Amorfs: The universal data format you've always wanted

Pete Chapman, Sydney, June 2018

I want to introduce a new data format that is simpler and richer than what we use today. It's called Amorfs because its uniform structure can be split or combined at any point without compromising its integrity, making it easy to share with other users and applications. Amorfs can represent highly complex information using extremely simple structures which can be nested to capture increasing levels of detail.

Being simpler makes the structure easy to explain and the data easy to read. Being richer means more matches and discrepancies can be handled automatically. Amorfs is a universal way to share information between any application or user, which means zero integration and less time spent rekeying details into online forms. Once in an Amorfs structure, data can be viewed as tables, forms, graphs, text, JSON, CSV, XML or any other visualisation.

Having an infinitely rich representation model shifts the paradigm away from individual applications having to own, understand and thereby control data. Information creators, the original owners, can now control how their data is shared and reused. At a trivial level that means no more rekeying credit card data for online purchases. Ultimately a step change in the data sharing economy can be realised by enhancing applications and integration tools to support Amorfs structures and make data meaningful in its own right.

Applications

Here are some of the applications that can be transformed using Amorfs data:

- Populating online forms using data stored locally like contact or payment details
- Transmitting product information like serial number or extended warranty along with payment details in online bank statements
- Sharing data across parties up and down the supply chain
- Enriching shared contact details and calendar invitations with readable content like location, attendees, rich agenda items, room configuration, AV equipment
- Embed additional relevant data into streaming media and photos
- Empower IoT sensors and devices to store extended details about location, maintenance history, operation, device specification, etc which may be outside existing standards
- Make web based product catalogues machine readable including specifications and purchasing instructions to enable greater automation

These activities can be performed using today's data models but only for predefined data that is already anticipated by all the applications involved. Using Amorfs, any data can be shared in any scenario. Even if it is not used by a particular application today it is still available for other applications or future requirements.

What is the data model that makes this possible?

Basic Text

To understand the underlying data model here is some example data. It is represented as "basic text" which is adequate for traditional data but uses only a fraction of the expressive power available.

Basic text is inspired by RDF 1.1 Turtle[1] which was developed as part of the semantic web movement in the mid 2000s.

```
Bible verse [Seek and ye shall find
             + book [Matthew | Mt ]
             + chapter [7]
             + verse [7]
             - version [King James Version | KJV ]
             - study tool [<biblehub.com/matthew/7-7.htm>,
                           <biblestudytools.com/matthew/7-7.html>,
                           <topverses.com/Bible/Matthew/7/7>
                          ]
            ]
```

In this example a Bible verse is recorded with its scriptural reference and some relevant study tools. Choosing something unusual illustrates how easily Amorfs can adapt to meet any requirement. The basic text captures the associations between underlying concepts as well as any expressions that could be used to represent them.

An easy way to read the basic text above is to substitute the symbols for specific phrases. The following symbols operate as boundaries between concepts.

symbol	*phrase*
+	Always has a (intrinsic association)
-	May have a (contextual association)
[...]	Which is
,	And

An additional symbol operates as a boundary between expressions representing the same concept:

symbol	*phrase*
|	Or

Using these substitutions the example above can be read as "The Bible verse which is 'Seek and ye shall find' always has a book which is Matthew, a chapter which is 7, and a verse which is also 7. It may also have study tools which are..."

The basic text contains all the necessary information items and the structured relationships between them. The data structure, field names and field values are all provided together in one place. Everything the application needs to parse and extract relevant information is provided.

Here is what the same data could look like represented as a form (figure 1).

