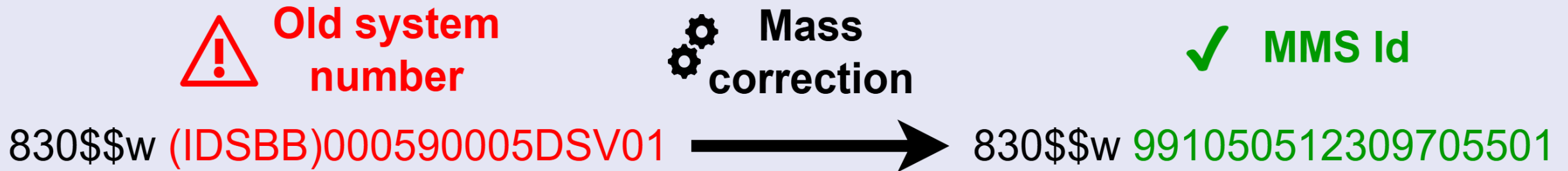
1.a Examples

GND ID with lower case x
Link to DNB is broken ➔ display in swisscovery and other services affected

Broken link

Mass correction

✔ **Repaired link**

100$$0 (DE-588)102464121x ➔ 100$$0 (DE-588)102464121X

1.a Examples

Replace $w identifiers with MMS ID
In fields 76x, 77x, 78x, 800, 810, 811, 830, replace the "old" networks identifiers with the MMS ID

⚠ **Old system number**   ⚙ **Mass correction**   ✔ **MMS Id**

830$$w (IDSBB)000590005DSV01 ➡ 830$$w 99105051230970 5501

1.a Examples

772: 18'372

773: 1'402'324 **!**

780: 87'811

785: 73'442

800: 330'568

830: > 3 million **!!**

How to do it ?

1.b Organizational Context
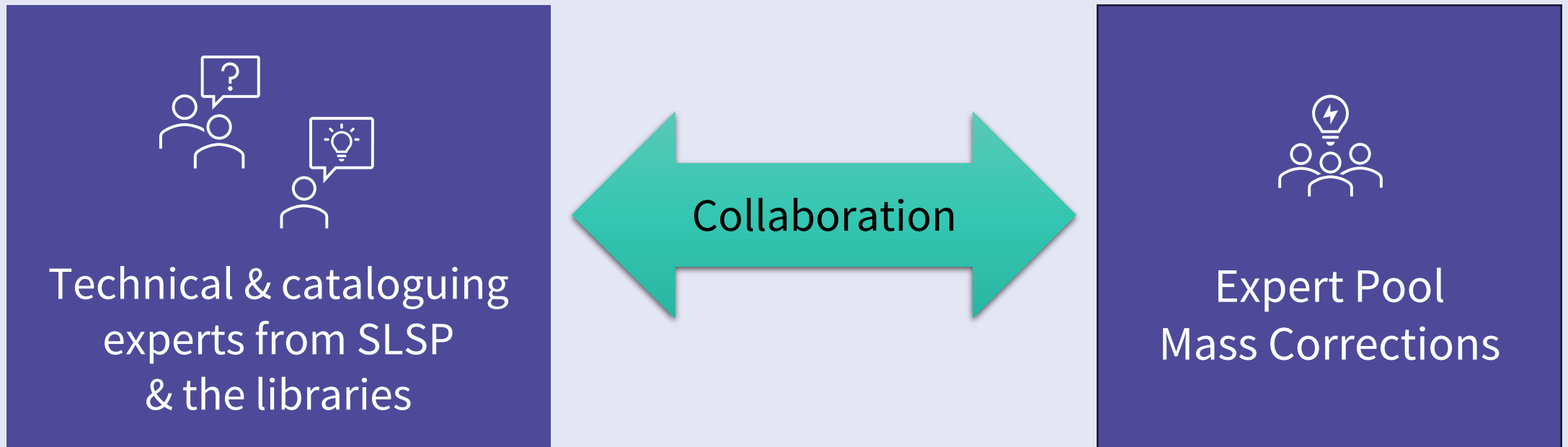
SLSP
Swiss Library
Service Platform

