Identifiers for the 21st century: How to design, provision, and reuse persistent identifiers to maximize utility and impact of life science data

Julie A McMurry[1], Nick Juty[2], Niklas Blomberg[4], Tony Burdett[2], Tom Conlin[1], Nathalie Conte[2], Mélanie Courtot[2], John Deck[3], Michel Dumontier[5], Donal K Fellows[6], Alejandra Gonzalez-Beltran[7], Philipp Gormanns[8], Jeffrey Grethe[9], Janna Hastings[13], Henning Hermjakob[2], Jean-Karim Hériché[10], Jon C Ison[11], Rafael C Jimenez[4], Simon Jupp[2], John Kunze[12], Camille Laibe[2], Nicolas Le Novère[2,13], James Malone[2], Maria Jesus Martin[2], Johanna R McEntyre[2], Chris Morris[14], Juha Muilu[15], Wolfgang Müller[16], Philippe Rocca-Serra[7], Susanna-Assunta Sansone[7], Murat Sariyar[18,19], Jacky L Snoep[20,21], Natalie J Stanford[6], Stian Soiland-Reyes[6], Neil Swainston[22], Nicole Washington[17], Alan R Williams6, Sarala Wimalaratne[2], Lilly Winfree[1], Katherine Wolstencroft[23], Carole Goble[6], Christopher J Mungall[17], Melissa A Haendel[1], Helen Parkinson[2]

1. Department of Medical Informatics and Epidemiology and OHSU Library, Oregon Health & Science University, Portland, USA.
2. European Bioinformatics Institute (EMBL-EBI), European Molecular Biology Laboratory, Wellcome Trust Genome Campus, Hinxton, Cambridge, United Kingdom
3. Berkeley Natural History Museums, University of California at Berkeley
4. ELIXIR Hub, Wellcome Trust Genome Campus, Hinxton, Cambridge, United Kingdom
5. Center for Biomedical Informatics Research, Stanford University, Stanford, California, USA
6. School of Computer Science, The University of Manchester, Manchester, United Kingdom
7. Oxford e-Research Centre, University of Oxford, Oxford, United Kingdom
8. Institute of Experimental Genetics, Helmholtz Centre Munich -German Research Center for Environmental Health (GmbH), Neuherberg, Germany
9. Center for Research in Biological Systems, University of California San Diego, La Jolla, California, USA
10. European Molecular Biology Laboratory, Heidelberg, Germany
11. Center for Biological Sequence Analysis, Department of Systems Biology, Technical University of Denmark, Lyngby, Denmark
12. California Digital Library (CDL)
13. Babraham Institute, Cambridge, United Kingdom
14. STFC, Daresbury Laboratory, Warrington, United Kingdom
15. Genomics Coordination Ctr, Dept of Genetics, University Medical Center Groningen and Groningen Bioinformatics Center, U of Groningen, Netherlands
16. SDBV, HITS, Heidelberg, Germany
17. Genomics Division, Lawrence Berkeley National Laboratory, Berkeley, CA, USA
18. Institute of Pathology, Charite – University Medicine Berlin, Berlin, Germany
19. TMF – Technologie- und Methodenplattform e. V. Berlin, Germany
20. MIB, University of Manchester, Manchester, UK
21. Department of Biochemistry, Stellenbosch University, Stellenbosch, South Africa
22. Manchester Centre for Synthetic Biology of Fine and Speciality Chemicals (SYNBIOCHEM), University of Manchester, Manchester, UK.
23. Leiden Institute of Advanced Computer Science, Leiden University, Leiden, Netherlands
24. Center for Research in Biological Systems, University of California San Diego, La Jolla, California, USA

Abstract

In many disciplines, data is highly decentralized across thousands of online databases (repositories, registries, and knowledgebases). Wringing value from such databases depends on the discipline of data science and on the humble bricks and mortar that make integration possible; identifiers are a core component of this integration infrastructure. Drawing on our experience and on work by other groups, we outline ten lessons we have learned about the identifier qualities and best practices that facilitate large-scale data integration. Specifically, we propose actions that identifier practitioners (database providers) should take in the design, provision and reuse of identifiers; we also outline important considerations for those referencing identifiers in various circumstances, including by authors and data generators. While the importance and relevance of each lesson will vary by context, there is a need for increased awareness about how to avoid and manage common identifier problems, especially those related to persistence and web-accessibility/resolvability. We focus strongly on web-based identifiers in the life sciences; however, the principles are broadly relevant to other disciplines.

Introduction

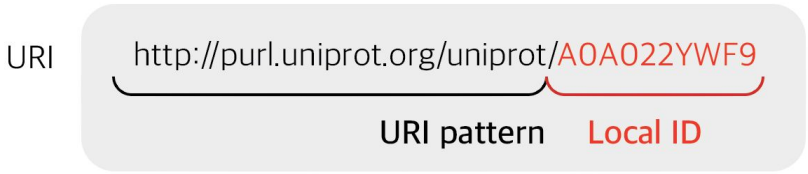
The issue is as old as scholarship itself: readers have always required `persistent identifiers` in order to efficiently and reliably retrieve cited works. ‘Desultory citation practices’ have been thwarting scholarship for millennia[1]. While the Internet has revolutionized the *efficiency* of retrieving sources, the same can not be said for reliability: it is well established that a significant percentage of cited web addresses go "dead"[2]. This process is commonly referred to as `link rot` because availability of cited works decays with time[3,4]. Although `link rot` threatens to erode the utility and reproducibility of scholarship[5], it is not inevitable: link persistence has been the recognized solution since the dawn of the Internet [6]. However, this problem, as we

will discuss, is not at all limited to referencing journal articles. The life sciences have changed a lot over the past decade as the data have evolved to be ever larger, more distributed, more interdependent, and more natively web-based. This transformation has fundamentally altered what it even means to 'reference' a resource; it has diversified both the actors doing the referencing and the entities being referenced. Moreover, the challenges are compounded by a lack of shared terminology about what an 'identifier' even is. Box 1 delineates the key components of an identifier used throughout this paper; all technical terms are in fixed width font and defined in the glossary.

Box 1. Anatomy of a persistent identifier

An **identifier** is a sequence of characters that identifies an entity. The term '**persistent identifier**' is usually used in the context of digital objects that are accessible over the Internet. Typically, such an identifier is not only persistent but also actionable[7]: it is a **Uniform Resource Identifier (URI)**[8], usually of type http/s, that you can paste in a web browser address bar and be taken to the identified source.

An example of an exemplary **URI** is below; it is comprised of ASCII characters and follow a pattern that starts with a fixed set of characters (URI pattern). That URI pattern is followed by a **Local ID**--an identifier which, by itself, is only guaranteed to be locally unique within the database or source. A local ID is sometimes referred to as an 'accession'.



Formally breaking down a URI into into these two components (URI pattern and local ID) makes it possible for meta resolvers to 'resolve' entities to their source. This practice also facilitates representation of a URI as a **compact URI (CURIE),** an identifier comprised of <**Prefix**>:<**Local ID**> wherein **prefix** is deterministically convertible to **a URI pattern** and vice-versa. For instance, the above URI could be represented as uniprot:A0A022YWF9. This deterministic conversion makes it easy for meta resolvers as well, e.g., http://identifiers.org/uniprot:A0A022YWF9.

Suboptimal identifier practice is artificially constraining what can and cannot be done with the underlying data: it not only hampers adherence to FAIR principles (findability, accessibility, interoperability, and reuse)[9], but also compromises mechanisms for credit and attribution. This article seeks to provide pragmatic guidance and examples for how actors in life science research lifecycle should handle identifiers. Optimizing web-based persistent identifiers is harder than it appears; there are a number of approaches that may be used for this purpose, but no single one is perfect: Identifiers are reused in different ways for different reasons, by different consumers. Moreover, digital entities (e.g., files, such as an article), physical entities (e.g., tissue specimens), living entities (e.g. Dolly the sheep), and descriptive entities (e.g., 'mitosis') have different requirements for identifiers[10].

The problem of identifier management is hardly unique to the life sciences; it afflicts every discipline from astronomy[3] to law[11]. Towards this end, several groups (**Supplemental Text S1**) have been converging on identifier standards that are broadly applicable [9,12–14]. Building on these efforts and drawing on our experience in integrating and accessing data from a large number of sources, we outline the identifier qualities and best practices we consider particularly important in the context of large-scale data integration in the life sciences. In **Lessons 1-9** (**Table 1**) we propose actions for data providers when designing new identifiers, maintaining existing identifiers, as well as when reusing and referencing identifiers from other datasets. In **Lesson 10**, we conclude with guidance for data integrators and redistributors on how best to reference multiple identifiers from diverse sources. More often than not, life science data providers often invent or organically grow their own identifier systems without a firm grasp of the lasting implications. Data providers are urged to take a long-term view of the scope and lifecycle of data and the identifiers that they issue, and to consider using existing identifier platforms and services [13] where appropriate.

Throughout this document, the word “must” is reserved for practices that ensure against the collision, ambiguity, or inaccessibility of items referenced by identifiers; instances of “must” are also often specific to particular design choices. We use the word “should” to convey that the tradeoffs must be understood and carefully weighed before choosing a different course (eg. consistent with IETF RFC2119 [15]). Terms that appear in `fixed-width font` are defined in the supplemental glossary (**Supplemental Table S2**).

There is no one in science that is unaffected by identifiers. **Table 1** details three basic roles one might play in the scholarly landscape and how identifiers are relevant in these contexts. Who are designers and creators? These are databases, but also those that submit supplemental data to archives, and anyone creating structured data. Who are the providers and maintainers? These are databases as well, but also services and indices that support web resolution and data validation. Who are the reusers and referencers? These are the “Research Data Parasites”[16], but also your average author: while authors may specify an identifier for a resource (e.g. a gene or antibody), more often identifiers are contextually inferred by the journals or curators, whether pre- or post-publication.

Legend
◒ Indirectly relevant for
● Directly relevant for

Lesson	Types of actors: Designers & creators	Providers & mainteners	Reusers & referencers
Lesson 1. Credit any derived content using its original identifier	●	◒	●
Lesson 2. Help local identifiers travel well: document prefix and patterns	●	●	●
Lesson 3. Opt for simple, durable web resolution	●	◒	◒
Lesson 4. Avoid embedding meaning, or relying on it for uniqueness	●	●	◒
Lesson 5. Design new identifiers for diverse uses by others	●	●	◒
Lesson 6. Implement a version-management policy	●	●	◒
Lesson 7. Do not reassign or delete identifiers	●	●	◒
Lesson 8. Make URIs clear and findable	●	●	◒
Lesson 9. Document the identifiers you issue and use	●	●	●
Lesson 10. Reference and display responsibly	●	●	●

Table 1. A summary of the 10 recommendations, their direct or indirect impact on different kinds of identifier actions.

