Appendix 1

Identification in the digital content network

This document follows the structure of the LCC **Principles of Identification** document and elaborates the recommendations made there.

The *indecs* principles of identification

The following existing four principles of identification are adopted from the indecs metadata framework[1] and endorsed by LCC:

- *The principle of Unique Identification:* Every entity should be uniquely identified within an identified namespace.
- *The principle of Functional Granularity:* It should be possible to identify an entity whenever it needs to be distinguished
- *The principle of Designated Authority:* The author of an item of metadata should be securely identified.
- *The principle of Appropriate Access:* Everyone requires access to the metadata on which they depend, and privacy and confidentiality for their own metadata from those who are not dependent on it

Prologue: Digital networks

Network technology

With the creation of the internet, and notably the WWW use of it from 1994, interactive applications changed dramatically over the last two decades. Major web applications emerged with significant scale increases in:

- The number of concurrent users and access points to data (via PCs on the web and later on mobile devices);
- The amount of data collected and processed, as it became easier and increasingly valuable to capture all kinds of data, including unstructured or semi-structured data;
- Cloud based services, outpacing relational database technology since relational databases are essentially architected to run on a single machine which is a mis-match for linked content, networked and fully cloud-based services[2].

The hypertext model that was selected for the Web, http/html, operates at the file level; so denoting a “document” not by a first class identifier but by a substitute (its address on a server) constrained the (web) identifier world to the use of the Domain Name System. DNS was designed for ease of redirection at the server level of IP addresses for delivery of packets of data; to get to some specific point within a file using http requires a further second mechanism dependent on the file address. This hard-wires intelligence into an identifier string in a single parent domain, with implications for persistence and relationships to other identifiers.

[1] http://www.doi.org/topics/indecs/indecs_framework_2000.pdf
[2] See e.g.”Why NoSQL: Three trends disrupting the database status quo” http://info.couchbase.com/WhyNoSQLWhitepaper.html

It was then recognised (drawing lessons from non-network identifiers such as the "information and documentation" identifiers of ISO TC46/SC9) that identifiers should be first class objects that is, have an identity independent of any other item, including any protocols used to resolve the identifier, and so be free to have relationships which could be dynamic, i.e. fully contextual. A piece of content could then be identified independently of the server where it was (currently) to be found – avoiding the most common cause of lack of persistence, the infamous "404 not found" or link rot.

The logical distinction of *reference* and *resolution* (i.e. a referent is not necessarily the result of resolution) was not always appreciated, which led to much confusion in early web discussions of naming and addressing (see also 1.3 above). DNS provides resolution of IP packet addresses (URL) but is not ideal from the point of view of managing names as references (URI, URN) or from the point of view of the levels of persistence, scalability and security required in content naming.

In the 1990's, recognition of these issues led to proposals[3] for managing access to digital information not via the addressing of the component bits (packets of data), but as *digital objects*, a data structure for identifying and organizing information for access over a communication network. Adding to this the concept of *stated operations* (that is, defined types of operations that may be performed on a digital object) allowed for the possibility of identified objects to be distributed across networks, available for multiple uses in various ways, whilst maintaining complete independence of the underlying physical packets and wiring. This is therefore a level of abstraction from the underlying digital network to an information network view.

Digital networks and information networks

At the level of commerce and intellectual analysis, we are concerned with referents of all forms, notably abstractions. But at the level of technology, we are concerned simply with digital bits and how these are processed. Therefore persistent identifiers whose referent is of any form (digital, physical or abstract) are used in architectures whose sole focus is on digital objects ("bags of bits") which are processed in the same way. The link between the two views is provided through abstraction: an identifier may have a referent in any form, yet use an underlying digital technology to process and deliver either (1) a digital object which is a representation of (some aspects of) a physical object – e.g. a painting as discussed above; or (2) a digital object which provides sufficient information for representation for the purposes of the application – e.g. a data page about a person or a work.

Fig 1a shows the well-known hourglass model of the internet[4] . This model views an IP address as the "hourglass waist" or central common switching point between the layers above and below. The architectural aim is to keep this waist as thin and efficient as possible and not burden or ossify it with application detail (in the layers above) or technical implementation (in the layers below)[5], thereby allowing a single mechanism as the internet[6] (the TCP/IP protocols)[7]. Fig 1b[8] is an analogous view

[3] Managing Access to Digital Information: Cross-Industry Working team, 1997: http://www.xiwt.org/documents/ManagAccess.html

[4] See http://everything2.com/title/Hourglass+model

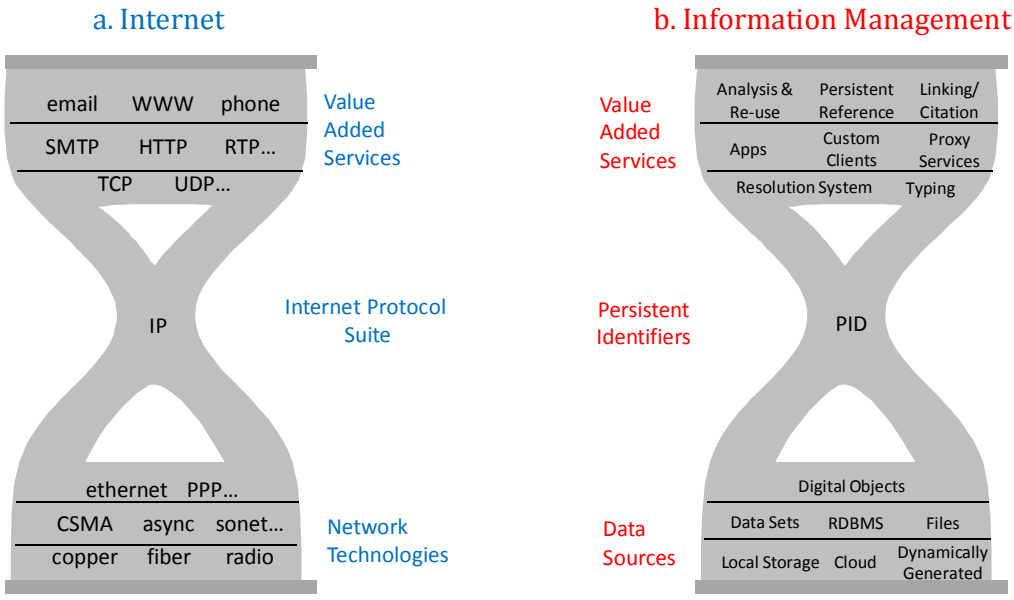
[5] Steve Deering: "Watching the Waist of the Protocol Hourglass": http://www.iab.org/wp-content/IAB-uploads/2011/03/hourglass-london-ietf.pdf

[6] "What Is The Internet (And What Makes It Work)": Dec 1999; Robert E. Kahn and Vinton G. Cerf http://www.cnri.reston.va.us/what_is_internet.html

[7] " "Internet" refers to the global information system that (i) is logically linked together by a globally unique address space based on the Internet Protocol (IP) or its subsequent extensions/follow-ons; (ii) is able to support communications using

from the perspective of information management rather than switching technology. This views persistent identifiers as the common interchange core, the waist layer of the hourglass, for value added services in the layers above and data sources in the layers below. Analogies may be drawn with the earlier hourglass model of the internet: there are benefits in keeping this waist slim, and not burdening identifier systems with data sources or added value services which are logically separable. In each case, some technologies may choose to bundle some of these layers into packages for ease of use, but this should not compromise the ability of others who choose not to do such bundling. In implementing services one can choose to place functions in various design components (e.g. the services managed in a resolution system versus those pointed at by the resolution system, or having services centralized or distributed). The hourglass is a useful reminder of the design principle of keeping core identification as simple as possible.

Fig 1: Hourglass Models



Corporation for National Research Initiatives

There is one notable difference between the two hourglass considerations. The waist of the hourglass in the "information management" model (persistent identifiers) is populated by a variety of identifier schemes from different sectors; whereas the waist in the "internet" version (IP) has only the single precise Internet Protocol specification. There have been attempts to define what level of interoperability or even fungibility might be possible among persistent identifier schemes to provide a single waistglass pinch point. However the requirements of interoperability required in

---

the Transmission Control Protocol/Internet Protocol (TCP/IP) suite or its subsequent extensions/follow-ons, and/or other IP-compatible protocols; and (iii) provides, uses or makes accessible, either publicly or privately, high level services layered on the communications and related infrastructure described herein." Resolution of US Federal Networking Council (cited in Kahn &Cerf op.cit.).

