

Lecture 20 – Identifiers

Open Data Management & the Cloud

(Data Science & Scientific Computing / UniTS – DMG)

Goal of this lecture



- **What are identifiers**
 - Definition
 - Uniqueness
 - Persistence

- **Why we use identifiers**
 - Identify objects (mainly digital objects)

What is an identifier?



- Definitions:

an **identifier** may be a word, number, letter, symbol, or any combination of those that identifies (i.e. labels the identity of) either a unique object or a unique class of objects

identifiers (IDs) are symbols having the function to identify an entity or a data-set. The concept of identifier and its function are analogous with the name of a person.

- When an identifier (word, number, letter, symbol, or any combination of those) follows an encoding system, it is often referred to as a **code** or **ID code**.



Identifier definition in different contexts



Definition in different contexts:

- In computer programming languages, an identifier is lexical token that names the language's entities.
- In IBM Integrated Analytics System (a high-performance, integrated data warehouse appliance):

an identifier is a token that is used to form a name. An identifier in an SQL statement is either an SQL identifier or a host identifier.

- MAC address (Media Access Control address) is a unique identifier assigned to a Network Interface Controller (NIC)
- IMEI (International Mobile Equipment Identity) number is a unique identification or serial number that all mobile phones and smartphones have.



Example “In computer programming languages” - python

Define a variable



A Python variable is a reserved memory location to store values. The variable must be defined assigning it a value:

```
>>> a=3 #works
```

```
>>> b = 3 #works
```

```
>>> a
```

```
3
```

```
>>> c # does not work
```

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

NameError: name 'c' is not defined

Rules for Python variable names: A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume):

- a variable name must start with a letter or the underscore character;
- a variable name cannot start with a number;
- a variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _);
- white spaces and signs with special meanings, as "+" and "-" are not allowed;
- variable names are case-sensitive (age, Age and AGE are three different variables).

Define a variable



<https://www.tutorialspoint.com/what-is-an-identifier-in-c-language>

Rules for naming identifiers

The rules that must be followed while naming the identifiers are as follows –

The case of alphabetic characters is significant. For example, using "TUTORIAL" for a variable is not the same as using "tutorial" and neither of them is the same as using "TutoRial" for a variable. All three refer to different variables.

There is no rule on how long an identifier can be. We can run into problems in some compilers, if an identifier is longer than 31 characters. This is different for the different compilers.

A valid identifier can have letters (both uppercase and lowercase letters), digits and underscores.

The first letter of an identifier should be either a letter or an underscore.

You cannot use keywords like int, while etc. as identifiers.

Identifiers must be unique

- **Unique identifier:**

A unique identifier (UID) is an identifier that is guaranteed to be unique among all identifiers used for those objects and for a specific purpose

- **Persistent identifier:**

A persistent identifier (PI or PID) is a long-lasting reference to a document, file, web page, or other digital object.

Most PIDs have a unique identifier which is linked to the current address of the metadata or content. Unlike URLs, PIDs are often provided by services that allow you to update the location of the object so that the identifier consistently points to the right place without breaking (paper DOI and Zenodo examples).

Uniqueness/Persistence



- **Java example**

```
public class Main {  
    public static void main(String[] args) {  
        int myNum = 15;  
        myNum = 20; // myNum is now 20  
        System.out.println(myNum);  
        String myNum = "venti";  
    }  
}
```

Try in <https://www.w3schools.com/java/tryjava.asp>

- **Python example:**

```
x = 10  
  
x = "pippo"
```

Try in <https://www.w3schools.com/python/trypython.asp>

Uniqueness/Persistence



- **tinyurl.com**

is a service where I can create as short URL
corresponding to a much longer one

At the end I obtain two URLs pointing to the same object.

- It is an example of not unique identifiers

- The main developer of the ARK (Archival Resource Keys) system, John Kunze, has suggested that persistence simply means that
 - “an identifier is valid for **long enough**”
- Better reformulation:
 - persistent identifiers should only be assigned to resources that will be preserved for long term
 - that is, over several hardware and software generations
 - a persistent identifier and the services it provides should be at least as persistent as the resource identified
 - The resource may undergo several migrations and the outdated versions may no longer be accessible and/or usable
 - A user who has a persistent identifier of an old manifestation of a resource should be redirected to the latest version available, or to work level metadata, which may enable acquisition of the work in some other form, such as print

Archival Resource Key (ARK) Introduction



Archival Resource Key (ARK) identifiers are persistent URLs designed to support long-term access to information objects. Introduced in 2001, ARK identifiers were designed to identify objects of any type:

- digital objects – documents, databases, images, software, websites, etc.
- physical objects – books, bones, statues, etc.
- living beings and groups – people, animals, companies, orchestras, etc.
- intangible objects – places, chemicals, diseases, vocabulary terms, performances, etc.

