Title

Implementing Controlled Vocabularies for Computer Game Platforms and Media Formats in SKOS

Journal

Journal of Library Metadata, 16(1)

ISSN

1938-6389 1937-5034

Authors

Kaltman, Eric
Wardrip-fruin, Noah
Mastroni, Mitch
et al.

Publication Date

2016-06-21

DOI

10.1080/19386389.2016.1167494

Peer reviewed

Authors

Eric Kaltman, Graduate Researcher, University of California, Santa Cruz, ekaltman@soe.ucsc.edu

Noah Wardrip-Fruin, Associate Professor of Computational Media, University of California, Santa Cruz, nwf@soe.ucsc.edu

Mitch Mastroni, Undergraduate Researcher, University of California, Santa Cruz, memastro@ucsc.edu

Henry Lowood, Curator for the History of Science and Technology, and Film and Media Collections, Stanford University Libraries, lowood@stanford.edu

Greta de Groat, Metadata Librarian, Stanford University Libraries, gdegroat@stanford.edu

Glynn Edwards, Head of Technical Services (Special Collections), Stanford University Libraries, gedwards@stanford.edu

Marcia Barrett, Head of Metadata Services, University of California, Santa Cruz, University Library, barrett@ucsc.edu

Christy Caldwell, Science and Engineering Librarian, University of California, Santa Cruz, University Library, caldwell@ucsc.edu

Keywords

Computer games, video games, controlled vocabulary, semantic web, media format, platform

Abstract

This article outlines the creation of computer game platform and media format Linked Open Data (LOD) controlled vocabularies by the Game Metadata and Citation Project (GAMECIP). We discuss the need for more consistent and accurate information in computer game library records, define what we mean by "platform" and "media format", and then elaborate on our research process and issues encountered along the way. Our vocabularies were constructed with the Simple Knowledge Organization System to take advantage of its properties and hierarchical structure. After describing our LOD schema in detail, we conclude with a discussion of future work and a call for more collaboration.

Introduction

As computer and video games, and general software have worked their way into library collections over the past few decades, cataloging standards and collection management practices have not adequately kept up. The Game Metadata and Citation Project (GAMECIP), a collaboration between the University Libraries at Stanford and the University of California, Santa Cruz (UCSC), is aimed at relieving confusion in library records that describe game software, and helping ensure better access to legacy software through authority control and guidelines for game cataloging and citation. Through the progress of our work, which was initially aimed at new recommendations for game description in discovery metadata, we realized that there was no authoritative source for computer game platforms or for the media formats on which they store their content. This paper describes

the GAMECIP team's development of Linked Open Data (LOD) controlled vocabularies for computer game platforms and media formats and the issues encountered along the way.[1]

Many OPACs and other discovery systems do provide descriptive information related to system requirements, media formats and other technical details, but either lack explicit authority control, or provide descriptors that do not support adequate identification. This is a problem as information about a game's platform and format is paramount for a user to locate that resource, and for a library to provide information or equipment to enable its use. As such, GAMECIP's description and cataloging recommendations forced us to address this lack of specificity in game catalog records (deGroat et al. 2015). Stanford and UCSC have some of the most extensive university library game collections in North America and they also suffered from a lack of coherent authority control.

Clarity in game records is paramount to ensuring future access to historical works that are dependent on legacy computational systems (McDonough et al. 2010). Whether present in physical form in a collection (as optical discs, floppy disks, etc.) or as organized bits of data in a digital library, any computer game needs correct description to ascertain how it can be represented on legacy or modern hardware. Our specific focus on computer game platforms and media formats is based on the needs of our own collections. However, the

[1] For the rest of this work, we refer to "computer and video games" as "computer games", as we are referring to software that is computational and feel that in the future "video games" as a term will become more and more anachronistic.

vocabularies and methodology below can be directly applied to any collection of software objects, as many of the more general computing platforms also support game software.

Our vocabularies aim to help non-specialist catalogers and other library staff organize consistent and historically secure descriptions of the games in their collections. A unified reference point for each platform and media format would improve game records and allow future patrons and scholars more informed access to software works. To further this unified ideal, we chose to create our vocabularies as Linked Open Data (LOD), specifically as Simple Knowledge Organization System (SKOS) concepts. This provided three significant advantages. First, using Semantic Web constructs allowed us to directly link computer platforms and their media formats, making it easier to include terms from the vocabularies into records. Second, it provided a means of 'future-proofing' our vocabulary, as we expect Semantic Web functionality and integration in libraries to increase in the coming decade (Southwick, 2015). And third, Semantic Web technologies align with the larger descriptive goals of our project. Our metadata element recommendations are also modeled with SKOS Concepts, and we are working to expand our descriptions of platforms and media formats into a larger and more detailed technical ontology for computer games. Therefore, we can easily link our SKOS vocabularies to our expanded Resource Description Framework (RDF) ontology and our metadata schema when they become available.

This article begins with a detailed description of "platform" and "media format", including a discussion of the criteria used to delineate between vocabulary terms. This is followed by a description of our organizational methodology and research plan and how that led to our

current LOD vocabulary terms. Our SKOS schema and property usage is then presented, along with the integration of our terms into the MARC standard. We conclude with future plans for expanding the vocabularies and how future digital libraries can benefit from this work.