[8] The Information Management model was created by Larry Lannom, CNRI.

information management (not only syntactic but semantic and community considerations) make this a much harder problem than is usually appreciated.

LCC: Ten targets of a digital identifier network

In 2013 The Linked Content Coalition (LCC) project was formed to scope what was needed to enable the digital network to function effectively for those who are creating, publishing and using content under any business model (or none). In 2014, the LCC (now a permanent consortium of standards bodies from all content sectors) published the following:

The effective operation of the digital content market relies needs the establishment of a global **identifier network** in which parties, creations, rights and usages are identified and linked in the internet in a way that enables the automated discovery of rightsholdings, and the licensing and reporting of usage.

The Linked Content Coalition has identified what it understands to be the essential elements of this network, and sets out below **ten targets** for data standards which, if fully implemented, would provide the necessary infrastructure. Most, though not all, of these are partly in place at the beginning of 2014. A primary role of the LCC is to promote their completion.

1. **A global Party ID "hub".** Rightsholders and "Asserters" should be identified with an identifier linked to the ISNI "hub".

   A Party is a person or an organization (this includes different "public identities" of parties, such as pseudonyms adopted by creators). Unambiguous identification of Rightsholders and those who assert Rights declarations is most basic building block of the rights data network. The ISNI (International Standard Name Identifier) is a relatively new ISO standard identifier which can be used as an ID in its own right, but whose main role is to be a global "hub" to which different IDs for the same party to be linked together so that they can be automatically matched to or substituted for one another in systems when necessary. ISNI does not therefore *replace* other IDs, but enables them to interoperate with one another.

2. ***Creation IDs for all.*** Creations of all types should be identified to any required level of granularity.

   Public identifiers, supported by minimum metadata, are essential for Creations of all types in which rights are asserted (physical and abstract works as well as digital, because rights in all these are assigned in the digital network). Identifiers are needed at whatever level of granularity (sets, parts, fragments or derivations) specific rights are assigned for. Not all types of Creation have public ID standards, and those which do are not all as fully implemented as needed.

3. ***Right IDs.*** Content rights should be identified distinct from, but linked to, the Creations to which they relate.

   A "Right ID" which identifies a Right as a distinct data entity, separate from the Creation(s) it applies to and the agreements or policies which bring it into existence, is the most significant gap in the network's data. Because rights data is changeable, it cannot be reliably embedded into digital content itself, but should be accessible separately via linked identifiers.

4. ***Resolvable IDs.*** Identifiers should have a URI form which may be persistently and predictably resolved to multiple services within the internet.

   A resolvable identifier is one that enables a system to locate the identified resource, or some information about it, such as metadata or a service related to it, elsewhere in the network. Some identifiers, such as DOI and EIDR, are already resolvable, but many standard IDs do not yet have an expression in a URI format.

5. ***Linked IDs.*** "Cross-standard" links between identifiers should use interoperable terms and be authorised by interested parties at both ends of the link.

   Where one Creation (for example, a sound recording identified by an ISRC) has a dependent relationship with another (for example, a musical work which it contains, identified by an ISWC) then the vocabulary term describing that relationship should be standardised in some public schema, and it should be possible for Creators or Rightsholders of either of the identified Creations to agree or dispute the validity of the link under some registry procedure.

6. ***Interoperable metadata.*** Standard content and rights metadata schemas and vocabularies should have authorised, public mappings which enable terms and data to be automatically transformed from one standard into another.

   As with other identifiers[9], it is neither possible nor necessary for everyone to use the same schemas and terms, although the more common usage there is, the better. What is needed is for authoritative mappings (authorised by those who govern the schemas) available as services supporting automated "translation" of metadata.

7. ***Provenance of rights data.*** The provenance ("Asserter") of Rights declarations should be made explicit.

   In a distributed data network like the internet, the provenance of rights declarations must be explicit if systems or users are to be able to trust it (or not). The Asserter of a statement of Right may or may not be the same party as the Rightsholder. Without the ability to identify the Asserter of a Right (with or via an ISNI), there is no basis for secure automated identification of Rights in the network, or for the identification and management of conflicts (see target 9).

8. ***Digital Content Declarations.*** Anyone should be able to make standardised, machine-interpretable public statements about Creations and the Rights and permissions which apply to them.

   Using the elements described in 1-7 above, Rightsholders and their agents require a means by which any Party can simply identify and describe themselves, their content and their Rights in a Web or other network environment. This is especially useful for the huge volume of "direct-to-Web" publishing which now takes place, but can be applied by anyone. The standard should be built into services which support the publication and management of content and related IP in the network.

[9] Note that terms in controlled vocabularies are identifiers, as they are unique names within their domain and type. When expressed as URIs they just become more identifiers in linked data.

9. ***Dispute management*.** Conflicts between public Rights declarations should be automatically identifiable so that their resolution can be managed.

   Conflict or dispute management has always been an important task for CMOs (collective rights management organizations) because they receive conflicting rights claims from different Parties. Where rights data moves out into the more "open" linked data, the same issues occur, but will be on a larger scale and not always under control of a single organization. Standard ways are needed of identifying and tracking these. *[link].*

10. ***Linked fingerprints*.** Digital "fingerprints" should be mapped to registered Creation identifiers.

    Proprietary digital content recognition systems[10] (for example, Content ID, Picscout, Soundmouse and Attributor) provide the means for a variety of functions, including the tracking of digital usage. Linking these IDs to registered Creation identifiers ensures that such functions can be fully integrated with the rights data network.

1 Entities: what should be identified?

Public, persistent identification of key supply chain entities is essential.

1.1 Identity is functional

The main reason for assigning an identifier is to separate things which are the same as each other from things which differ from them. If two things are different they require separate identifiers. To avoid the paradox of identity[11] we must add the qualification "…which differ from them *for some purpose*", and establish the criteria being used to make the distinction.

Identity is therefore never **absolute** ("A is the same as B") but **contextual** or functional ("A is the same as B *for the purpose of* C").

For example: two pencils on a table have been taken from the same packet of newly purchased pencils. For the purposes of writing, or of re-ordering more pencils by quoting the manufacturer's number printed on the side of each pencil, the two are *fungible* (that is, indistinguishable) and do not need to be separately identified. But if one pencil has been handled by a criminal whose fingerprints on it are crucial evidence of his presence at the scene of the crime, then for the purpose of forensic analysis the two pencils are no longer indistinguishable and need to be separately identified (for example, by adding an evidence tag).

In a second example, from information management, two editions of *Robinson Crusoe* may be indistinguishable for the purpose of a textual citation of the Defoe work (that is, they have identical

[10] For example, proprietary systems such as Content ID (video), PicScout (images), Soundmouse (audio) and Attributor (text)

[11] Liebniz's law of identity, or the indiscernibility of identicals ("X is identical with Y if and only if every property of X is a property of Y and every property of Y is a property of X"), leads to the conclusion that "Roughly speaking, to say of two things that they are identical is nonsense, and to say of one thing that it is identical with itself is to say nothing at all." (Ludwig Wittgenstein)

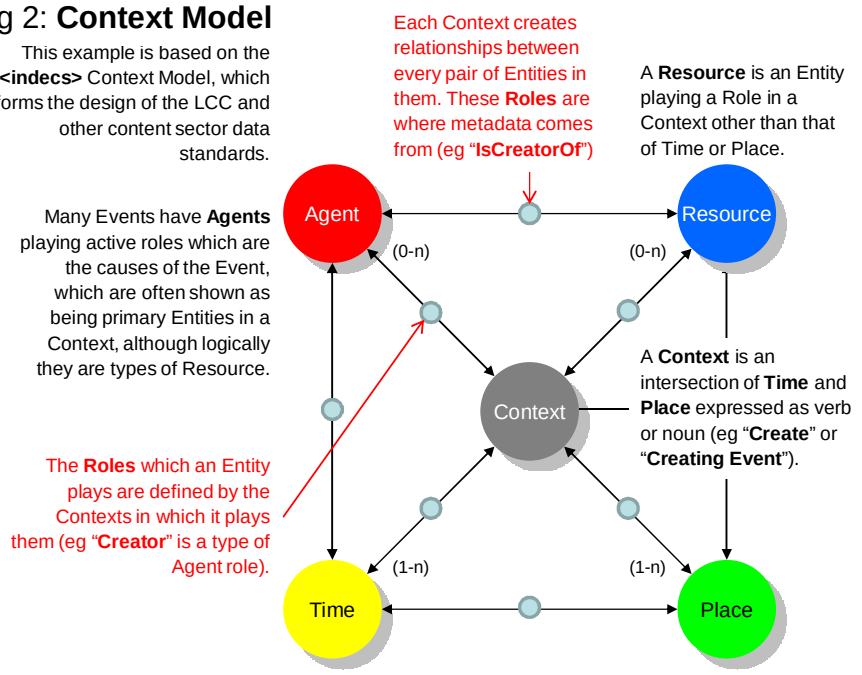
content) even if one is an e-Book version and the other a leather bound presentation copy, but they must be separately identified for purposes of ordering a replacement copy.

