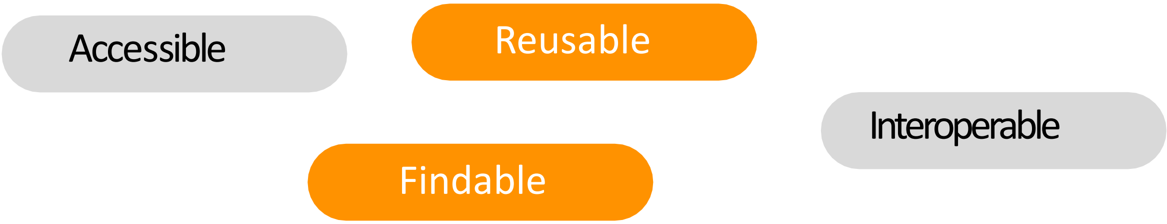
Identifiers

Science Data

- Data generation is getting easier/cheaper
- Complexity-shift from data generation to data processing & analysis
- Amount of data output is increasing, quality is getting better
  - How to stimulate reuse and enable reproducibility?
- Data needs to be:



Accessible

Reusable

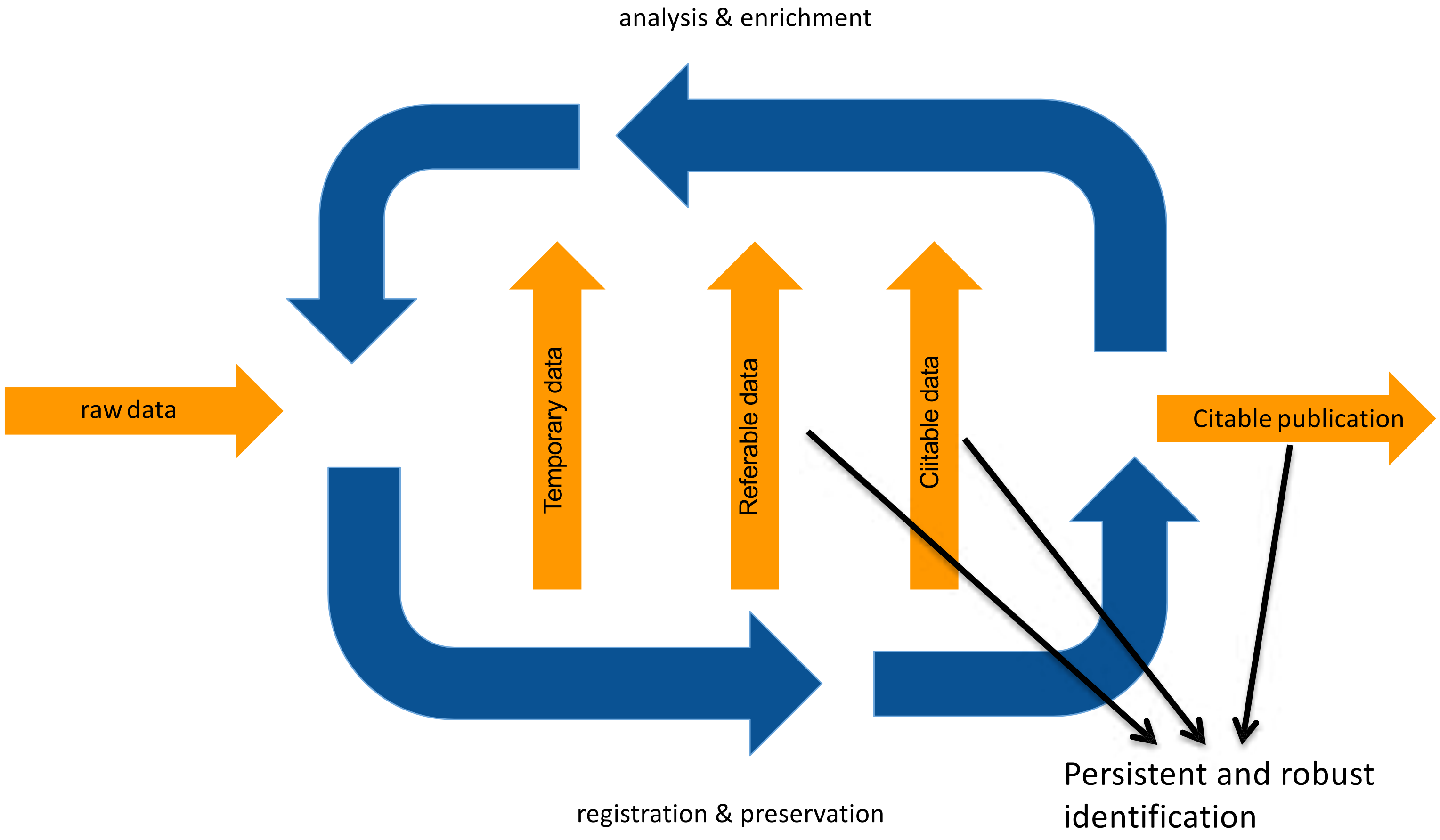
Interoperable

Findable

What are Permanent IDs (PIDS)?