It is widely used by libraries, data centers, archives, museums, publishers, and government agencies to provide reliable references to scholarly, scientific, and cultural objects. In 2019 it was registered as a Uniform Resource Identifier (URI).

Archival Resource Key (ARK) Introduction



A URL that is an ARK is distinguished by the label ark: after the URL's hostname, which sets the expectation that, when submitted to a web browser,

the URL terminated by '?' returns a brief metadata record, and

the URL terminated by '??' returns metadata that includes a commitment statement from the current service provider.

The ARK and its inflections ('?' and '??') provide access to three facets of a provider's ability to provide persistence.

Examples:

<https://n2t.net/ark:/67531/metadc107835/>

<https://n2t.net/ark:/67531/metadc107835/?>

<https://n2t.net/ark:/67531/metadc107835/??>

Archival Resource Key (ARK) Introduction



Implicit in the design of the ARK scheme is that persistence is purely a matter of service and not a property of a naming syntax. Moreover, that a "persistent identifier" cannot be born persistent, but an identifier from any scheme may only be proved persistent over time. The inflections provide information with which to judge an identifier's likelihood of persistence.

ARKs can be maintained and resolved locally using open source software

Uniqueness/Persistence



Passport: AA00000000 ← numero del documento
Italian Tax Card (Carta d'Identità): CODICE FISCALE: RSS MRA 70A41 F205Z; COGNOME: ROSSI; NOME: MARIA; LUOGO DI NASCITA: MILANO; DATA DI NASCITA: 01/01/1970
Social Security Card: 000-00-0000; JOHN DOE

Idem garr aai
RESETTA CREDENZIALI PER ACCEDERE AI SERVIZI IDEM
 This form will allow you to reset your password. A mail will be generated and sent to the email address on file for you.
 User ID:

 If you know your current password and want to change it, please visit the [password change](#) page.



Falls Sie diese Reiseinformation nicht oder nur teilweise lesen können, öffnen Sie bitte die angehängte PDF-Version.

Dieses ist eine automatisch erzeugte e-Mail. Bitte antworten Sie nicht

hierauf. **Lufthansa**

Buchungscode:
8RS1TB

Lufthansa Service Center

Persistent Identifiers in the Digital Era



- Actionability (resolve-ability)
 - In connection to digital objects and the internet
 - URLs → URIs → PIDs
 - URL: Uniform Resource Locator
 - URI: Universal Resource Identifier
 - PID: Persistent Identifier
 - ISBN “ISBN 951-45-9942-X” does not resolve to a work
 - But if it is available a resolver for it, like
<http://urn.fi/URN:ISBN:951-45-9942-X>
 - it does resolve to a digital object

Article Talk

Read View source View history

Search Wikipedia

International Standard Book Number

From Wikipedia, the free encyclopedia

Not to be confused with ICCU SBN.

The **International Standard Book Number** (**ISBN**) is a numeric commercial **book identifier** which is intended to be unique.^{[a][b]} Publishers purchase ISBNs from an affiliate of the International ISBN Agency.^[1]

An ISBN is assigned to each separate edition and variation (except reprintings) of a publication. For example, an **e-book**, a **paperback** and a **hardcover** edition of the same book will each have a different ISBN. The ISBN is ten digits long if assigned before 2007, and thirteen digits long if assigned on or after 1 January 2007.^[c] The method of assigning an ISBN is nation-specific and varies between countries, often depending on how large the publishing industry is within a country.

The initial ISBN identification format was devised in 1967, based upon the 9-digit **Standard Book Numbering** (**SBN**) created in 1966. The 10-digit ISBN format was developed by the **International Organization for Standardization** (ISO) and was published in 1970 as international standard ISO 2108 (the 9-digit SBN code can be converted to a 10-digit ISBN by prefixing it with a zero digit '0').

Privately published books sometimes appear without an ISBN. The International ISBN Agency sometimes assigns such books ISBNs on its own initiative.^[3]

Another identifier, the **International Standard Serial Number** (ISSN), identifies periodical publications such as **magazines** and **newspapers**. The **International Standard Music Number** (ISMN) covers **musical scores**.

Contents [hide]

- 1 History
- 2 Overview
 - 2.1 How ISBNs are issued
 - 2.2 Registration group element
 - 2.3 Registrant element
 - 2.3.1 Pattern for English language ISBNs
- 3 Check digits
 - 3.1 ISBN-10 check digits

International Standard Book Number

ISBN 978-3-16-148410-0



A 13-digit ISBN, 978-3-16-148410-0, as represented by an **EAN-13 bar code**

Acronym	ISBN
Organisation	International ISBN Agency
Introduced	1970; 51 years ago
No. of digits	13 (formerly 10)
Check digit	Weighted sum
Example	978-3-16-148410-0
Website	isbn-international.org

- Permalink: A permalink or **permanent link** is a type of persistent identifier, it is a URL that is intended to remain unchanged for many years into the future, yielding a hyperlink that is less susceptible to link rot.