Many of the following recommendations are applicable during the planning and identifier conceptualization phase, i.e. before any identifiers are created. The retrofitting (especially Lessons 1, 4, 5, and 6) of existing identifiers can sometimes be too difficult or may even make matters worse: for instance changing existing identifiers introduces the need for systems that can recognize the variations for what they are; such overhead can outweigh potential benefits. Each of the lessons is relevant to the basic classes of identifier actions (design, provision, reuse **Table 1**) within the ecosystem of data providers and integrators. These actions in turn are relevant to anyone on the spectrum of seven basic roles ranging from those that publish their own data to those that provide applications on top of others’ data (**Figure 1**). Even if we largely agree on what makes for a good persistent identifier (**Table 2**), actual implementation often falls short. No provider is perfect and no two are alike, hence the objective is to learn from each other’s diverse experiences. All of the negative examples herein are anonymized variations of real-world identifiers that we have had to work with.

Lesson 1. Credit any derived content using its original identifier

If you manage an online `database` (`repository`, `registry`, or `knowledgebase`), consider its role in identifying and referencing the knowledge that it publishes. We advise that you only create your own identifiers for new knowledge (**Figure 1**). Wherever you are referring to existing knowledge, do so using existing identifiers (Lesson 10): otherwise, wherever the 1:1 relationship of identifier:entity breaks down, costly mapping problems arise. Whether or not you create a new ID, it is vital to credit any derived content using its `indigenous identifiers` [10]; to facilitate data integration, all such identifiers should be machine processable and transparently mapped.

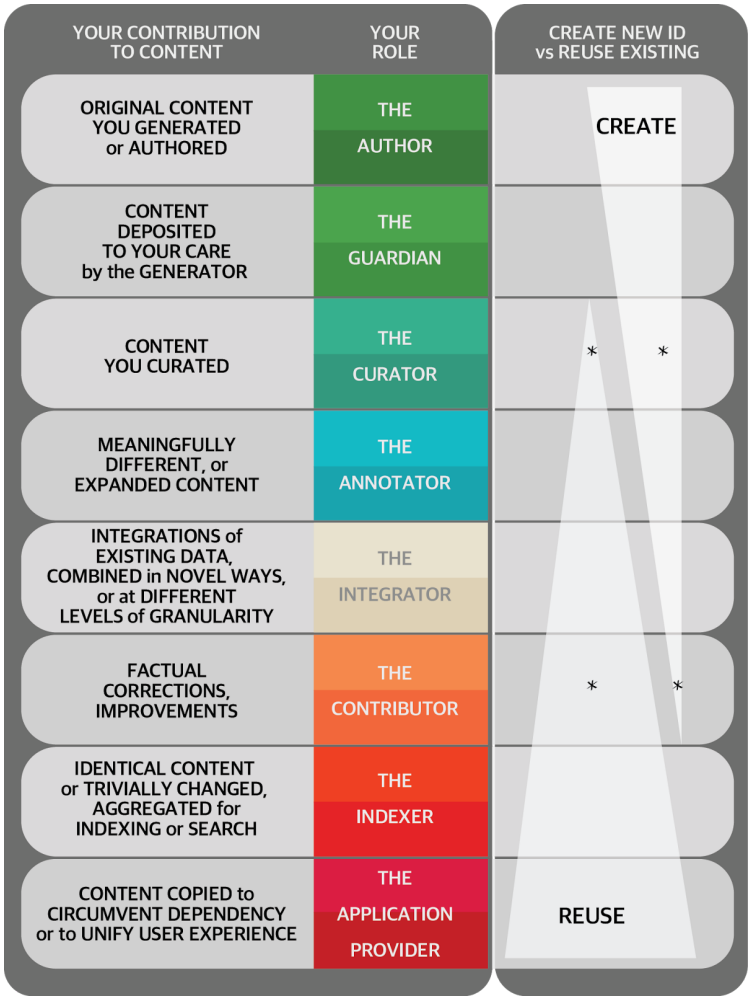


Figure 1. Contributions and roles related to content as they correspond to identifier creation vs reuse.
*The decision about whether to create a new identifier, or reuse an existing one depends on the role you play in the creation, editing, and republishing of content; for certain roles (and when several roles apply) that decision is a judgement call. Asterisks convey cases in which the best course of action is often to correct/improve the original record in collaboration with the original source; the guidance about ID creation versus reuse is meant to apply only when such collaboration is **not** practicable (and an alternate record is created). It is common that a given actor may have multiple roles along this spectrum; for instance, a given record in monarchinitiative.org may reflect a combination of a) corrections Monarch staff made in collaboration with the original data source, b) post-ingest curation by Monarch staff, b) expanded content integrated from multiple sources.*

Lesson 2. Help local identifiers travel well: document prefix and patterns

If you reference others' data, or anticipate your data *being referenced by others*, consider how you document your identifiers. Note that you may not know *a priori* how your data may be used. Data does not thrive in silos: it is most useful when reused, broken into parts and integrated with other data, for instance in database cross references ("`db xrefs`"). In spite of how important identifiers are to this process, the confusion with identifiers often starts with the basics, including what the "`identifier`" even is. A `local ID` (**Box 1**) is an identifier guaranteed only to be unique in a given local context (eg. a single provider, a single collection, etc.), and sometimes only within a specific version; as such, it is poorly suited to facilitate data integration because it can collide when considered in a more global landscape of many such identifiers. For instance, the `local ID` "9606" corresponds to numerous entities whose local accessions are based on simple digits, including: a Pubmed article, a CGNC gene, a PubChem chemical, as well as an NCBI taxon, a BOLD taxon, and a GRIN taxon. `Local IDs` therefore need to be contextualized in order to be understood and accessed (resolved) on

the web. This is often accomplished through the use of a `prefix`, which should be documented. If this is overwhelming, don't forget that there are meta resolvers and services built to help for exactly this reason (see **Lesson 3**).

`Uniform Resource Identifiers (URIs)` are identifiers that resolve on the web. "Cool URIs don't change" [6] because when, they *do change* (or disappear) all existing references break. In the context of academia alone, "`reference rot`" problem impacts one in five publications [4]. Despite `link rot` vulnerability, the global `http/s URI` (**Box 1**) is the best available identifier form for machine-driven global data integration because a) the `http URI` is a widely adopted IETF standard and b) the `http URI's` uniqueness is ensured by a single well-established name-granting process (DNS). However, the length of `URIs` can make them unwieldy for tasks involving human readability even within structured machine-parsable documents. `Compact URIs` (`CURIEs` [17], **Box 1**) are a mature W3C standard that is well established in some contexts (e.g. `JSON-LD` and `RDFa`) as they enable `URIs` to be understood and conveniently expressed. We the authors are not absolutist about anyone using CURIEs; however, we agree that the features that make for good URIs also happen to make CURIEs possible (for those who wish to use them) (Supplementary Text S3).

Thus if you are a database provider, it is in your best interests to document and preferably register a) the `prefix` (**Box 1**) that you would like others to use and b) its binding to a `URI pattern` (**Box 1**). Your chosen prefix should be unique, at least among datasets that are likely to be used in the same context. **Supplementary Table S4** contains a list of registries that may be suitable depending on the kind of data. PrefixCommons [18] is a platform designed to enable such registries to make more informed decisions about which prefixes to issue and utilized and for any given integrator to publish the mappings that they happen to use.

Table 2. Desirable characteristics for database identifiers in the life sciences

Characteristics	Definition	General rationale/impact on data integration	Specific example of a possible ramification due to non-adherence
Unambiguous	One **`Local ID`** must be associated to no more than one entity *locally.* One `URI` must be associated to no more than one entity *globally*	Avoids collisions that result in integrating on the wrong entity	A physician makes a wrongful diagnosis
Unique	One entity should ideally be identified by no more than one URI	1) Eliminates the cost of maintaining public mappings between equivalent identifiers 2) Avoids false negatives if data integrators do not leverage or know about a mapping	A researcher fails to make a pathway discovery because she does not realize that http://mydb.org/1234567 and http://mydb.org/q?=1234567 are in fact the same.
Stable (identifier)	The `URI, and by extension the local ID` should, wherever possible stay the same over time	Avoids `link rot`	A researcher is unable to reproduce an experiment because the link to a record is dead.
Stable (entity)	Identifier must NOT be reassigned to an altogether different entity, though the original entity may evolve provided a change history is documented	Avoids integrating on the wrong entity	A chemist uses the wrong chemical in a reaction.
Version-documented	If the entity's definition or essential metadata changes substantially, (Lesson 7) the `identifier` should, wherever possible be versioned and/or change history documented	Avoids integrating on the wrong entity state (specified through version)	A given experiment is not reproducible because the specific build version of a gene sequence was not specified.
Persistent	The identifier must NOT be deleted (but may be deprecated)	Avoids `link rot`	Information about a gene model is completely lost

Web-resolvable	The `URI` must be resolvable to a web address where the data or information about the entry can be accessed	Avoids the unnecessary proliferation of resolvable identifiers issued by third parties (for entities that are not resolvable and/or not identified in their native context) See also `surrogate identifier.`	A dozen different third party providers mint identifiers for entities that are not actually under their control. Harmonization between these off-brand identifiers is painful.
Convertible	The **`local ID`** and its `URI` counterpart must be inter-convertible by applying the **`URI pattern`** to the **`local ID.`** Note that in some communities (eg. ontologies), the `local ID` is often a `CURIE` by default.	Avoids the need for special handling of edge cases when integrating data at scale	Data integrators spend time cleaning identifiers and handling edge-cases instead of doing science.
Defined	The total set of assignable identifiers for the `database` must be describable through a formal pattern (regular expression)	Facilitates validation and extraction from scientific text, thus the pattern should be as tightly specified as possible (see Lesson 3)	Identifiers can not be validated and a provider may find it hard to assess their impact in the literature.
Web-friendly	The **`local ID`** should wherever possible be of a format that does not need special handling when used in `URLs` and common exchange formats (e.g. XML)	Avoids potential points of failure due to malformed URL, XML, etc.	Use of the identifier produces malformed XML and / or requires special detection and encoding.
Free to assign	The `identifier` should ideally be assigned at no cost to individuals depositing data in a repository	Lowers barriers for data generators to deposit data	Data generators become reluctant to deposit data in order to minimize costs.
Open access and use	The `identifier` and its `label` should be able to be transparently referenced and actioned (e.g. in a public index or search) anywhere by anyone and for any reason. Restrictions on associated data may apply but are not recommended.	Enables integration on the basis of scientific merit, rather than on the restrictions of the license	When there are license restrictions on the identifier and/or label (not just the content) it thwarts meaningful reuse and redistribution of whole datasets.
Documented	The identifier scheme should be documented	Encourages consistent use of existing identifiers by others and reduces the number of ways identifiers are represented.	Inconsistent informal approaches to referencing are difficult to harmonize post-hoc. By extension, impact is harder to assess.