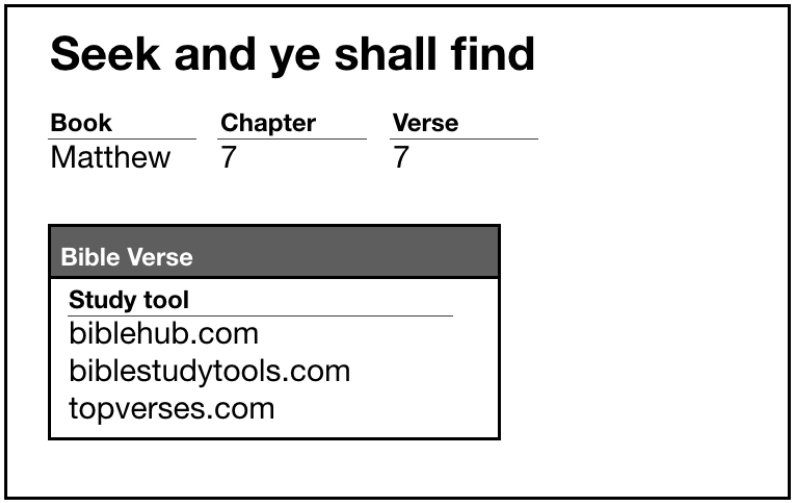

Figure 1. Basic text converted to a form

[1] https://www.w3.org/TR/turtle/

A core capability of Amorfs data is nesting. Any concept can be associated to additional details. In basic text this is represented by using nested square brackets. For example additional details about the book of Matthew can be included:

```
book [Matthew - title [The Gospel of Matthew]]
```

Any object in basic text can be enriched with additional detail. ed. nesting data inside square brackets down to as many levels as required. This is physically modelled in the underlying concept structure. Amorfs concepts and expressions are the building blocks that form the heart of an Amorfs data structure. Let's look at what they are and how they represent data.

A key feature of the Amorfs format is that it demarcates between abstract concepts and the physical expressions used to represent them.

A Bible verse, for instance, is an abstract concept. It can only be expressed in terms of other concepts like a book, chapter and verse. Or as text in one of hundreds of languages. These are all different and valid representations of the same underlying concept.

Data Model

The basic text example illustrates some of the capabilities of the underlying Amorfs data model, which operates at a conceptual level. Where possible concepts are linked to explicit expressions which are used to represent them. The model also captures the relationships that concepts have to each other and this provides the ability to capture nuanced meaning.

Concepts and expressions

Concepts are the building blocks of the Amorfs data model. Every concept represents a unique idea. There is no distinction between parameters, values, attributes, data and metadata or traditional constructs used to compute with data. Concepts are abstract points in an infinite knowledge graph and meaningless on their own.

Concepts are inherently temporal (located in a timeline), either explicitly, or implicitly based on their association to other time-bound concepts. The temporal assignment for a concept can be as vague or precise as necessary and captured as a beginning, ending, period, or event.

The model also contains the physical expressions that can be used to represent concepts and make them readable. These could be text, numbers, multimedia, web links (IRIs) etc. Expressions are explicitly linked to concepts so they can be represented to human and machine users (figure 2).

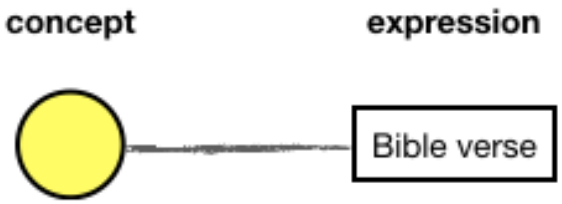

Figure 2. A concept with a linked expression

Typically a concept will be represented by one or more expressions. An expression represents an individual concept for machine and human users. There may be short form and long form texts, numbers, multimedia, web links or any digital format. If a linked expression is ambiguous then the associated concepts are available for clarification.

Languages can be supported by using different word expressions for the same concept. This isn't necessary for names and places if the proper noun remains the same between languages. In this way multiple languages can be supported

without compromising the readability or integrity of the underlying concept structure.

Being able to express the same information in many different ways means that the user experience can be optimised for a particular device or channel. Chatbot voice systems can operate with text and sound, while screen based interactions can make use of images and video. Machine users can consume external references to ontologies and standards by using IRIs. They can evaluate all of the linked and associated expressions to resolve ambiguity and assess potential matches.

Associated Concepts

Very importantly, concepts can associate other concepts. These associations are critical to capture the relationships between ideas necessary for context and ultimately meaning. This is a fundamental mechanism in an Amorfs structure. It is the essence of its expressive power. The concept that establishes the association is always more specific than the concepts it associates. This capability delivers levels of precision in the way information is captured.