Describing the contextual identity issue, the indecs project stated (as one of its four principles key to the management of identification) the *Principle of Functional Granularity*: "It should be possible to identify an entity whenever it needs to be distinguished". We might add "…for a particular purpose".

In everyday language the term "context" is often used loosely, but when applying identifiers the nature of the context should be definable in terms which can be analysed with precision, so that where necessary rules can be applied to determine granularity. The Linked Content Coalition's data model is based, like other interoperability initiatives, on a precise contextual model derived from the indecs project, where a context is defined as an intersection of *time* and *place*, in which *entities* may play *roles.* Fig 2 provides a summary of such a model. Each entity in a context needs an identifier.

Fig 2: **Context Model**

This example is based on the **<indecs>** Context Model, which informs the design of the LCC and other content sector data standards.



Contexts and meaning

Meaning arises through Events. Metadata describes relationships between Entities, all of which occur through Events, or from the States which follow them. All Events and States are Contexts.

Non-contextual metadata (for example, relationships between Creators and their Creations, or Creationbs and their Time or Place of Creating) are often shown as direct relationships without reference to the underlying Context, but this Context is the key to understanding their meaning and origin.

In the Context Model every element of metadata is a "link" between two Entities. In physical models, many links are "denormalised" and treated as Attributes belonging to an Entity. The LCC Entity Model provides one example of the formal representation of this.

Even when two identifiers have been securely determined to have the same referent, this does not necessarily imply that the two identifiers are fungible. They may have *semantic interoperability* (that is, they denote the same referent), but the social infrastructure in which they are used may have restrictions, e.g. resolution to some metadata or access to a repository may be restricted to a community of registered users in one or both systems, and they may lack *community interoperability*.

1.2 Forms of Referent

A referent is the entity denoted by an identifier. Unique identification requires that each identifier has one and only one referent. A referent may have (and usually does have) more than one identifier.

Referents may be of any form:

- **physical** (for example, a book identified with an ISBN, a building identified with a postal address or a human being identified with a Social Security Number)
- **digital** (for example, a digital file identified with a URI or a virtual location identified with a URL)
- **spatio-temporal** (for example, an event such as a rights agreement identified with a licence number[12], or a insurance policy identified with a policy number).
- **abstract** (such as a song identified with an ISWC, or a theme identified in a controlled vocabulary or code list).

1.3 Types of Referent

There are several types of entity for which public identifiers are essential in the digital content network. The LCC Rights Reference Model (RRM)[13] defines eight types of entity of interest in the rights data network, of which three (**party**, **creation**, **place**) commonly have public identifiers. The RRM also defines a **right** as a distinct entity and proposes that a public identifier is needed for it.

Appendix 2 of the LCC Principles of Identification provides details of the status of public identifiers for parties and creations.

1.4 Classes and individual manifestations

Identifiers of manifestations are often applied not to individual items but to classes of creations. For example, an ISBN is applied to the class of books which are functionally identical as published editions, whereas an individual copy of the book to be located within a library or a second-hand bookshop will have its own distinct local identifier.

1.5 Controlled vocabularies as identifiers

Controlled vocabularies (sometimes known as “code lists” or “allowed value sets”) are essential groups of public identifiers in the digital network, although they are often not recognised as such. In part this is because they are often presented as “words” in natural language, but a controlled vocabulary term is unique within the vocabulary and once it is given a namespace prefix to turn it into a URI (eg dc:creator, ddex:SoundRecording) then it becomes a global public identifier within the digital network. Controlled vocabulary terms are commonly to identify classes of entity.

[12] A “License ID” is often casually thought to refer to the document containing the details of the agreement, but in LCC is it recognised as an identifier of an event in which agreement was reached or permission unilaterally granted for some right to come into existence. The License document is a record of that event, and of the specific right(s) which exist as a result of it. The document itself may have a separate identity (indeed, there may be different copies of the document, as of any document, with distinct identifiers). The same principle applies to all manner of agreements or policies.

[13] http://www.linkedcontentcoalition.org/#!lccframe/c4nz

1.6 Digital representations of non-digital material

It is one of the ironies of digital identifiers such as DOIs (Digital Object Identifiers) that they often do not actually identify digital objects. A common role for digital materials is as a representation or manifestation of a *non*-digital entity (a creative work, a person etc.), because on digital networks such physical or abstract entities cannot be manifested directly. In linked content we are dealing not only with physical resources but with abstractions and digital materials. Abstractions and digital resources are just as "real" as physical entities, even though intangible: they interact with physical entities through actions in commerce and intellectual analysis[14].

A clear example is a digital representation of a physical painting, enabling users to see the painting without visiting the gallery in which it hangs. This example illustrates an important concept: in making a representation, some aspects of the original context are inevitably lost (as with the process of identification, representation is done *for some purpose):*

> " It's easy to forget that the very idea of a digital expression involves a trade-off […] A digital image of an oil painting is forever a representation… A real painting is a bottomless mystery, like any other real thing. An oil painting changes with time; cracks appear on its face. It has texture, odor, and a sense of presence and history.
>
> Another way to think about it is to recognize that there is no such thing as a digital object that isn't specialized. Digital representations can be very good, but you can never foresee all the ways a representation might need to be used. For instance, you could define a new MIDI-like standard for representing oil paintings that includes odors, cracks, and so on, but it will always turn out that you forgot something, like the weight or the tautness of the canvas.
>
> The definition of a digital object is based on assumptions of what aspects of it will turn out to be important. It will be a flat, mute nothing if you ask something of it that exceeds those expectations. If you didn't specify the weight of a digital painting in the original definition, it isn't just weightless, it is less than weightless. A physical object, on the other hand, will be fully rich and fully real whatever you do to it. It will respond to any experiment a scientist can conceive. …. it is impossible to represent it to completion. A digital image, or any other kind of digital fragment, is a useful compromise. It captures a certain limited measurement … within a standardized system that removes many of the original source's unique qualities.."[15]

2 Structure: what form should an identifier take?

2.1 Common identifier forms

Standard identifiers in the content supply chain have commonly incorporated some or all of the following elements:

- the **type** of identifier (eg ISBN)
- a code for the **issuer** of the identifier
- a code indicating the **place of issue** (typically a territory) of the identifier
- a date or datetime indicating the **time of issue** of the identifier

[14] An extensive discussion of this issue can be found in the work of Karl Popper, see e.g. Objective Knowledge: An Evolutionary Approach, 1972, Rev. ed., 1979, ISBN 0-19-875024-2

[15] Jaron Lanier, You Are Not a Gadget (2010), Ch.10

- one or more **check characters** calculated by an algorithm to validate the integrity of the identifier

The advent of the Web, and with it the URI (Uniform Resource Identifier) in its various specialized forms (which include URL, URN and DOI) has led to a more flexible and open-ended approach to identification, with uniqueness being secured by a globally unique common prefix (such as a DNS domain name www.anything.com/ or a DOI registration agency prefix "10.0001/") with a user's local identification model providing the remainder or suffix of the identifier.

At the same time there has been a steady move away from human-readable identifiers to **machine-readable** strings, a growing number of which are assigned by, and only readable by, computers. Increasingly important among these are digital "fingerprints", derived from the binary structure of the content of digital objects[16]. This ability to generate an identifier from content-in-hand is known as **affordance**: "a situation where an object's characteristics imply its functionality and use"[17]. "Afforded intelligence" identifiers are not normally as fragile as intentionally embedded intelligence: the object can have its identifier created (or recreated) from its invariant properties. However these are applicable only to unique physical objects (or unique digital objects in the form of hash signatures[18]) and are of no direct use as identifiers of abstractions such as works, concepts and classes, or physical referents outside the digital network, such as people or buildings.