- Pointers to data resources
  - Data files, metadata files, documents …
- Globally unique
- With infinite lifespan
- Can be used to identify and retrieve resources
- Can be resolved to the resource
- Examples: ISBN, DOIs, PURLs, Handles…

Data Creation Cycle



What is the Problem? Why not use simple URLs?

The URL specifies the location, on a particular server, from which the resource could be retrieved. Strictly network locations for digital resources.

BUT

"link rot"

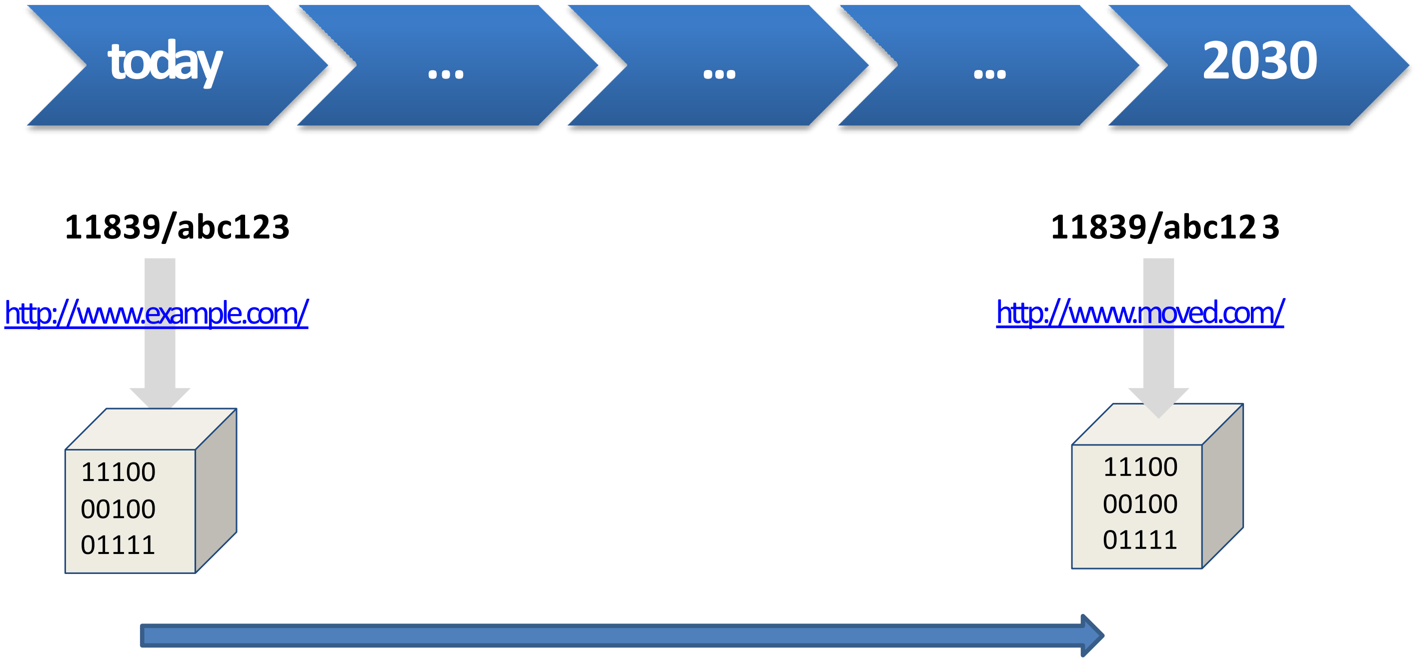
- domain may change
- resource may be relocated
- link may change

In the longterm

URLs a year later, often no longer work

Persistent over time

.. by design



today

...

...

...

2030

11839/abc123

http://www.example.com/

11100
00100
01111

11839/abc123

http://www.moved.com/

11100
00100
01111

Supports access to resource as it moves from one location to another.

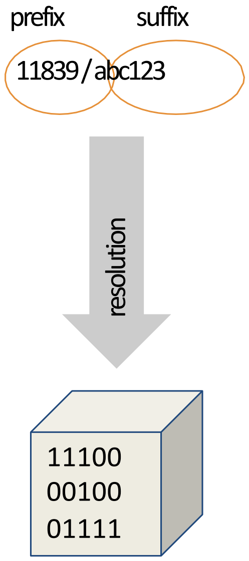
Why can Persistent Identifiers help?

- A Persistent Identifier is
- distinct from a URL
- not strictly bound to a specific server or filename

> *"A persistent identifier (PID) is a long-lasting reference to a digital object—a single file or set of files."*

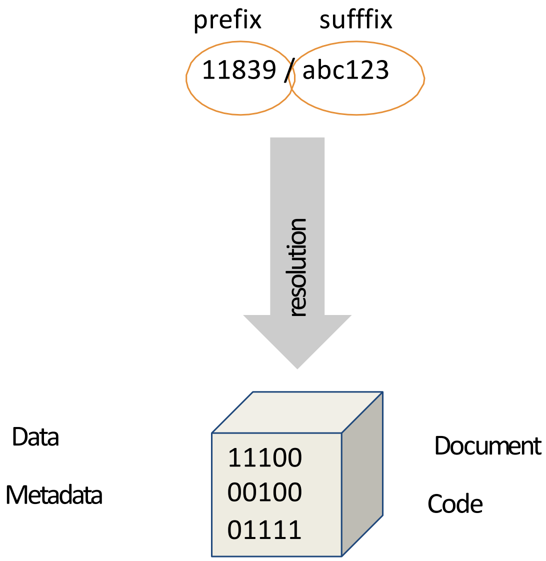
https://en.wikipedia.org/wiki/Persistent_identifier