An associative concept always combines three other concepts in a specific order: in a subject-verb-object relationship to one another (figure 3). This is a fundamental construct of natural language and cognitive science[2], and makes Amorfs structures highly intuitive because they mirror the way our own brains organise knowledge.

Basic Knowledge

The vast majority of digital data stored today relies on related metadata to be interpreted. The metadata is coded into applications and database structures. If the stored data becomes orphaned by being separated from its metadata then it is no longer readable. This makes it hard to share data between applications and users. An entire industry has grown around the need to integrate disparate systems by transforming data from one format to another.

Amorfs solves the orphaned data problem by reuniting data and metadata in a single package. It eliminates the artificial distinction between metadata and data by treating all data the same. A concept can be a "parameter" in one context and a "value" in another. This means that Amorfs data can be interpreted without knowing anything about the application where it was originally created. All the necessary information is stored together.

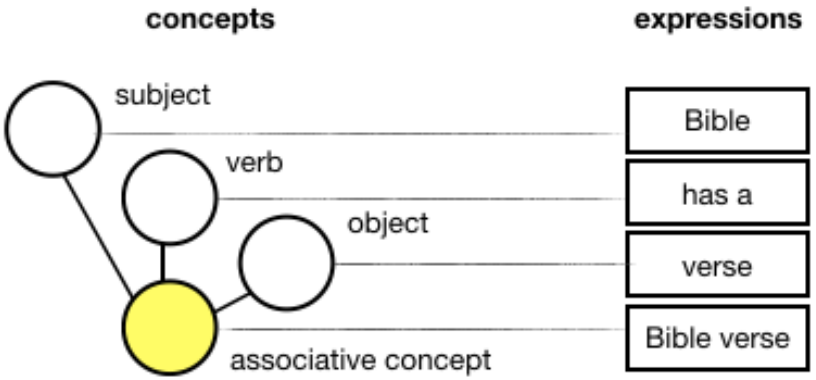

Figure 3. Concept structure with associated concepts

Incredibly, the parameter/value relationship between data and metadata can be represented in Amorfs using just two verb phrases:

- has a
- which is

[2] https://en.wikipedia.org/wiki/Subject%E2%80%93verb%E2%80%93object

These are referred to as the two basic verb phrases, and support all the concept structures represented in the basic text presented in this paper.

Nested concept structures utilising the two basic verb phrases are capable of representing any data from traditional data formats.

These two verbs only scratch the surface of the expressive power available in a full Amorfs model. However combined with nested concept structures they are a stepping stone for converting today's digital information into a universal format. It can then be leveraged into richer structures utilising additional verbs to provide the expressive power needed for the next generation of machine intelligence and cognitive processing.

Context

Some information is relevant no matter what. Your date of birth, for example, is always relevant whether you are at school, applying for a job or collecting welfare payments.

On the other hand, some information only makes sense within a particular context. Your student id, for example, only makes sense in relation to the corresponding school. Over a lifetime you may have many student ids because they change when you move school.

This means that birth date is associated directly with a person, but their student information needs to be associated together with the relevant school. Traditional structures are not good at capturing this distinction, and yet context is a critical for interpreting information correctly.

This next example of a snippet from a fictitious university record for a prominent English scholar shows the difference between contextual associations and intrinsic ones.

```
university [Oxford
             - fellow [C. S. Lewis
                       + birth date [19 Nov 1898]
                       - college [Magdalen College
                                  - address [ + city [Oxford]
                                              + postcode [OX1 4AU]
                                              + country [UK]
                                            ]
                                 ]
                       - photo [<upload.wikimedia.org/wikipedia/en/
1/1e/C.s.lewis3.JPG>]
                     ]
           ]
```