Lesson 3. Opt for simple, durable web resolution

A core component of persistent identification is redirection, the absence of which makes it extremely difficult to provide stable identifiers. When designing (or refining) your http URI strategy:

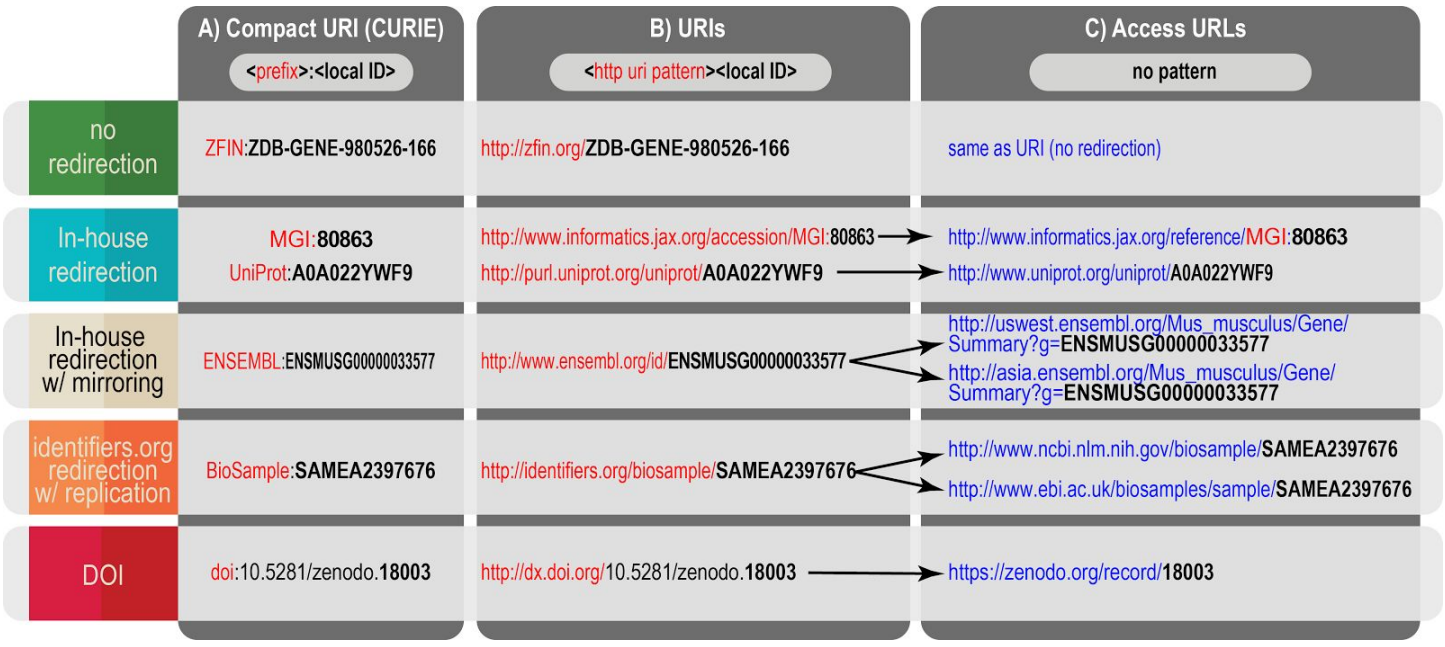
- **Consider a resolution provider before doing it yourself.** If you are a database provider, you must implement an `http URI pattern` (**Figure 1 panel B**) for **`local IDs`** to be resolvable to a web page. If you choose to outsource to a resolver service, use an approach that adheres to best practice[13] (e.g. DOI (DataCite, CrossRef), Identifiers.org, Handle.net, PURL (now via InternetArchive), EPIC, ARK) and be mindful of your constraints regarding cost, metadata ownership, turnaround time, etc. (See **Supplemental Text S5** for a more comprehensive list of considerations.) Some of these resolver services can even provide `content negotiation` for different encodings of your data[13] and make it easier to provide direct access to data, metadata, and persistence statements[19]. If you have the resources to support your own persistent URIs, design these to be “cool”[6]; this is most easily achieved by keeping URIs simple.
- **Avoid inclusion of anything that is likely to change or lapse**, including administrative details (e.g. grant name) or implementation details such as file extensions (‘`resource.html`’), query strings

('`param=value`'), and technology choices ('`.php`'). Never embed the **`local ID`** in the query part of a URI eg. `http://example.com/explore?record=A123456`.

- **Omit trailing characters after the local ID.** In all cases, the `URI pattern` must include the protocol (e.g. https://) and, if applicable, trailing slash or other delimiters. Trailing characters after the `local ID` are discouraged as they unnecessarily increase the variability with which the identifier is represented and also complicate straightforward appending of the `local ID` (requiring that tokens such as `$id` hold the place of the **`local ID`** in the URI pattern eg `http://example.com/$id/view.do`).
- **Avoid unnecessary detail.** Detail in 'persistent' identifiers creates complexity that must be managed in perpetuity. Make every attempt to limit the degree of path nestedness (eg. do `http://example.com/A123456` rather than `http://example.com/vertebrates/mammals/rodents/rat/white-rat/A123456`); see also Lesson 5 regarding types and meaning. The compact URI approach can work with any resolver(s): see for instance examples 4 and 5 in **Figure 2**. By choosing a single `URI pattern`, you make it possible for others to resolve your identifiers simply (**Figure 2 panel A**) without their having to know the type and its syntax in http URI. See also **Lesson 4** regarding omission of semantics.

Despite their differences, the examples in **Figure 2** share the most important features above.

Figure 2. Examples of provisioning resolvable URIs:
*Compact URIs (CURIEs) (Panel A), URIs (Panel B) and `Access URLs` (Panel C) with no redirection (ZFIN), in house redirection (UniProt, and Ensembl), and 3rd party resolvers (using identifiers.org and DOI). In each case, the URI can be algorithmically derived from the CURIE because the **`Local ID`** portion itself is included (unmodified) within the URI. `Access URL` design patterns differ substantially by provider and may change over time. As long as `access URLs (and other ephemeral links)` are not used as the referenced identifier, they can include prefix and colon (MGI) or not (Ensembl), they may include the entire `Local ID` (Biosample) or not (DOI), and they may include type (MGI) or not (ZFIN).*



Lesson 4. Avoid embedding meaning, or relying on it for uniqueness

When designing new `local IDs` or `http URIs`, avoid embedding meaning or relying on it for uniqueness. The structure and scope of collections evolve, as does scientific understanding; minimizing the meaning embedded in identifiers makes them less vulnerable to obsoletion. In human genetics many genes were initially identified based on disease association; later the identification, nomenclature, and function of genes were separated into different activities. Meaning should only be embedded if it is indisputable, unchangeable

and also useful to the data consumer (e.g. computer-processable). For instance, the type of entity imparts meaning to users and may fulfill these three criteria. When encountered, typing may be embedded, either within the **local ID** (ENSMUSG...), or within the http URI path (.../gene/12345), or both. In any case, if you opt to include type in the identifiers you issue, avoid relying on type for uniqueness: that is to say once a **local ID** eg. 12345 is assigned, it should never be recycled for another entity, even an entity of a different type for instance .../gene/12345 and .../patient/12345.

If you need the ability to convey meaning in a dense character space, you don't need to do so in the identifier itself; consider instead implementing an entity label, for instance as is done in model organism nomenclature such as by Mouse Genome Informatics (label: Kit^{W}/Kit$^{W\text{-}v}$, id: MGI:2171276). Labels are for human readability only; even if they are deemed durable, labels should not be treated as identifiers, nor should they appear within http URIs. URI patterns, if type-specific, require a corresponding type-specific prefix (e.g. for the Library of Integrated Network-based Cellular Signatures (LINCS), the prefix 'LINCS-cell' corresponds to http://lincs.hms.harvard.edu/db/cells/$id/ whereas the prefix 'LINCS-protein' corresponds to http://lincs.hms.harvard.edu/db/proteins/$id/). MGI implements both type-agnostic resolution (http://www.informatics.jax.org/accession/MGI:2442292) and type-specific destinations (http://www.informatics.jax.org/marker/MGI:2442292). Dual approaches like MGI's can be helpful to different kinds of consumers: type-agnostic resolution is useful in cases such as data citation in the literature where a) the type of the identified entity is not of primary importance, or b) the type of the entity is already conveyed contextually, and/or c) where resolution is done systematically at scale and/or involves many and varied or volunteer contributors that may be difficult to coordinate. Type-specific resolution is useful in cases like bioinformatic research pipelines where embedded type may facilitate the human-led debugging process. If you support both kinds of resolution, it is best to document a) whether you intend for both to be treated as persistent b) what mapping support you provide.

Whether or not your URIs or your local IDs include type, you should provide other ways for humans and machines to determine the type of entity that is being identified; this is most often achieved via webservices (eg. as done via Monarch API), but ideally also within metadata landing pages [19,20] if provided.

Lesson 5. Design new identifiers for diverse uses by others

Pre-existing identifiers should be referenced without modifications (see **Lesson 10**). However, if you create new **local IDs**, there are some design decisions that can facilitate their use in diverse contexts (spreadsheets, other databases, web applications, publications, etc.).

- **Avoid problematic characters.** Local IDs should, wherever possible comprise only letters, numbers and URL-safe delimiters. Omission of other special characters guards against corruption and mistranscription in many contexts; however, it is acceptable that the local ID be in CURIE format since modern browsers resolve colons without having to encode them. Although characters "/" and "?" are technically URL-safe, they are very problematic when used *within* the local ID as these characters are assumed to have special meaning and can complicate parsing of the identifiers, whatever forms they take. For the same reason, local IDs should ideally not contain '.' except to denote version where appropriate (see **Lesson 7**).
- **Define a formal pattern and stick to it.** Local IDs must adhere to a formal pattern (regular expression); this facilitates the validation of URIs and improves the accuracy of mining identifiers from scientific text. Consider a fixed length of 8-16 characters (according to the anticipated number of required **local IDs**). A pattern may be extended if all available identifiers are issued, but existing identifiers should not be changed. To minimize **local ID** collisions at global scale, it is considerate to tightly specify your pattern (e.g. using one or more fixed letters). The regular expression should include a fixed, documented case convention. In most cases, it is advised that identifiers not rely on case for their uniqueness: if you assign ab-12345 to one entity and AB-12345 to *a different* entity, collisions due to mistranscription are more likely. Case-sensitive patterns are best reserved for when brevity is a constraint (e.g. millions of IDs are required and each ID has to be short enough to be printed on a vial label).

- **Avoid problematic patterns.** Consider using both letters and numbers in the `local ID`. This avoids misinterpretation as numeric data (e.g. truncation of leading zeros or conversion to exponents in spreadsheets). Some patterns can result in misinterpretation/corruption whether as dates (e.g. “may-15”), exponents (e.g. “5e1234”)[21], or as unintended words (e.g. “bad-12”). Such issues in gene names alone have been shown to impact 19% of life sciences papers [22]. A historically common, if thorny, identifier pattern is that ‘_’ and ‘:’ are often interconverted and it has come to be understood as compact notation, delimiting the `prefix` from the rest of the identifier. Therefore ‘_’ or ‘:’ should a) occur no more than once per identifier and b) should only be used if **`local ID`** are intended to be deterministically expanded to a resolvable http URI. For instance, if your intended `prefix` is ‘MyDB’, then either `MyDB:gene-6622` or `MyDB_gene-6622` are acceptable patterns, but `MyDB_gene_6622` is problematic as it could result in three possible conversions by others, even if these are not intended: `MyDB_gene:6622`, `MyDB:gene_6622`, `MyDB:gene:6622`. Whatever pattern you adopt, document which variations you support resolution of, if any.