- Identifier **points to a resource** with no actual knowledge of the resource
- **Responsibility** of the PID owner to keep it up-to-date when the resource changes



Structure of a Persistent Identifier

points to a resource



prefix sufffix

11839 / abc123

resolution

11100
00100
01111

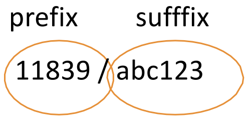
Data

Metadata

Document

Code

Is globally unique



prefix sufffix

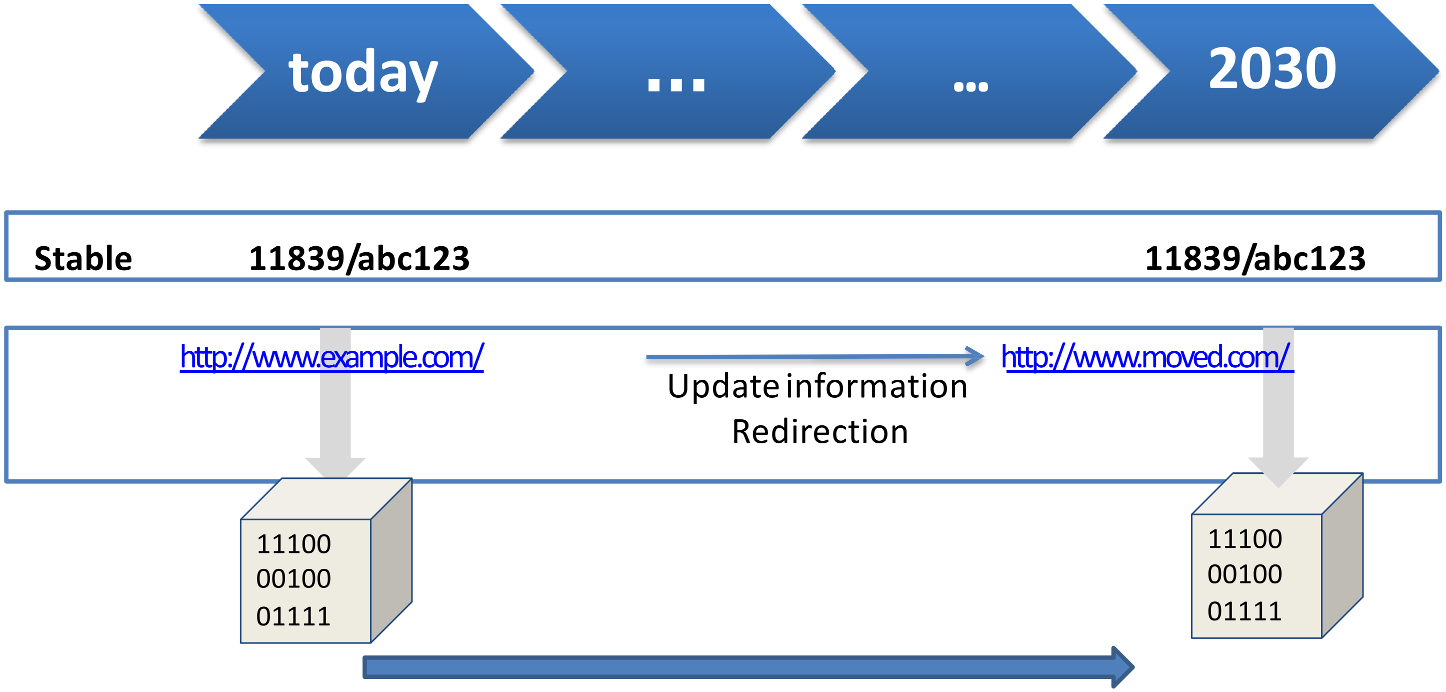
11839 / abc123

Prefix: designates administrative domain, comes from an issuing instance
Suffix: unique in the realm of the prefix

Once the PID is created, the resource is globally addressable.

Persistent over time

.. by design



today

...

...

2030

Stable 11839/abc123 11839/abc123

http://www.example.com/

Update information

Redirection

http://www.moved.com/

11100
00100
01111

11100
00100
01111

PID Benefits

- Persistent Identity via Indirection
  - **Static references into fluid systems over time** Data on networks moves Ownership/responsibility change
  - Formats change
- **Embedded IDs**
  - For data object in hand – current state data
    - Updates
    - New related entities
  - **Networks of Persistent Links** Data / metadata links Provenance chains

PID Costs

- Extra level of effort / cost on creation
  - Analysis – what to identify (granularity)
    - Folders, files
  - Single measurements in a time series experiment
  - Coordination across organizations
  - Maintain resolution system
  - Persistence requires sustained effort
    - Organizational discipline
    - Technology necessary but not sufficient
- Analyze cost/benefit ratio
  - Don’t start unless it is worthwhile
  - Is your data worth it?