Permalinks are often rendered simply, that is, as clean URLs, to be easier to type and remember.

Sometimes URL shortening is used to create them.

- Persistence requires management of resources identified
 - URIs can be used as persistent identifiers if they are properly managed
 - Domain Name Resolver / DNS

Persistent Identifiers & Services



- What an identifier resolves to
 - An object
 - A landing page including
 - Metadata
 - Links to actual resource
- To be persistent requires
 - Management
 - Translation services
 - Protocol?
 - Will HTTP be always there?
 - schema/prefix
- It mainly is an organizational matter

Examples of Identifiers



- URN: Uniform Resource Name
- Handle(.net) System
- DOI: Digital Object Identifier
- ARK: Archival Resource Key
- PURL: Persistent URL (Uniform Resource Locator)
- URI: Uniform Resource Identifier
- UUID: Universally Unique ID
- ORCID: Open Research Contributor ID
- ADS bibcode: Astrophysics Data System bibliographic code
- IVOID: IVOA ID

Uniform Resource Name



- A Uniform Resource Name (URN) is a Uniform Resource Identifier (URI) that uses the urn scheme.
- URN syntax

“urn:”<NID>”:<NSS>

schema: NamespaceIdentifier:NameSpaceSpecific

- where
 - <NID> is a namespace identifier
 - to distinguish between different identifier schemes
 - <NSS> is the namespace-specific string
 - UTF-8 (UCS/2)
- example
 - 'ISBN' is the NID for the ISBN
 - ISBN URN example:
 - URN:ISBN:951-45-9942-X
- Not directly resolvable/actionable
- Can be used inside other PIDs systems: e.g. DOI.
 - <http://dx.doi.org/10.1038/issn.1476-4687>
- Explicit statement for location preservation: proved unreliable

Example - urn:isbn:0451450523

- Introduced in 1994, formalized in 1997 and is now an IETF standard. No central governance, no central resolving infrastructure. Used by major national libraries in Europe. ISBNs for books are part of the URN system.
- No license costs involved for assigning URNs, but a URN registration agency needs to establish an assigning and a resolving infrastructure. The biggest initiative to harmonize URN registration in Europe is currently undertaken by the PersID project.

<https://project-thor.readme.io/docs/project-glossary>

- Handles consist of a
 - prefix which identifies a "naming authority"
 - suffix which gives the "local name" of a resource
- Examples
 - 20.1000/100
 - 2381/12345
 - arXiv:2103.08334
- Prefixes must be registered
- Handle is
 - Opaque: it encodes no information about the underlying resource
 - Provides only the means to retrieve metadata about the resource
 - UCS-2 based
- The Handle System is compatible with the Domain Name System (DNS)
 - but does not require it

Example - hdl:2381/12775

- Non-commercial decentralized identifier resolution system, established in 1995. Operated by CNRI. Used by many other higherlevel systems, e.g. DOI. A non-commercial Handle system that is
- operated by the Corporation for National Research Initiatives (CNRI). Different initiatives use commercial handle licenses to establish local handle system, such as the European Persistent Identifier Consortium (EPIC). Many existing content management systems, including institutional repositories, currently operate their own local handle system.

<https://project-thor.readme.io/docs/project-glossary>

DOI: Digital Object Identifier



- A **digital object identifier** (DOI) is a persistent identifier or handle used to identify objects uniquely, standardized by the International Organization for Standardization (ISO).

DOIs are in wide use mainly to identify academic, professional, and government information, such as journal articles, research reports, data sets, and official publications.

DOIs have also been used, however, to identify other types of information resources, such as commercial videos.

- A DOI aims to be "resolvable", actionable and interoperable
- The DOI for a document remains fixed over the lifetime of the document, whereas its location and other metadata may change. Referring to an online document by its DOI is supposed to provide a more stable link than simply using its URL. But every time a URL changes, the publisher has to update the metadata for the DOI to link to the new URL.
- The developer and administrator of the DOI system is the International DOI Foundation (IDF), which introduced it in 2000.

DOI: Digital Object Identifier



Example - doi:10.1186/2041-1480-3-9

- Combines a metadata model with the Handle system as the resolution infrastructure (i.e. DOIs are handles). First introduced in 1998 with the funding of the International DOI foundation (IDF).
- Became official ISO standard in 2012 (ISO 26324). The DOI system is built upon CNRI (Corporation for National Research Initiatives) Handles. DOI Registration agencies are responsible for assigning identifiers. They each have their own commercial or non-commercial business model for supporting the associated costs. The DOI system itself is maintained and advanced by the IDF, itself controlled by its registration agency members. Using
- the Handle system, there is a central free worldwide resolving mechanism for DOI names. DOI names from any registration agency can be resolved worldwide in every handle server; DOIs therefore are self-sufficient and their resolution does not depend on a single agency. A standard metadata kernel is defined for every DOI name. Assigning DOI names involves the payment of a license fee but their resolution is free.