An identifier may have multiple "designations".

2.2 Multiple forms ("designations") of the same identifier

It is increasingly common for a single identifier to be expressed in more than one standard form for different purposes. The book identifier ISBN provides a representative example. Its original, "human readable" standard form (known as the "ISBN-10") was of a 10-digit number with the common prefix "ISBN" (for example "ISBN 817525766-0"). This then became expressible as a 13-digit European Article Number (known as the "ISBN-13") suitable for encoding as a printed and machine-readable "barcode" ("9788175257665"). More recently the same ISBN has become expressible as a Digital Object Identifier (DOI) (known as an "ISBN-A", or "actionable ISBN) using the standard DOI syntax in the form "10.97881/75257665" to enable it to be resolved in a digital network. All three identifier forms have the same referent. In LCC terminology, these are different **designations** of the same identifier.

An identifier should not contain dynamic or confusing intelligence.

2.3 Meaning in identifier strings

Many disciplines over the years have learned that embedding attributes of the identified entity into the identifier string itself can produce a fragile identifier, subject to malfunctioning and

[16] Exemplified by products and and services such as (for example) PicScout (images), ContentID (Google/Youtube), Soundmouse (audio) and Digimarc Attributor (text).

[17] http://www.usabilityfirst.com/glossary/term_66.txl

[18] e.g. the proposed URI scheme "Naming Things with Hashes" http://tools.ietf.org/html/draft-farrell-decade-ni-10

misunderstanding. On the other hand, it is not the simple case that identifiers should be "dumb", but care is needed in embedding and using meaning in identifier strings.

Meaning in identifiers may be classified in one of three way: **vital intelligence** (see 2.4), **"risky" intelligence** that may be best avoided as it may be easily misunderstood (see 2.5), and **dynamic intelligence** which is changeable and so should always be avoided (see 2.6).

2.4 Vital intelligence

Vital intelligence in an identifier relates primarily to the process of allocation of the identifier itself, to ensure its global uniqueness typically by assigning a **set of prefixes** to one agent, which may then create its own unique namespace by further qualification, as in the ISBN system, or in the resolution of an identifier on a network (for example, assigning an internet protocol such as http: to precede the identifier string).

When identifiers are assigned on a federated model (see 3.8), it is normally essential to include such a prefix. Existing federated global identification standards of which LCC is aware (for example DOI, GS1, Media Access Control [MAC address], Internet Protocol [IP and IPv6], EIDR) use such a structured prefix solution in order to allow maximum possible flexibility of local members in the allocation of identities. We have not, in contrast, been able to find any examples of unstructured global federated identifiers. It appears that successful federated global identification standards have found it necessary to adopt a structured numbering assignment system[19].

The prefix may designate the **type** of identifier in the string itself (as in http:) or the **issuer** (as in a DOI prefix such as 10.1000). In human readable identifiers these may be presented transparently (as with "ISBN" or "DOI" preceding the rest of the identifier), or it may be encoded more cryptically: for example the three-digit code "978" at the beginning of a 13-digit EAN (European Article Number) means that the remaining 10 digits represent an ISBN.

It is also common for identifiers to include one or more **check characters** calculated by an algorithm, with which the validity of a particular string can be checked. For example, the ISBN has a single digit check character in the last position, and this varies according to whether the form of the identifier is a ten character ISBN or a 13-digit EAN.

2.5 "Risky" intelligence

Identifiers commonly include information about the **issuer** and sometimes the **date of issue** of the identifier. For example, the book identifier ISBN contains a group of 2-8 digits which designate the publishing group and publisher issuing the ISBN to the referent, and the sound recording identifier ISRC contains a two-digit code designating the year in which the ISRC was issued.

Such information may clearly be useful, but it is also easily and commonly misinterpreted. The issuer of the ISBN may be assumed to be the owner or publisher of the book in question, but may only be acting on their behalf; and the year of issue of the ISRC may be assumed to be the year in which the

[19] A recent paper on the FSB Legal Entity Identifier examines the issue of structure in identifier assignment and its role in federation mechanisms in great detail: *Braswell et al., Response to the Financial Stability Board's Request for an Engineering Study on the Best Approach to Managing the Structure and Issuance of Legal Entity Identifiers (LEIs)* (October 7, 2012): available at http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2197269

recording was released, but of course because a great deal of older recorded material is digitized, this is commonly incorrect.

It is also preferable that **persistent information** about the entity (that is, descriptive or other information that should not change) should **not** be encoded within the identifier, because (a) like all metadata, it may be interpreted differently in different contexts and (b) it may be found to be incorrect at a later date. Identifiers commonly include information about the type or form of the referent, or (in the case of creations) information about links to associated creations (such as the original of which the referent is a version or excerpt).

All such persistent information is best declared as metadata and stored in a repository to which the identifier may resolve (see section 4).

Identifiers with no embedded attributes derived from, or dependent on, another entity are also sometimes called **first class identifiers**.

2.6 Dynamic intelligence

Dynamic information about the entity (that is, information which may only be true for a particular time or in a particular context) should **never** be encoded in an identifier. This includes

- **location** information (such as a URL) for the referent (unless the referent is itself the location, of course);
- **status** information (such as the availability of a product); and
- information on ownership of, or **rightsholdings** in the referent, as this may change over time, place and for different purposes.

This principle has also been recommended by W3C for URIs: "*Good Practice: Resource metadata that will change SHOULD NOT be encoded in a URI…*"[20] Note that this principle is not the same as defining an identifier specification "that contains no embedded intelligence".

3 Assignment: how should an identifier be issued?

An identifier should be issued under well-defined registry procedures and policies.

3.1 Registries

A registry is a database or other information structure providing a definitive record of identifiers issued within a particular namespace, typically with associated data such as basic metadata about each referent, the registrant, and "audit trail" information about when and by whom the identifier was issued, and so on.

3.2 Scope of identifier registries

An identifier registry will have a defined scope, such that certain entities are within its scope and others outside it. For creations, in pre-internet days, when much material in libraries was physical in format (books, periodicals, CDs, tapes, etc.), classification was simple, on the basis of format

[20] The use of Metadata in URIs. TAG Finding 2 January 2007 http://tinyurl.com/ydd9yf

(although even then mixed media items caused problems); and within this by e.g. subject (e.g. "books whose subject is poetry"[21]. The definition of scope may be translated into the presence of certain attributes of a referent, which define the associated metadata to be registered. This assists the practical development and operation of some registries, especially when their scope is the more traditional library material of physical objects or material which is expressed via physical objects. Typically, ISO TC46 SC9 identifier schemes arose from such library-like documentation considerations, and creation identifier standards (ISBN, ISRC, ISWC, ISSN, ISAN etc) have been

However, this approach is increasingly inadequate. Pre-internet, a few exceptional identifier schemes allowed for abstract classification of any entity (for example, OID). With the rise of the internet, where all entities becomes represented as digital objects ("bags of bits") which are processed in the same way, the limitation of traditional classification has become clearer and the scope of identifiers such as URI, URN, HDL or DOI is recognised as potentially arbitrarily wide. For practical reasons, limitations of registry scope must be introduced to provide a basis of social infrastructure, but this tends to determined by functionality rather than form, so (for example) DOIs are allocated by different registration agencies each of which have a defined coverage or scope normally based on application (e.g. "scholarly citation") rather than referent format.

Further adding to the problem of classifying by format, not only is the amount of data requiring identification growing rapidly, but the nature of data is changing as well: it is estimated that over 80% of data generated today is unstructured or semi-structured and so does not fit readily into a pre-formed classification.

The view that the world of information can be categorised into non-overlapping independent silos in one definitive manner is now recognised as mistaken. All classifications are arbitrary; since all classifications are a form of identification through grouping into identified classes, they are subject to the same choices of functional granularity as with single identification. There are no absolutes, just mechanisms and human understanding:

> *"These ambiguities, redundancies, and deficiencies recall those attributed by Dr. Franz Kuhn to a certain Chinese encyclopedia entitled Celestial Emporium of Benevolent Knowledge. On those remote pages it is written that animals are divided into (a) those that belong to the Emperor, (b) embalmed ones, (c) those that are trained, (d) suckling pigs, (e) mermaids, (f) fabulous ones, (g) stray dogs, (h) those that are included in this classification, (i) those that tremble as if they were mad, (j) innumerable ones, (k) those drawn with a very fine camel's hair brush, (l) others, (m) those that have just broken a flower vase, (n) those that resemble flies from a distance".*[22]

3.3 Assignment at the point of creation or derivation

At what point should an identifier be issued (or "minted")? The answer is "the earliest possible point", but this is not always straightforward.