Persistent Identifier structure

- Every persistent identifier consists of two parts: its prefix and a unique local name under the prefix known as its suffix
  - Prefix - designates administrative domain, is generated by an issuer, which makes sure that all prefixes are unique
  - Suffix - local name must be unique under its prefix.
- The uniqueness of a prefix and the local name under that prefix ensure that any identifier is globally unique within the context of the System.

< PREFIX > / < SUFFIX > (e.g. 11111/123456745)

PID Systems

a

Persistent URLs (PURLs)

purl: GPO/gpo46189

Cost: no

Metadata: No additional metadata

c

Archival Resource Key (ARK)

ark: /12025/654xz321

Cost: no

Metadata: ERC (Electronic Resource Citation) metadata

Based on: Handle System

b

EPIC System

hdl:11210/123

Cost: $50 annual fee per prefix

Metadata: Associate any metadata

d

Digital Object Identifier (DOI)

DOI: 10.1000/182

Cost: fee per DOI +annual fee

Metadata: The INDECS schema, stored in separate database

PID system Requirements

- Attach multiple URLs to a PID
- Allow part identifiers for complex objects. Granularity issue
- Allow attaching of extra metadata to the PID (MD5 check, etc)
- Actionable (i.e. converted to URL) PIDs
- HTTP proxy for resolving (use port 80 only)
- Controlled by community
- Programmable interface for administration of PIDs from applications
- Delegation of PID administration to other organizations
- Distributed, robust, highly-available, scalable
- No single-point of failure, distributed system with mirroring
- Acceptable non-commercial business model

Identifier String Requirements

- Not based on any changeable attributes of the entity, e.g.:
  - Location, Ownership
  - Any other attribute that may change without changing identity
- Unique
- Avoid conflicts and referential uncertainty
- A good PID system should not allow you to use the same suffix twice

- Opaque, preferably a"dumb number"
  - A well known pattern invites assumptions that may be misleading
  - Meaningful semantics invite IP wars, language problems
  - Nice to have Human-readable Cut-able, paste-able
  - Fits common systems, e.g.
  - URI specification

Persistent URLs (PURLS)

- PURLs **are** URLs.
- A PURL has three parts:
  - (1) a ***protocol***,
  - (2) a ***resolver address***, and
  - (3) a ***name***. The following PURL examples use the same access protocol (*http*) to connect to the same PURL Resolver (*purl.oclc.org*) to resolve *different* names:

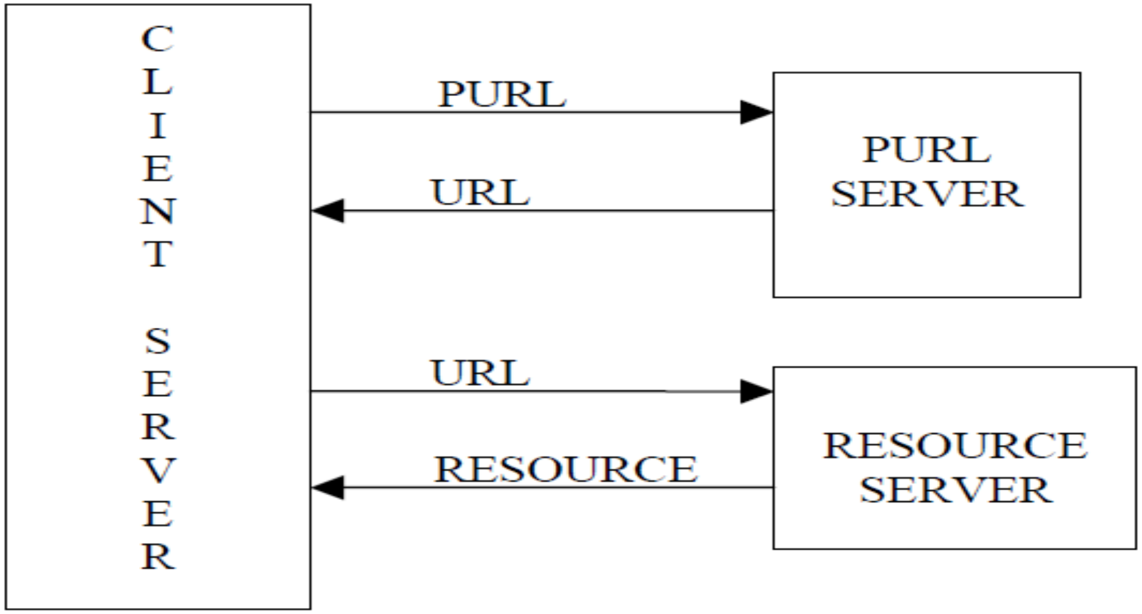
Example PURLs

- protocol resolver address name
- http://purl.oclc.org/keith/home
  http://purl.oclc.org/OCLC/PURL/FAQ
  http://purl.oclc.org/OCLC/OLUC/32127398/1

PURL Domains

- Top-level domains, as their name implies,
  - occupy the top-level of the name space on a PURL Resolver.
  - Requires manual intervention
- Subdomains exist within top-level domains or other subdomains
  - Users create subdomains in any domain for which they can write
- The PURL <URL:http://purl.fake.com/A/B/C/document>
  - has three domains, *A*, *B*, and *C*. *A* is a top-level domain, *B* is a subdomain of *A* and *C* is a subdomain of *B*.