Lesson 6. Implement a version-management policy

Whether you produce original data, or reference others’ data, consider the impact of changes. The nature, extent, and speed of data changes impact how data can be referenced and used. Document your chosen version management practice: If you issue identifiers, the change history for the entity should be either documented or queryable. Alternatively, the identifier itself can be versioned whether or not change history is also supported.

Explicit identifier versioning is recommended if the prevailing use of an *unversioned* identifier results in “breaking changes” (e.g., a change in the hypothesized cause of a disease). However, if new information about the entity emerges slowly and the changes are “non-breaking”, it is reasonable to instead maintain a machine-actionable change history wherein the changes are listed, and where they may also be categorized (eg. minor versus major changes). Versioning and change history work well together, especially when multiple types of changes overlap. Even where previous records are entirely removed, the URI should continue to resolve, but to a **“**`tombstone`**”** `page` (**Lesson 7**). A resource should communicate clearly what a version change refers to. UniProt and RefSeq use versions to reflect changes in sequence. Ensembl uses versions to reflect changes in sequence and splicing for transcript records but sequence alone for protein records. In each of these examples changes in annotation attached to a record does not alter the version.

There are two approaches to versioning, record-level and release-level; the latter is more common in the life sciences. Release-level versioning is usually performed for defined data releases. However, use cases vary; some user communities need to resolve individual archived entities via a deterministically-versioned URI pattern, for example as is done in Ensembl eg. http://e85.ensembl.org/id/ENSMUSG00000033577. In either case, whether or not you have the ability (or common use case) to maintain individually resolvable archived records, we strongly recommend supporting export to files so that users can archive the records they need. We also recommend making snapshots available for the database, whether in whole or in parts.[23]

If you version identifiers at the level of the individual record, you should version in the **`local ID`** after the ‘dot’, as per UniProt in **Table 3**; this provides continuity in your site and also enables a single `prefix` to be used with any version: UniProt:P12345.3 → http://www.uniprot.org/uniprot/P12345**.3.** If you do record-level versioning but dot suffixing is not practicable, we strongly recommend providing a transparent mapping between identifiers together with a mechanism for obtaining the latest version of the record (e.g. by inserting `/latest/` in the URI path). Maintaining mappings between identifier versions without use of the dot is possible, but so difficult that few providers do it well. Other groups have discussed change management consideration and `‘content drift’` in more depth [2,24,25].

Table 3. Recommendation for versioning

	Recommendation	UniProt	RefSeq	Ensembl
General versioning practices	Primary versioning strategy	Record level	Record level	Release level
	Past versions are accessible	All versions of individual records are accessible http://www.uniprot.org/uniprot/P12345?version=* http://www.ebi.ac.uk/uniprot/unisave/app/#/	All versions of individual records are accessible https://www.ncbi.nlm.nih.gov/nuccore/NM_004333.4?report=girevhist	Maintains all archives for at least five years; some key releases may be maintained for longer. All databases maintained for at least 10 years (currently all databases available from 2004) http://www.ensembl.org/info/website/archives/index.html
	Release versioning available	ftp.ebi.ac.uk/pub/databases/uniprot/previous_releases	No past releases available	ftp.ensembl.org/pub and archive sites
	Documentation exists regarding what kinds of record changes prompt a new version to be issued.	http://www.uniprot.org/help/entry_history http://www.uniprot.org/help/uniprotkb http://www.uniprot.org/help/fasta-headers	https://www.ncbi.nlm.nih.gov/books/NBK50679/#RefSeqFAQ.what_causes_the_version_number	http://www.ensembl.org/info/genome/stable_ids/index.html
URL versions	The base identifier (the one with no explicit version) should resolve (302 redirect) to most recent version	http://www.uniprot.org/uniprot/P12345	https://www.ncbi.nlm.nih.gov/nuccore/NM_004333	http://ensembl.org/id/ENSMUSG00000033577
	`Base identifier` should be deterministically convertible from any other version	Remove dot suffix from the Local ID eg: http://www.uniprot.org/uniprot/P12345.**1** to http://www.uniprot.org/uniprot/P12345	Remove dot suffix from the Local ID eg: https://www.ncbi.nlm.nih.gov/nuccore/NM_004333.4 to https://www.ncbi.nlm.nih.gov/nuccore/NM_004333	Remove build number from the URI, eg: http://e85.ensembl.org/id/ENSMUSG00000033577 to http://ensembl.org/id/ENSMUSG00000033577
	Older versions must resolve	http://www.uniprot.org/uniprot/P12345.**1**	https://www.ncbi.nlm.nih.gov/nuccore/NM_004333.1	http://e85.ensembl.org/id/ENSMUSG00000033577
	Illegal or invalid version should produce an informative http error code and a HTML page explaining the error.	http://www.uniprot.org/uniprot/P12345.302 returns a 400 bad request and brief description	https://www.ncbi.nlm.nih.gov/nuccore/NM_004333.302 returns a 404 page not found	Error not returned
	A list of all previous versions should be available	See 'history' tab in user interface	See format dropdown in user interface	http://www.ensembl.org/info/website/archives/assembly.html
	Link from older version to current version should ideally be provided	P12345.**3**	Link available at the top of the page	Plans to support
	Two versions (or dates) should ideally be comparable	Record history provides comparison	Record history provides comparison	Unsupported

Lesson 7. Do not reassign or delete identifiers

Identifiers that you have exposed publicly, whether as http URIs or via APIs may be deprecated but must never be deleted or reassigned to another record. If you issue identifiers, consider their full lifecycle: there is a

fundamental difference between identifiers which point to experimental datasets (GenBank/ENA/DDBJ, PRIDE, etc.) and identifiers which point to a current understanding of a biological concept (Ensembl Gene, UniProt record, etc.). While experimental records are less likely to change, concept descriptions may evolve rapidly; even the nature and number of the relevant metadata fields change over time. Moreover, the very notion of identity is often strongly impacted by relationships (e.g., between concepts or processes).

Extensive changes cannot be captured with numerical suffixing alone. For instance, taxonomists may split or merge species, pathologists may split or merge diseases, or hypothesized entities may be proven not to exist (e.g. vaccine-induced autism). Global initiatives (**Supplemental Text S1**) are actively exploring identifier strategies for such use cases. In the meantime, consider **Table 4** recommendations.

Table 4. Recommendations for identifier lifecycle management

Recommended handling	Example
Obsoletion: If an entry has been removed or deprecated, the original identifier must still resolve to a `'tombstone page'`. Reasons for obsolescence should be indicated. If the obsoleted ID is replaced by another ID, the replacement must be present and also described as automatic or suggested, preferably using the ontology properties `iao:replaced_by` and `obo:consider`, respectively. The obsoleted ID must never be reassigned to another entity. A list of obsoleted IDs should be maintained.	Single obsoleted identifier: http://www.uniprot.org/uniprot/A0AV18 List of obsoleted identifiers: uniprot.org/help/deleted_accessions
Merging: When two or more identifiers are merged, a new recipient identifier should be designated as the primary (citable) one and should contain information about the legacy identifiers it encompasses. Any legacy identifiers should continue to resolve via redirection to the primary identifier.	UniProt entries Q57339 and O08022 have been merged into Q00626. Q57339 and O08022 are redirected to Q00626.
Splitting: If an identifier is split (demerged) into two or more new ones, new identifiers should be assigned to all the new entries. The legacy identifier must be marked as obsolete, but must also still resolve, providing a warning and pointers to the new ones as per above.	UniProt entry P29358 has been split into P68250 and P68251. P29358 displays a warning and links to the demerged entries: http://www.uniprot.org/uniprot/P29358

Lesson 8. Make URIs clear and findable

Persistent URIs almost always differ from the ephemeral URLs to which users are ultimately directed (Figure 2). Therefore, whether you produce original data, or reference others' data, make persistent URIs obvious to users so that they are less inclined to ABC (Address Bar Copy). As a group, the best practitioners of this lesson are currently academic journals; they prominently advertise the DOI corresponding to each article. In situations where the version of a data record matters, advertise the corresponding "permanent link" (`permalink`) together with a statement about persistence. E.g.

> *"The permanent link to this page, which will not change with the next release of Ensembl is: http://e85.ensembl.org/id/ENSMUSG00000033577 We aim to maintain all archives for at least five years; some key releases may be maintained for longer"*

For archived records that are *out of date,* make this clear to the user and provide a link to the updated version (see `http://www.uniprot.org/uniprot/P12345.1`, for instance). Although it is good practice for each database website to include general citation guidance for users [26], it is increasingly important to provide a pre-populated citation *at the level of each record*. When it comes to making record-level citation clear on every page, eagle-i[27] provides the best example of a primary data source that we know of (outside of providers that issue DOIs) **(Figure 3)**. Additional features that are useful in such widgets are that full references should be copy-pastable, integrated with reference managers, and pre-populated with the version information and access date.

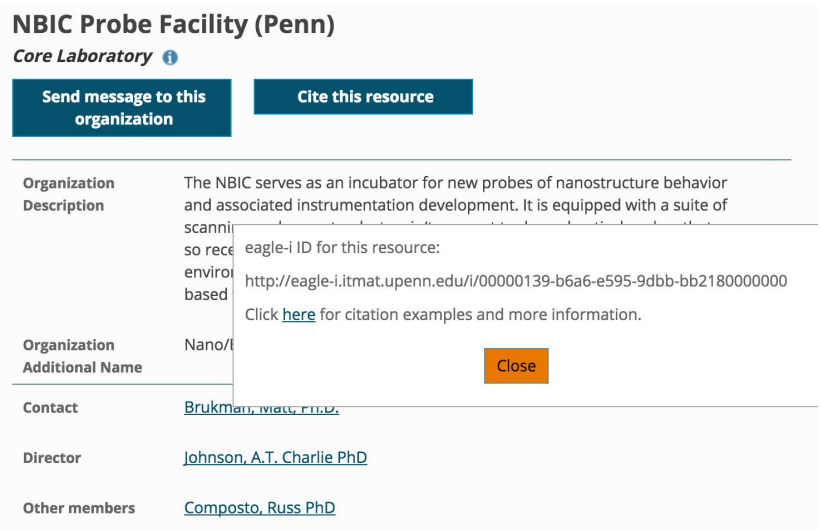


Figure 3. eagle-i record-level citation widget

Lesson 9. Document the identifiers you issue and use

The global-scale identification cycle is a shared responsibility and provider/consumer roles often overlap in the context of data integration. Whether you issue your own identifiers or just reference those of others, you should document your identifier policies. **Supplemental Table S6** provides a set of questions that data providers and re-distributors can use to develop such documentation. Documentation should be published alongside and/or included together in a dataset description, for instance, as outlined in the recommendations for Dataset Descriptions developed by the W3C Semantic Web in the Health Care and Life Sciences Interest Group [28]. For examples of such documentation see ChEMBL[29] and Monarch[30]; the format may vary.