Consortium of over 500 Swiss academic libraries

Search portal swisscovery

31 IZs + 1 NZ
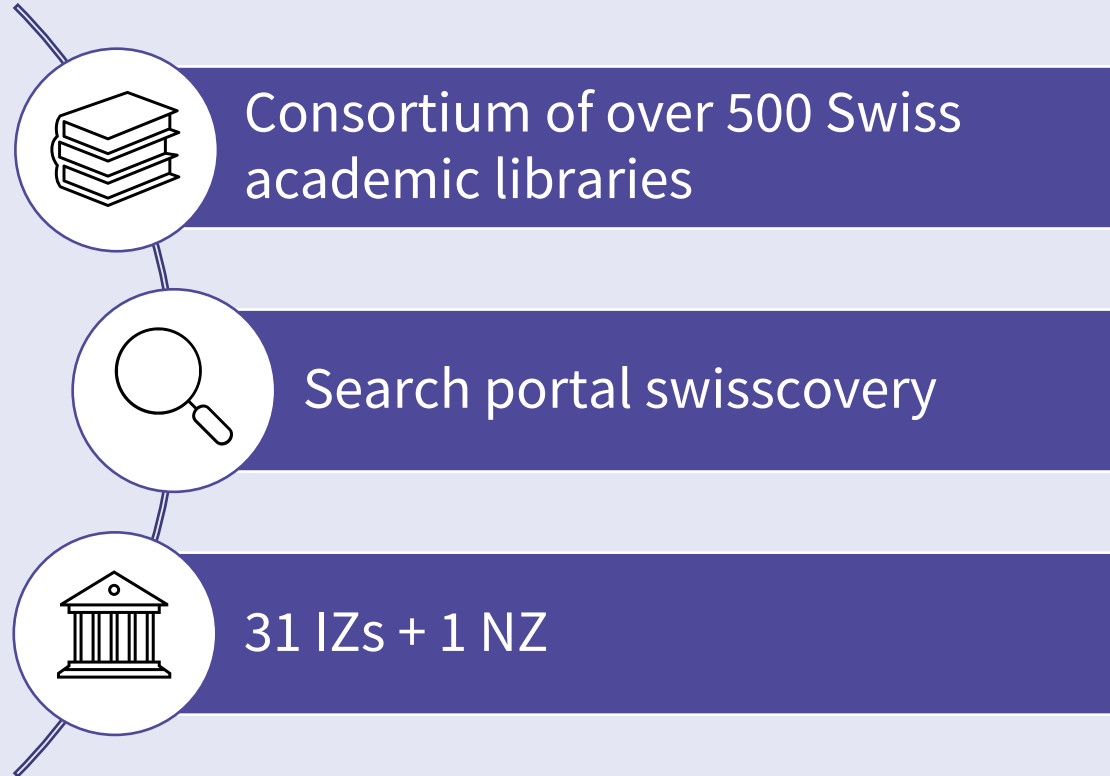
