Global Unique ID Framework for Copyright Material

LATE Florence 1, Shinji KONO2,Akira KAMIZATO3

The digital content distribution is becoming more complex and diverse with related issues like privacy of personal information and Internet shopping. One of the problems is the missing link of URL, and the other problem is interaction of copyrighted material and Internet, where everyhing is copy free. In this paper, we introduce a framework in which a global unique identifier (GUID) is assigned to each digital content. Using GUID, we can avoid the dangling pointer problem. GUID can be used for automatic sharing, semantic search and content exchange, and copyrights managements. To design the GUID framework, we need basic performance of our GUID registration server. We measured its performance using simple implementation by the size of the binary and the number of clients.

1 Introduction

Popularization of the Internet and progress in image, music and other content coding technology now make it possible to distribute digital content over networks easily and inexpensively. The revenue potential of network-based distribution had not yet been realized for the following reasons. There is no single established platforms existed which delivered digital content in fully secure manner to the satisfaction of digital content right owners. There is not a mechanism which allowed a digital content consumer to determine their rights in respect of particular items of digital content. There is not an effective mechanisms existed for charging recipients and the subsequent settlement of payments to all the legitimate participants in the digital content distribution chain. Also not an existing mechanism which enabled the cross-referencing of uniquely identified digital content because the IDs and any associated metadata were located in many different databases.

Even though significant advances have been made in the current web, the difficulties outlined here remain to be effectively resolved. In this paper, we propose a global unique identifier (GUID) Framework shows in figure1for allocation to packages of digital content (whether containing one or many items of digital content) intended for distribution. As part of a common global system, the GUID would be persistently associated with the package of digital content. We believe that the persistent association of the unique identifier with packages of digital content will provide far-reaching benefits to all participants in the digital content distribution chain. We will first explain how GUID is useful for the copyright point of view. Then we will present our simple implementation and the simulation of the GUID framework. We also estimated how many distributed ID server in the world.

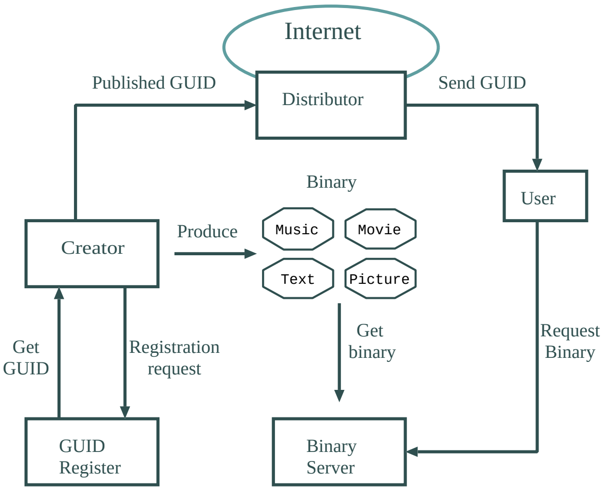

Figure 1: GUID Framework Overview

2 Copyright view point

How often do you need access to electronic information, be it an article, an image, a bibliography or a video, that you do manage to find it, how do you know whether it is the current version and what the copyright implications are for using any of the content? In the fast changing world of content production, keeping track of the location and ownership of electronic files is difficult. Wouldn't be really useful if you could link to the specific content you are searching for and also be notified of the copyright implications? A copyright is a form of protection provided by the laws, to the

[1] Intelligent System Engineering, Graduate School of Engineering and Science, University of Ryukyu
late@cr.ie.u-ryukyu.ac.jp

[2] Infomation Engineering, faculty of engineering, University of Ryukyu
kono@ie.u-ryukyu.ac.jp

[3] Infomation Engineering, faculty of engineering, University of Ryukyu
akira@ie.u-ryukyu.ac.jp

authors of ”original works of authorship”, including literary, dramatic, musical, artistic and certain other intellectual works. In case of United State law,the copyright registration is a legal formality intended to make a public record of the basic facts of a particular copyright. However, registration is not a condition of copyright protection. Even though registration is not a requirement for protection, the copyright law provides several inducements or advantages to encourage copyright owners to make registration. To declare copyright is a mechanism enabling creators to declare their copyrights in network based environment and to clarify the complicated structure of copyright ownership (who owns what has in growing content reuses). GUID should be used to clearly declare copyright ownership on contents in a standard manner. To obtain the exact contents is difficult to specify the content(similar titles, similar authors, similar description). GUID should be the key to specifying the contents and to obtaining the contents. GUID is a global unique identifier persistently associated with each digital content of any kind. It’s a set of well defined metadata describing a material and its distribution attributes, including it global ID. A set of metadata may be also associated with each content for appropriate content distribution. Identifier uniquely assigned to each instance of content metadata set describing a content and its distribution attributes referred in semantic searching, rights management, charging, transaction, etc. It’s required:

- Uniqueness No duplicate numbers be issued in global environments.
- Persistency Maintained by their issuers or their successors.
- Location resolvability Address of metadata or content be obtained from ID.
- Inseparability Firmly bound to content.
- Interoperability Co-exist with other ID systems.

3 GUID Framework System Model

A simple functional model illustrates the process from production of a work in the form of an item of digital content through assignment of rights relating to it, the aggregation or arrangement of work to create a new item of digital content, the issuing of a license to exploit the item of digital content in the form of a package of digital content for distribution and exploitation, and the procedure for invoicing, collecting and distributing royalties or fees. A simple player may also perform multiple functions. For example, a right owner may as well as having rights in the item of digital content, perform the function of the Aggregator and Distributor. Record companies, frequently perform all three of these functions today. Each service in the system model can be available to the players in other models. The players in the system model are the Electronic Authentication Authority (EAA), the GUID Issuing Center (GIC), the DB Center (DBC) and the Charging and Transaction System (CTS). The right owner, the Aggregator and the distributor register themselves with the EAA. other players in the system layer authenticate the identities of rights owners, aggregates or distributors in each service with the cooperation of the EAA. The GIC issues GUID in response to requests made by the rights owner or Aggregator. The metadata attributes that uniquely describe an item or package of digital content are kept in the DB and can be accessed by the right owner and Aggregator as necessary. The CTS charges either some value (fee) for the exploitation of packages of packages of digital content by the consumer or, a rights royalty or fee from the Aggregator or distributor. In both cases the CTS passes the royalty or fee to the relevant right owner. Figure 2shows the services involve in the system model. In the case of multiple rights owners, the DBC can be referred to for right ownership metadata, which identifies the royalty or fee each rights owner should receive.

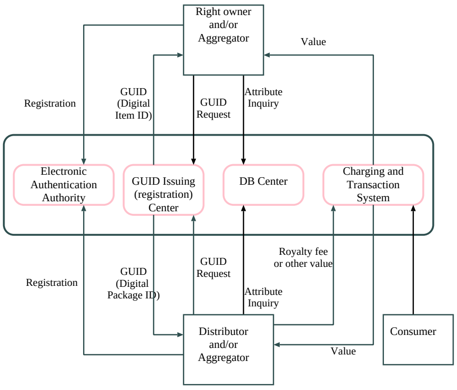
Right owner and/or Aggregator
Value
Registration
GUID (Digital Item ID)
GUID Request
Attribute Inquiry
Electronic Authentication Authority
GUID Issuing (registration) Center
DB Center
Charging and Transaction System
Royalty fee or other value
Registration
GUID (Digital Package ID)
GUID Request
Attribute Inquiry
Distributor and/or Aggregator
Value
Consumer

Figure 2: Services in the System Model

3.1 GUID issuing (registration) and Payment Transaction Processing

An example of the GUID issuing and exploitation process is illustrated in figure 3 from the right owners to the end users.

- The Creator produces an item of digital content(binary data).
- The Creator sends the binary to the GUID registration center, and enters a registration request.

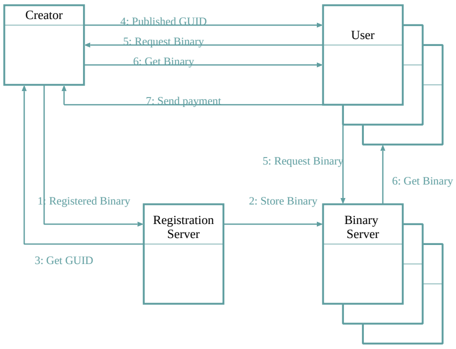

Figure 3: Example of GUID processing

- The GUID Registration center issues a GUID identifying the binary received from the creator.
- The creator grants a license to exploit the binary to a user by publishing the GUID.

The user can request the binary corresponding to this GUID, either directly from the creator or through the binary server.

- The creator or binary server sends the binary to the user.
- The user send the payment to the Creator.

