SIM: the Static Incremental Method

An Archiwwwe Methodology for the
Incremental Archiving of Websites

Mariya Lysenkova,
Lead Developer
mariya.lysenkova@archiwwwe.se

Matias Vangsnes,
CTO and founder
matias.vangsnes@archiwwwe.se

Archiwwwe, a leader in the digital archiving industry, has developed SIM, the Static Incremental Method, allowing organisations to archive the incremental history of their websites in a free, open and persistent format.

1. Executive Summary

The average organisation redesigns their website every two to three years, and far from all retain their old content on the web. This poses a problem for communications directors, legal advisors, compliance officers and other professionals who need to know what the organisation said on the web, and when.

Incremental Website Archiving is becoming a requirement for organisations who need a historical paper trail of the web content they have published. Unfortunately, website archiving is a new field with few universally accepted standards. It is also a field that poses some serious challenges, including the format of the archives.

How should the archive of a website be stored over time?

While solutions for incremental archives exist, using them carries a risk. If the archive requires proprietary software to be read, its ultimate longevity depends on the survival of the software and its ability to be executed on computers years into the future. If the archive is stored in a format other than the original, it must also be converted at regular intervals to keep up with evolving standards.

Archiwwwe, a leader in the digital archiving industry, has developed SIM, the Static Incremental Method, that allows organisations to archive the incremental history of their websites in a free, open and persistent format.

2. Business or Technical Problem

Why do we need incremental archives?

The average organisation redesigns their website every two to three years, and far from all retain their old content on the web. There are many reasons why it is necessary to track the evolution of a website over time. They include:

- Marketing
- Compliance & Litigation
- Research & Development

There are numerous questions that an incremental archive can answer. They include:

- When did we first begin research on topic X?
- Can we can show a history of innovation in area Y?
- When did we publicly announce the implementation of policy X, in accordance with legislation Y?
- In what year did we discontinue support for product X? Was this in accordance with our User Agreement?
- What version of our Terms of Use did we publish online back in year X?

The history of a website can be of critical importance to communications directors, legal advisors, compliance officers and other professionals who need to know what the organisation said on the web, and when.

3. Existing solutions

WARC: The current standard for incremental archiving

Of the few solutions available today, the most commonly used is Web ARChive (WARC), an ISO standard which allows for the incremental storage of crawled web pages.

WARC files are an aggregation of multiple digital resources, and as such, they require third-party software to open and browse. There are few software tools available to open WARC files, and most require advanced technical knowledge to use. As such, WARC is a poor choice for archives that need to be viewed by lay audiences.

In addition, there is an inherent risk in storing archives in a non-native format. A particular version of the WARC standard, or any other file format, will likely become outdated with time. An archive created with WARC today relies on reading software that implements that specific version of the standard.

This means that an organisation must be pro-active in maintaining healthy WARC archives, either by maintaining compatible reader software, or regularly converting their archives to the latest version of the WARC standard. This can quickly become expensive, especially as archives grow with time.

For these reasons, Archiwwwe has developed an alternative for storing incremental archives in their native format.

4. Better Solution

4.1. SIM: A new solution for incremental archiving

SIM, the Static Incremental Method, was developed by Archiwwwe as a solution to the problem of storing incremental archives. It is proposed as a standard for archives that accurately reflect the evolution of a website over time.

SIM is a collection of file storage and naming conventions that allows organisations to keep incremental web archives in their native format, organised in a flat directory structure.

SIM is a standard that is free, open and persistent. Its persistence comes from the philosophy that third-party software should not be necessary to view an incremental archive. That means that a SIM archive created today will not depend on additional software that may become outdated in the future.

4.1.1. SIM Pre-requisites Intervallic Web Crawling

SIM is a method of organising HTML, text and binary files that are collected from a given URL at various points in time.

These files are typically output from a web crawler, such as HTTrack or Heritrix, that is run against the target URL at arbitrary intervals. SIM is crawler-agnostic.

4.1.2 Snapshot Identification

Each version of a crawled website is a “snapshot” created at a particular date and time. SIM depends on a unique identifier that

links the files from a particular snapshot. In our case study, we use Unix timestamps based on the GMT date-time when a particular crawl was initiated. Other possibilities include SHA1, md5, or simple integer identifiers, depending on how the organisation may want to link archive snapshots to other data.[1]

SIM is agnostic as to the implementation of these identifiers.

