Identification of objects with Uniform Resource Identifier (URI): recommendations for application in plant phenotyping

Jean-Eudes Hollebecq, Llorenç Cabrera Bosquet, Romain David, Marie-Claude Quidoz, Francois Tardieu, Anne Tireau, Pascal Neveu

Identification of objects with Uniform Resource Identifier (URI): recommendations for application in plant phenotyping.

- Jean-Eudes Hollebecq, MISTEA, INRAE, Montpellier SupAgro, Université de Montpellier, Montpellier, France, jean-eudes.hollebecq@inra.fr ORCID: 0000-0002-5089-9179
- Llorenç Cabrera-Bosquet, LEPSE, INRAE, Montpellier SupAgro, Université de Montpellier, Montpellier, France, llorenc.cabrera-bosquet@inra.fr ORCID: 0000-0002-2147-2846
- Romain David, MISTEA, INRAE, Montpellier SupAgro, Université de Montpellier, Montpellier, France, romain.david@inra.fr ORCID: 0000-0003-4073-7456
- Marie-Claude Quidoz, CEFE, UMR 5175 du CNRS, Montpellier, France, Marie-claude.QUIDOZ@cefe.cnrs.fr
- François Tardieu, LEPSE, INRAE, Montpellier SupAgro, Université de Montpellier, Montpellier, France, francois.tardieu@inra.fr ORCID: 0000-0002-7287-0094
- Anne Tireau, MISTEA, INRAE, Montpellier SupAgro, Université de Montpellier, Montpellier, France, anne.tireau@inra.fr ORCID: 0000-0001-8501-6922
- Pascal Neveu, MISTEA, INRAE, Montpellier SupAgro, Université de Montpellier, Montpellier, France, pascal.neveu@inra.fr

How to cite this resource: Hollebecq J-E., Cabrera-Bosquet L., David R., Quidoz M-C., Tardieu F., Neveu P., December 2019, “Identification of objects with Uniform Resource Identifier (URI): recommendations for application in phenotyping installations”, EPPN2020 - Emphasis Guideline / Technical report V1.0, DOI:10.5281/zenodo.3560629, 27 pp.

One page guide for EPPN^{2020} and EMPHASIS users

Tracking all objects involved in phenomic experiment and representing relationships between them is essential in a high throughput context where thousands of plots, plants or sensors are involved. The FAIR requirements need a proper strategy allowing to individually identify each specific object as well as semantic properties for creating relationships between objects. This strategy is based on URIs (Uniform Resource Identifier). URIs follow a standard used in genetics, chemistry, Internet of Things, life sciences, etc. They identify objects (virtual or real) with a syntax that allows non-ambiguousness, unicity, persistence, stability and resolvability (see annex for definitions).

We present herewith the rules already discussed in EPPN^{2020} or EMPHASIS PREP meetings, which need to be followed in such a way that phenomic data can be re-used.

1- All objects involved in experiments (e.g. plants, pots, sensors) need to have a URI, with an overall syntax :

http:// national-infrastructure/local infrastructure/installation/Identifier

for instance

http://www.phenome-fppn.fr/m3p/arch/2017/c17000915

2- These URI need to be defined by the responsible of the local infrastructure, for consistency between all installations of a local infrastructure, following the general syntax above.

3- By default, this syntax is generated by a tool provided to the projects through PHIS.

4- However, many installations already use local URIs that cannot be replaced because this would interrupt the data traceability, although these URIs usually do not comply with the non-ambiguousness requirement. In this case, it is proposed to use a prefix to the URIs already used, by just adding

http:// national-infrastructure/local infrastructure/installation/

before the URI that is currently used (*Identifier)*

This supposes that the latter complies with the requirements of non-ambiguousness, persistence, stability and resolvability. The survey to installations shows that this is the case in some installations, but not all. In particular, the identification via numbers automatically generated by the installation supplier does not follow this requirement. In this case, it is suggested that the responsible of the local infrastructure interacts with JRA3 (EPPN^{2020}) or WP4 (Emphasis prep) :

- Llorenç Cabrera-Bosquet llorenc.cabrera-bosquet@inra.fr
- Pascal NEVEU Pascal.Neveu@inra.fr

If you are interested on how and why to build identifiers as cool URIs, you can read the rest of this document.

Abstract

Phenotyping experiments are made in various installations from greenhouses to lean field. These different experiments tend to answer agricultural challenges such as food security. The plant phenotyping handles a multitude of different objects, from biological and genetical material to weather data by phenotypic traits measures. This multitude of objects from their diversity but also from their large number need to be handled properly (each installation is generating up to 10 terabytes of photos in a month for one experiment). Identifying these objects is more and more important in the scope of shared and open data: the aim is to correctly understand the mechanisms at work behind the observations, making the links between events and effects.

In order to fill their purpose, identifiers must have some properties, namely non-ambiguousness, unicity, persistence, resolvability, stability.

These properties will ensure that the identifier will stay active long-enough and allow the data associated to it to be used in a different domain where it was originally created.

In this document we present rules and good practices for object identification in the plant phenomics research domain and propose an adaptation of the rules designed by computer and web scientists. Some key characteristics of this schema are i) the ability to use semantics within the identifier and ii) making it resolvable through browser. Semantics is a desired quality for the technical staff of phenotyping platforms to manipulate the identifiers. However, it is also a big issue when designing a persistent identifier (e.g. usable in more than twenty years) and an important drawback if people put too much semantics in it. The first recommendation is to use only the minimum information in identifiers, and the date of creation is a good one.

Table of Contents

Illustration table:

All objects in phenotyping experiments need to be identified

Phenotyping experiments cover many different objects (e.g. physical sensors, biological material, pots) that need to be individually identified. Having these objects identified allows to create connections between them and *in fine* knowledge. This is enhanced in a context of high throughput where thousands of plants or plots are involved. These connections not only benefit the current experiments by creating added scientific value but also benefit the future experiments that will be able to reuse these data.

For instance, such knowledge could be the history of positions of a given sensor (e.g. soil tensiometer or thermometer) along with the different calibrations of this sensor. Having this knowledge allows us to deconstruct the confusion that may hide in the data. The list of unexpected events is very long, from sensor replacement to plant repositioning by emergency antifungal treatment. These events will be used in long term analysis where several data provenances will be aggregated.