Lesson 10. Reference and display responsibly

The final lesson describes referencing recommendations for data redistributors: data aggregators, who collect information from different sources and re-display it; data publishers, who disseminate scientific knowledge through publications; and online reference material such as WikiData[31].

When external entities are referenced in narrative online text, they should be hyperlinked to their URIs or to pages/metadata containing their URIs. `Access URLs` are volatile (see **Lesson 4**) and must not be used for referencing or linking in any context intended to persist.

Broader issues associated with citation of data and software in the traditional literature are outside of the scope of this paper, but **Text S1** lists relevant complementary efforts. Our recommendations regarding data citation in the literature are circumscribed: within static documents of record (eg. in PDFs), or in situations where link updates are costly/difficult, we strongly advocate always using the URLs of well-established third-party resolvers, whether they be primary resolvers such as doi.org or hdl.net or meta-resolvers such as identifiers.org, or n2t.net (Supplemental **Text S4**). Each provider has a corresponding URI pattern; however, those URIs can and do change over time. Third-party resolvers are not immune to change; the fact that the PURL.org resolver recently nearly sunset into “read-only” mode illustrates a) the importance of sustained community buy-in and governance and b) that reliance on 3rd parties for resolution is not without its risks. Nevertheless, the risk that URIs will break because of resolver change is modest and easier to mitigate compared to the risk that any *single* referenced collection will move or disappear. It is incumbent on meta-resolvers to be vigilant about detecting and updating their redirection rules in the face of provider changes. identifiers.org is able to redirect to one of a few potential provider destinations based on an algorithm that considers a) provider uptime, b) whether a given provider is a ‘primary’ source of the data in that collection. N2T.net and Identifiers.org recently joined forces[32] to harmonize identifiers in the same way, using the same prefixes. As part of this partnership, they have both have adopted simple syntax that gives users

finer grained control, to request to be directed to a specific source of the data; for instance specifying the primary source of the data whether or not it has the best record of up-time.

Redistributors of data should monitor their references to other sources; any 'dead' links should be reported to the original data provider. If the original provider does not fix the broken link, your reference to it should be marked obsolete both visibly (for user interaction/interpretation), and within any accompanying metadata (for computational interaction/propagation). Differentiate identifiers linked internally within your application from identifiers linked outside your application; one way to do this is by using the linkout icon; consider opening all external links in a new browser window or tab in order to avoid confusion.

Conclusion

Better identifier design, provisioning, documentation, and referencing can address many of the identifier problems encountered in the life science data cycle - leading to more efficient and effective science. However, it is well established that just because it is broadly agreed that a practice would be beneficial to the community, does not mean that it is adopted; to have an impact, the adoption of best practice has to be both easy *and* rewarding. In the broader context of scholarly publishing, this is just what DOIs afford; DOIs succeeded because they were well aligned with journals' business goals (tracking citations) and because the cost was worth it to them. However, in the current world where everyone is a data provider, alignment with business goals is still being explored: meta resolvers can provide a use case for journals and websites seeking easier access to content, while software applications leverage these identifier links to mine for knowledge.

We recognize that improvements to the quality, diversity, and uptake of identifier tooling would lower barriers to adoption of the lessons presented here (**Text S7**). Those that issue data identifiers face different challenges than do those referencing data identifiers; we understand there are ecosystem-wide challenges that need will undertake to address these gaps in the relevant initiatives (**Text S1**). We also recognize the need for formal software-engineering specifications of identifier formats and/or alignment between existing specifications. Here, we implore all participants in the scholarly ecosystem - authors, data creators, data integrators, publishers, software developers, resolvers - to aid in the dream of identifier harmony and hope that this paper can catalyze such efforts.

Acknowledgments

JA McMurry, T Burdett, N Juty, S Jupp, and C Morris were supported in part by the BioMedBridges project, which is funded by the European Union Seventh Framework Programme within Research Infrastructures of the FP7 Capacities Specific Programme, grant agreement number 284209. EMBL-EBI core funds supported H Parkinson, MJ Martin, J McEntyre, H Hermjakob, J Malone, M Courtot. ELIXIR core funds supported N Blomberg, R Jimenez. The European Commission provided additional support for Simon Jupp under grant number 601043 ("DIACHRON") and for N Juty and H Hermjakob under grant number 312455 ("Infrastructure for Systems Biology - Europe (ISBE)"). The Drug Disease Model Resources grant number DDMoRe 115156 ("Innovative Medicines Initiative") supported C. Laibe. Support was also received from the following BBSRC grants: BB/L005050/1 ("ELIXIR-UK, Manchester") for SA Sansone, A Gonzalez-Beltran and C Goble; BB/M013189/1 ("DMMCore") for C Goble, J Snoep, and N Stanford; BB/K019783/1 ("Continued development of ChEBI") and BB/M006891/1 ("EMPATHY") for N Swainston; BB/M017702/1 ("SYNBIOCHEM") for N Swainson and D Fellows; BBS/E/B/000C0419 ("A systems approach to understanding lipid, Ca2+ and MAPK signalling networks") for N Le Novère; BB/L005069/1 ("ELIXIR-UK, Oxford") for SA Sansone, A Gonzalez-Beltran and P Rocca-Serra. NIH support was provided from the following grants: U41HG007822 ("UniProt") for MJ Martin; U24AI117966-01 ("bioCADDIE") for SA Sansone, A Gonzalez-Beltran and P Rocca-Serra; U54AI117925 ("CEDAR") for M Dumontier, SA Sansone, A Gonzalez-Beltran and P Rocca-Serra; R24OD011883 ("Monarch Initiative") for CJ Mungall, MA Haendel, JA McMurry and NL

Washington; NHGRI P41HG002273-09 ("Gene Ontology Consortium") for CJ Mungall. Additional support for CJ Mungall and NL Washington was received from the Director, Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under [Contract No. DE-AC02-05CH11231].

The authors wish to thank Mary Todd Bergman, Ewan Birney, Fiona Cunningham, Richard Cyganiak, Adam Faulconbridge, Andrew M Jenkinson, Sirarat Sarntivijai, Stephanie Suhr, Eleanor Williams, Martin Fenner, and Tim Clark for their valuable feedback and suggestions. We also wish to thank the BioMedBridges Scientific Advisory Board for the suggestion to address this important issue and the reviewers for their constructive comments.

References

1. Pitcher L. Writing Ancient History: An Introduction to Classical Historiography. I.B.Tauris; 2010.
2. Sanderson R, Phillips M, Van de Sompel H. Analyzing the Persistence of Referenced Web Resources with Memento [Internet]. arXiv [cs.DL]. 2011. Available: http://arxiv.org/abs/1105.3459
3. Pepe A, Goodman A, Muench A, Crosas M, Erdmann C. How do astronomers share data? Reliability and persistence of datasets linked in AAS publications and a qualitative study of data practices among US astronomers. PLoS One. 2014;9: e104798.
4. Klein M, Van de Sompel H, Sanderson R, Shankar H, Balakireva L, Zhou K, et al. Scholarly context not found: one in five articles suffers from reference rot. PLoS One. 2014;9: e115253.
5. Bugeja MJ, Dimitrova DV. Vanishing Act: The Erosion of Online Footnotes and Implications for Scholarship in the Digital Age. 2010.
6. Berners-Lee T. Cool URIs don't change, 1998. URL: http://www w3 org/Provider/Style/URI (25 07 2009). 2009;
7. Wikipedia contributors. Persistent identifier. In: Wikipedia, The Free Encyclopedia [Internet]. 8 Oct 2016 [cited 13 Feb 2017]. Available: https://en.wikipedia.org/w/index.php?title=Persistent_identifier&oldid=743128540
8. Berners-Lee T. Uniform Resource Locators: A unifying syntax for the expression of names and addresses of objects on the network. 1993; Available: https://www.w3.org/Addressing/URL/uri-spec.html
9. The FAIR Data Principles. In: FORCE11 [Internet]. 3 Sep 2014 [cited 8 Feb 2017]. Available: https://www.force11.org/group/fairgroup/fairprinciples
10. Guralnick RP, Cellinese N, Deck J, Pyle RL, Kunze J, Penev L, et al. Community next steps for making globally unique identifiers work for biocollections data. Zookeys. 2015; 133–154.
11. Zittrain J, Albert K, Lessig L. Perma: Scoping and addressing the problem of link and reference rot in legal citations. Legal Information Management. Cambridge Univ Press; 2014;14: 88–99.
12. Altman M, Crosas M. The evolution of data citation: From principles to implementation. IASSIST Q. 2013;37. Available: http://www.iassistdata.org/iq/evolution-data-citation-principles-implementation
13. Starr J, Castro E, Crosas M, Dumontier M, Downs RR, Duerr R, et al. Achieving human and machine accessibility of cited data in scholarly publications. PeerJ Comput Sci. 2015;1. doi:10.7717/peerj-cs.1
14. Bandrowski A, Brush M, Grethe JS, Haendel MA, Kennedy DN, Hill S, et al. The Resource Identification Initiative: A cultural shift in publishing. F1000Res. 2015;4: 134.
15. Bradner S. Key words for use in RFCs to Indicate Requirement Levels. Harvard; 1997 Mar.
16. Emmert-Streib F, Dehmer M, Yli-Harja O. Against Dataism and for Data Sharing of Big Biomedical and Clinical Data with Research Parasites. Front Genet. Frontiers; 2016;7. doi:10.3389/fgene.2016.00154
17. Birbeck M, McCarron S. CURIE Syntax 1.0. W3C Candidate Recommendation CR-curie-20090116. 2009; Available:

https://www.w3.org/TR/curie/

18. prefixcommons. prefixcommons/biocontext. In: GitHub [Internet]. [cited 8 Feb 2017]. Available: https://github.com/prefixcommons/biocontext

19. CDL. ARK Specification. http://www.cdlib.org/services/uc3/arkspec.pdf: California Digital Library; 2008 May.

20. Joint Declaration of Data Citation Principles - FINAL. In: FORCE11 [Internet]. 30 Oct 2013 [cited 8 Feb 2017]. Available: https://www.force11.org/group/joint-declaration-data-citation-principles-final

21. Zeeberg BR, Riss J, Kane DW, Bussey KJ, Uchio E, Linehan WM, et al. Mistaken identifiers: gene name errors can be introduced inadvertently when using Excel in bioinformatics. BMC Bioinformatics. 2004;5: 80.

22. Ziemann M, Eren Y, El-Osta A. Gene name errors are widespread in the scientific literature. Genome Biol. 2016;17: 177.