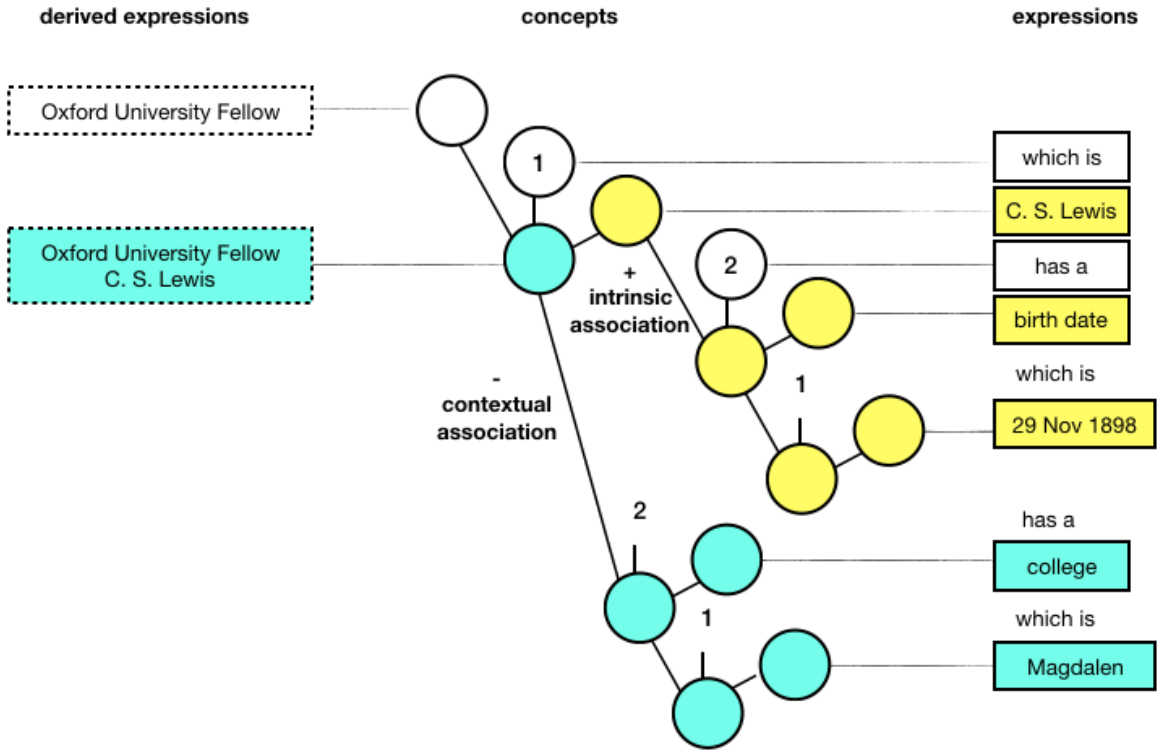

Figure 4. Intrinsic and contextual concept associations

The birthdate for C. S. Lewis is associated using "+" because it is relevant even outside the context of his tenure at Oxford university. His residence at Magdalen College on the other hand is associated using "-" because it is very much part of the university context.

The underlying concept structures reflect the distinction (figure 4). The intrinsic birthdate concepts are associated directly to C. S. Lewis whereas contextual details like his college information is attached indirectly. Notice how the "+" and "-" symbols in basic text generate different context structures to distinguish between intrinsic and contextual associations.

Intrinsic associations are the identifying characteristics or unique keys that set a concept apart from all others. If a concept from merged data shares the same intrinsic characteristics then there is a high degree of confidence that they are the same and can be merged.

In the Bible verse example, the book, chapter and verse references were intrinsically associated to the verse concept. These never change. The text on the other hand was shows as contextual. It will be different depending on contextual factors like language and translation version.

Implied Concepts

Amorfs concept structures make it easy to represent the implied contextual nuances that are difficult to capture in traditional metadata or express precisely.

The Bible verse described in the first example contained some assumptions about the verse being in english and from the King James translation. This was adequate for a particular purpose but a more nuanced understanding would recognises that verses can be translated into many languages and versions.

As an aside, when Amorfs data is merged from different sources a degree of "machine intelligence" is required to rearrange concepts into the structure that is most fit for purpose. In the process nuance may be lost from received data, or added to retained data. This process bears uncanny similarities to human learning and data acquisition.