<https://project-thor.readme.io/docs/project-glossary>

Handle vs DOI



- The handle system provides resolution of identifiers
- The DOI system is an implementation on top of the handle system that maintains a registry of metadata related to the identifiers

IDF: International DOI Foundation



The DOI® System

ISO 26324




This is the web site of the International DOI Foundation (IDF), a not-for-profit [membership organization](#) that is the governance and management body for the [federation of Registration Agencies](#) providing Digital Object Identifier (DOI) services and registration, and is the registration authority for the ISO standard (ISO 26324) for the DOI system. The DOI system provides a technical and social infrastructure for the registration and use of persistent interoperable identifiers, called DOIs, for use on digital networks.

Resolve a DOI Name

Type or paste a DOI name, e.g., 10.1000/xyz123, into the text box below. (Be sure to enter all of the characters before and after the slash. Do not include extra characters, or sentence punctuation marks.)

Clicking on a DOI link (try this one: <https://doi.org/10.1109/5.771073>) takes you to one or more current URLs or other services related to a single resource. If the URLs or services change over time, e.g., the resource moves, this same DOI will continue to resolve to the correct resources or services at their new locations.

Check the current status of the DOI system at doi.statuspage.io.

DRIVEN BY  Enhance the value of your content.
Join the DOI Community.
[Watch a video, get the facts, and find out how.](#)

Updated May 13, 2021

contact@doi.org

doi®, DOI®, DOI.ORG®, and shortDOI® are trademarks of the International DOI Foundation.

Digital Object Identifiers



- DOI syntax

prefix/suffix

- 10.1002/joc.1130 is a valid DOI
- 10.1002. prefix composed of
 - 10: DOI identifier within the Handle system
 - 1002: identifier of the organization that has assigned the DOI
- joc.1130: suffix which identifies the resource
- DOIs are Handle persistent identifiers
- In practice, DOIs are usually expressed in the Web as hyperlinks:
 - <http://dx.doi.org/10.1002/joc.1130>
 - Directly resolvable PID

Digital Object Identifiers: usage



CV-ESP-Europass-20180829-Bertocco-IT.odt
CV_SaraBertocco.pdf 59,6%

europass Curriculum vitae Bertocco Sara

- Sistemi di storage: OpenStack-Swift (buono), Ceph (discreto), VOSpace (eccellente)
- Piattaforme Cloud: OpenStack (buono)
- Sistemi operativi: Linux CentOS e Ubuntu (eccellente sistemistico e utente). Nel corso della mia carriera ho lavorato anche con Unix Sun e HP-UX e con DOS

ULTERIORI INFORMAZIONI

Publicazioni **Articoli da rivista referati:**

"Cloud access to interoperable IVOA-compliant VOSpace storage"
S. Bertocco, P. Dowler, S. Gaudet, B. Major, F. Pasian G. Taffoni
Astronomy and Computing, Vol. 24, July 2018, pp. 36-44, Elsevier, ISSN: 2213-1337
DOI: <https://doi.org/10.1016/j.asco>

"Design and Implementation of the gLite CREAM Job Management Service"
C. Aiftimiei, P. Andreetto, S. Bertocco, S. Dalla Fina, A. Dorigo, E. Frizziero, A. Gianelle, M. Marzolla, M. Mazzucato, M. Sgaravatto, S. Traldi, L. Zangrando
Future Generation Computer Systems, Vol. 26, Issue 4, April 2010, pp. 654-667, Elsevier, ISSN: 0167-739X
DOI: <https://doi.org/10.1016/j.future.2009.12.006>

Articoli da rivista non referati:

"Torii, an Open Portal over Open Archives"
S. Bertocco
High Energy Physics Libraries Webzine, Giugno 2001 n. 4
URL: <http://library.cern.ch/HEPLW/4/papers/4/>

Proceedings referati

"Performance and Energy Footprint Assessment of FPGAs and GPUs on HPC Systems Using Astrophysics Application"
D. Goz, G. Ieronymakis, V. Papaefstathiou, N. Dimou, S. Bertocco, F. Simula, A. Ragagnin, L. Tornatore, I. Coretti, G. Taffoni
Computation 2020. Special Issue "Energy-Efficient Computing on Parallel Architectures"
DOI: <https://doi.org/10.3390/computation8020034>

"Measuring the energy footprint of Astrophysics HPC simulations and Exascale platforms"

DOI: landing page example



Computation | Free Full-Text | Performance and Energy Footprint Assessment of FPGAs and GPUs on HPC Systems Using Astrophysics Application — Mozilla Firefox

Unifc | Goog | Abilit | Pythc | vars() | What | numj | herb. | herb. | DCM | Main | Co. X

https://www.mdpi.com/2079-3197/8/2/34

Getting Started | Other Bookmarks

MDPI

Search:

Journals / Computation / Volume 8 / Issue 2 / 10.3390/computation8020034

computation

Article Menu

Article Overview

- Abstract
- Open Access and Permissions
- Share and Cite
- Article Metrics
- Order Article Reprints

Article Versions

Related Info Links

Open Access Article

Performance and Energy Footprint Assessment of FPGAs and GPUs on HPC Systems Using Astrophysics Application

by David Goz ^{1,*†}, Georgios Ieronymakis ^{2,*†}, Vassilis Papaefstathiou ², Nikolaos Dimou ², Sara Bertocco ¹, Francesco Simula ³, Antonio Ragagnin ¹, Luca Tornatore ¹, Igor Coretti ¹ and Giuliano Taffoni ¹

¹ INAF-Osservatorio Astronomico di Trieste, 35122 Padova, Italy
² ICS-FORTH, GR-700 13 Heraklion, Crete, Greece
³ INFN-Sezione di Roma, 00185 Rome, Italy
* Authors to whom correspondence should be addressed.
† These authors contributed equally to this work.

Computation **2020**, *8*(2), 34; <https://doi.org/10.3390/computation8020034>

Received: 6 March 2020 / Revised: 8 April 2020 / Accepted: 11 April 2020 / Published: 17 April 2020

(This article belongs to the Special Issue Energy-Efficient Computing on Parallel Architectures)

We use cookies on our website to ensure you get the best experience. [Read more about our cookies here.](#)

DOI: more general usage



- wiki.ivoa.net/internal/IVOA/InterOpOct2019DCP/sara_bertocco_IVOAInterop_Groningen2019.pdf
www.adass2019.nl/wp-content/uploads/adass-pdf/Poster239.pdf
- Resource linking management example:
indico.in2p3.fr/event/22995/contributions/91954/attachments/62596/85887/ESCAPE-WP3-HCG16-v2.pdf

Archival Resource Key



- ARK syntax

[<http://NMAH/>]ark:/NAAN/Name[Qualifier]

- NAAN: Name Assigning Authority Number
 - mandatory unique identifier of the organization that originally named the object
- NMAH: Name Mapping Authority Host
 - optional and replaceable hostname of an organization that currently provides service for the object
- Qualifier: optional string that extends the base ARK to support access
 - to individual hierarchical subcomponents of an object (using a slash)
 - to variants (versions, languages, formats) of components (using dots)
- ARK example
 - <http://example.org/ark:/12025/654xz321/s3/f8.05v.tiff>
- Qualifier is the most prominent part of this PID
- ? and ?? added to retrieve metadata (brief) and preservation statement

Archival Resource Key



Example - ark:/13030/tf5p30086k

- Introduced in 1995. Not a formal standard but all ARKs follow the same structure and workflows. No central resolver - organisations can sign up to become Name Assigning Authority Numbers (NAANs) and run their own resolution infrastructure for ARKs. System is run by the California Digital Library with dozens of NAANs worldwide through a combined ARK/DOI infrastructure EZID. This EZID infrastructure allows interoperability between ARKs and DOI names.

<https://project-thor.readme.io/docs/project-glossary>

https://www.project-freya.eu/en/deliverables/odin/odin_pids4data.pdf

Persistent Uniform Resource Locator



- PURL
 - uniform resource locator (URL, i.e., location-based URI)
 - used to redirect to the location of the requested web resource
 - PURLs redirect HTTP clients using HTTP status codes.
- The PURL concept is generic and can be used to designate any redirection service (named PURL resolver) given
 - a "root URL" as the resolver reference
 - `http://myPurlResolver.example`
 - "names" in the root URL
 - `http://myPurlResolver.example/name22`
 - means to provide
 - to associate each name with its URL to be redirected
 - to update this redirection-URL
 - persistence of the "root URL" and the PURL resolver

Uniform Resource Identifier (1)



- A URI provides a simple and extensible means for identifying a resource
 - Uniform
 - different types of resource identifiers can be used together
 - semantic interpretation of common syntactic conventions
 - introduction of new types of resource identifiers seamlessly
 - Resource
 - widest general sense
 - electronic document, image, source of information, service, collection of other resources
 - not necessarily accessible via the Internet: human beings, corporations, books
 - abstract concepts: operators, operands, relationship types, numeric values, ...
 - Identifier
 - distinguishing one resource from all other resources
 - an identifier does not define or embodies the identity of what is referenced
 - a system using URIs will not necessarily access the resource identified
 - "one" resource identified might not be singular in nature