4 Related Work

The Digital Object Identifier (DOI) is a system for identifying and exchanging intellectual property in the digital environment, being implemented and developed by the International DOI Foundation (IDF). DOIs are persistent identifiers assigned to content in the form of digital objects, accompanied by a set of interoperable well-structured metadata and a managed distributed resolution system. Large scale applications of DOI such as CrossRef are now live and registration agencies are being established internationally. The IDF is interesting in the importance of unique identification in enabling commerce, including that of copyrighted items; the importance of naming content rather than locations such as URL; the convergence of all media types (images, video, audio, text, etc.) in a managed digital network, requiring interoperable standards for identification description and rights; the need for associated resolution systems (e.g. The Handle System developed by CNRI) to associate identifiers with current metadata, which may articulate services; the need for consistent policy development including the development of commercially feasible self-financing operation of identifier systems. The difference between DOI with our GUID is that, we use automatic rgistration server and assign the GUID globally.

5 GUID Framework Simple Registration Implementation

In order to experiment and check our design, we implemented two protocols: the registration protocol and the read protocol.

5.1 Registration Protocol

This section talks about the registration process and it implementation.

The registration protocol is a protocol use for making the registration in GUID framework. It's consist of a:

- Server for receiving registration request from client.
- Client for making a registration request.
- Database to store the binary data from the clients

Figure4 shows the registration process. After the connection is established, GUID server receives a registration request from the client by sending a binary data and its associated metadata. Here the binary data can be an email, a mp3 file, mpg, etc...

The server starts the registration by calculating the MD5 hash value, write the binary to a database directory, calculate the GUID number and then return the GUID number to the client.

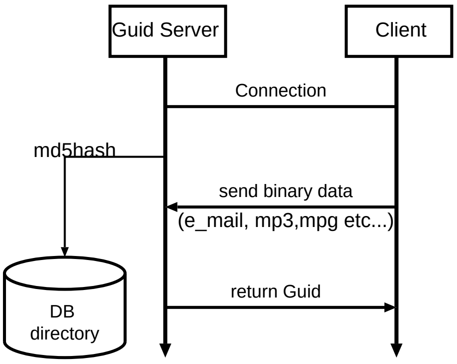

Figure 4: Entire flow of Registration Protocol

Below is an output example in the registration procedure. This output is in case of sending a simple file like md5.h.

5.1.1 Md5 hash calculation

MD5 (Message-Digest algorithm 5) is a widely-used cryptographic hash function with a 128-bit hash value. As an Internet standard (RFC 1321), MD5 has been employed in a wide variety of security applications, and is also commonly used to check the integrity of files. MD5 takes the content of a file and forms a number from it in such a way that:

- it is not possible to tell the contents of the original file just by looking at the hash value and;

```
sh-2.05b$./Server
put
From: kono@ie.u-ryukyu.ac.jp (Shinji KONO)
Organization: Information Engineering,
University of the Ryukyus
MIME-Version: 1.0
Content-Type: text/plain; charset="iso-
2022-jp"
Content-ID: <623.1120459220.1@leo.ie.u-
ryukyu.ac.jp>
Content-Transfer-Encoding: 8bit
Date: Mon, 04 Jul 2005 15:39:52 +0900
Message-ID: <616.1120459192@insigna.ie.u-
ryukyu.ac.jp>
MD5: cc7072859a8a83e195a0385bbfbee4a4
Length: 8336
finish writing
dirname: cc7072859a8a83e195a0385bbfbee4a4
```

Figure 5: Output from the server side during execution

```
sh-2.05b$./regist md5.h
port Number: 10547
Reply-Guid:
Guid: cc7072859a8a83e195a0385bbfbee4a4-
aae05884326b34df761d9bfea0154048-8336-1
Content-ID: <623.1120459220.1@leo.ie.u-
ryukyu.ac.jp>
Date: Mon, 04 Jul 2005 15:39:52 +0900
Message-ID: <616.1120459192@insigna.ie.u-
ryukyu.ac.jp>
```

Figure 6: Output from the client side during execution

- it is not reasonably practicable to generate a file that will give a particular hash.