Definition of "Platform"

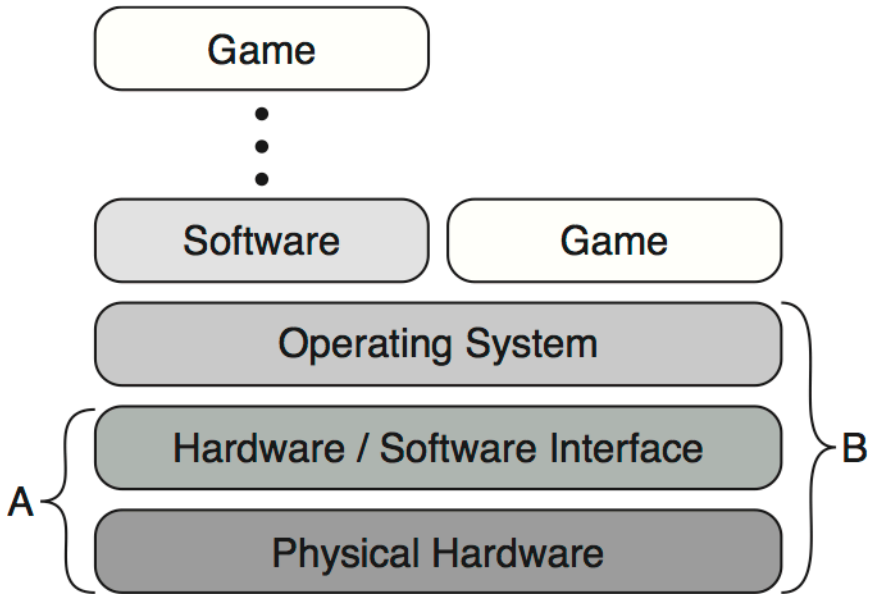
The definition of a computational "platform" is notoriously confusing and indistinct, even among experts in the game studies and computing communities. Game scholars Ian Bogost and Nick Montfort, in the preface to their edited series on "Platform Studies," do not provide a definitive definition of the term, but their discussion illuminates some key considerations of our definition below.

"A platform in its purest form is an abstraction, a particular standard or specification before any particular implementation of it. To be used by people and to take part in our culture directly, a platform must take material form...This can be done by means of the chips, boards, peripherals, controllers, and other components that make up the hardware of a physical computer system. The platforms that are most clearly encapsulated are those that are sold as a complete hardware system in a packaged form, ready to accept media such as cartridges...In other cases, a platform includes an operating system. It is often useful to think of a programming language or environment on top of an operating system as a platform, too. Whatever the programmer takes for granted when developing, and whatever, from another side, the user is required to have working in order to use particular

software, is the platform. In general, platforms are layered—from hardware through operating system and into other software layers—and they relate to modular components, such as optional controllers and cards.“ (Bogost and Montfort, 2009)

They are intentionally broad in their designation of platform; it is apparently up to each author to argue for a system’s existence as a “platform” for computational expression. Bogost and Montfort’s definition does, however, contain specific ideas that we incorporate into our own. Platforms are abstractions of computer hardware and software. They are layered, accept and interpret media, and exist as a network of components that, when unified, enable a user to interact with a specific set of software. Obviously, if we are to organize concise and identifiable designations for the terms in our vocabulary, we need some framework for disentangling the technical complexity of platforms and cementing coherent relationships between them. Their “aboutness”, their distinct ontological identity is necessary to guide and support our vocabularies (Taylor, 2008).

Figure 1. Platform Abstraction Layers Diagram



In figure 1 above, we present a simplification of the common layers of abstraction in a computational system. Each layer maintains a specific set of interfaces with those above and below, however their specific interactions are outside the scope of this article. The important thing to note is that a platform can exist as multiple combinations of these layers, and those different combinations have a distinct place in the hierarchy of our vocabulary terms.

The bottom layer, **physical hardware**, includes the physical components needed to form computational systems. In addition to a processor, memory, storage, and display components, this layer also includes physical media interface devices and any peripherals needed to provide user input (keyboard, mouse, game controller, etc.) Above that is the **hardware / software interface**. This layer allows for communication between program

code and the physical hardware, and can take many different forms. In our simplification, this layer represents the most basic interface a programmer can use to interact with a platform. It is also the initial abstraction that allows for the manipulation of electronic signals into a computational expression. The **operating system** layer is next level and is essentially a software program that manages the hardware / software interface for other software programs. Many of the basic tasks in programming rely on highly repetitive and complex management of the interfaces to various hardware components. An operating system abstracts away those processes and allows for the creation of even more complex software. That **software** layer sits atop the operating system, and can either represent game software itself, or provide another abstraction for a **game** layer above. In actuality, the layers of software can extend for many layers (theoretically infinite) above the operating system. However, for the purposes of our definitions, we will stop here.

A platform consists of a specific combination of layers for which a game is designed and on which it can be represented to a user. We delineate two basic types of platforms: **hardware** and **software**. Each type can encapsulate different combinations of the layers in our computational system diagram. A hardware platform is either a combination of physical hardware and a hardware / software interface (designated as hardware A in the diagram), or a combination of physical hardware, a hardware / software interface, and an operating system (hardware B). The second combination may be a little confusing, considering that an operating system is software; however, there are many computer game platforms in our vocabulary that are almost always explicitly referred to, and labeled in reference to their physical hardware and form factor, without mentioning their operating systems. This is the

case primarily with platforms that are dedicated to gaming and that accept a single media format. In these situations there is an operating system present, but the specifications of the platform are controlled at a level where the operating system is not relevant to the average user. We consider both hardware A and B as a single kind and use the designation of hardware platform to cover both concepts in the remainder of this paper.

A software platform is either an operating system, or single or multiple software layers. Our distinction of operating system as a software platform is based on the common requirements present on software packaging. In many cases the operating system is the major component necessary to run a game, since it is assumed that if the operating system is present, it will have access, through abstraction, to a correct set of hardware / software interfaces and physical hardware. A software layer as platform is simply any abstraction above the operating system that provides a holistic interface to run a game or other program. Web browsers can be considered a software platform for web-based games, since the specific operating system supporting the browser is irrelevant to the content contained within it.

A computer game platform is then any computational system capable of supporting an instance of game software. Since most general computing devices also have game software, they are also considered computer game platforms for the purposes of our vocabulary.