How PURLS Work



Digital Object Idenifiers

- DOI = Digital Object Identifier (system) ®
- International DOI Foundation ("IDF")
  - Common operations and governing organisation: www.doi.org
- RAs = DOI Registration Agencies
  - members of IDF offering the DOI system
  - to customers who wish to assign DOIs
  - to offer a DOI-based service to users

Status

- Foundation launched to develop system in 1998.
- An ISO standard: **ISO 26324**
- Currently used by c. 11,000 naming authorities (assigners)
- e.g. 3,000 STM publishers, science data sets, entertainment industry, EU documents, etc.
- 87 million DOIs assigned to date
- Via 9+ RAs (international)
- DOI services provided by RAs: build on DOI system
- Initial applications mainly are simple redirection to a URL.
- More sophisticated functionality available e.g. multiple resolution

Scope

- Digital Identifier of an Object
- Object = any entity (thing: physical, digital, or abstract)
- Resources, parties, licences, etc.
- Initial focus was documents/media e.g. articles, data sets.
- Now also moving into parties and licences.
- Extending to other sectors
- Digital Identifier = network actionable identifier (“click on it and do something”)
- Extensible by design: not intended as a publishing-only solution (digital convergence)
- Work with existing tools and data
- International: RAs worldwide.

What it Does

- provides a resolvable, persistent, interoperable link:
- resolvable – standard identifier syntax + network resolution mechanism (Handle System)
- persistent – through:
  - ***technical infrastructure*** (registry database, proxy support, etc)
  - ***social infrastructure*** (obligations by Registration Agencies)
- interoperable - through a data model (semantic interoperability)

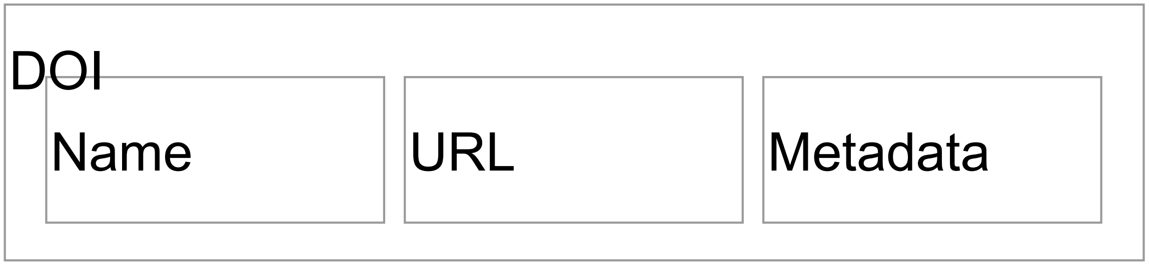
Technical Infrastructure

- Handle system: persistent identification in digital networks
- Data model: principles for interoperability of data in e-commerce systems

- Both used elsewhere: aim was to not re-invent the wheel
- Handle: *www.handle.net*
- Data Model: indecs. Linked Content Coalition

DOI: Digital Object Identifier

- A DOI is a serial code used to **uniquely identify** content of various types of entities. The DOI system is particularly used for electronic documents such as journal articles or datasets.
- What is digital is the identifier, not the object!



DOI

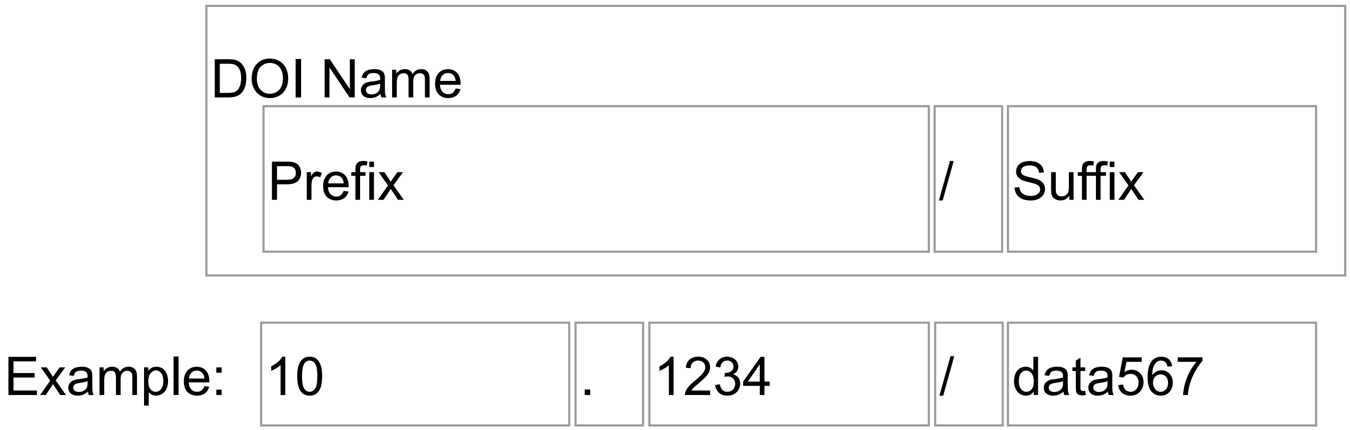
Name

URL

Metadata

DOI Names

- Prefixes are assigned to different services. Each one of them manages the ‘suffix’ namespace freely.