For example, suppose a creation (a product of some kind) passes through a supply chain. Parties ("nodes") in this chain are denoted A, B, etc., and an identifier assigned at point X will be denoted i_X. In a simple supply chain, the product made by the producer at A is supplied to a consumer at C via a

[21] Svenonius, E. (2000) The Intellectual Foundation of Information Organization. MIT Press, Cambridge, Massachusetts
[22] J.L Borges: "The Analytical Language of John Wilkins". See also other Borges discussions of metadata quoted at www.tertius.co.ltd.uk, "Orbis Tertius".

retailer at B, and the chain can be summarised like this: A → B → C. If the consumer is to manage the entity down this chain and reliably reference it by a unique identifier, the identifier used in the chain must be the identifier i_A, which is assigned at point A, because

- the product at A is supplied unaltered to C, so the same thing is present at A, B and C,
- the same product might be supplied via an alternative retailer D: A → D →C , so B can have no role in determining identity, and
- C may wish to cross-reference material supplied by both B and D.

So all parties in this simple supply chain should use i_A. A is the **point of entry** to the supply chain, which may or may not be the same thing as the **point of creation**, but in any case it is the earliest possible opportunity in this model. In fact, i_A may even have been issued even before the creation of the product, if there is a workflow at A which requires identification at the beginning of the production process.

Another party in the chain may of course decide to use an additional identifier for their own internal processes, and store it as a "mapped identifier" to i_A. This brings risk, though, as such local identifiers may be mis-used and "leak out": for external interoperability this is not efficient and creates a potential source of delay or breakage in the chain.

A level of complexity occurs when B creates and supplies a version or **derivation** of what is provided by A (for example, a different format, or a product containing some additional feature added by B, such as language subtitles). Now, when is a derivation something new? What A considers to be "the same thing" is not what C may consider to be "the same thing"[23].

If it is determined that the version is indeed a new entity, this becomes a subset of a more general case: when the chain is not of simple supply but of re-purposing or modification: A → E → C, where a party or machine at E processes the entity from A to create a new entity which is supplied to C (for example, A creates an audiovisual work and E adds subtitles to it). It is clear that here both i_A and i_E are necessary, with E being the **point of derivation** since A may be supplying the work to others to provide similar opportunities for value creation, and it is also clear that the relationship of i_A to i_E must be declared. C may be purchasing other, different, works derived from A by another supplier F (for example, with subtitles in a different language). This requires i_F, and a declaration enabling a distinction from i_E to be ascertained. Two chains may wish to determine if they are supplying the same customer C; and so on.

The point at which a new identity, and therefore a new identifier, comes into existence will vary according to the type and nature of the entity, and will be a matter of agreement, policy or custom. For example, the point at which minor changes and corrections in the text of a book warrant a new edition will be a matter of editorial judgment. Up to that point, re-publication of the amended book may be with the same ISBN, and after that point there will be a new ISBN.

It is necessary to have distinct identifiers at the point at which a new entity is made available, and the relationship to the contributing entities in a new creation must be declared in a usable way. It

[23] Those working with identifiers will be aware of the common reaction from users faced with such complexities: "but we all know what we mean by…". Indeed, you may know what you mean, and I may know what I mean – but it does not follow from this that you mean what I mean.

should be possible to describe how two different entities derived from a common underlying component (e.g. i_E and i_F) relate to each other. In all of these cases, the principle that identifiers should be assigned at the earliest possible point (where the new entities come into existence or into the supply chain) minimises error and maximises efficiency.

This is still much less complex than reality. In a linked content world, simple linear chains as these simple examples will intersect and form branching and reuniting networks. Material produced by one party may be re-used by third parties without direct knowledge of the originator. Standardised principles enabling unique identification and data models enabling objective description of relationships are essential. Even this is a simplification and there will be complications due (for example) to legacy systems, or there will be multiple identification for legitimate reasons (such as privacy).

Such chains are multiplying and may be subject to constant change. Many products which were once viewed as fixed resources are now more likely to be continuous resources with frequent changes (news websites, for example).

3.4 Co-reference

Co-reference is the term for occurrence of multiple or inconsistent identifiers for a single resource. *"Much of the Semantic Web relies upon open and unhindered interoperability between diverse systems; the successful convergence of multiple ontologies and referencing schemes is key. However, this is hampered by the difficult problem of co-reference…"*[24]

Co-reference is unavoidable for many entities, where there is not a single, clear "early as possible" point of assignment of an identifier. For example, people and organizations ("parties") are identified in many different systems in different domains and for different purposes. It is not practical to impose a single global party identifier.

However, it is also unnecessary. Co-reference can be effectively managed through **mapped identifiers**, where any number of different identifiers for the same referent are associated directly with one another in "same as" relationships. For example, the International Standard Name Identifier for parties (ISNI) operates on precisely this basis. Some ISNIs may not be used publicly, but will act as central identifiers for the purpose of many-to-many mappings. Identifiers a, b and c may all be mapped to a single ISNI, and are therefore mapped as the "same as" one another and can be automatically substituted for one other to enable data to pass from one domain to another, where different identification systems are used.

It is not uncommon for two or more identifiers *of the same type* to be assigned to the same referent in error. To deal with such cases, the registry procedures must support the deprecation of one identifier by its "merging" with another. The **merged identifier** does not cease to exist, but remains "mapped" to the active identifier so that any usage of the "merged" identifier can be substituted for usage of the active one.

Some identifier frameworks offer the ability to express an existing identifier in the syntax, or as a "same as" metadata link to another system: for example, ISBNs may be expressed as GS1 bar

[24] Glaser, Hugh, Lewy, Tim, Millard, Ian and Dowling, Ben (2007) On Coreference and the Semantic Web. http://eprints.soton.ac.uk/265245/

codes[25]; ISO identifiers may be expressed as DOIs[26]. This confers the advantage of being able to embody equivalence, but is also open to the risk of embodying incorrect equivalence which cannot then be rectified if a registry has not captured sufficient information with a specific registration. Effective processes to discover incorrect co-reference, and to amend data effectively when it is discovered, are essential for a registry.

3.5 False co-reference

Common language provides, and sometimes makes assumptions about, context which needs to be carefully translated into precise entity attributes and relationships when referents are abstracted to be computed. Without the underlying analysis, one may not be identifying the correct referent.

A pitfall arises if a clear delineation is not made of the difference between a datum, the symbolic name of that datum, the address at which the datum is stored, and the symbolic name of that address[27]. It is not always necessary to separately distinguish these (each may be reified if necessary; this is an issue of functional granularity), but confusing them when they are distinguishable leads to problems. For example, an address may be reified, which requires identification (naming), so that the reification then has a name; a name is an address in a name space called "addresses"; and this is at the heart of much confusion over naming and addressing in discussions of URN, URL, URI and discussions of their usage as persistent "identifiers".

3.6 Ambiguous identifiers

Ambiguity is much more serious that co-reference, as it may be systematically unresolvable once it has occurred. An **ambiguous identifier** is one that has been wrongly applied to two or more different referents in error. This happens most commonly when it is not recognised at the time of assignment that two things which were thought identical are in fact different (for example, two different people called "John Smith"). It may also occur in error when a system re-issues an identifier to a new referent without recognising that it is already assigned to another. There is no easy answer to this: the trail of identification must be reviewed and the correct identifiers assigned where possible.

3.7 Functional granularity in creations

Functional granularity means that "it should be possible to identify an entity whenever it needs to be distinguished for some purpose". For example, a book may have a single ISBN, but if individual chapters or illustrations are extracted from it and published separately elsewhere, they may require distinct identifiers of their own; or when a new edition or translation is produced, it will require a new ISBN; or if a 30-second audio or video clip is taken from a longer recording for promotional purposes it will require its own identifier as it is deployed in different systems.

Functional granularity applies to the identification of any entity type, but it is especially significant in creations, and become more so as digital content can be fragmented, aggregated and transformed in an ever increasing number of ways, so the examples given here are all of creations.