Media Formats and Compatibility

A computer game media format is any collection of game data encapsulated in a physical medium that requires a computer game platform for presentation to and interpretation by a human being. Media formats generally consist of some means for non-volatile storage of data, and a physical interface for connecting to a platform. For example, a floppy disk is a magnetic storage disc (non-volatile storage), enclosed in a square plastic sheath (physical interface) that can be inserted into a floppy disk drive (connection to platform) and interpreted. Other formats, like optical discs, are a singular combination of storage (reflective markings on a substrate) and physical interface (plastic disc).

The relationship between a media format and its platform is essential to the delineation of the platform terms in our vocabulary. A game resource in a collection consists of a media format storing the game's data and complementary packaging that, hopefully, provides some information about the resource's system requirements or its host platform. Because the game resource, and by extension, the structure of its media format dictate the platform that supports it, a platform is essentially a specific computational configuration that supports a class of media formats. Looked at this way, a platform, whether hardware or software, is defined through the media formats or software that it can support and not solely through its own technical components or interfaces.

A platform that can interface with a media format and interpret its data is said to be **compatible** with that format and data structure. Compatibility is the main criterion we use

for deciding whether the differences between two computational systems require them to be considered different platforms. Many computer game platforms (specifically hardware platforms) only accept a single physical media format, which makes their compatibility easier to establish. It is also another reason that their operating system is rarely noted, as the single format they accept, with its specific data configuration, abstracts away that dependency from the user. Hardware platforms that support a single format are defined as **dedicated** computer game platforms in our vocabulary, whereas platforms that can support multiple media formats, and also non-game software, are defined as **general** computational platforms.

Establishing a game resource's media format and platform also cements its compatibility with any future system that might be able to interpret it. As data is migrated from physical formats into future digital libraries, maintaining an identifying link to a resource's initial compatible platform and format will make it easier to interpret the resource in the future. Additionally, many general computational platforms can emulate (imitate through software) the hardware / software interface to other systems. A focus on identification of platform and media format through compatibility will also help future institutions know what items in their collections are currently usable, and allow knowledgeable patrons the ability to search based on the types of compatible software they can use.

Vocabulary Development

The creation of authoritative terms for our vocabularies is the result of a great deal of research and synthesis of information distributed across the Internet, our own collections, and the expertise of our project team. This section outlines the methods we used to research, disambiguate, and validate the items in our current vocabulary, which consists of a pilot-set of more common platforms and formats. We discuss the initial accumulation of information, how we whittled it down to the more manageable pilot-research set, and some issues encountered along the way. As stated, our goal for the vocabularies was to get them in a state that would be consistent and viable for non-specialist cataloging, as well as easily integrable into LOD structures. The consistency requirement involved determining the level of specificity and detail with which to treat each platform and format and how that specificity affected the general compatibility of different systems. Luckily, the hierarchical nature of SKOS obviated some of these issues (as discussed in the next section), but the vast range of computational systems and formats did initially catch us off guard.

To begin the investigation, we decided to collate all the information we could find about how computer game platforms were referenced, both within our own collections, and online. We also chose, initially, to only focus on hardware platforms to reduce the scale of the research. Our initial listing aggregated information from our finding aids, and from a collection of online resources that we thought would cover most of the colloquial and technical names for platforms likely to be encountered by a potential vocabulary user. For the online sources, we split them into three categories, Wikipedia, game community sites,

and commercial sites. Below is a table of the number of individual platform references for the sites we reviewed (Wikipedia may in fact contain significantly more references, but their explicit categorization of computer game platforms is where we derived the number shown), along with statistics on the collections at UCSC Library and in Stanford's Cabrinety Collection.

Table 1. Aggregated Number of Potential Platforms and Formats per Collection[2]

Source / Collection	Platforms	Operating Systems (if separately listed)	Media Formats (if separately listed)
Wikipedia[3]	182	Over 1000	-
Amazon	44	-	-
EBay	43	-	-
GameStop	29	-	-
Giant Bomb	142	-	-
MobyGames	168	-	-
PlayAsia	46	-	-
Universal Videogames List	179	-	-
Video Game Console Library	110	-	-
UCSC Library	27	13	-
Stanford Stephen Cabrinety Collection	446[4]	790	53
GAMECIP Aggregated Listing	410	-	150

We then cross-referenced all the platform names to form a unified listing of potential platforms. For each, we did further research into their media formats, peripherals, alternate names and versions, and any other salient information that could lead to better

[2] The numbers here illustrate the breadth of platforms and their number. No two resources used the same level of technical granularity; therefore the operating system numbers likely include many sub-versions that will eventually be removed. Also, many sites did not distinguish between platforms and operating systems.

[3] Wikipedia's number of platforms is specifically dedicated computer game hardware entries. Virtually every computing device ever made has an article, which is many thousands, but they are not counted here. Also, Wikipedia does record media formats, but there is no singular listing of storage media available on the site.

[4] Due to an archival setup that does not distinguish between types of hardware, the platform listing for the Cabrinety Collection is of the total number of hardware items in the collection. Potential platforms will likely be much lower (around 200).

identification and disambiguation. An example of this information is shown in Table 2 for the Atari Video Computer System, or as more commonly known the Atari 2600.

Table 2. Example research entry for a platform (before disambiguation)

Platform	Alternate Name	Alternate Version	Format	Operating System	Peripherals
Atari 2600	Atari Video Computer System (VCS); Sears Video Arcade	Atari 2600 Jr.; Atari 2800; (Japan); Coleco Gemini (clone)	ROM cartridge; Cassette tape	Atari OS	Joystick; Paddle

It became apparent due to the sheer number of potential platforms that we would need to limit the initial vocabularies to items that we could verify with our own collections, and that would be available in a general, non-specialized library collection. Also, our decision to avoid software platforms in the initial investigation proved untenable. If we relied on deriving platform identification from packaging, operating systems (like Microsoft Windows) provided the only effective means of recording compatibility for a resource.