In our system, the following program example is use to calculate the md5 hash value.

```
static void recv_meta(){
  char meta[343];
    struct md5_ctx ctx;
    u8 out[MD5_HASH_WORDS*sizeof(u32)];
    int i;
    char god[7];
    FILE *fd;
    i = read(new_sockfd,meta,343);
     md5_init(&ctx);
     md5_update(&ctx,meta,i);
     md5_final(&ctx,out);
     strcat(diname,"-");
      for(i=0;i<MD5_HASH_WORDS*
        sizeof(u32);i++) {
        sprintf(god,"%02x",out[i]);
        strcat(diname,god);
      }
     strcat(diname,"-");
      i = sprintf(god,"%d",total_size);
      strcat(diname,god);
      strcat(diname,"-1");
      fd = open("1.meta", O_WRONLY|
       O_CREAT|O_TRUNC,0666);
        if(fd == -1)
          {
            perror( fd );
            exit(1);
          }
      write(fd,"guid: ",5);
      write(fd,diname,72);
      write(fd,"\n",1);
      write(fd,meta,343);
      write(fd,"\n",1);
      write(fd,"size: ",5);
      write(fd,god,i);
      write(fd,"\n",1);
      write(fd,"md5: ",5);
      write(fd,dirname,32);
      close(fd);
      write(new_sockfd,diname,72);
  }
```

5.1.2 Database Structure

```
DB structure


  path/md5-value/1.data/seq
  path/md5-value/1.meta/seq
  path/md5-value/2.data/seq
  path/md5-value/2.meta/seq
```

The system uses a directory name to store the binary data. The data is store in a md5 value directory name in two sequences:

- 1.data which contains the original file and it content.
- 1.meta which is the metadata(data of data). It contains information concerning the file and the author or the sender.

The metadata looks like this:

```
guid:c894489b9470efb1943a129238d9d557-
10d82e5c56c570e791c88221fad99d89-1790-1
From:akira@ie.u-ryukyu.ac.jp
Organization: Information Engineering,
University of the Ryukyus
MIME-Version: 1.0
Content-Type: text/plain; charset="iso-
```

```
2022-jp"
Content-ID: <623.1120459220.1@leo.ie.u-
ryukyu.ac.jp>
Content-Transfer-Encoding: 8bit
Date: Mon, 04 Jul 2005 15:39:52 +0900
Message-ID: <616.1120459192@insigna.ie.u-
ryukyu.ac.jp
size:1790
md5: -10d82e5c56c570e791c88221fad99d8
```

5.1.3 GUID number generation and it structure It not so easy to generate possible correct GUID but easy key for binaries in the server. It contains the metadata information. It obtains by:

md5-value + metadata-md5-value + binary-data-length + sequence. It looks like:

```
guid:c894489b9470efb1943a129238d9d557-
10d82e5c56c570e791c88221fad99d89-1790-1
```

6 Read protocol

This section is about the reading process and it implementation. The read protocol consist of a:

- Binary server for receiving a reading request
- Client for making a reading request
- Database to return the database directory

Figure 7 describes the reading process. The binary server receives a reading request from a client by enter a GUID number. The binary server transfers the corresponding binary and the metadata associate to this GUID number to the client.

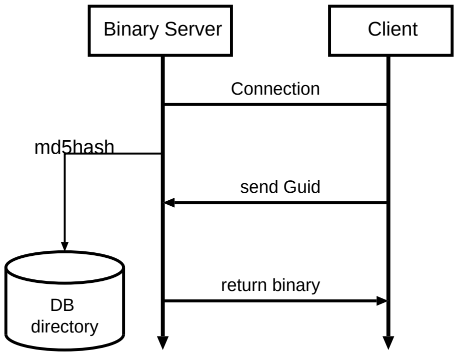

Figure 7: Entire flow of Read Protocol

7 Simulation Results

The Registration protocol was simulated on a PC cluster with a capacity of 250GB for the server. The clients were the ordinary personal computer in the laboratory. The graph in figure 8 is the time to registration by size of 100MB, up to 40 clients can send simultaneously a data to the server. The graph in figure 9 can support up to 40 clients request with a data size of 500MB. And in figure 10 only 20 clients can register simultaneously their data of 1.1GB.