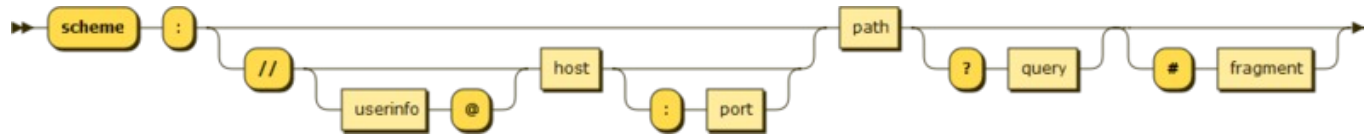
URI (2)



- URI Syntax

URI = scheme:[//authority]path[?query][#fragment]

authority = [userinfo@]host[:port]



```
      userinfo      host      port
      |             |             |
https://john.doe@www.example.com:123/forum/questions/?tag=networking&order=newest#top
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
scheme authority path query fragment

      userinfo      host      port
      |             |             |
ldap://[2001:db8::7]/c=GB?objectClass?one
|-----|-----|-----|-----|
scheme authority path query

mailto:John.Doe@example.com
|-----|-----|
scheme path

news:comp.infosystems.www.servers.unix
|-----|-----|
scheme path

tel:+1-816-555-1212
|-----|-----|
scheme path

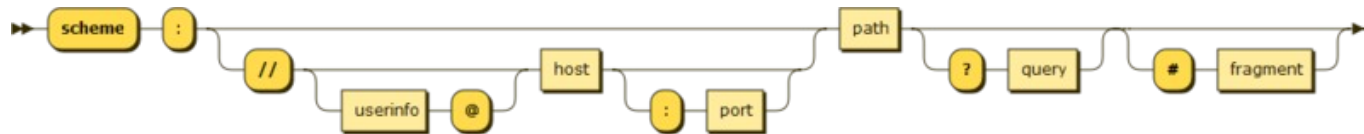
telnet://192.0.2.16:80/
|-----|-----|-----|
scheme authority path

urn:oasis:names:specification:docbook:dtd:xml:4.1.2
|-----|-----|-----|-----|
scheme path
```

URI (3)



- URI embeds
 - URL: explicitly serves a location to access the resource
 - URN: preserves the specification about “named” resources
- Involves identification/interaction distinction
 - Is used for identification
- URI is hierarchical
 - Starting from the “scheme” part down



```
reserved    = gen-delims / sub-delims
gen-delims  = ":" / "/" / "?" / "#" / "[" / "]" / "@"
sub-delims  = "!" / "$" / "&" / "'" / "(" / ")"
             / "*" / "+" / "," / ";" / "="
```

Universally Unique Identifier



- UUID is a 128-bit number used to identify information in computer systems
 - UUIDs are for practical purposes unique
 - uniqueness independent of a central registration authority or coordination
 - The probability that a UUID will be duplicated is not zero, it is close enough to zero to be negligible.
- Canonical textual representation
 - xxxxxxxx-xxxx-Mxxx-Nxxx-xxxxxxxxxxxx
 - the sixteen octets represented as 32 hexadecimal (base 16) digits
 - displayed in five groups separated by hyphens [8-4-4-4-12]
 - Example: 123e4567-e89b-12d3-a456-426655440000
 - M indicate the UUID version, N indicate the UUID variant
 - i.e. a way to get the meaning of the numbers, if needed

Open Research Contributor Identifier



- **ORCID**
 - an https URI with a 16-digit number
 - e.g. <https://orcid.org/0000-0001-2345-6789>
 - ISNI compatible / URN valid
 - Random generated 16 digits
- Identifies a person
 - Uniqueness in author/paper cross-reference
 - Not fail-proof but should really be unique
- Attaches to a record of that person

Open Research Contributor Identifier



- ORCID is an open, non-profit, community-driven effort to create and maintain a registry of unique researcher identifiers and a transparent method of linking research activities and outputs to these identifiers. Identifiers issued by ORCID are known as ORCID IDs.
- ORCID enables a transparent linking mechanism between ORCID and other current researcher ID schemes. These identifiers, and the relationships among them, can be linked to the researcher's output to enhance the scientific discovery process and to improve the efficiency of research funding and collaboration within the research community.
- ORCID provides two core functions: (1) a registry to obtain a unique person identifier and manage a record of activities, and (2) APIs that support system-to-system communication and authentication.

<https://project-thor.readme.io/docs/project-glossary>

- Astrophysics Data System (abstracts service) bibliographic codes
 - YYYYJJJJVVVVMPPPPA – always 19 characters, “.” padded
 - YYYY: year of publication
 - JJJJJ: standard abbreviation for the journal (left padded)
 - ApJ, AJ, MNRAS, Sci, PASP, ... internally controlled
 - VVVV: volume number (for a serial) or a type abbreviation (right padded)
 - conf, meet, book, coll, proc
 - M: Qualifier for publication:
 - E: Electronic Abstract (usually a counter, not a page number)
 - L: Letter
 - P: Pink page
 - Q-Z: Unduplicating character for identical codes
 - PPPP: Page number (right padded)
 - Note that for page numbers greater than 9999, the page number is continued in the m column.
 - A: The first letter of the last name of the first author
 - 1992ApJ...400L...1W
 - Astrophysical Journal Letters volume 400, page L1 (Windhorst).
 - Resolver: <http://adsabs.harvard.edu/abs/1992ApJ...400L...1W>