In reconciling the hardware and software platforms derived from our initial list and filtered through our collections, we encountered two types of incompatibility that are reflected in the current vocabulary: **version incompatibility** and **region incompatibility**.

Version incompatibility refers to situations where a modification to a specific platform results in it being unable to support software created before the modification. This is most

present in the versioning of operating systems or other software platforms, where the main version number stands as an indicator of compatibility. The granularity of a versioning change can have a variety of effects on its broader compatibility with games and therefore there are many instances where the new version should be considered a totally new platform. Many systems do provide for **backwards compatibility** to allow older games to run on newer hardware or software platforms.

Region incompatibility is when a platform is modified into mutually incompatible versions based on its region of availability. This incompatibility is usually the result of business decisions to prevent cross-market sales, or because of conflicting technical standards for physical hardware components. Our vocabulary mostly consists of North American platforms, however, we included numerous international versions because of their presence in our collections.

After sorting through the various incompatibilities and researching how each of the abbreviated platforms are described on each item's packaging, we settled on a naming convention for each platform.

```
{Company / Corporation Name} {Platform Full Commercial Name} {Region (if applicable)} {Version (if applicable)}
```

An entry for a platform, in this case the Sony PlayStation 2 released in North America would read:

```
Sony (Company) PlayStation 2 (Platform) NTSC-U/C (Region)
```

Each vocabulary item is prefaced by the company or corporation responsible for its manufacture. Many packaging styles we examined feature truncated or alternate names and in some cases the manufacturer is not explicitly mentioned. We felt that future search needs of users would most definitely include the company names associated with the platforms and wanted to make sure they appeared in each record. The truncated or contemporary colloquial names available on packaging were also sometimes not distinct, so we enforced use of the full commercial name for a platform when possible. The region and version distinctions are derived from the above research into their respective compatibilities, and also from information reflecting those distinctions present on game packaging.

The media format vocabulary was a natural extension of the work on platforms. After aggregating the initial information about each platform's media formats, we decided to break those items off as a separate vocabulary. As we intended to construct the vocabulary as LOD, adding a complementary format list let us illustrate some of the relationships between platforms and media formats that would be more difficult outside of a Semantic Web structure. Since some general hardware platforms are designed to accept multiple types of formats, we could easily link each format to the appropriate platform supporting it. We culled our media format vocabulary to match the platforms in the abbreviated platform vocabulary. In some cases, a media format can have many different types of configurations (like additional storage capacity or digital rights management features) that are currently

unaccounted for in our vocabulary. We hope to expand the specificity of the media format vocabulary throughout the remainder of the project.

Semantic Web and SKOS Integration

The Semantic Web represents a rich opportunity for library discovery in the coming decade. As such, we decided to incorporate our vocabularies into Semantic Web structures from the outset. This ensures the future compatibility of our descriptive information with semantic digital libraries and semantic finding aids and allows for expression of some of the hierarchical and interrelated aspects of the platforms and formats. There are many means for incorporating new triples into the Semantic Web, and given the basic identification and disambiguation goals of the vocabulary, we felt that the Simple Knowledge Organization System (SKOS) was the correct means for conveying our vocabulary to the wider Semantic Web. SKOS is aimed at creating thesauri and is used for a whole range of controlled vocabularies. Its organization of linked SKOS Concept nodes with preferred labels directly addressed the disambiguation needs arising from our vocabulary research.

In order to get our vocabulary embedded into the Semantic Web expeditiously, we opted to use the Open Metadata Registry to host our vocabularies. The Open Metadata Registry provides maintenance and support tools for SKOS controlled vocabularies and metadata element sets. One drawback is that the Registry only supports a basic set of SKOS properties and concepts and our current vocabularies reflect that limitation. We expect our

initial, simplified SKOS vocabulary to eventually expand through the creation of a university-backed Semantic Web server that will interface with the basic vocabulary items on the Registry.

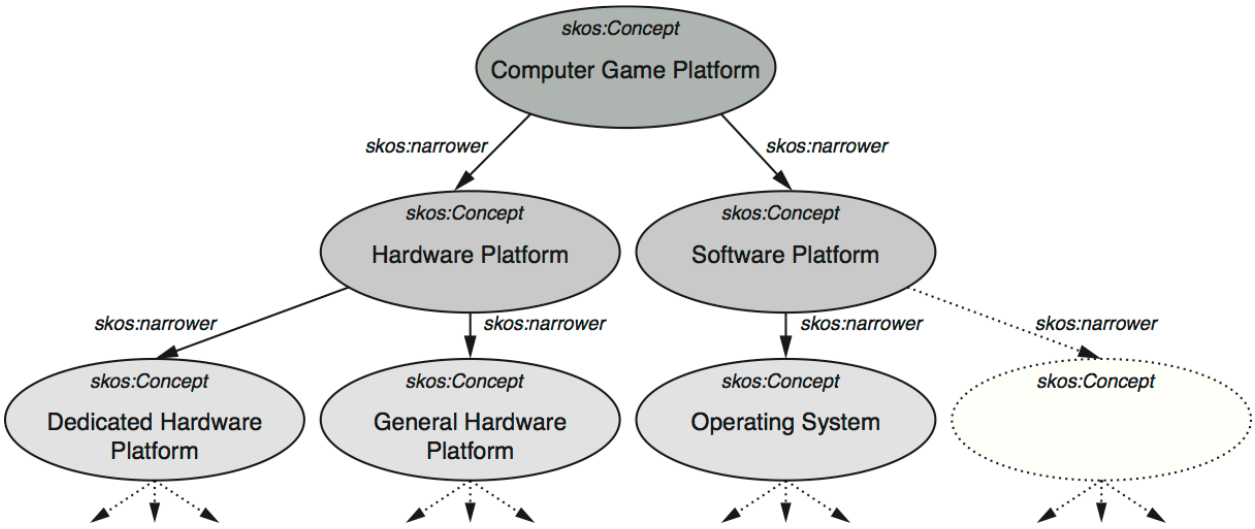
Other benefits of Open Metadata Registry, aside from hosting the vocabularies, is that it functions as a SPARQL endpoint, it enforces use and maintenance of distinct URIs for each vocabulary item, and it provides automatic download of triples in RDF-XML form. The SPARQL integration makes our vocabularies accessible to Semantic Web queries and immediately incorporates our work in the larger, searchable Semantic Web. Our initial schema used Registry provided URI, in the form of http://metadataregistry.org/XXXX/1001. In cleaning up the URIs, we decided to keep the unique, numeric identifier and transferred the URI to a new, cleaner form as: http://gamemetadata.org/uri/platform/1001 and http://gamemetadata.org/uri/media/1001. The gamemetadata.org server simply redirects all web requests back to the Open Metadata Registry hosting. In the future, we plan on creating a fully functional Semantic Web server to handle both RDF and HTML (human-readable) descriptions of the concepts in our vocabulary.