To illustrate this document, we will have a look at PHIS (Phenotyping Hybrid Information System), and eventually propose a strategy to identify all objects involved in a phenotyping environment. Objects in PHIS (Neveu, 2019) are identified with Uniform Resource Identifiers (URI). You can explore PHIS at (www.phis.inra.fr). This identification ensures the traceability of every object in space and time. URIs are generated via the information system automatically. The strategy adopted is a semantic identification including year (e.g. /localInstallation/2019/experiment/plant736) coupled with usage of QR-codes so that any terminal can access the objects and drawing links between them. For instance, one can scan the QR code of a plant then the QR code of a carrier to link the two objects.

1 - Identification challenges

What is a URI?

URI stands for **Uniform Resource Identifier**.It is a standard used in a large variety of domains: genetics, chemistry, Internet of Things, life sciences, etc. URI purpose is to identify objects (virtual or real) and has to be designed properly to avoid issues. As an identifier, URIs must have some properties: non-ambiguousness, unicity, persistence, stability and resolvability.

Here is an example of a URI:

http://www.phenome-fppn.fr/m3p/arch/2017/c17000915

With the following schema:

http://authority/path/LocalIdentifier

Short definitions of expected properties:

- Non-ambiguousness: The URI must be associated with only one resource.
- Unicity: One resource should have only one URI, or have a tool to resolve equivalent URI.
- Persistence: One should not replace or delete the URI.
- Stability: URI has to remain the longest time possible (at least 20 years), and not be reassigned to another resource. The definition is close to the persistence; stability is persistence over long time.
- Resolvability: URI should be used through internet browser to find information about the resource or the resource itself.

The properties and problems if URIs don't follow the rules, are explained in the Table 1. For more details about URI design and properties, it has been described in RFC 3986 from W3C (World Wide Web Consortium) in 2005.[1]

[1] The World Wide Web Consortium is led by 3 organizations (the MIT Computer Science and Artificial Intelligence Laboratory (United States), Keio University (Japan) and INRIA (Institut National de Recherche en Informatique et Automatique) France. Its role is only advisory.

What are the things to identify?

Ideally we want to identify everything, but we have very different resources, do we identify them in the same way? Are URIs the best option to identify these resources? That is a question you should think about before designing URI.

For instance measurements collected by a sensor can be gathered in a dataset and require one URI for the dataset, or even can be aggregated in a database. Then, the daily measurements are identified with a data URI.

Other identifiers already exist and are well established such as DOI for documents, ISBN for books, ORCID for researchers, etc.

Upgrade an existing identifier?

Moving from an existing identifier (as a primary key) to a URI is not difficult, in theory it is just extending the character string to include a resolvable part and a few more information.

The aim is to have a good identifier in the end. So if your existing identifier is not designed as a good identifier, it's better to replace it with a brand new URI.

An improper design for identifiers is for instance, including the XY coordinates of a given object such as: '*potX014Y110*' because position can change over time and potX014Y110 might be somewhere else than the X014 and Y110 coordinates in the future, leading to an ambiguous URI.

If your precedent Identifier was 'sensor_019' then it could be transformed as a URI concatenating: *http://www.NationalInstitution/LocalInfrastructure/2019/sensors/sensor_019*

Such an identifier is sensitive to change over time, such as INRA changing its name.

So an alternative is to use an opaque name and bind it with inra.fr resolution *BC23A786DF5/m3p/arch/2019/sensors/sensor_019*

What problems do URIs face?

Table 1: A few examples of what problems can happen if the properties are not fulfilled. Semantic and opaque identifier are neither a good solution, but an intelligent composition of these two characteristics gives good identifiers.

Problem	Source of error	Consequence
Ambiguous	**Plant984** designate two different plants	Collision between two plants, user will sow the wrong genotype
Not unique	**Camera12** and **NDVI_sensor_2** designate the same sensor measuring the same NDVI	Correlation between yield and NDVI will not be understood because of name confusion
Not persistent	**Plant984** stands for a different plant every year	Confusion when doing multi-year analysis. **Plant 984** is different every year
Not stable	Link rot returns the 404 error, an empty page with no information	Lost the ability to do multi-year analysis due to lost information
Not resolvable	Identifier not findable on the Internet	Inability to contact the author due to lack of access to the metadata
Opaque	*http://9089d5.net/f7cf8baa2fd* gives absolutely zero information about the resource it might identify.	The operator may encounter great difficulties if he has to find a dozen plants among several hundred
Semantic	*http://www.phenome-fppn.fr/montpellier/phenoarch/2017/maize/water-deficient/jeanne/pot170/plante521/hand-report/biomass-measure/915* is not a good option either.	Too much semantic leading to an unnecessary big identifier, likely to become obsolete rapidly and difficult to copy/manipulate without errors

In the next paragraphs we will see how to design a cool URI.