1.b Organizational Context
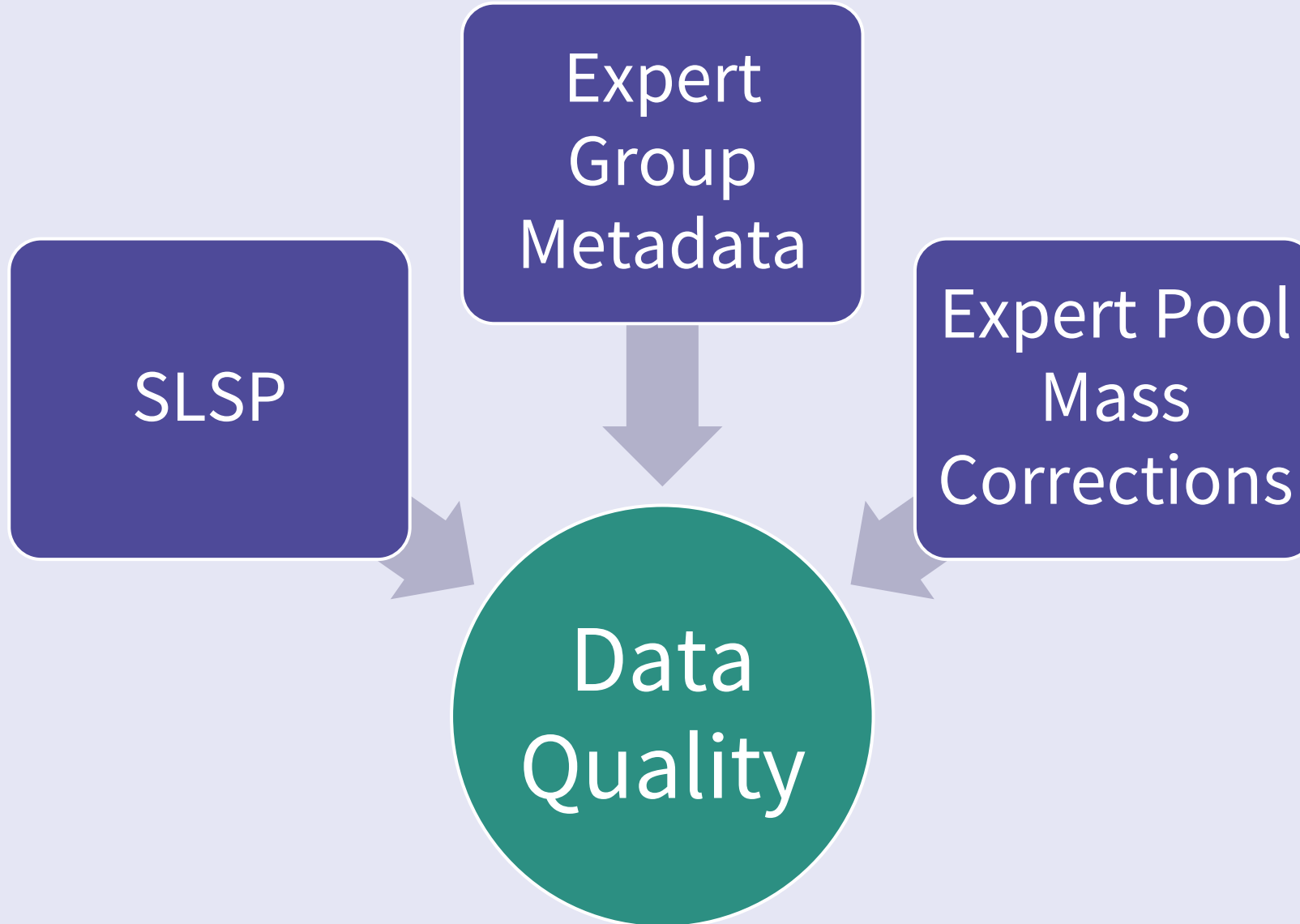
1.b Organizational Context