SKOS Models and Properties

Our SKOS models reflect the constraints of the Open Metadata Registry structure, and incorporate the previously discussed issues of compatibility and specificity at the heart of platform and media format identification. SKOS makes use of narrower and broader properties to signify if a concept has a more specific definition available, or is part of a

broader class of concepts. For our purposes, the terms we recommend for usage are the narrowest possible. That is, all the leaves of our SKOS hierarchical trees should be considered first for the identification of a platform or media format. We organized the hierarchy to specifically allow for the addition of narrower terms because our research has consistently revealed minor compatibility issues that could benefit from entries in the vocabulary. We expect those using the vocabulary to match against each concept's properties to find the appropriate preferred name. At this time, the platform hierarchy is already cemented, the hierarchy for media formats is less complete and will be amended greatly over the remainder of the project.

Figure 2. Platform SKOS Hierarchy



The root and umbrella concepts of the platform tree are illustrated in the Figure 2 above. Each concept node provides a basic definition of the concepts to be found in its respective sub-tree, and then partitions the platforms into software and hardware, and then again into

the specific categories discussed in the above platform definition section. Hardware is split into general and dedicated platforms, while software platforms currently have a single sub-concept of operating system. The hierarchy presented is not attempting to capture the multi-layered platform model discussed earlier, but to isolate concepts that relate best to the goal of describing the distinct compatibility of a resource. The media format listing is only split into two subcategories of general and dedicated formats as a way to signify that the general formats may relate to multiple platforms. Dedicated media formats primarily link to dedicated hardware platforms, and are easier to define. General media formats are currently a result of the limitations in our research, as there are many types of data formatting and encoding schemes that do affect compatibility but we have yet to fully classify. Regardless, we feel the current format list is a significant extension beyond any currently available to the community.