2 - Design a cool URI

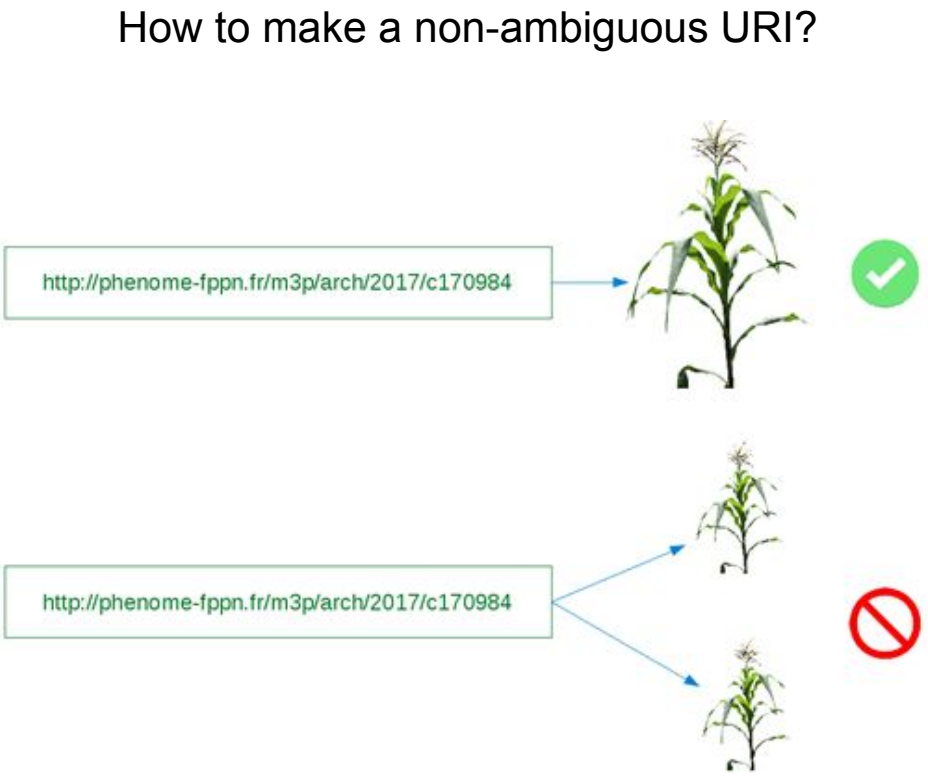


Figure 1: A non-ambiguous identifier associated with only one object and an ambiguous identifier associated with two different objects.

By non-ambiguous we mean that an identifier only stands for one resource. **Two objects cannot have the same identifier (Fig 1).** Identifiers can be generated with several strategies. For instance, PHIS uses a semantic strategy with an incremental number (the number of the plant) and the year, prefixed with a letter that helps to manipulate the URI.

Technical focus on non-ambiguous URI

The name of the URI should be in two parts: **pattern** and **local ID**.

http://phenome-fppn.fr/m3p/arch/2017 is **the pattern** and gives information about the object nature and helps about the understanding. Information will necessarily be missing in this part, provide only the most relevant one.

C170984 is a **local ID** with some minimal semantics to guarantee the non-ambiguity.

There are a few ways to generate a local ID. The easiest one is the **incremental option**, with the assumption that the pattern of the URI will help avoid collision with other "plant N°984" somewhere else.

The next step is to add a **semantics on top of the incremental ID**, to specify that it is a "type c" created in "2017" object. There is obviously a need for a dictionary to understand what "c" refers to.

Another option is to rely on **external identifiers**, like ePIC, DOI, identifiers.org, w3id.org, handle.net. Being external services and large scale identifiers, they will provide a good unique URI, with the drawback to be dependent on a third party.

handle.net is a registry run by Corporation for National Research Initiatives (CNRI), assigning identifiers, or handles to information resources. Handle System identifiers are opaque, meaning they have no information about the resource they identify, being bound only to metadata regarding the resource. Consequently, the handles are not rendered invalid by changes to the metadata.

Useful identifier should be user friendly and incorporate some semantic to make the identifier meaningful.

How to make a unique URI?

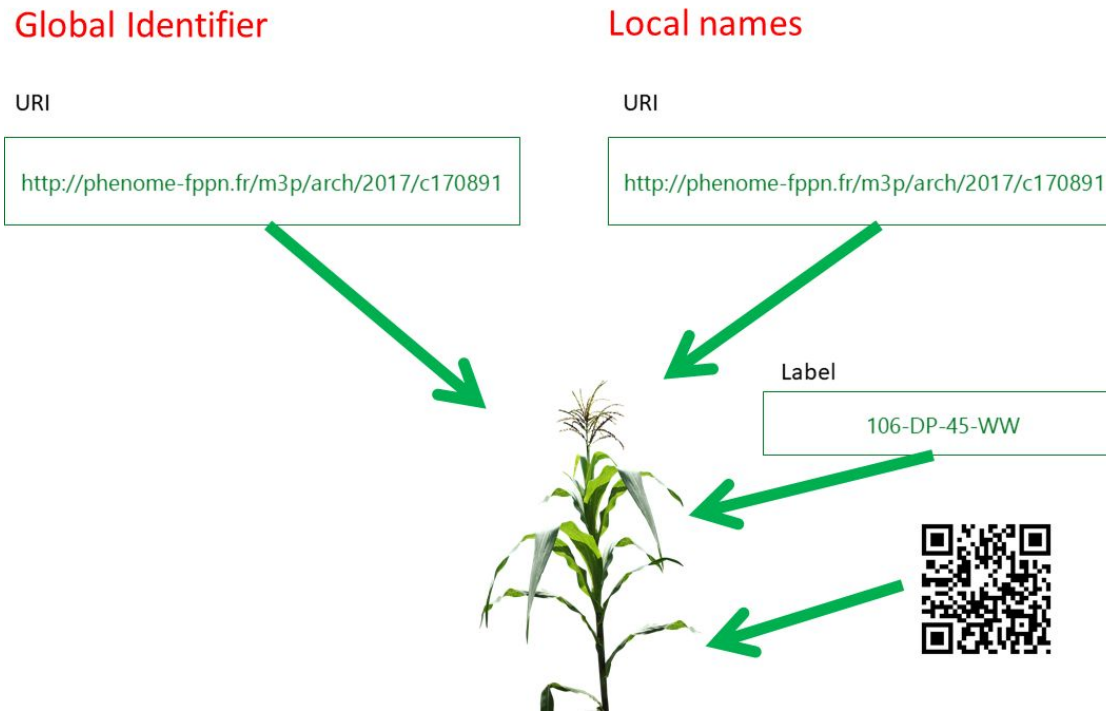


Figure 2: A unique global scale identifier on one side, and on the other side multiple local identifiers for limited scale

Unicity means that there should be only **one identifier for one resource**. This property is hardly compatible with real life cases, as URI can be difficult to read or manipulate. If using multiple URI for the same object is wrong and leads to errors, one can can use **aliases and labels** for URI. Humans do this all the time: you probably have yourself different nicknames. It is possible to use multiple names for the same resource (Fig 2). One of those names should follow the rules for “cool URI” and the others could be shorter identifiers and more human friendly. The biggest difference between the aliases and the URI is the scale of the identification. If used in a local scale (station or platform), and for just a limited time, the recommendations are lowered and aliases are suitable. But for long term identification, and global scale, URIs are mandatory.