Here is amore nuanced example showing how the Bible verse can be expressed in more than one version.

```
Bible verse [
             + book [Matthew | Mt ]
             + chapter [7]
             + verse [7]
             - text [Seek and ye shall find - version [KJV],
                     Seek and you will find - version [NIV]
                    ]
           ]
```

In this example no expression is provided for the Bible verse itself (figure 5), just the associated concepts for book, chapter verses and text. In the underlying model a concept for the verse is still generated. It is just not linked directly to an expression. This is an implied concept.

The new concept, not being explicitly linked with an expression, can only be represented in terms of other associated concepts. An expression can be derived from those concepts and their linked expressions - something like "Matthew 7:7". Derived expressions allow implied concepts with no linked expression to be represented.

A very common example of an implied concept is an address. Conceptually an address is one specific location but it's typically expressed in terms of other concepts: street number, street, suburb, postcode, country and so forth. It can also be expressed in other ways, such as a GPS location or map reference.

```
address[
        + street [Edwin Flack Ave]
        + city [Sydney]
        + state [NSW]
        + postcode [2127]

        + GPS location[
                       + latitude [-33.844364]
                       + longitude [151.062145]
```

```
                    + elevation [14m]
                  ]
    ]
```

In the text above the address item is left empty. Instead, associated concepts for street, city, state, postcode and GPS location are provided. A concept is still created for the address but it won't have any linked expression. The GPS location uses a similar construct. The GPS location item is left empty and associated concepts for latitude, longitude and elevation are provided. The resulting concept structure is shown below (figure 6).

Derived Expressions

The table in this section provides some simple rules to derive an expression for an implied concept by using associated concepts and the expressions linked to them. Other forms of expression like audio and video can also be linked and processed by appropriate algorithms to support different modes of interaction. Keeping the underlying concept structure independent from supported modes of expression allows innovation on new technologies without losing the integrity or backward compatibility of data.

Operation	Basic Text	Construct (SVO)	Inverse Construct	Derived Expression
Identification	**A [B]** *college [Magdalen]*	A which is B *college which is Magdalen*	B is a A *Magdalen is a college*	B A *Magdalen college*
Attribution	**X - Y** *Oxford - fellow*	X has a Y *Oxford has a fellow*	Y of X *fellow of Oxford*	X Y *Oxford fellow*
Possession	**X - Y [Z]** *Oxford - fellow [C. S. Lewis]*	X has a Y which is Z *Oxford has a fellow which is C. S. Lewis*	Z is a Y of X *C. S. Lewis is a fellow of Oxford*	Z (X Y) or X Y Z *C.S.Lewis (Oxford fellow) or Oxford fellow C.S. Lewis*

More advanced computational linguistics will be able improve on these simple constructs.