Figure 3. SKOS Compatibility Examples

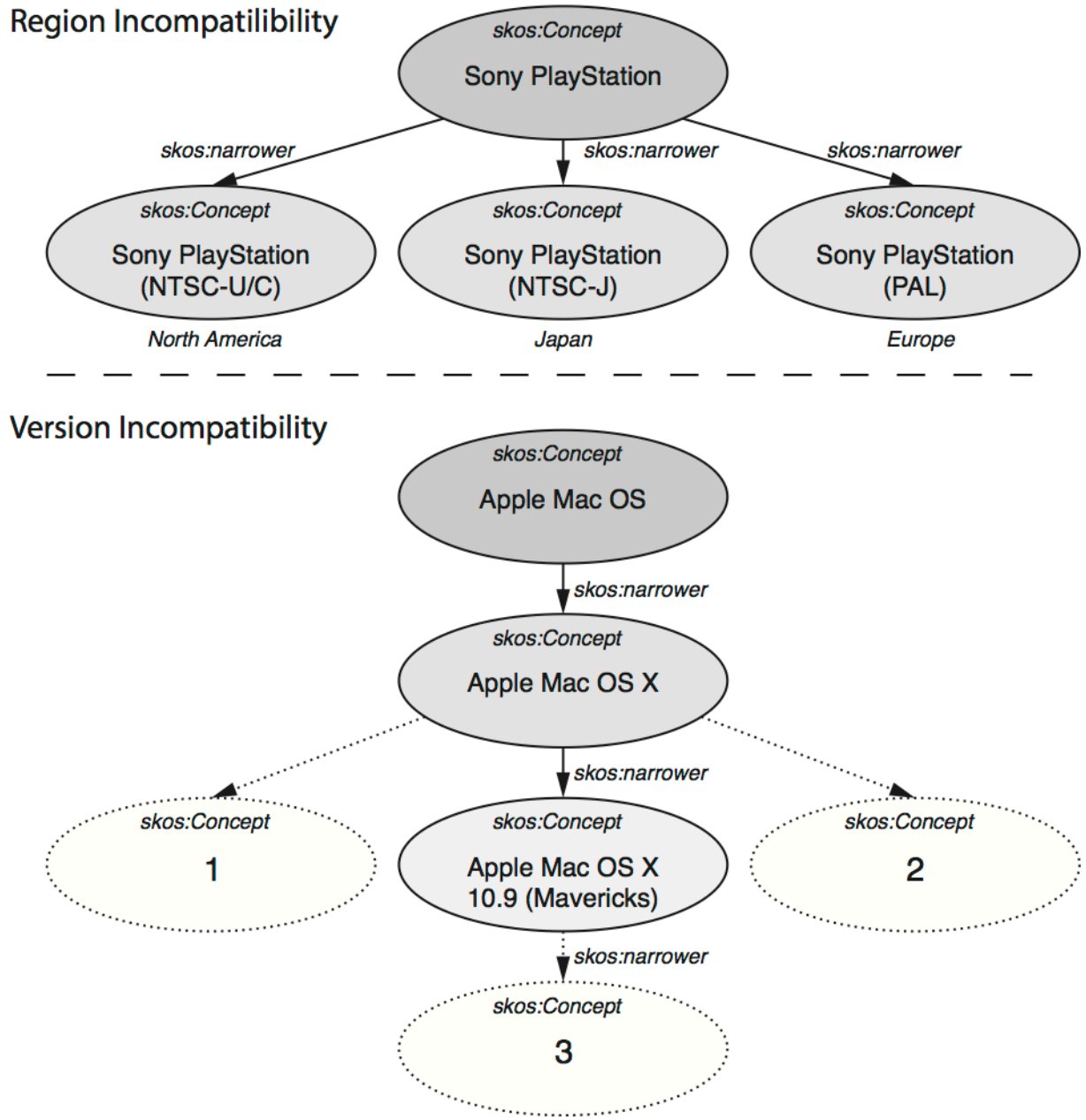


Figure 3 shows how version and region incompatibility are expressed in the vocabulary. The entry for the Sony PlayStation is a broader concept with narrower ones identifying regional variations caused by differences in their video refresh rates. When defining an operating system entry, many times a designation of its name and major version (in this

case Apple Mac OS X) is enough, but due to certain compatibility issues, more specific sub-version numbers should also be used if they are found on packaging. If multiple versions are listed, we recommend choosing the earliest version available, as that will allow future users a wider range of options for compatible systems. There is a potential for similar version and regional compatibility issues in media formats and we are working to extend our research to cover that eventuality.

Below are more detailed descriptions of each SKOS property used in our vocabularies. Each of these properties is implicitly tied to a SKOS Concept, since such Concepts represent the nodes in the SKOS LOD topology.

Computer Game Platform Properties

`<skos:prefLabel>`

This is the authoritative designation for a platform, and consists of 4 separate descriptors to help with identification and discovery. First is the corporate entity associated with the platform and then its canonical commercial name as ascertained from previous research. Following that are two optional descriptors that only appear in entries needing further disambiguation. The first optional parameter is the regional designation or standard with which the platform is compatible. This information is also generally recorded in MARC 347 $e according to RDA standards on regional encoding (RDA 3.19.6) and broadcast standards (RDA 3.18.3). We attach it here to make sure that regional compatibility is explicitly noted along with the platform. This also ties the regional compatibility to an accepted international standard since many game systems have been linked to the broadcast

standards required by a region's televisions. The second optional parameter, which is mainly for software platforms as they have more distinct version compatibility issues, is the canonical version number for each major release. In some instances, certain versions are left off of our pilot vocabulary due to our focus on more commonly held game resources. Additionally, for the major, contemporary versions of Apple Mac OS and Microsoft Windows, we include significantly more explicit versions due to their larger presence in our collections. A fuller or expanded listing of our vocabulary would most certainly include all known compatibility classes for every software platform.

`<skos:altLabel>`

The alternate label identifier attempts to capture well-known variants of a platform's name. Many computer game platforms have numerous colloquial and abbreviated distinctions, in some cases only numbers, as in the "2600" for the "Atari Video Computer System."[5] Still, since this vocabulary is aiming to standardized platform terminology, all alternate names listed are mostly for the benefit of non-experts simply trying to match a label on the packaging to a label in the vocabulary. This property generally has numerous entries.

[5] The Atari Video Computer System (VCS) represents a specific edge case in which the platform is now mostly known as the "Atari 2600". This was done to disambiguate the VCS from the Atari 5200 after the 5200's release. As a result, the vocabulary uses the "Atari 2600" as the preferred name for the platform. Sometimes a colloquial name becomes canonical.

`<skos:definition>`

This is a basic definition of the platform, including information about its time of production and basic media formats. If a platform is a successor to a previous platform, or maintains some amount of backward compatibility, it is noted in this field.

`<skos:related>`

The related property is currently reserved for bi-directional links to items on our media format vocabulary. Other uses for this property could include alternate and international versions of the platform or other platforms related through backward compatibility or dependency. For instance, the property could capture the dependency between a hardware platform and its operating system, or an operating system and the software platform abstraction layers that it supports. Due to the possibilities of LOD, we expect the related property to expand in scope and potentially include a linking of conceptualizations of a similar platform across multiple websites and online resources.

Computer Game Media Format Properties

`<skos:prefLabel>`

The preferred label property for media formats is meant as an augmentation to the standard RDA carrier-types. Our labels attempted to explicitly describe each media format with reference to its commonly used name, and where applicable, the platform to which it is linked. For example, in RDA a Nintendo Entertainment System ROM cartridge would be referred to as a "computer chip cartridge", which is rather ambiguous. Our preferred label is "Nintendo Entertainment System GamePak". This ties the media format to its specific

platform and provides a clear connection and search term for any future patron or researcher. Right now, using discovery tools to search on media format types is essentially useless, but we hope with more strictly defined items, this vocabulary can provide another way to find items in software collections. For more general computing devices, we have limited the vocabulary to items reflecting different form factors, as finding compatibility between multiple versions of a general media format, like a 5.25" floppy disk, will require further research.

`<skos:altLabel>`

The media format alternate label is reserved for other common terms applied to a format, but which are either not complete or non-canonical (in that they differ from a manufacturer's definition of the format). Many formats may share similar alternate names, and we are looking means for better disambiguation on for this property.

`<skos:definition>`

The definition for a media format item will mention its dimensions and notable characteristics, as well as the general frame of its retail availability. If the format is linked to a specific platform, that platform and linkage will also be noted.

