UPS ReDIF Conversion Report

Thomas Krichel and Victor M. Lyapunov *

October 1999

1 Introduction

The conversion of the e-print metadata dumps to ReDIF
was funded by the WoPEc project (an eLib project funded
by JISC) as a donation to the UPS protoproto work. The
work was supervised by Thomas Krichel, the project di-
rector of WoPEc, at the University of Surrey. He opened
the Acmes mailing list where persons interested in the pro-
cess contributed comments. These included Sune Karls-
son, Michael L. Nelson and Herbert Van de Sompel.

2 An introduction to ReDIF

The aim of ReDIF is to describe the output aspects of an
academic discipline. These include the resources that it
creates, the creators of these resources as well as the in-
stitutions that support the creation process. The term "dis-
cipline" should be understood here as any group that uses
ReDIF to document its activities using a common identi-
fier space. Therefore disciplines can be viewed as naming
authorities who ensure that the elements in the dataset that
have a different identifier really are different.

The version 1 of ReDIF is a relational metadata format.
It defines a finite set of record types "Paper", "Article",
"Person", "Institution" etc. Each record has an identifier.
Records can use the identifiers of other records. Example

```
Template-Type: ReDIF-paper 1.0
Author-Name: Bill Clinton
Author-Email: president@whitehouse.gov
Handle: paper1
```

Another (better) way to encode the same information

```
Template-Type: ReDIF-paper 1.0
Author-Name: Bill Clinton
Author-Person: us1
Handle: paper1

Template-Type: ReDIF-person 1.0
Name-First: Bill
Name-Last: Clinton
Email: president@whitehouse.gov
Handle: us1
```

ReDIF is formally defined in a specification file. Basically,
for each record type, the set of fields is defined, as well as

their optionality, repeatability, and value syntax. The spec-
ification is used in a perl module rr.pm to read and validate
ReDIF. Using a different specification file allows commu-
nities to design a new vocabulary for ReDIF. The ReDIF
specification format and the rr.pm software are the work of
Ivan Y. Kurmanov.

The main limitation of ReDIF version 1 is the flat struc-
ture of record types. The second version of ReDIF will
object oriented. Record types will be classes. For exam-
ple a preprint record type will be defined as subclass of
a document types. Thus communities using ReDIF will
be able to refine existing types as they wish. ReDIF ver-
sion 2 will be defined in a specific language code-named
"Zhuleb". ReDIF records will be syntax independent.
Thomas Krichel and Ivan Y. Kurmanov are working on this
project.

3 Conceptual work

ReDIF was developed to encode the RePEc dataset about
academic Economics. It is more general than a format to
use for the internal documentation of a preprint collection.
It is sufficiently powerful to encode the aspects of reality
described in the preprints data sets. However it had to be
enlarged for the purpose to accommodate for the needs of
datasets that were used in the UPS protoproto.

3.1 Relations to external manifestations

Many datasets describe e-prints that have also been made
available through a traditional publication channel. The
metadata for this other channel is included in the meta-
data of the e-print. The CogPrints dataset is particularly
rich in this area. For example they will have an indica-
tion (in an extra field) on where the conference was held
if the paper in the archive "is" a conference paper. As far
as ReDIF is concerned the location of a conference is an
attribute of the conference and not of the papers presented
there. And again as far a ReDIF is concerned the paper in
the conference in not the same thing as the paper in the e-
print archive. Both are different manifestations of the same
resource. Nevertheless all these fielded data are included
using cluster nesting. It is possible in ReDIF, but it is ugly.

3.2 Identifier structures

Another adjustment was made to the structure of the identi-
fiers. Within RePEc, all document identifiers have the form

`RePEc:`*aaa*`:`*sssss*`:`*other_stuff*

where *aaa* is a three letter code of the archive, *sssss* is a six letter code for the series of documents, and *other_stuff* is a sequence of chars—none of which may be blank—to reflect the position of the paper within the series. This form has two advantages. First, it avoids mnemonic identifiers. Second, the syntactic constraint makes easier to distinguish resource identifiers from other identifiers (e.g. person or institution identifiers).