Unicity is using concepts that are described in the RFC4122 describing the Universally Unique IDentifier (UUID).

The URI can be used in a form of a QR-code to help the interoperability and machine readable aspect of the URI. For example, if you scan the URI of a camera then the URI of an Unmanned Aerial Vehicle (UAV), you can link the two with the relation “installed” or “uninstalled”.

Technical focus on unique URI

Identification being a bijection. This name should also give enough information to identify the nature of the object behind the URI and should **be understood by humans as well as machines**. Words have to be used carefully as they often change or create confusion. It is advisable to use the creation date of the object or its intrinsic nature.
There is a trade-off between a URI that describes perfectly the resource and the persistence of the identifier.

The URI can be a quite long character string and is not perfectly suited in everyday work. To make this work easier, one can create a label or alias for the resource (e.g. a plant) and work with this alias. In the end there will be two columns that would identify the object, the 'URI' for long term and long scale identification and the 'label/alias' for short term and short scale.

Data linking: having different identifier for the same resource is a challenge encountered in data linking, when merging several databases to gather information. Obviously this alignment of identifiers, in order to say that it identifies the same resource could be made automatically. Some tools can help you if a resource has been given multiple URIs, and create an equivalence between the two. This kind of challenge is a scientific subject by himself in data mining sciences and ontologies uses (David *et al*., 2019, Ferrara *et al*., 2011).

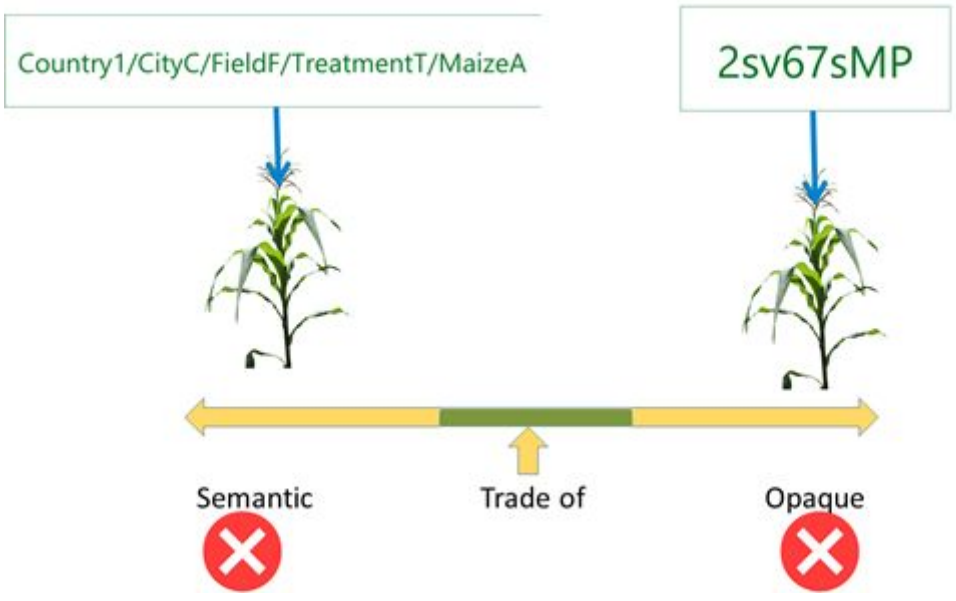


Figure 3: Trade-of between a semantic identifier and an opaque identifier.

Technical focus on short URL and manipulations

The URL refers to real things and helps scientists register them in databases. Thus **reading** and **manipulating** the URL **should be facilitated** with short URLs, or with the use of QR-code/RFID that can be scanned. Several services already propose to shorten your URL and create **QR-code or barcode** with it. a QR code is just another way of writing the URI, it is used to facilitate physical world interoperations.

How to make a resolvable URI?

Figure 4: Making a URI resolvable enables internet browser to action it and retrieve the information behind it.

The term resolvable (also means dereferenceable) means that the resource is accessible through internet. For example: a browser should open a page with the resource or information about the resource (Fig 4). This is achieved by using **Internet Protocol**. It also means that you have to provide different output to this resource. If you are using a GET request for an image as a png format you will get the image. But if you make the same request for an XML format you will get the metadata associated to it.

Tipping http://purl.obolibrary.org/obo/foodon.owl in your browser redirects you to : https://raw.githubusercontent.com/FoodOntology/foodon/master/foodon.owl where you can find information about the resource *foodon.owl*.

Technical focus on resolvable URI

The W3C recommend to use HTTP-S URIs in the scope of Linked Open Data. The resolvability comes along with some HTTP status. The most well-known is the *404 not found* status (no page behind an URL). But there are also Redirection status *300* and most of the time *200* status when everything is OK. These *300* status redirect from one URL to another URL and helps dealing with 404 errors.

The 300 status is a family of HTTP response concerning redirection. It means that the URL has been moved, and the request is redirected to the new URL. You can have different fragrances, such as 301 been permanently moved, when 302 is temporarily moved.

How to make a persistent URI?