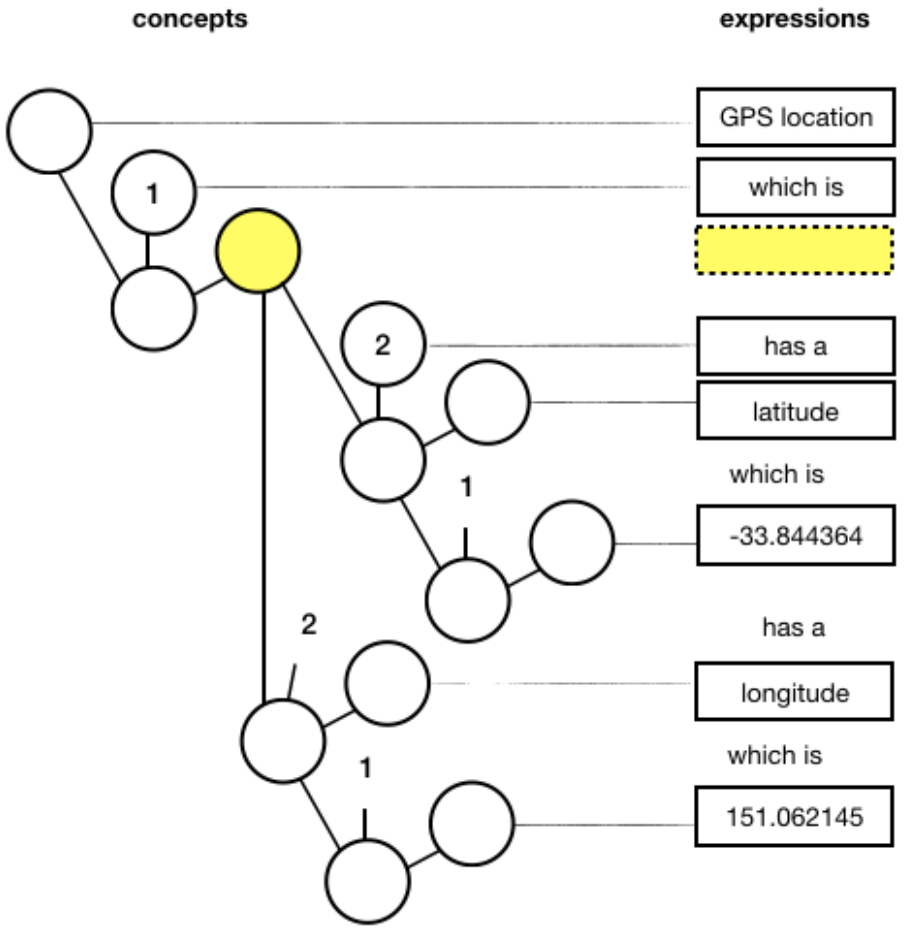

Figure 6. GPS location is an implied concept

Conventions

The rich expressive power available in Amorfs concept structures makes it possible to represent similar information in different ways. This is in stark

contrast with the rigidity of traditional data formats and encourages greater sophistication in the ways data can be consumed.

One reason for differences is the level detail. Applications will capture greater detail for information in specialist areas and less detail for periphery areas.

Another reason is perspective. Subject-verb-object sequence is determined by the context of the observer. Applications will organise information differently according to the functions being performed.

These differences are also evident in natural systems and suggest that Amorfs structures more closely reflect human knowledge representation that traditional data structures.

Some strategies are necessary to make processing as deterministic as possible by minimising unnecessary differences.

1. General to specific

Sometimes the ordering of concepts is not obvious. Does a state have a city or does the city have a state? A helpful principle is to move from the general to the specific. Since the state will likely have many cities it should be the subject for the association. One state has many cities. One set of parents has many children. One employer has many employees.

This principle makes it easier to include the relevant associated concepts for context when exporting a subset of data . Including associated parent concepts and their intrinsic associated concepts in an exported package will make it easier to incorporate the information into target concept structures by matching them to existing concepts.

Applying this principle to an address shows that the original basic text was an over simplification. Working from general to specific would link the street representation all the way back to a national context, which is actually much more meaningful and creates a useful geographic reference framework as each new address is added.

```
country [Australia
         - state [NSW
             - postcode [2127
                   - suburb [Newington
                         - street [Edwin Flack Ave] @efa
                            ]
                        ]
                  ]
       ]
```

Notice how @efa receives a reference to the concept linking Edwin Flack Ave as a street up to its parent suburb, postcode state and country. The @efa pointer can now be used as the address and even assigned a GPS location.

```
address [@efa
         + GPS location [
                         - latitude [-33.844364]
                         - longitude [151.062145]
                         - elevation [14m]
                        ]
        ]
```

2. Familiar to unfamiliar

Often, however there is no obvious direction for a relationship. Does a wife have a husband, or a husband have a wife? In this case the subject should be the most familiar or best known concept. If only the wife is known then the most salient fact is that she has a husband. In the event that both concepts are frequently referenced then it's appropriate to generate two complementary associations, one for each direction.

The freedom to represent similar concepts in different ways forces consumers to apply a degree of intelligence when interpreting query results or ingesting new data. The patterns that emerge from analysing aggregated Amorfs data sets will

be useful for reasoning about source system priorities, identifying potential discrepancies, refactoring misaligned constructs and predicting user responses.