For the converted metadata, it was decided to propose that the colon should be a special character. Each resource identifier consists out of 5 components.

authority_id : *archive_id* : *series_id* : *number_id* : *manif_id*

There are some details

- All components obey to the perl regex `[^:]*`.

- For some authorities, some components are null or irrelevant. For example, NCSTRL does not know about series. As such an identifier is e.g. ncstrl:ucmp::id.

- For the centralised archives the authority and the archive are the same: xxx:xxx:hep-th:19998898374.

- For the xxx and cogprints data, the series is at the level of the category identifier before the dot.

- The manifestation id might later be be used to put together different manifestations of a publication (cf the bucket concept). For the moment it is not used. It is not yet present in the data that we have.

3.3 Subject classification

The classification scheme proposed in the NASA TechReport TM-1998-208955. has been adopted to broadly classify all records under the name "Ila". These classification data are added at the series level.

The existing classification schemes are kept in tags `Classification-`*name*`-`*yyyy*`:`
where *name* is an identifier for the scheme and *yyyy* is the revision year of that scheme.

4 Technical work

The computations are the work of Victor M. Lyapunov at the Siberian Branch of the Russian Academy of Science. He did not keep a precise account of how many hours he spent on each part of the job. However the following table gives a rough idea on the time spent on each archive.

			days
Jul 13	CogPrints	arrived	
Jul 16	xxx	arrived	
Jul 30	CogPrints	released	17
Aug 11	NCSTRL	arrived	
Aug 20	xxx	released	20
Sep 1	NCSTRL	released	11
Sep 10	NDLTD	arrived	
Sep 17	NDLTD	released	7
Sep 27	NDLTD full text links	arrived	
Sep 30	NDLTD full text links	released	

In the following, we report on particular datasets, before we report on plans for further technical work.

4.1 CogPrints

Summary: 743 input records, 743 output records, 750 lines of perl code
CogPrints is the most accurately converted archive, since it has been started first and the intermediate results were discussed and criticised.

The only technical problem is determining the correspondence between the e-mail address and the author name. The source record contains exactly one contact e-mail and one or more authors. In the case of one author the situation is trivial. Otherwise the converter uses an algorithm which appreciates the similarity between the email and author name. Besides, the established author-address pairs are saved and are used if the algorithm fails to determine the correspondence. The latter option is obviously more efficient if the archive is totally rebuild. The address which is not matched to any author is put into the "Contact-Email" ReDIF field, which can give a general contact for the paper. This concerns 13 of 237 multi-author entries.

4.2 xxx

Summary: 129,000 input records, 88063 output records, 600 lines of perl code
Up to October 1994, records contain a combined title-author field which does not have an explicit delimiter between title and list of authors. Most of them are in the form *Title Authors* but a few are are in a reverse form, that is *Authors Title*. There are about 13,800 such records. An experimental algorithm (another 350 lines of code) separates correctly Title and Author subfields in about 90% of these combined fields, but it can not appreciate itself the correctness of the separation. However the WoPEc project is offering to manually separate the records as a donation to xxx. Technical details on how that should be done will have to be agreed. At the moment the non-separated records are not converted into ReDIF.

There is also a problem with the records in the format adopted since 1994. The list of authors does not contain any delimiter between the author names and institutions. However, the algorithm separates these more successfully. The wrongly parsed author lists are approximately 2%.

The significant (or even most) part of flaws being happened while extracting the author's workplace. As a rule, the source author+workplace string is presented in two forms. The first form is *Author* (*Workplace*). An example for the first form is

```
paper: astro-ph/9801002
authors: Patricia A Whitelock (South
 African Astronomical Observatory)
```

In the second form, the author workplace data takes the form `Author` (*footnote*) and the author list is appended with a (not always) bracketed workplace list in the form.