DOI Name

Prefix / Suffix

Example: 10 . 1234 / data567

DOI Resolution

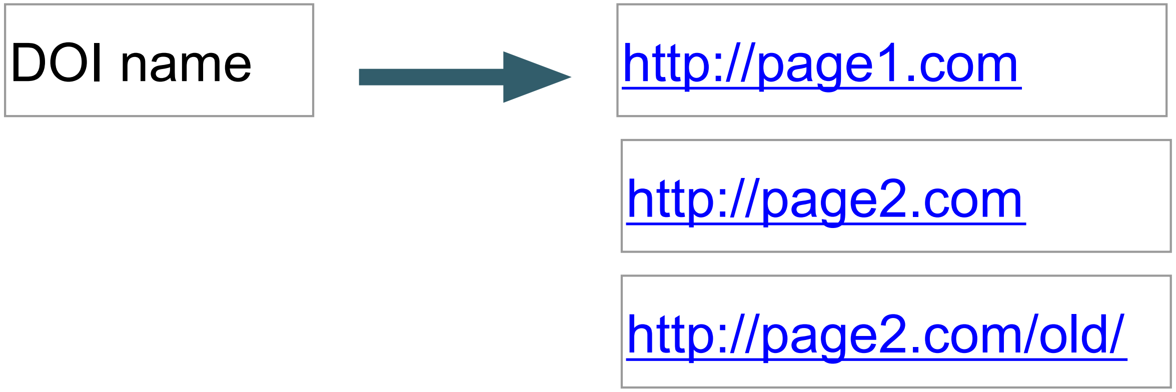
- Ensures persistence by resolving the DOI to associated value such as URL
- Resolution may be:
  - Simple resolution
  - Multiple resolution

The Handle System

- The Handle System is a technology specification for assigning, managing, and resolving persistent identifiers for digital objects and other resources.
- The protocols specified enable a distributed computer system to store identifiers (names, known as Handles) of digital resources and resolve those Handles to the information necessary to locate, access, and otherwise make use of the resources.
- That information can be changed as needed to reflect the current state or location of the identified resource without changing the Handle.

URL

- The DOI name points to a URL, and it can be repointed as many times as needed.
- This URL can be the object itself or a landing page displaying metadata and how to access the object.