IVOA Identifier (1)



- An IVOA identifier, or IVOID, is a special sort of URI
 - “ivo” as the scheme
 - “Registry part” including authority and path
 - “local part” including query and fragment
- `ivo://<authority><path>?<query>#<fragment>`
 - Registry part → `ivo://<authority><path>`
 - Registry reference, it must be resolvable in the Registry
 - Resource key → path
 - Authority → `ivo://<authority>`
 - Local part → `?query#fragment`
- IVOID example: `ivo://example.org/svc?voc.xml#Term`

IVOID (2)



- Restrictions to URI specification apply to
 - Authority
 - it MUST be at least three characters long
 - it MUST begin with an alpha-numeric character
 - it MUST NOT contain percent-encoded characters
 - it MUST NOT contain characters outside of <unreserved>, with the tilde strongly discouraged
 - there are no <userinfo> or <port> components
 - Resource key
 - No percent encoding
 - No sub-delimiters, unless explicitly specified by other IVOA REC
- Query & fragments follows URI
 - Including semantic meaning of fragments

ivo://nasa.heasarc/user/STScI_1/1a-7z.u → OK

ivo://a2/data!g-vo.org → KO (but...)

- Resolving IVOIDs
 - It is done through a Registry query (bootstrap: you need the Registry endpoint)
 - Registry references → resolve directly
 - IVOID with query part
 - Resolve the Registry reference (strip from ? on)
 - Identify the service dereferencing the full IVOID
 - Depends on resource context
 - IVOID with fragment
 - Same as IVOID up to the query part
 - Fragment specification depends on the specific resource
 - i.e. it's up to the service/application
- IVOIDs comparison (restriction wrt URI)
 - As no hierarchy is implied in any IVOID part, no path segment normalization
 - As IVOIDs must not percent-encode characters that do not need to be encoded, no percent-encoding normalization is ever performed on IVOIDs
 - the resource key is also compared case-insensitively

- Same rules as IVOID, i.e. URI with ivo:// scheme and restrictions
 - Dataset Identifiers
 - DIDs / PubDIDs / CreatorDIDs
 - Answering the need to reference datasets
 - They are IVOIDs where the query part identifies (locally) the dataset
 - No need to register them
 - No need to have a service resolving them
 - Fragment part can be used to identify sub-parts
 - Standard Identifiers
 - To identify the IVOA endorsed standards
 - And use in annotating resources
 - Fall inside the ivo://ivoa.net authority
 - Have resource key and local part as
 - /std/<standard-ref> "#"<key-name> "-"<version>
 - e.g. ivo://ivoa.net/std/exampleProto#model-1.0

Goals for Identifiers



- Resource/thing identification
 - Scientific work
 - Research data
 - Involved people
 - ...
- Allow reference among identified objects
 - Relationships can be identified too
- Be useful if interaction/resolving is needed
 - Persistence of the identifier
 - Flexibility on the location and protocol
- Allow granularity if needed

RDA Consolidated Assertions on PIDs



- PIDs are increasingly important and are being applied almost everywhere across sectors and disciplines, and for all types of digital objects. (Here, the term "sectors" covers science, industry, governments, health care, etc.)
- Data management experts are becoming increasingly dependent on the availability of functioning persistent identifiers which
 - are uniquely identifying a specific Digital Object
 - in general consist of a name space indicator (prefix) and a local identifier suffix)
 - are actionable on the web, by extending it to a fully defined URI, if required
 - can be persistently resolved to state information and/or a landing page
 - are associated with a persistent resolution system
 - are issued and managed by a clearly specified registration authority

- Data Citation of Evolving Data

- Recommendations of the WG on Data Citation - RDA

Goals of this WG are to create identification mechanisms that:

- allows us to identify and cite arbitrary views of data, from a single record to an entire data set in a precise, machine-actionable manner
- allows us to cite and retrieve that data as it existed at a certain point in time, whether the database is static or highly dynamic
- is stable across different technologies and technological changes

Solution: The WG recommends solving this challenge by:

- ensuring that data is stored in a versioned and timestamped manner.
- identifying data sets by storing and assigning persistent identifiers (PIDs) to timestamped queries that can be re-executed against the timestamped data store.

A. *Preparing the Data and the Query Store*

Prepare existing data sources and provide the required infrastructure, which is needed for implementing the query based approach.

- **R1 – Data Versioning:** Apply versioning to ensure earlier states of data sets can be retrieved.
- **R2 – Timestamping:** Ensure that operations on data are timestamped, i.e. any additions, deletions are marked with a timestamp.
- **R3 – Query Store Facilities:** Provide means for storing queries and the associated metadata in order to re-execute them in the future.