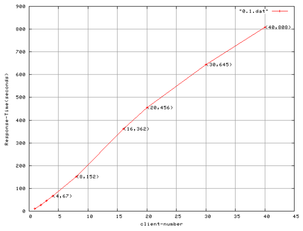

Figure 8: Time to registration by size and by process of 100MB

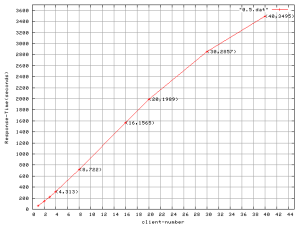

Figure 9: Time to registration by size and by process of 500MB

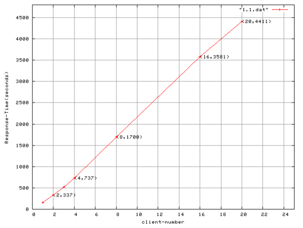

Figure 10: Time to registration by size and by process of 1.1GB

7.1 GUID Distributed ID Server Estimation We have to use one unique server to provide unique ID. If we have one unique server, it cannot be reliable. We need a distributed unique ID server. But how to maintain the uniqueness and how many server for our implementation? According to Peter Lyman and Hal R. Varian from the School of Information Management and Systems at the University of California at Berkeley, roughly 90 000 CDs music items per year in the world. The United States holds a 37percent share of the world music market and releases about 33 100 items per year. To estimate how many server we need for our implementation, we used the number of CDs released regarding world market. In case of our server registration implementation, one server can support up to 20GB. If we have 90 000 items CDs Uncompressed 650 GB/item, the total size is 58 terabytes (58 000GB). If we we multiplie 90000 CDs item by 100 to have a roughly idea in all particular item we obtain 9000000 items copyright material per year. That means roughtly 24657 items per day, 1027 items every 1 hour.i A total size of 9000000 cna be roughtly estimated to 580TB. Du to the network traffic and transaction transfert, we can estimate 290000 distributed server in the world.

8 Business Model of the Content Commerce The business model spells-out how a company makes money by specifying where it is positioned in the value chain. It draws on a multitude on business subjects including entrepreneurship, strategy, economics, finance, operations, and marketing. In the most basic sense, a business model is the method of doing business by which a company can sustain itself, that is to generate revenue. The owner of digital content will go out of business unless legal and technical means are imposed to prevent the rampant piracy.

8.1 Proposals for the Conceptual Business Model Development After reviewing the Conceptual Business Model for Content Delivery and Rights Management, we have noticed that the MPEG model is good enough in terms of representing the current right management in the complexity of the business model of the content commerce. A comprehensive standard framework for networked digital multimedia designed by the Moving Picture Experts Group. MPEG-21 includes an REL and a Rights Data Dictionary. Unlike other MPEG standards that describe compression coding methods, MPEG-21 describes a standard that defines the description of content and also processes for accessing, searching, storing and protecting the copyrights of content.

A role of Editor or Secondary Creator who reuses the whole or part of other creators creations to make his/her own creations is quite common. The value and information flows of the secondary creations have to be different from those of the original creations because the copyright of secondary creations is owned by both original creators and editors. The unique numbers of the secondary creations should have a nesting structure where secondary creations are associated with the original creations. We believe that MPEG-21 should consider GUID as one of the urgent topics to standardize in MPEG-21.

9 Conclusion The Web requires new tools that can be used in new ways. One important use will be the GUID, allowing people to dynamically create and use digital information. Building such system will solve the URLs dangling problems and the copyright point, and we described some ideas aimed at providing this basis. Thus, GUID the registration and the read protocol described in this paper are examples of some of the basic technologies that are needed to create the Internet Next Generation.

References

[1] R. R Larson. Design and development of a network-based electronic library. in *Proc. ASIS Mid-year Meeting, pages 95-114, Portland, Oregon, May 1994.*

[2] S. Browne and J. Dongarra. Location-Independent Naming for vitual Distributed Software Repositories.

[3] H. Sakamoto, H. Fuji and J. Kishigami Digital Item Identification and Description based on Content ID Technology.

[4] J. W.Moore. The use of encryption to ensure the integrity of reusable software components in *Proc. Third International conference on Software Reusability. IEEE Computer Society Press, Nov. 1992.*

[5] K. Sollins and L. Masinter. Function requirements for Uniform Resource Names. in Internet RFC 1737, Dec. 1994

[6] Dr. Hiroshi Yasuda. Content ID Forum. March 21, 2000 at Noordwijkerhout, Netherlands.

[7] Junichi Takayashi, Takayuki Kushida, IBM Research, Tokyo Research Laboratory, Japan.

[8] Peter Lyman and Hal R. Varian. How much information? http://www.sims.berkeley.edu/research/projects/how-much-info/optical.html