DOI name

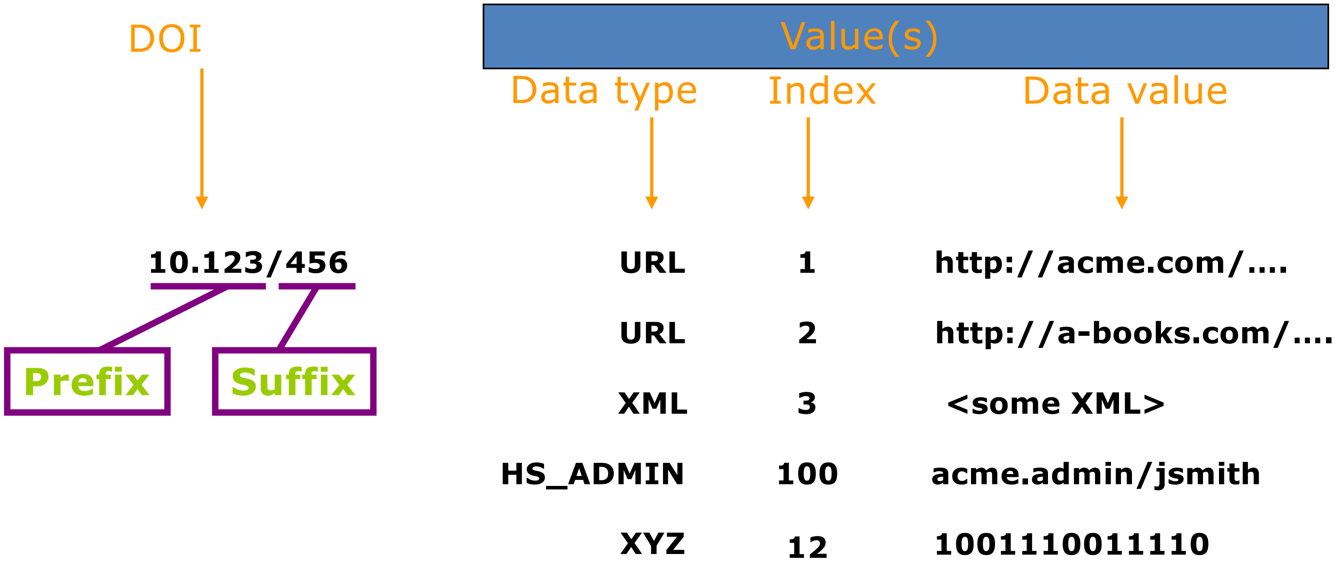
http://page1.com

http://page2.com

http://page2.com/old/

DOI Uses Handle to Resolve to Data

Schematic of a DOI Handle record



DOI

10.123/456

Prefix

Suffix

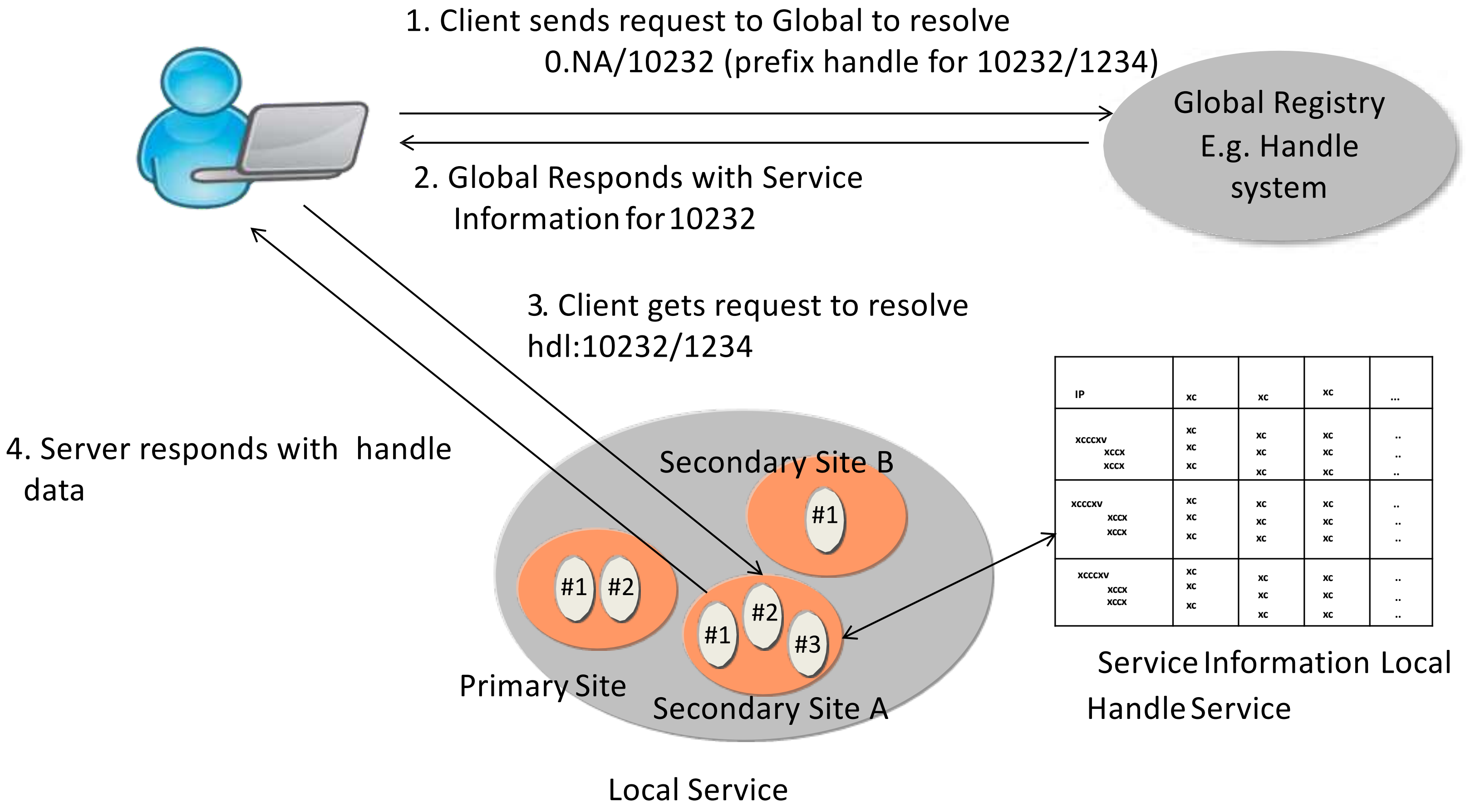
Value(s)

Data type	Index	Data value
URL	1	http://acme.com/….
URL	2	http://a-books.com/….
XML	3	<some XML>
HS_ADMIN	100	acme.admin/jsmith
XYZ	12	1001110011110

Each DOI has one or more Handle values (type:value)