23. Monarch Data Release Archive. In: Monarch Initiative [Internet]. 20 Feb 2017. Available: https://archive.monarchinitiative.org

24. Kratz J, Strasser C. Data publication consensus and controversies. F1000Res. 2014;3: 94.

25. Van de Sompel H, Sanderson R, Shankar H, Klein M. Persistent Identifiers for Scholarly Assets and the Web: The Need for an Unambiguous Mapping. International Journal of Digital Curation. 2014;9: 331–342.

26. Haendel M. OBOFoundry Citation Policy. In: http://www.obofoundry.org [Internet]. [cited 15 Feb 2017]. Available: http://www.obofoundry.org/docs/Citation.html

27. Vasilevsky N, Johnson T, Corday K, Torniai C, Brush M, Segerdell E, et al. Research resources: curating the new eagle-i discovery system. Database . 2012;2012: bar067.

28. W3C. Dataset Descriptions: HCLS Community Profile [Internet]. 2015. Available: https://htmlpreview.github.io/?https://github.com/indiedotkim/HCLSDatasetDescriptions/blob/master/Overview.html#s6_3

29. WC3 Interest Group. Complete Example of a Dataset Description [Internet]. 14 May, 2015. Available: http://www.w3.org/TR/hcls-dataset/#appendix_1

30. McMurry J, Washington N, Shefcheck K, Conlin T. DIPPER: The Monarch Data Ingest Pipeline Identifier Documentation [Internet]. 2015. Available: https://github.com/monarch-initiative/dipper/blob/master/README.md#identifiers

31. Wikidata [Internet]. [cited 8 Feb 2017]. Available: https://www.wikidata.org/wiki/Wikidata:Main_Page

32. Wimalaratne S, Juty N, Kunze J, Janée G, McMurry JA, Beard N, et al. Uniform Resolution of Compact Identifiers for Biomedical Data [Internet]. bioRxiv. 2017. p. 101279. doi:10.1101/101279

33. BD2K Home Page | Data Science at NIH [Internet]. [cited 6 Mar 2017]. Available: http://bd2k.nih.gov/

34. Home | BioMedBridges [Internet]. [cited 6 Mar 2017]. Available: http://www.biomedbridges.eu/

35. DataCite Team. Welcome to DataCite [Internet]. [cited 6 Mar 2017]. Available: https://www.datacite.org

36. Data Citation Implementation Pilot (DCIP). In: FORCE11 [Internet]. 28 Sep 2015 [cited 6 Mar 2017]. Available: https://www.force11.org/group/dcip

37. My Site [Internet]. [cited 6 Mar 2017]. Available: http://www.diachron-fp7.eu/

38. ELIXIR. ELIXIR Data for life [Internet]. [cited 6 Mar 2017]. Available: http://elixir-europe.org/

39. FORCE11. In: FORCE11 [Internet]. [cited 6 Mar 2017]. Available: https://www.force11.org/

40. Welcome to Monarch [Internet]. [cited 6 Mar 2017]. Available: http://monarchinitiative.org/

41. RDA | Research Data Sharing without barriers [Internet]. [cited 6 Mar 2017]. Available: https://rd-alliance.org/

42. Scott Marshall M, Stephens S. Semantic Web Health Care and Life Sciences (HCLS) Interest Group [Internet]. [cited 6 Mar 2017]. Available: http://www.w3.org/blog/hcls/

43. Wg OT. The OBO Foundry [Internet]. [cited 6 Mar 2017]. Available: http://obofoundry.org

44. Hanedel Ma Mungall C. Identifier Citation Policy [Internet]. 2015 [cited 6 Mar 2017]. Available: http://obofoundry.org/id-policy.html

45. Home | Global Alliance for Genomics and Health [Internet]. [cited 6 Mar 2017]. Available: http://genomicsandhealth.org

46. Mietchen D, McEntyre J, Beck J, Maloney C, Force11 Data Citation Implementation Group. Adapting JATS to support data citation. National Center for Biotechnology Information (US); 2015.

47. Journal Article Tag Suite [Internet]. [cited 6 Mar 2017]. Available: https://jats.nlm.nih.gov/

48. Wikipedia contributors. ASCII. In: Wikipedia, The Free Encyclopedia [Internet]. 3 Mar 2017 [cited 6 Mar 2017]. Available: http://en.wikipedia.org/w/index.php?title=ASCII&oldid=768432447

49. Wikipedia contributors. Content negotiation. In: Wikipedia, The Free Encyclopedia [Internet]. 26 Feb 2017 [cited 6 Mar 2017]. Available: http://en.wikipedia.org/w/index.php?title=Content_negotiation&oldid=767497641

50. Berrueta D, Fernández S, Frade I. Cooking HTTP content negotiation with Vapour. of 4th Workshop on Scripting for the …. Citeseer; 2008; Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.142.8172&rep=rep1&type=pdf

51. Wikipedia contributors. Domain Name System. In: Wikipedia, The Free Encyclopedia [Internet]. 4 Mar 2017 [cited 6 Mar 2017]. Available: https://en.wikipedia.org/w/index.php?title=Domain_Name_System&oldid=768585053

52. Leach PJ, Berners-Lee T, Mogul JC, Masinter L, Fielding RT, Gettys J. Hypertext Transfer Protocol--HTTP/1.1. 1999; Available: https://tools.ietf.org/html/rfc2616

53. JSON-LD 1.0 [Internet]. [cited 6 Mar 2017]. Available: https://www.w3.org/TR/json-ld/

54. Wikipedia contributors. Link rot. In: Wikipedia, The Free Encyclopedia [Internet]. 9 Feb 2017 [cited 6 Mar 2017]. Available: https://en.wikipedia.org/w/index.php?title=Link_rot&oldid=764507867

55. Wikipedia contributors. Permalink. In: Wikipedia, The Free Encyclopedia [Internet]. 15 Sep 2016 [cited 6 Mar 2017]. Available: https://en.wikipedia.org/w/index.php?title=Permalink&oldid=739572533

56. EMBL-EBI. Biosamples < EMBL-EBI [Internet]. [cited 6 Mar 2017]. Available: https://www.ebi.ac.uk/biosamples

57. Wikipedia contributors. Uniform Resource Identifier. In: Wikipedia, The Free Encyclopedia [Internet]. 6 Mar 2017 [cited 6 Mar 2017]. Available: https://en.wikipedia.org/w/index.php?title=Uniform_Resource_Identifier&oldid=768870095

58. The Antibody Registry [Internet]. [cited 6 Mar 2017]. Available: http://antibodyregistry.org/

59. Frontiers | An antibody registry for biological sciences [Internet]. [cited 17 Feb 2017]. doi:10.3389/conf.fninf.2011.08.00067

60. Wikipedia contributors. Web resource. In: Wikipedia, The Free Encyclopedia [Internet]. 15 Jan 2017 [cited 6 Mar 2017]. Available: http://en.wikipedia.org/w/index.php?title=Web_resource&oldid=760120815

61. Wikipedia contributors. URL normalization. In: Wikipedia, The Free Encyclopedia [Internet]. 13 Jan 2017 [cited 14 Feb 2017]. Available: https://en.wikipedia.org/w/index.php?title=URL_normalization&oldid=759921577

62. Juty N, Le Novère N, Laibe C. Identifiers.org and MIRIAM Registry: community resources to provide persistent identification. Nucleic Acids Res. 2012;40: D580–6.

63. Welcome - the Datahub [Internet]. [cited 6 Mar 2017]. Available: http://datahub.io

64. Cyganiak R, Jentzsch A. Linking open data cloud diagram. LOD Community (http://lod-cloud net/). 2011;12. Available: http://lod-cloud.net/

65. Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nat Biotechnol. 2007;25: 1251–1255.

66. Whetzel PL, Noy NF, Shah NH, Alexander PR, Nyulas C, Tudorache T, et al. BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. Nucleic Acids Res. 2011;39: W541–5.

67. namespace lookup for RDF developers | prefix.cc [Internet]. [cited 6 Mar 2017]. Available: http://prefix.cc

68. Vandenbussche P-Y, Atemezing GA, Poveda-Villalón M, Vatant B. Linked Open Vocabularies (LOV): a gateway to reusable semantic vocabularies on the Web. Semantic Web. IOS Press; 2017;8: 437–452.

69. Joint Declaration of Data Citation Principles - FINAL. In: FORCE11 [Internet]. 30 Oct 2013 [cited 6 Mar 2017]. Available: https://www.force11.org/datacitation

70. Crossref team. Metadata enables connections - Crossref. In: www.crossref.org [Internet]. [cited 6 Mar 2017]. Available: http://www.crossref.org/

71. EMBL-EBI. Identifiers.org < EMBL-EBI [Internet]. [cited 6 Mar 2017]. Available: http://identifiers.org/

72. Handle.Net Registry [Internet]. [cited 6 Mar 2017]. Available: http://handle.net/

73. PURL Administration [Internet]. [cited 6 Mar 2017]. Available: https://archive.org/services/purl/

74. Persistent Identifiers for eResearch [Internet]. [cited 6 Mar 2017]. Available: http://www.pidconsortium.eu/

75. Name-to-Thing (N2T) Identifier Resolver Home Page [Internet]. [cited 6 Mar 2017]. Available: http://n2t.net/

76. BioModels.net team. MIRIAM Registry [Internet]. [cited 6 Mar 2017]. Available: http://identifiers.org/nbn/

77. Persistent Identifiers for eResearch [Internet]. [cited 6 Mar 2017]. Available: http://www.pidconsortium.eu/

78. w3id.org - Permanent Identifiers for the Web [Internet]. [cited 6 Mar 2017]. Available: https://w3id.org/

79. Internet Archive: Digital Library of Free Books, Movies, Music & Wayback Machine [Internet]. [cited 6 Mar 2017]. Available: https://archive.org/

80. Graham M. Persistent URL Service, purl.org, Now Run by the Internet Archive | Internet Archive Blogs [Internet]. [cited 6 Mar 2017]. Available: https://blog.archive.org/2016/09/27/persistent-url-service-purl-org-now-run-by-the-internet-archive/

81. EBI Web Team. Protein Identifier Cross-Reference Service [Internet]. [cited 8 Feb 2017]. Available: http://www.ebi.ac.uk/Tools/picr/RESTDocumentation.do

82. Journal Article Tag Suite [Internet]. [cited 8 Feb 2017]. Available: https://jats.nlm.nih.gov/

83. BioSchemas [Internet]. [cited 8 Feb 2017]. Available: http://bioschemas.org/

Supporting information

Supplementary Text S1. Initiatives relevant to identifiers

- **BD2K (Big Data 2 Knowledge)**[33]**.** This US program supports a variety of initiatives aimed at making better use of the diversity of biomedical data, including various data integration efforts.
- **BioMedBridges**[34]. This is an implementation-driven project to integrate data that facilitates translational research[34].
- **DataCite**[35]**:** DataCite is interested in enabling the persistent identification of data, and develops and supports the standards required to achieve this[35].
- **DCIP**[36]**:** The Data Citation Implementation Pilot goal is to provide basic coordination between publishers, repositories and identifier / metadata services for early adopters of data citation according to the JDDCP[36].
- **Diachron**[37]: DIACHRON intends to address and cope with certain issues arising from the evolution of and identification of data in a web environment.
- **ELIXIR**[38]: A pan-European research infrastructure tasked with safeguarding and managing biological data.
- **Force11**[39]: This international pan-disciplinary organization is a forum for innovations in scholarly communication, including citation of data, research resources, and other web artifacts such as software.
- **Monarch Initiative**[40]**:** A global consortium dedicated to integrating cross-species genotype-phenotype data for disease discovery.
- **RDA**[41]**:** The Research Data Alliance is a globally active alliance interested in achieving the open sharing of data across countries, technologies and research domains.
- **W3C HCLS**[42]: The World Wide Web Healthcare and Life Sciences Interest group aims to develop semantic standards for interoperability.
- **OBO Foundry**[43]**:** The OBO Foundry consortium is a collaborative of ontology developers adhering to common best parctices and shared principles to ensure interoperability, including a common identifier and citation policy [44].
- **GA4GH**[45]**:** The members of the Global Alliance for Genomics and Health work towards integrating and analysing genomic data.
- **JATS**[46]**:** The Journal Article Tag Suite is an application of NISO Z39.96-2015, which defines a set of XML elements and attributes for tagging journal articles and describes three article models. JATS is a continuation of the NLM Archiving and Interchange DTD work begun in 2002 by NCBI[47]. It can also be used to cite data in journals.