`<skos:related>`

The related identifier is currently only used to link a media format and the platform(s) that accept it. For more general media formats, this one-to-many linkage may be quite large. A second application of this property ties each media format to the RDA carrier-type with

which it is most associated. The RDA carrier-types themselves are not particularly effective in their designation of media formats and that deficiency is one of the main motivations for our media format list. RDA lists seven potential formats for a software resource and the historical provenance of these definitions is sometimes difficult to determine or correlate with currently available formats. We hope that the terms in our vocabulary, which are linked to the RDA definitions through the related property will help with future queries seeking to determine which carrier-type best fits a specific resource. We plan to release more work on this aspect of the vocabularies in the future, along with a historical discussion of carrier-type provenance and development.

The vocabularies are intended to standardize the references to platforms and media formats that are already present in common library catalogs. For more forward looking metadata schemas that allow for semantic web URIs to be directly applied to a record, as with a MODS valueURI attribute like in a MODS note field, we recommend including links to our vocabularies somewhere in the record. Since this vocabulary developed as an extension from our core metadata set recommendations, we direct the interested reader to our project website for more information on integrating our terms into some other common standards.[6]

Many standards call for copying system requirements from packaging into some field in a schema. Usually, this is a verbatim copy of text on the packaging. For MARC records, the

[6] http://gamecip.soe.ucsc.edu

538 field is the general dumping ground for system requirements. We recommend that in addition to the technical specifications on the box, a cataloger match against our vocabulary items and place the appropriate preferred label text in the 538 field. Additionally, we are recommending that the 753 field also feature terminology from our vocabulary. Theoretically, new media format designations could go into the 338 field for carrier terms. We intend to apply for a new MARC source code to help add more authoritative terms to that field. Game resource packaging uses varying system requirements descriptions. We want to ensure that, as much as possible, there is some standard naming convention for at least the platform and media format component of a resource's technical specifications and system requirements.[7]

Issues Encountered

The research into platforms and media formats presented numerous research challenges and potential inconsistencies. Paramount is the lack of any explicit authority for the information in the vocabulary. Our work is a combination of personal technical knowledge and information available on a cross-section of professional and semi-professional websites. Much of the necessary information to verify full compatibility of specific game software objects is currently unavailable or needs be taken with a grain of salt. The level of expertise and specificity in the community is quite high, but sometimes inconsistencies arise that could only be settled by acquiring the correct hardware and running a specific game on it. Since this is outside of the resource capabilities for our institutions, which

[7] For more detailed information on cataloging games in MARC, please refer to our recommendations on game cataloging (deGroat et al. 2015).

specialize in game content and not hardware, we expect that it will be difficult to fully verify all historical compatibility without resorting to unverified online expertise.

Another issue for the organization of the vocabulary was deciding on what level of specificity to consider for each platform and media format, as well as creating a distinct enough definition of "platform" and "media format." As shown in Table 1, our extended list of platforms topped out at 410 discrete listings, and 150 media formats. In order to whittle these down we needed to create exclusion criteria to exempt certain potential platforms and concentrate on those that would form a more unified collection of terms. The criteria are listed in the following table, along with their rational.

Table 3. Problematic Criteria for Computer Game Platforms

Problematic Category	**Number in Listing**	**Basic Definition**	**Example(s)**
Too Broad	16	Too general and requires further clarification.	Nintendo
Non-existent	6	Incorrect source. In most cases this seems to be a product of editorial error.	Amiga ST[8]
Model Instead of Platform	2	An instance of a larger set of compatible systems.	Intellivision II
Upgrade	3	A modification of a previous platform. It may also constitute a new platform based on compatibility.	Apple IIE, Apple IIGS
Alternate Name	6	Alternate name of a platform	Acorn Archimedes[9]
Alternate Version	14	Alternate version of a platform.	Atari 2600 Jr.

[8] Combination of the Commodore Amiga and the Atari ST
[9] Umbrella term for the line of 32-bit Acorn platforms

Unreleased	18	May have been prototyped or advertised but was never commercially available.	Atari Panther
Operating System	17	General operating system and needs specific version information.	Windows
Development Platform / Framework	8	For creating other software.	Java Platform, Enterprise Edition
Built In	7	Combination device with a built-in platform.	Sega TeraDrive
Emulation Machine	5	Plays emulated games from other platforms.	GP2X
Clone	12	An illegal copy or approved regional variant of another platform.	Dendy Junior
Convergence Device	4	Designed to interface with multiple legacy formats.	Retron5
Digital Distribution System	7	A software distribution system for a different platform.	DSiWare
Peripheral Only	8	Dependent peripheral for a different platform.	Family Computer Disk System
Children's Platform	8	Advanced children's toy.	Leapster
Single Game Machine	32	Designed to specifically run a single game or collection.	Color TV Game, Game and Watch
Download Devices	10	Only plays games from a digital distribution service, no physical media involved.	PSP Go
Programming Language	1	Programming language.	Chip 8
Not Reconstructable	3	Unlikely to be physically reconstructed without significant effort. Simulators may exist.	DEC PDP-1, PLATO System
Secondary Platform	3	Designed for a primary function other than computer games.	Texas Instruments Calculators, iPod Models

Of note in the criteria is the designation of "clones" or "compatible copies." Numerous game platforms existed in various international versions. However, given the popularity of some of the platforms, companies and individuals in regions with no ability to purchase these

machines decided to copy their specifications and produce hardware "clones." For example, the Nintendo Entertainment System was not available in Eastern Europe, Russia and other USSR territories. As a result, the Taiwanese company, Steepler, released a hardware clone for the region known as the "Dendy" that supported the NES's cartridge based format and could process its data. This platform was illegal, but most of the region is historically familiar with the Dendy as opposed to the NES. Where do historically significant, yet unofficial platforms fit on this list? Many regions in Asia also supported significant hardware and software clones and rampant piracy. Should technically illegal items be inserted into an authority list based on their relevance to a particular region?