Resolution can return all values, or all values of one type

Resolving Handle Record



1. Client sends request to Global to resolve
0.NA/10232 (prefix handle for 10232/1234)

Global Registry
E.g. Handle
system

2. Global Responds with Service
Information for 10232

3. Client gets request to resolve
hdl:10232/1234

4. Server responds with handle
data

Secondary Site B

Primary Site

Secondary Site A

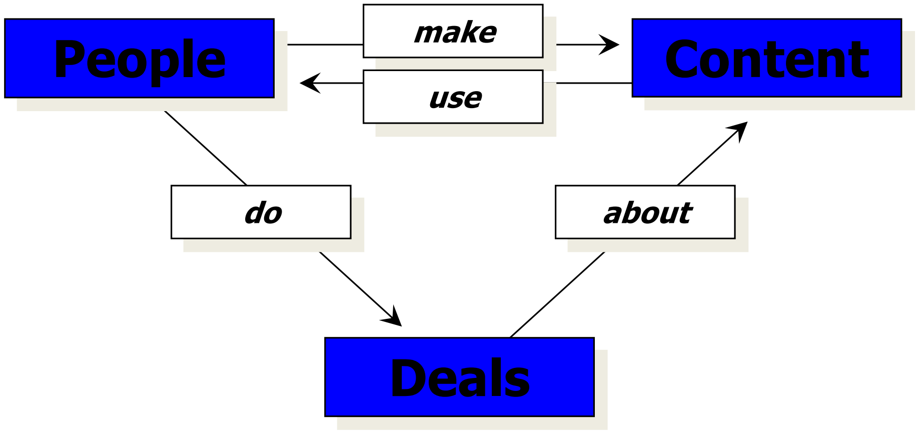
Local Service

IP	xc	xc	xc	...
xcccxv xccx xccx	xc xc xc	xc xc xc	xc xc xc	.. .. ..
xcccxv xccx xccx	xc xc xc	xc xc xc	xc xc xc	.. .. ..
xcccxv xccx xccx	xc xc xc	xc xc xc	xc xc xc	.. .. ..

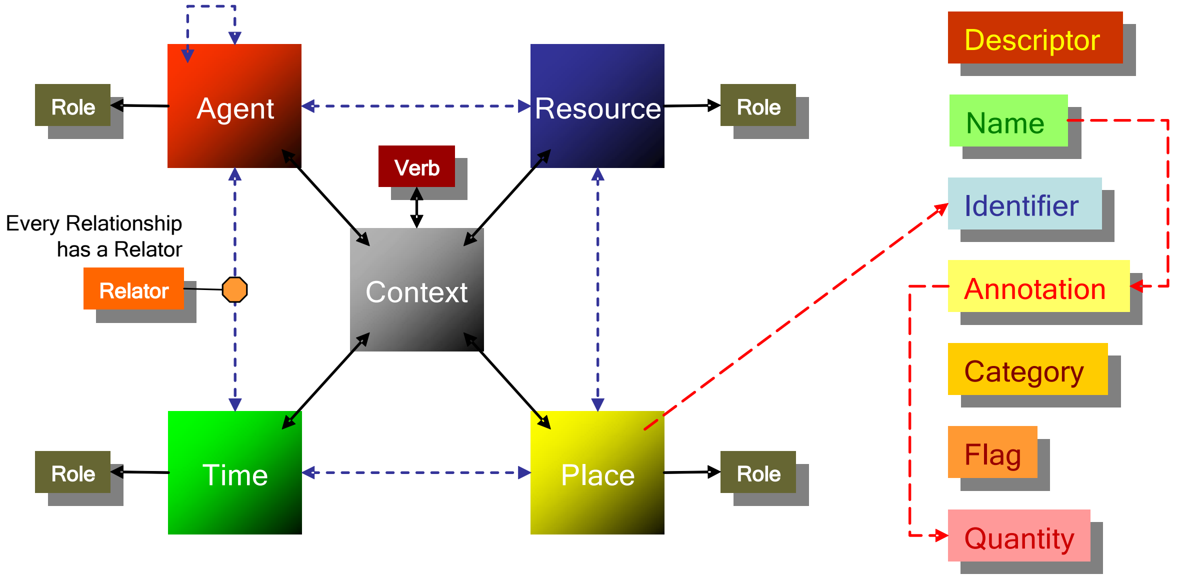
Service Information Local
Handle Service

Data Model: High Level

- High level model (from indecs)



Data Model: More Detail



EntityTypes
An Entity may have typed relationships
with Entities of any kind
(including those of its own kind)

AttributeTypes
An Entity may have Attributes of any kind.
(Attributes, which are a type of Resource,
may have their own Attributes).

⟷ Contextual Relationships

◂- - - - ▸ Non Contextual Relationships (illustrative: any Type of Entity may relate to any other)

– – – – ▸ Attributes (illustrative: any Entity or Attribute may have Attributes of any type)

Data Model – End Result

- Each DOI has some basic metadata
  - All DOIs have this “kernel”
- Metadata is held and managed by the RA
  - Common model for DOI System
- More metadata can be added
  - Appropriate to an RA or DOI service
  - Some groups of DOIs will have the same metadata terms
- Extensible to any level needed
- Can use existing metadata and map it to DOI
- DOIs with the same service or same metadata can be grouped and managed as a class

Metadata

- A metadata schema is a list of core metadata properties chosen for the accurate and consistent identification of a resource.

Mandatory	**Recommended**	**Optional**
Identifier	Subject	Language
Creator	Contributor	Alternate ID
Title	Date	Size
Publisher	Related identifier	Format
Publication year	Description	Version
Resource Type	GeoLocation	Rights

https://schema.datacite.org

Current version 4.0
XML examples available

Social Infrastructure

- Shared development – e.g. APIs, etc
- Shared tools e.g. running mirror servers
- Obligations for persistence:
  - To customers (within the RA)
  - In event of failure, etc.(beyond the RA)
- Collaborate
- Enable shared DOI services where practical
  - A customer could use more than one DOI service