Supplementary Table S2. Glossary of web technology terms

Term	Definition
Access URL	The URL of the page to which the http URI is ultimately redirected. Such a page is often referred to as a `landing page` (see below). While experts may differ about what is and is not a landing page, the most important definition.
Alternate identifier	A 3rd-party-issued identifier that refers to an entity that already has its own (`indigenous`) identifier. See also `Surrogate identifier`.
ASCII	ASCII is a 8-bit character encoding, the first 7-bits define a stable set containing 128 characters[48]. It contains the numbers from 0-9, the uppercase and lowercase English letters from A to Z, and some special characters. UTF-8 keeps the first 7-bits of ASCII as is and includes non-ASCII characters that may be used in Internationalized Resource Identifiers (IRI), however since non-ASCII characters are not allowed in URIs, ASCII is the least problematic choice.
Base identifier	An identifier that intentionally has no version information embedded. For databases that have an entity-level versioning policy, the “base identifier” would have no versioning embedded local part (through dot suffixing); eg, in UniProt, a base identifier would be http://www.uniprot.org/uniprot/P12345 and a corresponding versioned identifier http://www.uniprot.org/uniprot/P12345.1. For databases that have release-level versioning, the base base resource http://ensembl.org/id/ENSMUSG00000033577 would be http://e85.ensembl.org/id/ENSMUSG00000033577. The base identifier serves two purposes. 1) redirection to the most current version, and 2) convenience of omitting version if that level of detail is not important for a particular use case.
content drift	“The resource identified by a URI may change over time and hence, the content at the end of the URI may evolve, even to such an extent that it ceases to be representative of the content that was originally referenced.”[4]
Content negotiation	“Content negotiation is a mechanism defined in the HTTP specification that makes it possible to serve different versions of a document (or more generally, a resource representation) at the same URI, so that user agents can specify which version fit their capabilities the best.”[49] [50]
Cross reference	A reference to an `entity` in a 3rd party `database, repository, registry, or ontology`; classical cross references are not accompanied by recapitulated data from the native `entity`.
CURIE prefix (see also prefixed URI)	• deterministically expandable to a URI pattern (see below) which is the basis for the CURIE’s global uniqueness • a mnemonic that helps in human communication • documented and aspirationally globally unique • documented in terms of its case convention • conforms to the rules of an XML QName (e.g. does not contain ‘:’)
Domain Name System (DNS)	The Domain Name System (DNS) is a hierarchical distributed naming system for computers, services, or any resource connected to the Internet or a private network. It associates various information with domain names assigned to each of the participating entities.[51]
Entity	An identifiable unit, for instance in a database, registry, repository, or ontology. Entities can be of different types: Digital entities include files, images, video, etc. Physical entities include things like preserved specimens, individual living specimens, and strains or lines of living specimens.
HTTP Status codes	When a web resource is requested, the response falls into one of five high-level categories, or “HTTP status codes”: 1) informational, 2) success, 3) redirection, 4) client error, 5) server error. For instance, a ‘302 redirect’ means that the resource has moved temporarily; ‘301 redirect’ means the

	move is permanent. This distinction enables search engines to keep the old page, or replace it with the one at the new location.[52]
JSON-LD	JSON-LD, or JavaScript Object Notation for Linked Data, is a method of encoding Linked Data using JSON. It was a goal to require as little effort as possible from developers to transform their existing JSON to JSON-LD. This allows data to be serialized in a way that is similar to traditional JSON. [53]
label	A human-readable version of a resource's name. Labels should be displayed where human comprehension is important, but labels should be backed by identifiers. In the context of the Semantic Web, labels are often instances of rdf:Property.
landing page	The JDDCP recommends that citations be human *and* machine readable. It's very hard to ensure that all machines (or people) are ready to consume, interpret or access the data. A landing page provides any additional information that is required for these points. A landing page also can serve as the intermediary for complex data packages, e.g., .zip, .tar, gz, to provide a unique point of access. Landing pages should ensure that both the metadata and the data are "Machine accessible", i.e., that the landing page provides access by well-documented Web services to data and metadata stored in a robust repository, independent of browser access by humans. Specific recommendations for how to achieve these goals may be found in Starr et al (2015)[13]. The DCIP Expert Group on Repository Metadata will also be issuing a set of guidelines in the near future. The URL that corresponds to the landing page is an "`access URL`".
Local Identifier (Local ID)	An identifier that is only guaranteed to be unique within a single database. (See Box 1 and Figure 1). While the concept has historical precedent, we are introducing the term itself for the first time here. In prior versions of this paper, we referred to it as LRI (Local Resource Identifier).
link rot	"Link rot (or linkrot), also known as link death, link breaking or reference rot, refers to the process by which hyperlinks on individual websites or the Internet in general point to web pages, servers or other resources that have become permanently unavailable."[54]
Persistent identifier	The term '**`persistent identifier`**' is usually used in the context of digital objects that are accessible over the Internet. Typically, such an identifier is not only persistent but also actionable[7]: it is a **`Uniform Resource Identifier (URI)`**`[8]`, usually of type http/s, that you can paste in a web browser address bar and be taken to the identified source.
Permalink	A permalink or permanent link 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 friendly URLs, so as to be easy for people to type and remember. Most modern blogging and content-syndication software systems support such links. Sometimes URL shortening is used to create them. A permalink is a type of `persistent identifier` and the word permalink is sometimes used as a synonym of persistent identifier. More often, though, permalink is applied to persistent identifiers which are generated by a content management system for pages served by that system. This usage is especially common in the blogosphere. Such links are not maintained by an outside authority, and their persistence is dependent on the durability of the content management system itself. [55]
Indigenous identifier (aka native identifier)	An identifier issued by an entity's original or authoritative source (eg. original `database, repository, or registry`), also referred to as native identifier[10]. See also `surrogate identifier` and `alternate identifier`.
prefixed URI or "CURIE" (see also CURIE prefix)	A compact URI comprised of `<Prefix>:<Local ID>` wherein **`prefix`** is deterministically expandable to **`a URI pattern`** to yield the http URI which *alone* is the basis for the CURIE's global uniqueness. An example of a CURIE is `UniProtKB:A0A022YWF9.` Occasionally, the CURIE is the ID form that is actually used locally (see MGI, figure 2) and thus functions as a Local ID. [17]

Registries and Repositories	Databases may be classified as registries, repositories, both or neither. A registry is an indexed list of entities with pointers to their external locations. A repository internally stores the actual entities and assumes primary responsibility for them. Knowledge bases synthesize information from diverse sources. In practice, most databases combine features of the these three categories and can be differently classified depending on the entity in question: for instance, BioSamples DB[56] is a *repository* of BioSample information but a *registry* of the experimental data associated with those samples.
URI	An identifier that is guaranteed to be both uniform and globally unique. In this paper, we define a URI as an ASCII string that uniquely identifies a Web (not localhost) resource and also resolves to (provides or redirects to) a webpage containing information about the identified entity. Such URIs are generally of the HTTP protocol but may be other (e.g. HTTPS). Although according to their original specification, URIs may either be of type URN or URL, common usage of the term 'URI' almost always means those of type URL only. We have further distinguished between a URIs and 'access URLs', not because their anatomy or technical specification differs, but because their purpose differs. URIs may and should be used for identification purposes because they are designed to be persistent. Access URLs on the other hand are ephemeral and should therefore not be used for identification purposes. It can be difficult or impossible for a user to determine whether a given URL is an access URL or a URI. In native resolution (**ZFIN, Fig. 1a**), access URL and URI are exactly the same; this approach reduces the likelihood that an ephemeral address will be used for identification purposes. Providers that choose redirection strategies (**Fig 1b-1e**) for their URIs must be vigilant about documentation for users. [57]
URI pattern	A URI pattern (sometimes referred to as a "resolving namespace" a fixed sequence of characters that can be used to resolve a database's local IDs. In this paper, we mean "URI pattern" to mean the simplest scenario wherein the pattern can be prepended to the local ID (or to the part of the CURIE that follows the colon, if different). See **Fig. 2 for examples**. In all cases, the **URI pattern** must be exactly as it appears in the URI: it must include the protocol (e.g. http://) and, if applicable, trailing slash or other delimiters. Some providers require additional characters *after* the Local ID is appended; this should be strongly avoided as it requires the URI patterns to contain tokens that are replaced eg. example.org/$id/view; token replacement works fine in custom code but is not supported in normal contexts such as JSON-LD, XML etc. The combination of documented URI patterns and local ID regular expressions makes it possible for consumers/integrators to validate any referenced http URIs they happen to be using.
Surrogate identifier	A 3rd-party-issued identifier that refers to an entity that does not already have its own (native) identifier. See also Alternative identifier. Surrogate identifiers are most often issued when the identifier that is needed by third parties is more granular, or less granular than the one provided by the native source. For instance, many antibody manufacturers have an online catalogs with a single PDF containing hundreds of antibodies each with a catalog number. However, it is rare for the manufacturers to provide corresponding webpages for each product. Thus in order to ensure that the identifiers are uniquely referenceable and resolvable to a webpage, the http://antibodyregistry.org/ created surrogate URIs containing the local catalog numbers as advertised by the manufacturer[58]. This was done so that antibodies could be more reliably referenced in the literature and their usage better tracked.[59]
Tombstone page	A page which continues to resolve after the corresponding entity has been deleted. It should provide the reason that the object was deleted and some basic metadata about the object
Web Resource	"Every 'thing' or entity that can be identified, named, addressed or handled, in any way whatsoever, on the web at large, or in any networked information system." [60]
XRef	Also known as "external reference" or "cross reference", XRefs are references from one database to a record in another database.