---

[25] http://en.wikipedia.org/wiki/International_Article_Number_%28EAN%29#Bookland

[26] http://www.doi.org/factsheets/DOIIdentifiers.html; http://www.doi.org/factsheets/ISBN-A.html

[27] This issue was memorably dissected by Lewis Carroll (pseudonym of the logician C.L.Dodgson) in *The White Knight's Song* which appears in *Alice in Wonderland*: see the discussion of it by Eric Walker at http://www.alice-in-wonderland.net/?school/alice1020.html

There is a set of common relationships which may result in the need of new identifiers in relation to existing creations: **parts** (3.6.1), **aggregations** (3.6.2), **fragments** (3.6.3), **derivations** (3.6.4) and **manifestations** (3.6.5). There is no theoretical limit on the smallness of a fragment or the enormity of an aggregation which may be identified if there is a reason for doing so.

3.7.1 Parts

A part is a distinctly identified component of something. The entity may have been assembled from pre-existing parts (for example, an online "learning object" with a package of text, music, video and images) or it may be created as a whole in which parts are later identified (for example, the chapters of a book). The parts may be identified, described and managed separately, and become parts of other compound objects.

3.7.2 Aggregations

An **aggregation** is created out of several pre-existing entities, perhaps with new material added. The aggregation can then be represented, in part, by the relationships that exist between the distinc t identifiers of its components.

3.7.3 Fragments

A **fragment** is a resource that is subordinate to (or "contained within") another, primary resource. The fragment (such as an arbitrary clump of text or a 15-second excerpt from the middle of an audio track) is not inherently a first class object but instead its identity is defined as a subset of the primary resource: of course this may be identified as a new first class entity if there is a functional need.

A problem raised by fragment identifiers is the existence of an infinite set of possible ad hoc identifiers from one base primary resource (for example, time ranges in a video). In many cases today "fragment" is used in one specific sense: in http to refer to the piece of a URL that the server doesn't really know about and that the client hangs on to and then processes returned html to do the right thing. This is a function of the hypertext model that was initially selected for http/html, which operates at the file level; so to get to some specific point within a file using http requires a second mechanism. In the internet, fragment identifiers are well understood in principle, but not uniformly dealt with.

An example from music well illustrates the application of functional granularity to fragments. In a European music collecting society the royalty distribution rules were biased in favour of the number of different works used as background to a television program, rather than simply the duration of music, and it was therefore more rewarding to have six "musical works" of ten seconds each rather than one work of sixty seconds. As a result the standard practice of composers and publishers was to register large numbers of very short works with titles like "Man goes into room" and "Man goes out of room" rather than a single work "Background music from Man In A Room".

3.7.4 Derivations

Derivations cover all situations where a new creation is made by adapting, in some way, an existing creation. This covers all forms of versions, adaptations, arrangements, translations, transliterations, mashups, remixes, photoshoppings, director's cuts, edits, editions, and so on (and on). In the digital network the opportunity for new types of derivation grows constantly, and the volume of derived content along with it.

The question of "when do I need to assign an identifier to a new version" is the classic case of functional granularity. In principle, any change in the attributes of a creation may result in the identification of a new derivation, but the point at which a creator or publisher declares a "new" derived creation is arbitrary, and will commonly be a matter of convention: it is functional granularity at its purest – "I say something is a new version when I say it is".

Each application space defines its own rules, which may be guided by legal requirements or commercial opportunities. For example, laws or regulations may state that even an addition of a comma in a statement, even if it does not affect meaning, requires resubmission as a new item, whereas such an addition in a news article may be viewed as a proofreading correction that does not merit new identification. This necessarily requires not only tools for semantic interoperability but agreements on community interoperability. In scientific publishing, version control is significant for definitive attribution, so this community has agreed some rules for definitive version identification and tools to assist in their implementation[28].

An example from music will serve to illustrate the profound difference that a commercial opportunity may make to granularity. A music collecting society database contains hundreds of copyrighted "arrangements" of the Christmas carol "Stille Nacht/Silent Night" even though the musicological significance of the "arrangements" are generally trivial, because as the original work is long out of copyright, the original creator will not likely to dispute the claim. The same database contains only a single versions of Lennon & McCartney's "Yesterday", despite the fact that there are many musicologically distinct versions extant, for the simple reason that the original creators/owners will not recognise derivations. Using musicological, rather than commercial, criteria as a basis would result in a quite different pattern of functional granularity.

3.7.5 Manifestations and abstractions

The relation of manifestation to abstraction (or "work") is the most important and problematic when dealing with "compound creations". Music provides a clear model in the distinction between the underlying **work** (for example, a "song" like "Yesterday") and a **manifestation** of it in the form of a performance, recording or sheet music.

Works and manifestations have different rights, and often different rightsholders (for example, the composer and publisher of a work, and the performer and record company producing the sound recording). Although perhaps most clear in music, the manifestation/work split occurs in all types of content. Whenever something new is created, both a manifestation and a work will come into existence, with their accompanying rights. For example, a writer creating a poem is simultaneously creating an underlying work (the "poem")and the first manifestation (the document or "fixation" on the page or computer). Any number of new manifestations and performances of the poem may take place afterwards, but there remains a single abstracted "work" until such time as someone (for example, the poet) decides that sufficient change has happened to it to declare a new version (fort example, when it is translated in to another language).

The "parallel worlds" of perceivable manifestations and abstract works are increasingly reflected in identifier standards (for example, music with the ISRC and ISMN for manifestations, and the ISWC for works; text with the ISBN for books and the ISTC for textual works). Perhaps even more

[28] e.g. CrossMark: http://www.crossref.org/crossmark/

surprisingly, all Digital Object Identifiers (DOIs) issued to date have been to referents w hich are abstract works rather than digital manifestations. The maintenance of the fixed relationship between a manifestation and the work(s) it contains is in itself a critical registry function, which may be regarded as essential core metadata, but is as yet not commonplace in existing registries.

3.8 Federation of identifier systems

The governance of any identifier standard must consider the level and type of centralization desired in the system, ranging from a single monolithic registry to a more de-centralized federated system. Such considerations necessarily touch on both organizational/political considerations as well as network architecture issues. The creation or minting of an individual identifier (that is, the number or code itself as opposed to the reference metadata) may be most efficiently carried out by a federated set of registrars.

Federation is not a precisely-defined term, even within the context of the digital identifier network. Generally speaking, it describes an organizational structure somewhere in between a single entity or system and a set of completely independent entities or systems. Multiple entities or systems "federate" in order to jointly achieve some set of goals or functions while still maintaining some level of independence of action and governance. They do this by agreeing to co-operate with each other at some level, typically through the use of shared protocols or standards. The global telephone system is an example of this: there are many local and national telephone systems that work together, sometimes in relative ignorance of the details of each other's existence, by following a common set of technical protocols. The Internet can similarly be thought of as a set of local networks agreeing to a common address scheme (IP addresses) and implementing common network communication protocols such as TCP and UDP.

3.8.1 Federated identifier creation

Many global identifier systems use federated registrars to create valid and unique identifiers without the need to consult a central authority in each case. This is commonly done by sub-dividing the identifier space in some fashion and assigning or allocating the sub-divisions to various registrars. There are many examples of this approach being used successfully, some of which (Ethernet MAC addresses, IP addresses, GS1 product identifiers, and credit/debit card accounts) are outside the remit of LCC but provide working examples following the same principle. In the area of content management, the ISBN, ISRC, ISAN and ISWC are all managed through national registrars who assign numeric codes to content creators or publishers. In the Digital Object Identifier (DOI) system, an implementation of the Handle System, prefixes are allocated to organizations that then create identifiers by appending suffixes to those prefixes, but the distinctions between DOI registries are not based on nationality or territoriality but on the types of content or service being offered.

There are many advantages to the approach of minting identifiers on a global scale by subdividing the space and distributing the authority to create the identifiers to a collection of collaborating parties while still guaranteeing identifier uniqueness. Most of these advantages stem from the ability of the federated registrars to mint ids without consulting a central authority each and every time.

3.8.2 Avoiding co-reference in federated registries

A critical issue in the federation of the issuing of identifiers is to protect, as far as possible, against co-reference (the issue by of identifiers to the same content by two or more registries and, by

extension, the registration of duplicated descriptive or rights metadata). How this is done in any particular case will depend on the characteristics of the content and the sector affected: for example, the issue of unique ISWCs is generally guaranteed by its being tied to the collecting society membership of the creators (which in turn is made available by the management of another federated identifier, the IPI number for parties), whereas ISBN, ISAN or ISRC relies primarily on the necessary integrity of the internal processes of registering organizations (it is fundamentally in the interests of, say, a book publisher to ensure that a published edition has a single ISBN).