How to collaborate the best way?

Technical & cataloguing experts from SLSP & the libraries

Collaboration

Expert Pool
Mass Corrections

1.b Organizational Context

Lead

SLSP

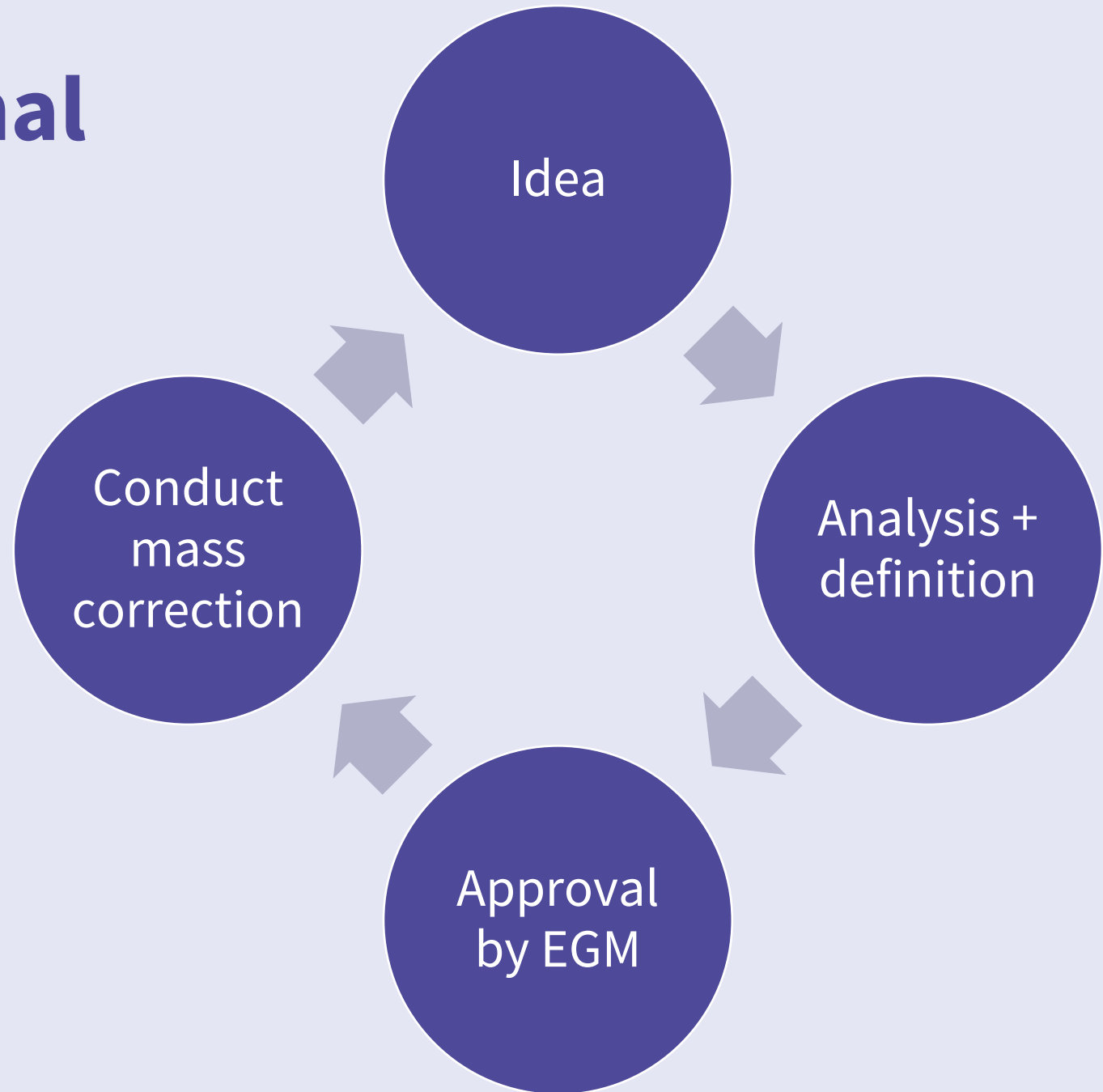**Members**

8 mass correction experts of 7 libraries

Opportunity

Several members have cataloguing and technical knowledge

1.b Organizational Context

- Iterative workflow
- Meetings: Twice a year
- Up to three corrections at a time

1.b Organizational Context

Ideas for corrections:

- Transparent to all
- Currently 55 in total, always 3 in progress
- New ideas can be added at any time
- Priority determined by voting

55 topics found

Sticky	Topic Name
📌	Wrong IDs in $w of series or linking entries
📌	Anchoring records without inventory in the French spe…
📌	Replace $w identifiers with MMS
📌	Monographs ("nam-Records") with 022 ISSN
📌	Records with nas in LDR: no ISBNs or DOIs allowed
📌	Field 655: delete field "655_7 $a Bilderbuch $2 idsmusi"
📌	GND ID with lower case x
📌	E-book recordings encoded in print
📌	Date only in 264, missing in 008
📌	Delete fields 883 and corresponding/orphaned $8

2. Tools

2.a Alma integrated tools

What is the best way to modify large volumes of data in complex use cases with clean and reliable logging?

Normalization rules & indication rules
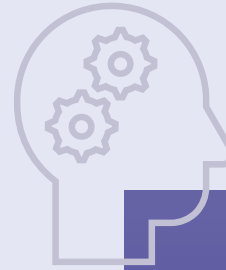
Import profile/ merge rules