Supplementary Text S3. Utility of CURIEs

The features that make for a good persistent URI also make for good CURIEs: desirable features include lack of semantics in both the `URI pattern` and the `local ID` **(Lesson 4)**, absence of characters after the local ID (**Lesson 5**), omission of problematic characters etc **(Lesson 5)**. `CURIEs` can complement http URIs in important ways for curators and data integrators:

A. **Brevity.** In the life sciences, prefixed identifier forms are traditionally favored over http URIs in curation tasks; for instance, within spreadsheets, online lab notebooks, and anywhere where identification is a core concern but where screen real estate is limited.
B. **Location-independence.** Third-party data integrators often add knowledge on top of existing identifiers, for instance as MonarchInitiative.org does with OMIM. But if Monarch's URI's instead included the embedded http URI of the OMIM source dataset it would look like https://monarchinitiative.org/uri=http://omim.org/entry/154700 instead of like https://monarchinitiative.org/OMIM:154700. If the OMIM ID were not converted to its CURIE form, the resulting URI in Monarch would be a) very long b) permanently vulnerable to any volatility in the original source URI. Encoding the prefix mappings for the sources dynamically provides both simplicity and
C. **Clues for collapsing equivalents.** Due to a lack of awareness and to evolving implementations and collection scope, it is exceptionally rare that only a single http URI is used for an entity. Although it is difficult to reliably 'normalize' equivalent URIs[61] that are syntactically different, the use of CURIEs can provide clues that facilitate it.

Supplementary Table S4. Prefix and URI pattern registries

Table S4a. Formal registries

Prefix registry	Scope	Registration URL	Note	Registers Native Prefix	Registers URI pattern	Functions as a resolver
Identifiers.org[62]	Life sciences	https://sourceforge.net/p/identifiers-org/new-collection/	Manually curated. Core OBO foundry namespaces are imported periodically.	yes	yes	yes
n2t.net[32]	Cross-domain Name-to-Thing resolver	http://n2t.net	Supports a combination of per-identifier and per-scheme (rule) redirects.	yes	yes	yes
Data Hub[63]	Cross-domain	http://datahub.io	Datasets may be uploaded or registered for free. The Data Hub registry is used to populate the linked open data cloud.[64]	yes	yes	no
OBO foundry[65]	Bio-Ontologies	http://www.obofoundry.org/join.shtml	Each ontology in OBO requires an "ID space" which is unique across all ontologies in OBO. Not all ontologies are eligible for inclusion.	yes ("ID space")	no	yes, OBO PURL only
Bioportal[66]	Bio-Ontologies	http://bioportal.bioontology.org/login?redirect=/ontologies/new	Each ontology in BioPortal requires a "Short ID" which is unique across all ontologies in BioPortal.	yes ("Short ID")	no	yes, Bioportal PURL only
Prefix.cc[67]	Cross-domain	http://prefix.cc/	Designed for Semantic Web practitioners. Accepts, short, memorable prefixes only, punctuation not allowed. Prefix assignments are ranked according to community voting.	yes	yes	no
Linked Open Vocabularies[68]	Cross-domain	http://lov.okfn.org/dataset/lov/	Designed for Semantic Web practitioners. Vocabularies relevant to linked data.	yes	yes	no

Table S4b. Hybrid registries

There are other databases and web applications that leverage/mirror the `prefixes/uri patterns` served from the above registries, but that do not register any new ones themselves. This hybrid approach requires post-hoc coordination. The ongoing coordination and aggregation of information from various prefixing authorities is important to further minimize collisions.

Prefix registry	Scope	Registration URL	Note	Registers Native Prefix	Registers URI pattern	Functions as a resolver
Prefix Commons Biocontext	Primarily lifesciences	https://github.com/prefixcommons/biocontext/blob/master/README.md	Enables any registry or integrator to declare the mappings they issue and happen to use			no
Ontology Lookup Service	Ontologies used by EMBL-EBI	n/a	Scope is for ontologies used in molecular biology	no	no	yes
BioSharing	Life sciences - policies, standards and databases	https://www.biosharing.org/new/	Manually curated crowd-sourcing approach. Periodically synchronized with other sources such as Identifiers.org. Each BioSharing record is registered with a short ID, which is unique across all of BioSharing.	yes	no	yes, BioSharing PURL only
Gene Ontology Prefix Registry	Identifiers that use GO or that are used by GO	https://github.com/geneontology/go-site/blob/master/metadata/db-xrefs.yaml (click on the 'edit' icon)	Manually curated YAML, managed in github. Pull requests accepted. Periodically manually synchronized with other sources such as Identifiers.org	yes	yes	Yes (in a limited context, eg. Amigo)

Supplementary Text S5. Things to consider when choosing a resolver approach

There are basically three kinds of approaches to serving URIs on the web: (a) “native” URIs that require no redirection at all (as in Fig. 1, ZFIN). (b) “in house” URIs that redirect internally (as in Fig. 1, Ensembl); and (c) schemes using an external resolving authority (as in Fig. 1, Biosamples). Representative resolver authorities that meet the JDDCP (https://www.force11.org/datacitation, [69]) criteria are e.g. DOI (DataCite[35], CrossRef[70]), Identifiers.org[71], Handle.net[72], PURL[73], EPIC[74], N2T[75] and NBN[76]; these are described in Starr et al[13]. Additional resolver authorities that meet the criteria but which are not described therein are EPIC[77] and w3id[78]. Note that PURLs under the authority of PURL.org had gone into read-only mode and were therefore no longer adherent to the JDDCP principles; however, the InternetArchive[79] has assumed responsibility for them as of September 2016[80].

Below are some additional criteria you may want to consider in choosing one of these resolvers.

- Does the resolver retain the native Local Resource Identifier that you issue (eg. identifiers.org, n2t.net), or does it instead issue a new one? (eg. DOI).
    - If the resolver *does* issue a new identifier, what is the typical turnaround time between request and fulfilment? Can you obtain an identifier before you yourself need to use it?
- Would you or your institution need to pay fixed/variable costs to have your identifiers resolved? If the service is free for those that need their identifiers resolved, who pays to maintain the service?
- Is the service capable of issuing and managing identifiers in the kinds of volume you would require?
- Change management policy
    - Will you need to change the data which is referenced by the URI, and if so does the resolving system under consideration permit such change?
    - Is the object to which the URI resolves allowed to be removed?
    - Does the resolver support numerical suffixing for versions of the LRI?
    - If new LRIs are issued for each version of an entity, how can versions be related to each other?
- Will you require the resolver to support multiple resolving locations (mirrors)?
- Does the resolver support content negotiation at resolver’s HTTP URI?
- Does the resolver collect, index, and/or curate metadata about individual entities?
    - If so, is the metadata that is collected relevant for the types of entities identified?
- Does the resolver collect, index, and/or curate metadata about collections of entities (e.g. whole databases)?
    - If so, is the metadata that is collected relevant for the type of collection?
- Does the resolver support controlled access for confidential data?
- Is the resolver cross-discipline?

Supplementary Table S6. Questions that good identifier documentation should answer

Scope	Question to answer	Recommendation
Provider	What types of entities are identified, what is the scope of these entities?*	Must include
Provider	What is your primary `URI pattern`, if only one exists? If multiple, equally-valid `URI patterns` co-exist, what are these? (e.g. INSDC.org has four such schemes as the entire dataset is fully represented by each of three authorities: NCBI, ENA, and DDBJ)	Must include
Provider	Are you aware of any alternate URIs (eg. different resolvers) that other groups use for your identifiers? (Even though alternates are not recommended for use, knowing what which URIs are equivalent facilitates data integration.)	Could include
Provider	What is the `prefix` you wish others to use if they reference your entities in an abbreviated way? If this `prefix` is registered, where? What is the compact URI you wish others to use?**	Must include
Provider	What is your persistence policy regarding maintenance of the URIs? What is your persistence policy regarding the corresponding entities and metadata?	Must include
Provider	Can machine-readable representations of your entities be accessed? If so, where and in what formats?	Must include
Provider	What is the regular expression of your `Local IDs` and URIs? What do your identifiers look like. If possible provide a strict pattern to describe these identifiers.	Must include
Provider	Are there relationships between your identifiers? Where are these described?*	Should include
Provider	Under what license are identifiers made available?	Should include
Provider	Does the lifecycle of the entities potentially include versioning, splitting, merging, or deprecation? How are these changes managed, communicated, and synchronized between those using that entity?*	Must include
Provider-Redistributor	Do you identify *entities* that are also identified by others? Who are these others? Where are these mappings found and who, if anyone, maintains them?	Strongly recommended
Provider-Redistributor	Do you reference *identifiers* that are issued by other authorities? If so, in what cases? How often are the identifiers synchronized?	Must include
Provider-Redistributor	If you reference *identifiers* that are issued by other authorities, what are the mappings used for `prefix-to-URI patterns`? What is the source of these mappings (e.g. manual or identifier service). Where can your mappings be found?	Must include

* Adapted from the Linked open data institute recommendations [LODI]

**If your Local IDs already have a colon, make it clear to users what your preferred corresponding compact URI syntax is. We recommend referencing the LRI as if it were already a compact URI. For instance, the case of GO:0007049, the prefix 'GO' can be expanded to http://purl.obolibrary.org/obo/GO_ and prepended to the numeric fragment to yield http://purl.obolibrary.org/obo/GO_0007049, in accordance with their documentation.

Supplementary Text S7. Current and future efforts that would help lower barriers to adoption

Current efforts

- **Registries, 3rd party resolvers:** A list of identifier resolvers and identifier registries is in **Supplemental Table S3**.
- **PICR** [81]**:** Protein Identifier Cross-Reference Service has a service that returns identifier mappings, optionally including deleted ones. PICR or a similar service could be developed to have broader scope.
- **HCLS** [28]**:** Health Care and Life Sciences dataset descriptions provide a standard representation of the original sources of data (and therefore identifiers) in any integrated dataset.
- **JATS** [82]**:** In the context of the literature, Journal Article Tag Suite provides a standard way for data citations to be represented in the literature, facilitating credit and reward mechanisms. However, outside of the literature, referencing and display is primarily an issue of increasing awareness.
- **BioSchemas.org**[83] is promoting more consistent adoption of schema.org markup in the life sciences. Markup can facilitate more transparent provenance and credit mechanisms of integrated data, as well as optimizing data for discovery by search engines, whether Google, or others.

Future efforts

- **Identifier validator:** Identifier designers could help data producers choose the design that best suits their particular use case, validators could determine whether an existing identifier is valid according to a published scheme.
- **Embeddable citation widgets or citation markup** could help providers display citation information, clearly and consistently.
- **Archiving services: For archival of content, client-facing services include the Memento web protocol**[2]. We authors of this paper are not aware of any existing platforms that providers can outsource their content archiving to, but such a service may be worthwhile. Another function for archival services is for maintaining a robust network of linked entities. In this case, full archival of content may not be needed. Rather, resolver and/or indexing services may provide “tombstone pages” with essential metadata so that these entities can still be resolved.