Here is an example of a specialist Bible verse representation representing a more sophisticated understanding of available translations and verse structure.

```
bible
- text [KJV text
        - version [KJV | King James Version |
                   KJB | King James Bible |
                   AV | Authorised Version]
        - language [english]
        - copyright [public domain]
        - first published [1525],
        NIV text
        - version [NIV | New International Version]
        - language [english]
        - copyright [ - year [1978]
                      - owner [Biblica]
                    ]
       ]
- [new testament
   - book [Matthew
           - chapter [7
                      - verse [7
                               - KJV text [Seek and ye shall find]
                               - NIV text [Seek and you will find]
                              ]
                     ]
          ],
   old testament
  ]
```

Technical Features

Superficially the structures resemble semantic web constructs like RDF triple sets[3] and innovations like IRIs[4], semantic schemas[5] and Princeton's word net web ontology can be utilised as disambiguating expressions making data easier to normalise automatically.

Unlike the semantic web technologies developed to date, however, two new characteristics underpin Amorfs capabilities: abstraction and recursion.

Abstraction is used to extract knowledge from natural language and represent it explicitly as concepts. This allows more than one representation to express the same concept (e.g. synonyms, or one idea in different languages). Traditional data formats rely on external metadata and transformation rules for the context needed to interpret it but Amorfs brings that context into the structure itself, making it meaningful in its own right, regardless of the application that creates or consumes it.

Recursion gives Amorfs concepts the ability to represent subtle nuances and precise meanings. The concepts recurse by referencing one another to create new concepts, which themselves can be referenced. Although simple to implement, this means the same set of vocabulary items can be represented many different ways, reflecting things like relative importance. Amorfs offers new dimensions of expressive power for representing knowledge across digital ecosystems and makes explicit the differences in perception that human cognition exhibits. The expressive power means information stored in existing data formats can be trivially converted to Amorfs.

Conclusion

A prerequisite for machine intelligence is a strong knowledge management foundation. It must allow intelligent processes to acquire novel concepts and

[3] https://www.w3.org/TR/rdf-concepts/

[4] https://en.wikipedia.org/wiki/Internationalized_Resource_Identifier

[5] e.g. schema.org, FOAF, etc.

support cognitive phenomena such as automaticity, nuances of meaning, attentional focus, conscious and unconscious operational modes and so forth.

Today's data models have barely moved from their roots in mainframe computers fifty years ago. Applications that cannot share data easily are a huge drag on innovation and hold back the step change required to realise aspirations for smart cities and ambient intelligence.

Bringing the focus on to knowledge representation and finding ways for data to be meaningful in its own right, regardless of the application where it originated, can finally break mindsets free from the mainframe computing paradigms inherited from previous generations. Monolithic everything-connected models are inefficient and unnatural, serving only those with an agenda to monopolise data collection. True digital democracy means leaving the central planners behind and empowering data makers to be owners too.

Much of the hype around block chain solutions is based on the misconception that they set data free from applications, allowing it to be used anywhere. For that to be possible a shared data model must still be agreed between all participants.

Adopting an Amorfs data model keeps all options open regarding the data that can be supported either now or in the future.

Terminology

Amorfs - a concept-based data model

Association - three concepts in a subject-verb-object relationship

Basic text - a subset of Amorfs data constrained to basic verb phrases shown as formally structured text

Basic verb phrases - "has a" and "which is"

Concept - an idea or association, as opposed to an expression

Concept structure - a network of associated concepts

Derived Expression - expressions linked to associated concepts used to represent an implied concept

Expression - data used to represent a concept to users

Has a construct - basic verb phrase used to associate a parameter with a traditional data object or class

Implied Concept - a concept with no linked expression of its own

IRI - Internationalised Resource Identifier - used as a machine readable expression to assist with disambiguation

Link - connection between a concept and a corresponding expression (many to many)

RDF - Resource Description Framework - a predecessor data model that associates expressions and inspired the semantic web but doesn't demarcate concepts or support recursion

Which is construct - basic verb phrase used to associate traditional values with parameters