3.9 Core metadata

An identifier should be supported by metadata for discovery and disambiguation.

It is normally essential that when an identifier is minted, sufficient metadata is registered to enable the identifier to be discovered, and for the referent to be uniquely identified. The essential metadata ill normally including “management” metadata which will include:

- The date/time of issue of the identifier
- The registration authority or party minting the identifier

and some basic information about the referent:

- At least one name of the entity
- At least one type of the entity

For creations, typical core metadata will include:

- A title of the creation
- The creator(s) or major contributor(s) to the creation, with their roles
- The date/time of creation
- The basic structural type of the creation (abstract, physical, digital, spatio-temporal)
- The mode(s) in which the creation is intended to be perceived (audio, visual, taste, smell, touch)
- The character of the content of the creation (words, image, music, data etc)
- A form or format of the creation
- A genre of the content
- One or more measurements (for example, duration, size in MB)
- Links to other creations from which this one is derived if it is a version, excerpt or manifestation
- Annotations describing unique characteristics for the purpose of disambiguation.

Several of these categories (such as structural type + mode + character) may be combined in a “creation type” category (for example, “video clip”). However, such a list can never be exhaustive, because each type of creation may have specific attributes which are vital for identity or disambiguation.

3.10 Persistence

Once assigned, an identifier should never be re-assigned to another referent. The attributes of the referent may change while its identity, and therefore its identifier, persists. For example, a human being may grow older or undergo a gender re-assignment, but its identity persists. Similarly, a collection of creations may be added to or reduced, but their identity persists, or a territory's borders may change, but its identity persists.

The core metadata may also be added to or corrected, where it is found to have been incorrect. Persistence should also be ensured through registry provisions for maintaining metadata after the original issuer or asserter is defunct or dead or otherwise unwilling to accept responsibility for it.

3.11 Identifying "Orphan" and other unadopted entities

There is a particular problem with issuing identifiers for, and then preserving the uniqueness of identity of, entities for whom no-one in the network has a primary reason for taking responsibility. In the content and rights network, this includes "orphan" and public domain works (those for whom there is no, or no known, rightsholder or responsible party) and parties who are deceased or defunct and whose works are out of copyright.

There are two opposite risks: that no-one will identify them, or that several people will, in incompatible ways. For each global standard, it becomes necessary for the registry or federation of registries to devise a sound method of enabling such content to be identified in a standard way.

4 Deployment: how should an identifier be used?

An identifier should be accessible.

4.1 Four components: registry, resolution, repository

Four components are logically needed for deployment of first-class identifiers on digital networks: a registry of the identifiers (see 4.2); resolution mechanisms to link the identifier to some data (see 4.3-4.6); repositories where data may be found (see 4.7), and interoperability between different identification systems (see 4.8).

4.2 Registry

Registries are described above (see 3.1).

4.3 Resolution

An identifier should be resolvable.

Resolution is the process by which an identifier is the input request to a network service to receive in return a specific output of one or more pieces of current information related to the identified entity.

4.3.1 Basic resolution

In the digital content network, to promote linking and discovery of content, identifiers should be resolvable to a minimum set of publicly declared data. This does not of course mean that all

metadata about and entity should be public, and does not preclude the development of added value services built on such identifiers which are marketed to communities of registered users, but it does recognise the limitations of all identifier registries.

4.3.2 Resolution and reference

Resolution and reference are not necessarily the same (though in some special cases might be). There is only one referent for a given unique identifier, but there may be several resolution results for that identifier, and these may change over time.

In the special case where the identifier referent is "this specific file on this specific server" then this may indeed resolve to the referent. Resolution of domain name based URLs to an IP address is an example (though even this is complicated in real implementations due to the possibility of re-direction, caching, proxy servers, aliases, etc.).

4.4 Persistent resolution

For interoperability, identifiers must be persistent, at least return returning a "tombstone" message such as "this identifier refers to X which has since been removed due to Y" (cf. ISBN for out of print book titles) when resolution is attempted. Identifiers should have well defined and public registry operations and policies likely to ensure persistence.

Persistence is the consistent availability over time (persistence has been called "interoperability with the future") of useful information about a specified entity. It is ultimately guaranteed by social infrastructure (through policy) and assisted by technology. The aim should be to not shoot oneself in the foot by adopting inappropriate technology choices which will then restrict the best possible social infrastructure to maximise persistence: the principle of largely "dumb" numbering is clearly one such technical step. Other steps include managed metadata and indirection through resolution which allows reference to an entity to be maintained in the face of legitimate, desirable, and unavoidable changes in associated data such as organization names, domain names, URLs, etc. ; and governance steps to facilitate persistence in the event of registry demise (e.g. by orderly transfer of records).

The key social infrastructure necessary to ensure persistence is a registry (see 3.x). Further long term persistence requires continuity planning: governance consideration of the future of the registry in the event of the registration authority being unable or unwilling to continue. In the case of ISO identifiers, a generic ISO Registration Authority agreement has recently[29] been substantially revised (notably those of ISO TC46/SC9) and provides a minimum set of requirements providing some reassurance to the user community that assigned identifiers will be maintained, but these minimum requirements are not sufficient to plan a full implementation.

The most widespread persistent identifier used in the content sectors, the DOI System, has developed a series of persistence requirements (together with governance and operational policies)

[29] 2011. ISO state that "the generic RAA template is available upon request. There are 67 RAAs among ISO's 19000 standard so this is not something that we put on our website. ISO Committees should only establish RAs for exceptional cases" (source: ISO Central Secretariat, 25 Oct 2012)

which may serve as a model for other registry authorities seeking to provide a comparable level of continuity[30].

4.5 Federated resolution

Identifiers may or may not be actionable within a single system or multiple systems, and those systems may or may not be tightly connected to whatever approach is used to create the identifiers. ISBN, for example, was created for supply chain management and came into widespread use before ubiquitous network availability. The system for identifier minting was not associated with a system for resolving the ISBN to the sort of standardized data that the Digital Identifier Network requires. Many such systems sprang up, but the tightly federated effort has pretty much been restricted to the minting of identifiers: it is thus actionable in some cases but not uniformly or consistently so across the entire collection of ISBNs.

Newer and global-scale identifier systems are more likely to include a federated resolution approach in addition to federated assignment. In a federated system, resolution is typically a multi-stage process. The resolution information will typically be distributed across multiple systems (controlled or used by the federates) and client software must first discover which of these to query. A common approach is for the identifier to be structured, or subdivided, such that client software knows which federate to query, or how to find out which federate to query, typically by the inclusion of a registry code within the larger identifier. In that sense, federated assignment and federated resolution fit together well. In the Domain Name System (DNS), for example, a set of root servers that are known to all DNS clients contain data that redirect clients to the appropriate lower-level DNS servers in a hierarchical fashion, going from right to left to ask, for example, the com server where to go for example.com and asking example.com where to go for www.example.com, and so on. The Handle System is more likely to have a two-level approach, with a set of root servers redirecting all queries that begin with a certain prefix, e.g., 10.1037 to a given set of servers to resolve, e.g., 10.1037/0003-066X.59.1.29.

The combination of a method for locating the federates responsible for a given subset of the overall namespace and an agreed-upon group of protocols enables the federating organizations to be addressed as a single virtual system. Client software does not need to know the location of every possible server ahead of time and can find the resolution data as it is needed, including in servers and systems that were only recently added to the federation. This gives a great deal of flexibility.

Further, the federated systems need only agree on providing those services that they are required to provide in common. Each of the federated systems can provide additional services to their constituencies, possibly increasing overall efficiencies. This has been the experience of the International DOI Foundation, in which growth has come through the various registration agencies agreeing to provide basic DOI resolution in common while each separately provides a customized set of services to their customers.