C. *Resolving PIDs and Retrieving the Data*

- **R11 – Landing Page:** Make the PIDs resolve to a human readable landing page that provides the data (via query re-execution) and metadata, including a link to the superset (PID of the data source) and citation text snippet.
- **R12 – Machine Actionability:** Provide an API / machine actionable landing page to access metadata and data via query re-execution.

D. *Upon Modifications to the Data Infrastructure*

- **R13 – Technology Migration:** When data is migrated to a new representation (e.g. new database system, a new schema or a completely different technology), migrate also the queries and associated fixity information.
- **R14 – Migration Verification:** Verify successful data and query migration, ensuring that queries can be re-executed correctly.

B. *Persistently Identify Specific Data Sets*

When a data set should be persisted, the following steps need to be applied:

- **R4 – Query Uniqueness:** Re-write the query to a normalised form so that identical queries can be detected. Compute a checksum of the normalized query to efficiently detect identical queries.
- **R5 – Stable Sorting:** Ensure that the sorting of the records in the data set is unambiguous and reproducible
- **R6 – Result Set Verification:** Compute fixity information (checksum) of the query result set to enable verification of the correctness of a result upon re-execution.
- **R7 – Query Timestamping:** Assign a timestamp to the query based on the last update to the entire database (or the last update to the selection of data affected by the query or the query execution time). This allows retrieving the data as it existed at the time a user issued a query.
- **R8 – Query PID:** Assign a new PID to the query if either the query is new or if the result set returned from an earlier identical query is different due to changes in the data. Otherwise, return the existing PID.
- **R9 – Store Query:** Store query and metadata (e.g. PID, original and normalized query, query & result set checksum, timestamp, superset PID, data set description, and other) in the query store.
- **R10 – Automated Citation Texts:** Generate citation texts in the format prevalent in the designated community for lowering the barrier for citing the data. Include the PID into the citation text snippet.

June 26, 2018

Dataset **Open Access**

VAMDC extraction with identifier = 5c91e7c7-e0c9-474f-b76a-62bff7b38468

VAMDC, Consortium

This is a dataset extracted from <http://vamdc.icb.cnrs.fr/gecasda/tap/> VAMDC node.

Query originating this dataset: `query=select * where (inchikey in ['quzpnffhzprkjd-aklpvkdbsa-n', 'quzpnffhzprkjd-bjudxgmsa-n', 'quzpnffhzprkjd-igmarmgpsa-n', 'quzpnffhzprkjd-oiobtwansa-n', 'quzpnffhzprkjd-oubtvsysa-n']);`

Data source version: 1

Data format: XSAMS 12.07

Query uuid in VAMDC query store: 5c91e7c7-e0c9-474f-b76a-62bff7b38468

1 views 0 downloads

[See more details...](#)

Indexed in

Publication date:
June 26, 2018

DOI:
[DOI 10.5281/zenodo.1620783](https://doi.org/10.5281/zenodo.1620783)

Related identifiers:
Identical to:
<https://cite.vamdc.eu/references.html?uuid=5c91e7c7-e0c9-474f-b76a-62bff7b38468>

License (for files):
[Creative Commons Zero v1.0 Universal](#)

Versions

Version 1	10.5281/zenodo.1620783	Jun 26, 2018
-----------	------------------------	--------------

Cite all versions? You can cite all versions by using the DOI [10.5281/zenodo.1620782](https://doi.org/10.5281/zenodo.1620782). This DOI represents all versions, and will always resolve to the latest one. [Read more.](#)

Preview

- [5c91e7c7-e0c9-474f-b76a-62bff7b38468.zip](#)
- [5c91e7c7-e0c9-474f-b76a-62bff7b38468.xsams](#) 133.6 MB

Files (2.1 MB)

Name	Size	
5c91e7c7-e0c9-474f-b76a-62bff7b38468.zip	2.1 MB	Preview Download

ADS data linking & ORCID



https://ui.adsabs.harvard.edu

astrophysics data system

Classic Form **Modern Form** Paper Form

QUICK FIELD: Author First Author Abstract Year Fulltext All Search Terms

author author:"huchra, john" citations citations(author:"huchra, j") ?

first author author:"^huchra, john" references references(author:"huchra, j") ?

abstract + title abs:"dark energy" reviews reviews("gamma-ray bursts") ?

year year:2000

year range year:2000-2005 refereed property:refereed ?

full text full:"gravitational waves" astronomy database:astronomy ?

publication bibstem:ApJ ? OR abs:(planet OR star) ?

Use a classic ADS-style form

Learn more about searching the ADS

Access ADS data with our API



- IVOID in Registry
 - Authority
 - datasets/collections
 - services/protocols
- StandardIDs
- Publisher DIDs