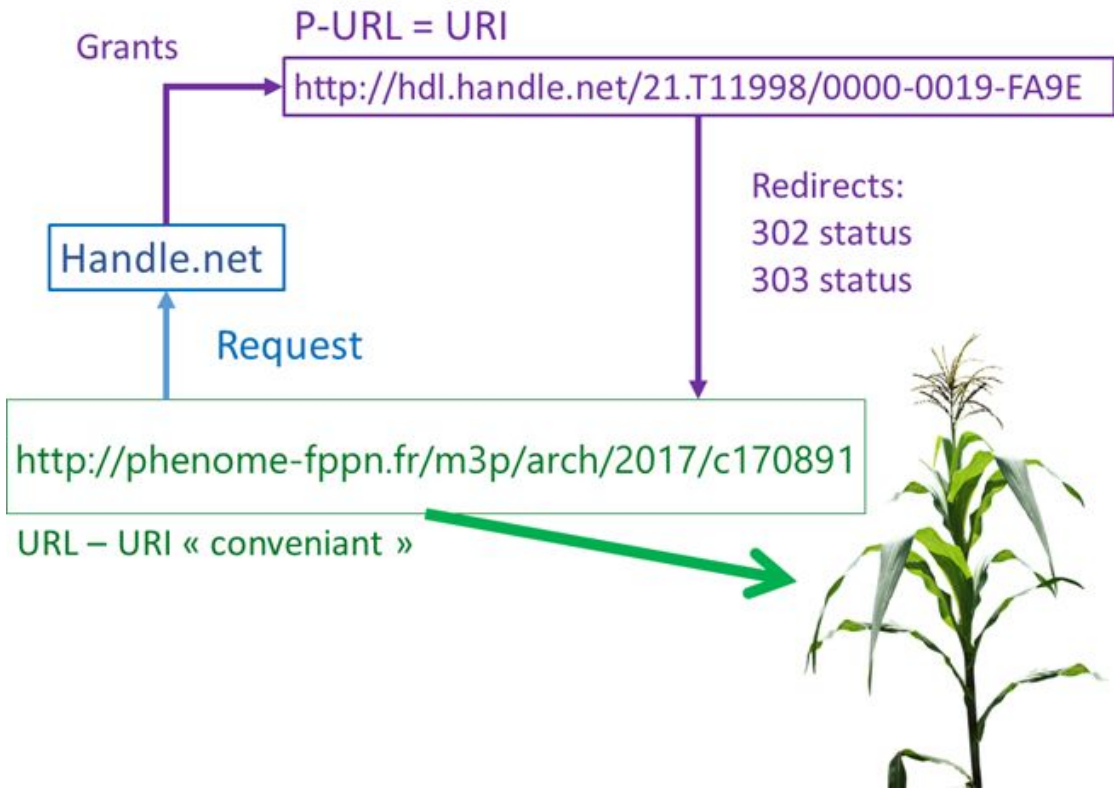


Figure 5: A persistent URI should request large scale identifier with 300 redirections between the URI and a convenient identifier.

Persistence means that once URI is given to a resource, it should not change. The way URI is written should not differ and the object associated with the URI should remain the same, even if the object itself does not exist any more. This purpose is harder to achieve because most of the time, information present in the URI change (e.g. ownership, rename, etc.), then the URI becomes either ambiguous or outdated. For this issue, persistent URL are commonly used. This persistent-URL should be the URI, written in a never changing

way and redirect to the appropriate content (see point 3). That means that the *http://handle.net/21.T11998/0000-0019-FA9E* redirects to the *http://phenome-fppn.fr/* link with information about the resource (Fig 5).

Different tools exist to create persistent-URL. We have already mentioned them as external identifiers ePIC, DOI, identifiers.org, w3id.org, handle.net. Most of them rely on the handle system and creates an opaque identifier along with a prefix that is linked to the data authority you provided (inra.fr for example).

Technical focus on persistent URI

Persistent URLs use HTTP 303 **See also** redirect status (Fig 5.). This way, we have a persistent URI associated with a non-persistent URL. This persistent URI will never change, however the non-persistent one can change, we can rename the domain name, reformulate the path, etc. There is just one constraint: to update the redirection of the persistent URI to the new URL to avoid broken links (HTTP 404).

For interoperability purposes, multiple names can be used to designate the same content. Some names can be non-persistent and use a common language, we call them aliases or rdfs:labels, and one identifier should be persistent and follow the recommendations proposed, and be called a URI.

URI must not attempt to describe or give excessive information about the object itself, this is the purpose of metadata. Almost every information related to the object are likely to change, even the authors can change if someone decides to review and comment on a document. Information present in the persistent URI should only be the never changing ones.

Words can change, date won't.

The difficulty when writing a persistent URI is to foresee how it could become non-persistent over the long term. Most of the problems around non-persistent URI come from the association between the name of the URI and the content of the URI. Therefore, removing names seems a good idea. The names can change over time, due to technological advancement, cultural shift, etc. Adding the date when it was created helps giving context of the creation. This recommendation is pleased by the W3C for persistent URI design (link to URI W3C recommendations). The recommendation is to **use minimal information** (Ćwiek-Kupczyńska, 2015) that would be sufficient to allow a persistent URI. Information suitable for URI are the creation date and location (this two will never change) and also the intrinsic quality of the resource (i.e. plant, plot, etc. but warning that it will not change, a seed growing to a plant for example...).

Technical focus on persistent data authority

One big issue with identifiers is the short life cycle of domain names. Domain names are often linked to organisms, those are prone to transformations and fusions. Thus domain names often change or disappear leading to non-persistent identifiers. Data should be stored in open infrastructures and decentralized services that are less likely to change or disappear. Even though relying on third party is sometimes not handy, primary and meta resolvers like: B2handle, pidconsortium.eu, doi.org, hdl.org, identifiers.org, n2t.net or PURL.org are well established. Their purpose is to generate and resolve high scale URI.

How to make a stable URI?