4.5.1 Approaches to federated resolution

Approaches to federation vary across systems, depending in part on community requirements and in part on the age and legacy constraints of each system but, as in the case of identifier issuance, the

[30] www.doi.org: see in particular http://www.doi.org/doi_handbook/6_Policies.html#6.5

advantages of the federated approach all derive from a common characteristic - some level of independence from a central authority. There are two important aspects to this:

- **Federating existing systems**. An advantage of federation is that existing systems can be included without seriously disrupting their current operations. Whatever functionality is required for each federate can be layered on top of, or selected from, existing functions, thus reducing the need for new efforts and leveraging the proven reliability of existing systems.
- **Organizational independence and scalability.** Creating a global system by federating a set of local systems makes it easier to reach global scale. Domains and regions can be integrated by adding another federate without dictating how that federate must be created, funded, managed, etc. As long as the global level functions are met, (for example, providing data or services at a certain level of accuracy and timeliness) then the underlying structures need not concern the global system. This also introduces more diversity of organizational and technical expertise and experimentation, making it less likely that the centralized system will stagnate.

4.5.2 Network architecture issues in federation

There are a number of issues in the consideration of federated/centralized structures which relate to the architecture of the networks on which the identifiers will be used.

The first is **distributed computing** over networks. This is commonly associated with Internet-based federations (the web could be considered a very loose federation) but "distributed" does not equal "federated". Both centralized and federated systems can be physically distributed to avoid problems of single points of failure in either hardware or connectivity and can use redundancy to improve reliability.

Another issue of particular importance to identifiers is **persistence**: will an identifier created today still support its intended function five, ten, or fifty years from now? The identifier technology and network architecture are important here in that they should provide the tools for persistence. These tools will work only if there is an organization dedicated to their application and committed to making the identifiers work over time. Federations do provide strength in numbers and in that sense are likely to provide better organizational persistence than any single organization.

Finally, **open architecture** is key to long-lived and extensible systems. Today's Internet is the outstanding example of that approach. The protocols and standards used in connecting new or existing services to the Internet are widely and freely available, and any organization that wishes to provide a new service that can be interconnected to other users and services over the Internet can do so with minimal barriers to overcome. This has allowed it to survive and prosper in the face of enormous technical change. An open architecture Digital Identifier Network, using public interfaces for both input and output, will be much more likely to find wide-spread support and corresponding growth and stability. Again, this issue can be considered separately from centralization/federation, but open architecture and federation fit together well.

An identifier should be capable of resolution to multiple locations.

4.6 Multiple resolution

Resolution is the process in which an identifier is the input — a request — to a network service to receive in return a specific output of one or more pieces of current information (state data) related to the identified entity: e.g., a location (URL). *Multiple* resolution is the return as output of *several* pieces of current information related to an entity, such as at least one URL plus defined data structures providing additional data, options, or allowing management. Linking one referent to several outputs, each of which could be separately managed through an identifier resolution system (that is, additional items can be added or changed at the registry) has clear advantages, and provides an obvious way to move from a monovalent system (where an identifier can only resolve to one thing: a chain) to a multivalent system (where multiple linked connections can be made: a network).

At its simplest, single resolution could return a list from which to make a manual choice (e.g. "further information is available from the following sources…"). However, this is not a scalable solution for an automated environment, which needs to enable processing without intermediate human involvement. For machines to make sense of a complex response, the response has to be structured such that machines can distinguish the alternatives and act on them as appropriate. This is probably best done with multiple returned values as opposed to a single complex value. There is a need at this point for a common understanding of the values and their consistent representation, shared between (a) the parties and communities involved in creating those returned values; and (b) the parties and communities involved in creating the services that automatically interpret and act on those values. This is fundamentally no different than the case of single resolution which is acted upon automatically (for example, redirecting a single identifier to a single location such that the user just sees the end result) but is more complex, extensible to added services, more scalable, and requires greater coordination around standards and best practices. Including content negotiation as one of multiple values, for example, a single identifier resolving to both (a) a single content page for a human user and (b) a structured set of metadata for that content for further processing, is a current common example of multiple resolution.

If the various resolution options are defined as type:value pairs (e.g. "type = URL": "value=www.acme.com/123"), then automated and extensible management becomes possible, by choosing a type or types on resolution. This requires a standard set of defined types (and an extensibility mechanism to build further types), such that users could (a) adopt an existing type to provide a stated function, rather than invent their own; (b) be able to interrogate a list of existing types to determine the properties of a defined type. This technical infrastructure could then be further facilitated by a set of tools allowing appropriate management within a community of the type:values (*Designated Authority* and *Appropriate Access*), so that the common understanding of the values and their consistent representation can easily be implemented. For example, a given registry could take control of all the type:value records for a given referent; or these could be delegated with specific separate permissions to separate managers; or intermediate models can be envisaged , such as a rights repository and a content repository collaborating such that one identifier infrastructure can be used to serve two problems (e.g. embedded identifiers in media to "indicate copyright" and "'identify the work for other reasons").

4.7 Repositories

A repository is one or more information structures providing definitive data associated with the referent, possibly including representations or instances of the referent.

4.7.1 Repositories and registries

Repositories are logically separable from registries, though some applications may in practice combine elements of both: for example, a registry of identifiers will typically carry core metadata describing the referent of each identifier, whereas a repository may carry much more diverse and extensive metadata or content.

4.7.2 Resolving to repositories

Identifiers resolve to one or more repositories. Different applications may require additional contextual data and so require the identifier to be used with a different repository. In linked content applications, metadata may be needed from multiple repositories to provide definitive contextual information: for example, product information (formats, price etc.) is typically not stored in the same repository as rights data (permissions, licenses, licensors, royalty agreements, etc.) and typically these sets of data are under different management or different policies (privacy, public or private access, etc.)

There is a rough analogy here with a traditional library: Index/Catalog/Stacks = Registry/Resolution/Repository. For example, the Registry is the way to find an identifier when you don't have it in hand, like the "Readers Guide to Periodical Literature". The analogy is loose, however, because of the confusion between architectural components and functions and the natural inclination to make them match one-to-one. Registries may be independent, or may be constructed to use a specific resolution method; resolution and registry tools may sometimes overlap (for example, the next version of the identifier tool the Handle System[31] (v8) will add reverse look-up so that each Handle Service can act as a mini-registry, searchable by the values in the handle records). The Digital Object Architecture implements this logical model (see Appendix 2 for details).

4.8 Interoperability

At least three types of identifier interoperability may be distinguished[32]:

4.8.1 Syntactic interoperability

Syntactic interoperability refers to the ability of systems to process a syntax string and recognise it (and initiate actions) as an identifier even if more than one such syntax occurs in the systems. This is relatively straightforward and trivial: the "bar code reader" level of interaction.

4.8.2 Semantic interoperability

Semantic interoperability refers to the ability of systems to determine whether two identifiers denote precisely the same referent, and if not, how the two referents are related.

Identifiers of the same entity should be interchangeable.

An important tool for interoperability within the digital network is the mapping of **alternative identifiers** for specific entities. This is a commonplace process in computer systems to enable data from one system (which uses identifier I_A to refer to entity A) to be integrated into another (which uses identifier I_B to refer to entity A). The system records that $I_{A=} I_{B,}$ and other alternative identifiers may be added to the set.

[31] www.handle.net
[32] "Identifier Interoperability": http://www.doi.org/factsheets/Identifier_Interoper.html;

For example, a rights management organization may have its own internal identifier for a particular creation, and a rightsholder of the creation may have their own identifier which they have notified to the organization and by which they wish to have royalties reported. The two identifiers are "mapped" within the organization's system, and it is therefore able to report royalties to the rightsholder using the appropriate identifier.

In some cases a **registry of mapped identifiers** is created to enable many-to-many mapping of alternative identifiers (the ISNI for party identities is a prominent example). In such a registry there is one "hub" identifier for to which all other alternatives are mapped. The registry can then provide services which enable people or machines to look up and "translate" from any identifier into an identifier of another specified type, making them effectively interchangeable. The "hub" identifier (for example, the ISNI) may or not be an identifier that is publicly used: its primary purpose is to support the mapping so that people of machines can substitute one for another for any purpose.

4.8.3 Community interoperability

Community interoperability refers to the ability of systems to collaborate and communicate using identifiers whilst respecting rights and restrictions on usage of data associated with those identifiers in the systems. This is the level of business interoperability: identifiers may well share the same syntax and semantics, but if the associated metadata has been costly to collect and manage, or where it is commercially or otherwise confidential to a restricted audience, there may be legitimate barriers to making this freely available. However, practical use of resolvable identifiers requires that some minimal set of associated metadata should be generally available to facilitate third party use[33].

[33] For an example of this in practise see the DOI System concept of the DOI Kernel at http://www.doi.org/doi_handbook/4_Data_Model.html#4.3.1