Following the issue of international piracy is another of international expertise. Computing devices, and by extension, game platforms have been created for specific regions (through globalization) and also by specific regions (as local expertise increased). This leads to the need for experts familiar with the language and computing customs of a specific region to make sense of its platforms and media formats, and their complementary histories. In addition to North America and Western Europe, our extended listing features a few platforms specific to Japan, but the language barrier prevented us from extending our vocabulary to better cover that region. Since SKOS and semantic web constructs can easily integrate multiple languages into their representations, it should be possible to find resident experts in those regions. As such, a current limit of the vocabulary is that it is very Western-centric, and only includes a few popular and therefore, better researched international platforms.

General computing devices are ubiquitous and a majority of them support some game-based items. Even a TI-86 graphing calculator and its complementary BASIC programming language have active game development communities.[10] This leads to the issue of whether to include essentially any computational device capable of supporting secondary software on future extensions of the vocabulary. We do not know whether many of these platforms will become historically significant, but it is possible that future, significant developers might get their starts on platforms that would not generally fall into our conceptualization as such.

Lastly, our vocabulary has specifically excluded ambiguous, though common, terminology like "PC" and "Mac", and we cannot be sure if some of the current definitions we use may be co-opted into general terms at some future point.[11] The drive for these identifiers and definitions should be towards more coverage and specificity, but not lose sight of being approachable to non-specialists.

Future Considerations

Our current vocabularies only cover a fraction of the platforms and media formats that have been created over the past 40 years. While the abbreviated set does contain most (if

[10] See http://www.ticalc.org/pub/86/basic/games/ for a listing of current TI-86 BASIC games.
[11] "PC" and "Mac" are common terms listed on packaging and in requirements, but do little to historically situate the resource. Either term's meaning can vary depending on the date of the resource, and we decided to replace each term with an explicit description of the resource's operating system. "IBM-PC" may be present in the vocabulary at some future point, but more research is needed to disambiguate and specify it.

not all) of the platforms and formats that will be available in non-specialized libraries, there are a few concerns from the past and future that will require further work.

First, is the shrinking place of dedicated hardware platforms and physical media formats in the modern world of broadband connections and cheap storage. Many games today, including those that will probably end up being the most historically significant, will never have a physical release. As collections are digitized and placed into digital repositories, much of the descriptive metadata, including the data's original format and platform, will be historically useful and necessary. Going forward, most, if not all platforms will be software platforms. The increase in power and penetration of general computing devices into society will continue apace and the main designation allowing legacy software to function will be its compatibility as defined through its original platform or media format.

Software distribution is now primarily a born-digital enterprise and we will be dedicating future project resources to help distributors correctly identify the compatibility requirements for their software. This issue is most dire for mobile computing devices, as single companies have almost total control over their operating system versioning, compatibility and updates. In a couple years it will be very difficult to find an old iPhone running its initial operating system, let alone any cell phone from the turn of the century. Potential future work would involve adding more mobile hardware and software platforms to the vocabulary and finding means to correctly identify and categorize streaming resources and other born-digital content. Given that there is currently no accepted solution

to the collection or remediation of born-digital game software in libraries, we feel that the longer potential solutions to this problem are ignored, the more information may be lost.

In using Semantic Web frameworks, we feel it will be possible to extend the approach we used here to add more specificity and depth to the identification of game software and hardware platforms. This may require creating new RDF properties and further guidelines to describe these types of works, at least until the larger library community develops reliable techniques and standards for hosting, sharing, and collection born-digital software and games. The Semantic Web provides a fantastic means for embedding any current vocabulary work into a network of information that will last into the future. The means for relating items through content-agnostic triples means that any future additions to our vocabulary, or vocabularies developed by others, can be sensibly linked and only increase, rather than obfuscate, the information available about a specific game resource and its dependent platforms and formats.

Conclusion

The work presented in this paper is the first step towards a more consistent, concrete, and historically secure identification and classification of computer game resources. Our pilot controlled vocabularies for platforms and media formats, constructed through significant research, and embedded into the Semantic Web of LOD, are the beginning of an evolving, and hopefully collaborative process between our institutions and others interested in the continued improvement of game and other software-related records. LOD provides a foundation on which to build a more expansive network of authoritative information about

computer game platforms, their media formats, and in the future, more detailed technical descriptions and ontologies.

Our basic research process resulted in an abundance of information regarding platforms and media formats that still require more effort to parse and categorize. We feel that our initial vocabularies will sufficiently cover the needs of most libraries and non-specialized collections, but we will continue adding to our initial pilot set for the remainder of the project. Furthermore, we hope that this effort leads others to understand the complexity of correctly describing software for future use. In the not-too-distant future, our descriptive efforts, if adopted, could help the ability of future libraries and their patrons to represent and experience legacy software.

There is still much to do in securing the historical position of game and other software-based resources in library collections, whether digital or physical. Standards and schemas must be updated to incorporate more descriptive and identifiable technical information; we hope that this work provides a small nudge in the direction of more extensive categorization of computer game platforms and formats.

References

Description of MODS 3.4 Changes. (2010, June). Retrieved from http://www.loc.gov/standards/mods/mods-changes-3-4.html

deGroat, G. (2015). Best Practices for Cataloging Video Games Using RDA and MARC21. Online Audiovisual Catalogers, Inc.

deGroat, G. (2015). Core Metadata Schema for Cataloging Video Games. GAMECIP Tech Report 1.

McDonough, J. P. (2010). Preserving virtual worlds: Final Report. Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign.

Montfort, N., & Bogost, I. (2009). Racing the beam: The Atari video computer system. Cambridge, Mass.: MIT Press.

Open Metadata Registry. (2015). Retrieved from http://metadataregistry.org

Southwick, S. B. (2015). A Guide for Transforming Digital Collections Metadata into Linked Data Using Open Source Technologies. Journal of Library Metadata, 15(1), 1–35.

Taylor, A. G. (2008). The organization of information. Westport, Conn.: Libraries Unlimited.

Term and Code List for RDA Carrier Types, (2012, March 9). Retrieved from http://www.loc.gov/standards/valuelist/rdacarrier.html