API
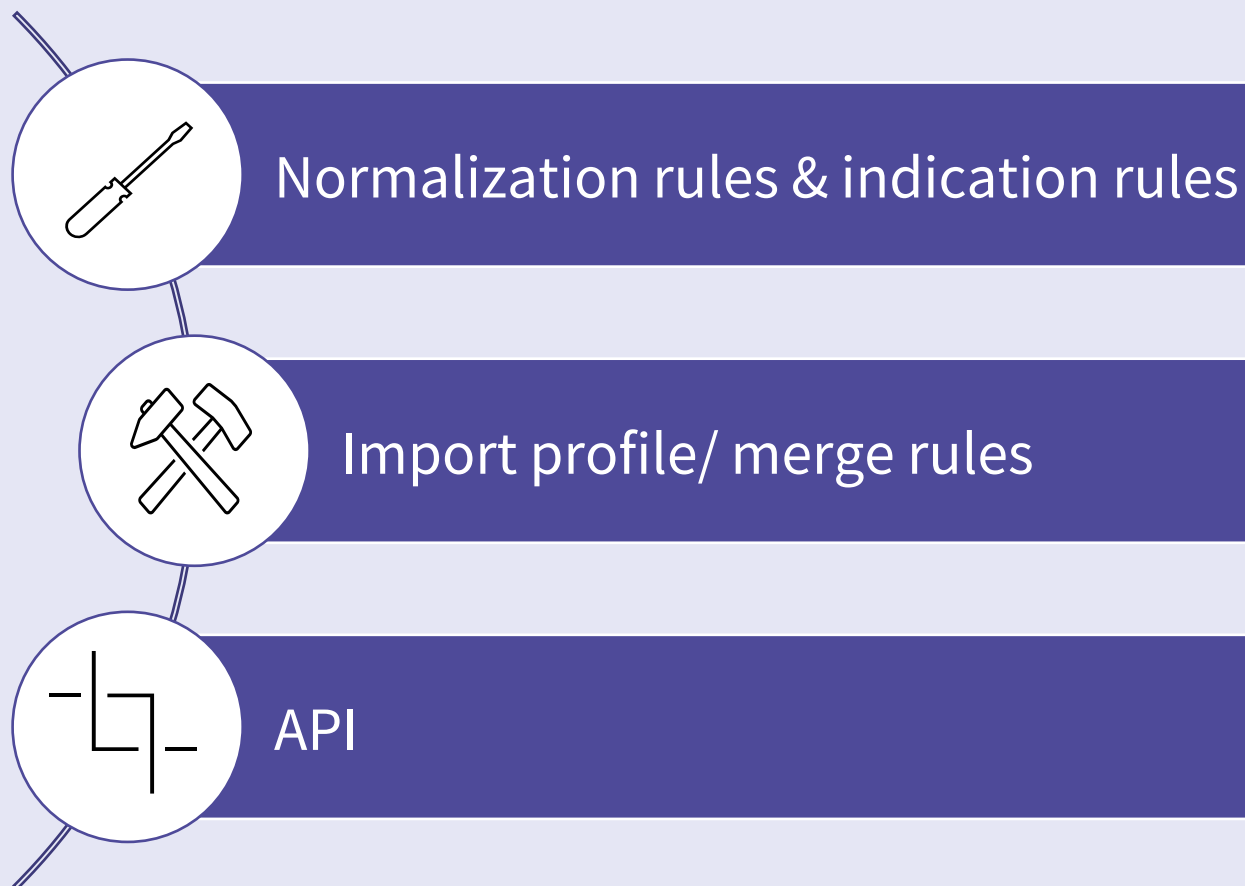
Normalization rules (and indication rules)

Chances

- Useful for simple, rule-based changes
- System functionality / native Alma
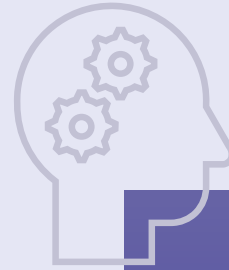- Easy to apply on sets
- Very quick

Limitations

- Limited amount of commands/replacements
- No easy mapping with normalization rules
- Sometimes unexpected behaviour
- Limited logging possiblities

Changes with Import Profile / merge rules
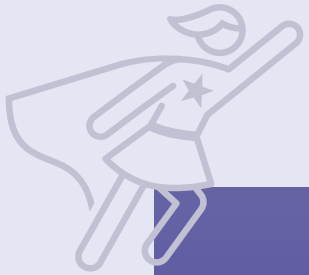
Chances

- Ideal for straightforward additions, e.g. inserting IDs in field 035 or applying local extensions