(*footnote*) `Workplace` An example for the second form is

```
paper: astro-ph/9801016
authors: M. J. Drinkwater (1)  M. D.
 Gregg (2) ((1) University of New
 South Wales, (2) University of
 California, Davis,  and Institute
 for Geophysics and   Planetary
 Physics, Lawrence Livermore
 National Laboratory)
```

These formats, and slight variants thereof, are handled correctly by the converter. However some records do not obey to either form. example

```
paper: chem-ph/9501002
authors: M. Vossen, F. Forstmann,
 Institut f"ur Theoretische Physik,
 Freie Universit"at Berlin,
 Arnimallee 14, 14195 Berlin, Germany
```

This type of records is prone to parsing errors. An obvious way to raise the quality of parsing this kind of strings is to provide the converter with a list of words which could never be the personal names: university, institution, laboratory, etc. That is not done in the current release.

There are also 24,000 duplicate entries in the source xxx archive. According to a suggestion by Herbert Van de Sompel, the duplicates carry the set of values of the source "subj-class" tag, which reflects them being posted into several sub-archives. This set of values is added to the ReDIF "Classification-xxx" tag. The resulting amount of values of "Classification-xxx" is 114348, which is close to the number of 'good' source records.

4.3 NCSTRL

Summary: 29905 input records, 29918 output records, 300 lines of perl code

87 input records lack either Author or Title tags, or those tags have no value. They were excluded from the data. 128 records were not included in the deadline NCSTRL ReDIF release, due to the bug in the conversion script. This bug is specific to the NCSTRL converter only.

Herbert Van de Sompel brought to our attention that that there is a overlap between of NCSTRL and xxx through the shared CORR facility. It was decided to ignore this duplication in the protoproto work.

No conversion problems were encountered.

4.4 NDLTD

Summary: 1590 input records, 1590 output records, 350 lines of perl code

Victor M. Lyapunov converted the source data from MARC format into plain text with ms-dos utility marcbrkr.exe. Markus Klink gave us a link to a perl MARC–>plain conversion module but 'as is' it didn't work properly. It will adapted later if there is to be an update the converters.

The initial source data archive did not contain the full-text links. The separate plain-text file with full-text links arrived later. Therefore the converter uses two functionally different input files.

4.5 Future work on the converters

We have not yet determined the environment in which the conversion algorithm should work, that is, the structures of input data and the ways of delivering the input and output. We are more than happy to make the converters available for initiatives to convert their data. Alternatively we can mirror the data, convert it and make the converted metadata available.

Currently every archive is processed by a specialized conversion script. These 4 scripts use some common subroutines. The nearest future job may be to bring the scripts together into universal converter, which would use specific configuration files for the tuning to the specific archive processing. The main difficulty in this job would be a diversity of input data formats. However it is possible to make such a general converter available.

5 Recommendations

To improve the quality of the metadata, we make the following suggestions. Some of these suggestions are really hard to implement, and of course these are only our own ideas. They are heavily influenced by the general ReDIF philosophy.

5.1 Add a creation date to all records

Some records (e.g. in RePEc) do not have a creation date for the resource that they describe. Others have such records, but they are difficult to parse. It would be desirable to have such data available in a homogeneous format for example of the type *yyyy-mm-dd*, followed by other information that could be valuable for local purposes.

5.2 Normalisation of the identifier structure

For many applications, for example SFX linking, it is highly desirable to have an identifier that is easy to parse. A normalisation of the identifiers would be desirable. Of course the converted data does provide such a uniform identifier structure.

5.3 Avoid metadata overload

Much of the difficulty that we have with the metadata in general comes from a tendency to overload the resource metadata with information that is not directly related to the resource. There are two main areas where this occurs. First the overloading of resource metadata with data about the creators of the resource. We recommend to separate the data on the creators of the resource from the metadata of the resource. This will make it easier to update the data. In

the same way we recommend to set aside the data on creators from the data about the institutions that these creators are affiliated with. The second instance of overload is the presence of data on manifestations. This is a much more difficult area.

5.4 Augment the document metadata with citation metadata

It would desirable to have citation metadata available with the metadata for each resource that is a document. This data could be provided by metadata creators or computed using a software like CiteSeer.