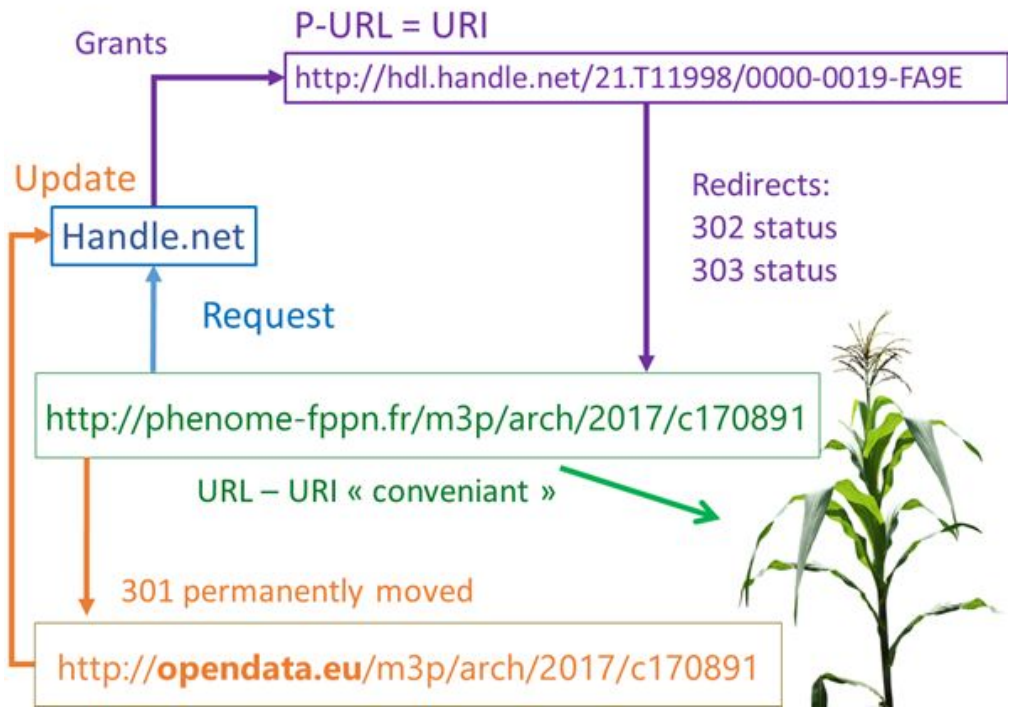


Figure 6: Ensuring stability of identifiers by updating the redirections 300 status

Stability helps you to rely on the URI over the long term, avoid reusing URI for another object. The first recommendation for stability is **the use of the date of creation** within the URI, making it non suitable for future object or past objects. Stability is achieved by using URI written with the minimal information, so that it will not be outdated or misleading due to re-name or ownership change.

Technical focus on stable URI

Stability is increased by using a persistent URL as mentioned above using the 303 redirect status. And further increased by the **link redirection** between the different URI thanks to the 301 permanently moved status (Fig 6). This way if the URL is changing, the redirection keeps the link alive. You have to be careful when multiplying the URIs, the cost of redirection will rise abruptly when using too many URIs.

If you analyse the handle schema (full documentation) you can see that it is composed with the following schema :

prefix/OPTIONAL_PRE-num1-num2-num3-c-OPTIONAL_APP

prefix is the handle prefix, allocated to your authority, it can be updated if your authority changes domain name..

num1-num2-num3 are 12 bytes, coded in uppercase hexadecimal digits with delimiters. This is the fingerprint of your document, a completely opaque identifier.

c is a checksum to ensure plausibility of the handle string.

OPTIONAL_PRE and OPTIONAL_APP are optional fields maximal 32 letter for uppercase alphanumeric characters that gives you some flexibility. However, you should use it only in rare case as it can disable the persistence of the URI.

The whole URI is linked to an URL through pidconsortium services. You can update the history of URLs (and you definitely should do it).

Technical focus on fingerprints in URI.

Seeing forward: Many databases are, from the initiative of their authors, or when old projects are taken over by other/bigger projects, aggregated into more homogeneous databases, and structured with common vocabularies in larger communities. During these aggregations, modifying the pathway or switching to a common vocabulary, may lead to a transformation of URIs (for example, the word “grain” replaced by “seed”: it is modified to have the same name in the database). It often happens that the old URI is not maintained / rerouted (301) to the new equivalent, making it non-persistent. But nonetheless still used by the other communities with the ancient URI. A mean to facilitate the matching of these URIs (also called data binding) is to insert in its schema, somewhere, for example at the end of the URI an opaque alphanumeric fingerprint with a sufficient size (10 to 12 characters) generated through cryptographic algorithm (SHA-256, md5, etc) (The hexadecimal format also avoids the type-zero and letter O-like characters). In case of moving of the database (and change in URI syntax), the search of this fingerprint with a search engine in the web of data allows to find over a very long term the probable matches between an object and an obsolete URI. At least it reduces the list of possible identical fingerprints to a very few elements as the probability to generate the exact same fingerprint is very low. This acts as a double security for the data traceability and data finding with old identifiers.

Example:

http://phenome-fppn.fr/m3p/arch/2017/c1700891-a1368bcf2hc19

Conclusion

Recommendations in image

The resource we are identifying is a maize plant. It is produced by the arch platform of m3p in year 2017. This plant originally has the URI in green, and a short URI for interoperability associated with a QR-code. This first URI has some **ownership information** in its name, the name of the domain is **susceptible to change**, the name of the platform (m3p/arch) can also change in long time perspective. For persistence purpose, handle system is required to create a Persistent-URI (purple).

To do so, **e-PIC service (pidconsortium)** is required to create a unique ID (blue in the figure 7) for the platform. When creating an identifier, a link is made between the new identifier (http://hdl.handle.net/*21.T11998/0000-0019-FA9E)* and *phenome-fppn.fr*. If later the domain name *phenome-fppn.fr* changes, then it just has to be updated in e-PIC's service.

e-PIC now grants a unique identifier and set a 303 redirect status associated with the URL. If anyone wants to find the plant *0000-0019-FA9E*, the PURL redirects to the phenome link, and finally to the plant associated. By the way e-PIC identifier is semantic free, that is not suitable for human friendliness. It is desirable in a context of phenotyping, where internet network may not be available at any time (field work) to have a minimum of semantic in the identifiers to help humans. This can be achieved by suffixing the e-PIC identifier with some semantic.

Imagine that for some reason (fusion, rename, etc.), the phenome-fppn/m3p/arch is changing. Then we should have a 301 permanently moved status on the previous URL to redirect to the new one (orange in figure 7). Meaning that the e-PIC URI will now redirect to the new URL from the same URI.

By adopting this strategy, we avoid rotting links due to URL rename. The PURL having minimal information will remain valid longer and not leading to ambiguity. The

301-302-303 redirect status has to be properly used to link all these URI/URL together.