Limitations

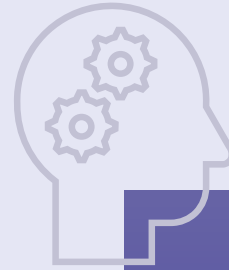- Creation of XML file is necessary
- Testing is required
- Limited file size, larger amount of bib records must be done in batches

Changes via API

Chances

- Ideal for applying record-specific updates through a Python script
- accurate
- Custom and robust logging
- For complex use cases

Limitations

- Working with large amounts of data takes considerable time
- Code or scripting is required

2.b External Database

Content:

- Current and older versions NZ records

- No IZ records, no authorities, no inventory

System: MongoDB®

Purpose:

- better analysis possibilities

- Data backup

- Deduplication

Access (SLSP):

Compass (UI) or Python

2.b External Database

Documents 18.6M   Aggregations   Schema   Indexes

{"marc.830.sub.w":"(IDSBB)000590005DSV01"}

ADD DATA ▾   EXPORT DATA ▾   UPDATE   DEL

_id: ObjectId('6685a16d71fa47eede4cee76')
mms_id : "991059787829705501"
▸ marc : Object
p_date : 2024-10-25T05:16:06.000+00:00
sup : false
c_date : 2020-11-09T06:26:44.000+00:00
u_date : 2024-10-24T14:27:44.000+00:00
format : "Book"
access : "P"

_id: ObjectId('6685a2f171fa47eede544edd')
mms_id : "991044927589705501"
▸ marc : Object
p_date : 2025-04-04T04:53:50.000+00:00
sup : false
c_date : 2020-11-09T10:36:59.000+00:00
u_date : 2025-03-29T05:27:23.000+00:00
format : "Book"
access : "P"

_id: ObjectId('6685a00d71fa47eede45c8a1')
mms_id : "991171018185705501"
▾ marc : Object
  ▾ 245 : Array (1)
    ▾ 0: Object
      ind1 : "0"
      ind2 : "0"
      ▾ sub : Array (3)
        ▾ 0: Object
          a : "<<The>> anthropology of power, agency, and morality"
        ▾ 1: Object
          b : "the enduring legacy of F.G. Bailey"
        ▾ 2: Object
          c : "edited by Victor C. de Munck and Elisa J. Sobo"
  ▾ 264 : Array (1)
    ▾ 0: Object
      ind1 : " "
      ind2 : "1"
      ▾ sub : Array (3)
        ▾ 0: Object
          a : "Manchester"
        ▾ 1: Object
          b : "Manchester University Press"
        ▾ 2: Object
          c : "2022"

https://www.mongodb.com/products/tools/compass

2.c Alma API Wrapper

Wide Alma API coverage

Request builder

API keys management

Errors & logs handling

Open source

Actively maintained

🏠 **Almapi Wrapper**
Available on Read the Docs

https://almapi-wrapper.readthedocs.io/en/latest

2.c Alma API Wrapper

```python
# Import libraries
from almapiwrapper.users import User, NewUser, fetch_users, fetch_user_in_all_iz, Fee, Loan
from almapiwrapper.inventory import IzBib, NzBib, Holding, Item, Collection
from almapiwrapper.config import RecSet, ItemizedSet, LogicalSet, NewLogicalSet, NewItemize
from almapiwrapper.config import Reminder, fetch_reminders, Library, fetch_libraries, Locat
from almapiwrapper.record import JsonData, XmlData
from almapiwrapper.analytics import AnalyticsReport
from almapiwrapper.acquisitions import POLine, Vendor, Invoice, fetch_invoices
from almapiwrapper.configlog import config_log
from almapiwrapper import ApiKeys

# Config logs
config_log()

# Get an item from barcode
item = Item(barcode='03124510', zone='HPH', env='S')

# Create a backup of the record
item.save()

# Update the barcode using the property barcode (library and location also can be changed t
item.barcode = '03124510_NEW'

# Update the internal note using the data property
# We use lxml.etree.ElementTree to find the field and change its value
item.data.find('item_data/internal_note_1').text = 'Note for testing'

# Make the update in Alma
item.update()
```

https://almapi-wrapper.readthedocs.io/en/latest/

3. Conclusion

Questions?

Thank you for your attention!

nikol.stepan@slsp.ch ;
raphael.rey@slsp.ch