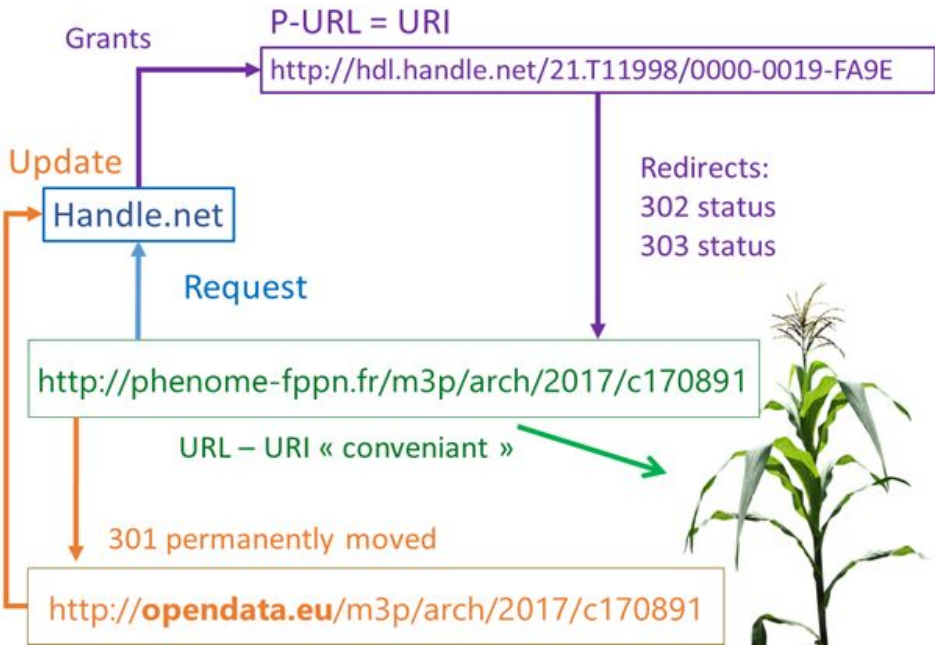


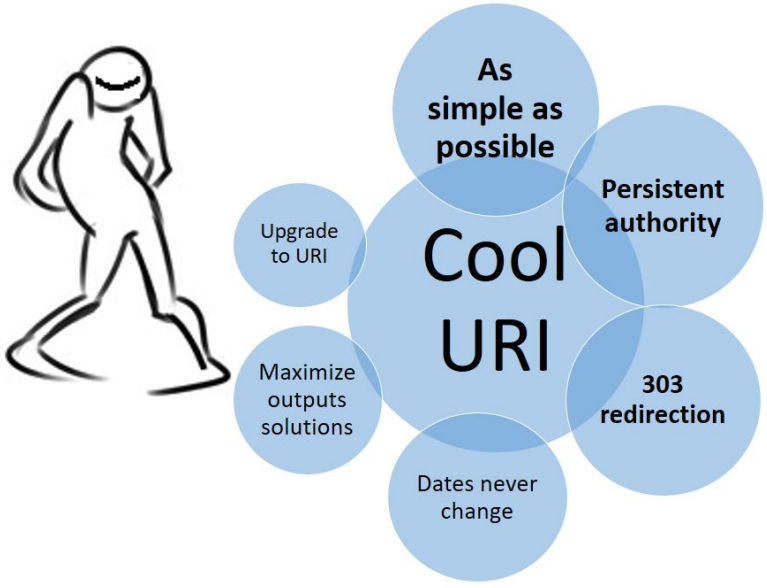
Figure 7: A convenient URI, with semantic information sensitive to change is granted a persistent URI via Handle. The redirection between the persistent URI and the convenient identifier is made with 302 or 303 status. Change of the convenient URI managed by 301.

Take home message

Here are a few rules one should follow in order to create a good URI:
Details about the points presented here are exposed above.

Things you should do

1. Use minimal information, get rid of everything that may change.
2. Require external identifier (B2HANDLE, e-PIC) if your authority is not persistent enough.
3. Use persistent-URL with 303 redirect status.
4. Associate creation date to help understanding.
5. Provide multiple output format (.txt, .html, .csv, etc.) and link them together, so the user will have the choice.
6. Integrate/upgrade already existing identifiers in a URI.



Things you should avoid

1. Avoid unnecessary metadata in the identifier.

2. Avoid ownership and other information that are likely to change over time, prefer nature of the resource.

3. Avoid unnecessary long identifiers with too much semantic.

4. Avoid entirely opaque identifier.

5. Avoid files extension in the URI (no `.extension` in the URI).

6. Avoid query (no ”?” in the URI).

7. Avoid misleading characters such as O and 0 or I and l, etc.

8. Avoid URI that are not the best way to identify the object you are looking at ?

References

Web resources

https://brapi.docs.apiary.io/
https://www.ebi.ac.uk/ols/ontologies/co 357/terms?iri=http%3A%2F%2F
https://www.ebi.ac.uk/services
https://www.ebi.ac.uk/rdf/documentation/good_practice_uri
http://europepmc.org/abstract/MED/271968
https://www.cyverse.org/products
https://archive.org/services/purl/
http://identifiers.org/
http://n2t.net/
http://www.phis.inra.fr/openphis/web/index.php
https://www.doi.org/
https://www.w3.org/
https://www.w3.org/OWL/
https://www.w3.org/RDF/
https://www.w3.org/TR/rdf-schema/
https://www.w3.org/TR/skos-reference/
https://www.w3.org/TR/rdf-sparql-query/
https://www.w3.org/wiki/URI
https://www.w3.org/TR/url/
https://www.w3.org/urn/
https://www.w3.org/wiki/UriSchemes/uuid
https://www.w3.org/Provider/Style/URI
https://www.w3.org/html/
https://www.w3.org/Protocols/
https://www.w3.org/wiki/LinkedData
https://www.w3.org/2009/08/skos-reference/skos.html#exactMatch
https://tools.ietf.org/html/rfc2616
https://tools.ietf.org/html/draft-kunze-ark-18
http://www.ietf.org/rfc/rfc3651.txt
http://www.ietf.org/rfc/rfc3652.txt
https://www.ietf.org/rfc/rfc3986.txt
https://www.ietf.org/rfc/rfc3987.txt
https://www.ietf.org/rfc/rfc4452.txt
https://tools.ietf.org/html/rfc7231
http://www.miappe.org/
https://plantmethods.biomedcentral.com/articles/10.1186/s13007-016-0144-4
http://www.cropontology.org/
http://purl.obolibrary.org/obo/BFO 0000001
http://obofoundry.org/id-policy
https://philarcher.org/diary/2013/uripersistence/
https://en.wikipedia.org/wiki/URL shortening
https://bitly.com
https://joinup.ec.europa.eu/document/10-rules-persistent-uris
https://5stardata.info/en/
https://wiki.pro-ibiosphere.eu/wiki/Best_practices for stable URIs
http://linkeddata.org/
http://www.purlz.org/

https://www.eudat.eu/services/b2handle
https://www.pidconsortium.eu/
https://github.com/EUDAT-B2SAFE/B2HANDLE
https://github.com/bioversity/Crop-Ontology
https://github.com/Planteome/plant-trait-ontology
https://en.wikipedia.org/wiki/Persistent_uniform_resource_locator
https://www.oasis-open.org/committees/xri/
https://fr.wikipedia.org/wiki/Globally_Unique_Identifier
http://www.lsid.info/
https://orcid.org/

Scientific References:

- Bafna, Sapna & Humphries, Julian & P Miranker, Daniel. (2008). Schema Driven Assignment and Implementation of Life
- Clark, Tim & Martin, Sean & Liefeld, Ted. (2004). Globally distributed object identification for biological knowledge bases. Briefings in bioinformatics. 5. 59-70. 10.1093/bib/5.1.59.
- Ćwiek-Kupczyńska H., Altmann T., Arend D., Arnaud E., Chen D., Cornut G., Fiorani F., Frohmberg W., Junker A., Klukas C., Lange M., Mazurek C., Nafissi A., Neveu P., van Oeveren J., Pommier C., Poorter H., Rocca-Serra P., Sansone S.-A., Scholz U., van Schriek M., Seren Ü., Usadel B., Weise S., Kersey P., Krajewski P. Measures for interoperability of phenotypic data: minimum information requirements and formatting. Plant Methods. 2016;12:44.
- Dappert, Angela, et al. « Connecting the Persistent Identifier Ecosystem: Building the Technical and Human Infrastructure for Open Research ». Data Science Journal, vol. 16, no 0, juin 2017, p. 28.
- David R., Hollebecq J-E., Cabrera-Bosquet L.,....Symeonidou D., Tardieu F., Tireau A., Neveu P., August, 2019, “Recommendations for references ontology uses in plant phenotyping”, EPPN2020 - Emphasis Guideline / Technical report V1.0, DOI:XXX, 40 pp.
- Dong Joon Lee & Besiki Stvilia. Identifier Schemas and Research Data. https://www.asis.org/asist2012/proceedings/Submissions/311.pdf
- Alfio Ferrara, Andriy Nikolov, and François Scharffe. Data linking for the semantic web. Int. J. Semantic Web Inf. Syst., 7(3):46–76, 2011
- Haak LL, Meadows A and Brown J (2018) Using ORCID, DOI, and Other Open Identifiers in Research Evaluation. *Front. Res. Metr. Anal.* 3:28. doi: 10.3389/frma.2018.00028 https://www.frontiersin.org/articles/10.3389/frma.2018.00028/full
- Hakala, Juha. (2010). Persistent identifiers – an overview.
- Halewood, Michael, et al. « Plant Genetic Resources for Food and Agriculture: Opportunities and Challenges Emerging from the Science and Information Technology Revolution ». The New Phytologist, vol. 217, no 4, mars 2018, p. 1407‑19.
- The Handle System, is a technical specification for assigning, managing, and resolving persistent identifiers assigned to digital objects and other Internet resources. handle.net. Retrieved 2019-12-03.
- Implementing Persistent Identifiers. Overview of concepts, guidelines and recommendations (report). Hans-Werner Hilse and Jochen Kothe. Consortium of European Research Libraries European Commission on Preservation and Access 2006 http://xml.coverpages.org/ECPA-PersistentIdentifiers.pdf
- Krajewski P., Chen D., Ćwiek H., van Dijk A.D.J., Fiorani F., Kersey P., Klukas C., Lange M., Markiewicz A., Nap J.P., van Oeveren J., Pommier C., Scholz U., van Schriek M., Usadel B., Weise S. Towards recommendations for metadata and data handling in plant phenotyping. J. Exp. Bot. 2015;66:5417–5427.
- McMurry, et al. (2017). "Identifiers for the 21st century: How to design, provision, and reuse persistent identifiers to maximize utility and impact of life science data". PLOS Biology. 15 (6): e2001414. doi:10.1371/journal.pbio.2001414. ISSN 1545-7885. PMC 5490878. PMID 28662064.

● Meadows A, Haak L L and Brown J, Persistent identifiers: the building blocks of the research information infrastructure, Insights, 2019, 32: 9, 1–6; DOI: https://doi.org/10.1629/uksg.457
● Neveu, Pascal, et al. « Dealing with Multi-Source and Multi-Scale Information in Plant Phenomics: The Ontology-Driven Phenotyping Hybrid Information System ». The New Phytologist, vol. 221, no 1, janvier 2019, p. 588‑601.
● Page, R. D. M. « Biodiversity Informatics: The Challenge of Linking Data and the Role of Shared Identifiers ». Briefings in Bioinformatics, vol. 9, no 5, avril 2008, p. 345‑54.
● Roitsch T, Cabrera-Bosquet L, Fournier A, Ghamkhar K, Jimenez-Berni J, Pinto F, Ober ES. 2019. Review: New sensors and data-driven approaches-A path to next generation phenomics. Plant Science 282: 2-10.
● Tzounis, Antonis, et al. « Internet of Things in Agriculture, Recent Advances and Future Challenges ». Biosystems Engineering, vol. 164, décembre 2017, p. 31‑48.
● Wilkinson, Mark D., et al. « The FAIR Guiding Principles for scientific data management and stewardship ». Scientific Data, vol. 3, mars 2016, p. 160018.
● Wolfert, Sjaak, et al. « Big Data in Smart Farming – A Review ». Agricultural Systems, vol. 153, mai 2017, p. 69‑80.