4.2 How is a SIM archive structured?

In the next section, we will crawl a simple website and show how to store the output according to SIM conventions.

4.21. SIM Case Study – Initial web archive

In this example, we will crawl a fictional website www.example.com, and store the output according to SIM conventions. This simple website contains a single index.html page that is linked to an image file stored at the relative path images/img.jpg. The contents of the original index.html look like this:

index.html

```
<html>
  <body>
    <img src="images/img.jpg">
  </body>
</html>
```

Let's take a look at the contents of the SIM archive after we have processed the crawler output.

After Crawl 1 – January 1, 2015 at 10:00:00

```
.
└── www.example.com
    ├── 1420106400
    │   ├── images
    │   │   └── img.jpg
    │   └── index.html
    └── .sim
```

Observe that the output of the crawl has been placed into a directory named with a unique identifier (in this case, a Unix timestamp representation of the snapshot date), denoting that these files reflect their content on the snapshot date-time of January 1, 2015 at 10:00:00. Also note the presence of the .sim directory, which will be used for SIM metadata later.

[1] When selecting a convention for snapshot identification, it is helpful to keep identifiers as short as possible. These identifiers will be used in filenames, and many operating systems have limits on filename lengths.

Finally, note that 1420106400/index.html uses relative paths for dependencies, and looks much like the original index.html:

1420106400/index.html

```
<html>
  <body>
    <img src="images/img.jpg">
  </body>
</html>
```

4.2.2. SIM – Archiving Subsequent Crawls

Let us look at what happens when we crawl www.example.com a second time, on February 1, 2015 at 10:00:00.

We find that index.html has changed; it now links to a new file called files/info.pdf. Here is how this updated information is reflected in our SIM archive:

After Crawl 2 – February 1, 2015 at 10:00:00

```
.
└── www.example.com
    ├── 1420106400
    │   ├── images
    │   │   └── img.jpg
    │   └── index.html
    ├── 1422784800
    │   ├── files
    │   │   └── info.pdf
    │   └── index.html
    └── .sim
```

The archive now contains:

- 1420106400/index.html – unchanged
- 1420106400/images/img.jpg – unchanged
- 1422784800 - a newly-added directory
- 1422784800/index.html – the updated version of index.html
- 1422784800/files/info.pdf – a newly-added PDF file
- .sim – still empty

Note that 1422784800/index.html links to the new PDF file, and also to the image which has remained unchanged since the January 1 snapshot. Thus, multiple versions live side by side in the same flat archive, and can be interlinked:

1422784800/index.html

```
<html>
  <body>
    <img src="../1420106400/images/img.jpg">
    <a href="files/info.pdf">Download info</a>
  </body>
</html>
```

4.2.3. SIM – Tracking Deletions

So far, we have covered how to track incremental additions and edits. But how do we handle deletions? No files are ever deleted from a SIM archive; metadata is therefore required to track when files disappear from a website.

Suppose we crawl www.example.com a third time on March 1, 2015, and we find that images/img.jpg and files/info.pdf have been deleted, and index.html no longer links to them.

After Crawl 3 – March 1, 2015 at 10:00:00

```
.
└── www.example.com
    ├── 1420106400
    │   ├── images
    │   │   └── img.jpg
    │   └── index.html
    ├── 1422784800
    │   ├── files
    │   │   └── info.pdf
    │   └── index.html
    ├── 1455710339
    │   └── index.html
    └── .sim
        └── 1455710339-deletions.txt
```

We have added 1455710339/index.html, as well as a metadata file .sim/1455710339-deletions.txt, which contains a list of files (one per line) that have been deleted as of that snapshot date:

.sim/1455710339-deletions.txt

```
images/img.jpg
files/info.pdf
```

5. Conclusion

The Static Incremental Method (SIM) allows organisations to store an accurate record of what was published on the web, and when, by defining a standard for storing incremental web archives in a format that is not dependent on third-party applications. It is suggested as an alternative to archive formats such as WARC, which depend on third-party software which runs the risk of becoming obsolete.

SIM is an open standard, and can be implemented using most file-manipulating programming languages. It is also possible and encouraged to convert archives from existing formats, such as WARC, to SIM.

For additional information on the practical implementation of SIM, please visit **https://archiwwwe.com**.

About Archiwwwe

The Archiwwwe development team hopes that this white paper has been helpful in planning the incremental archive for your organisation's websites.

Archiwwwe, based in Stockholm, Sweden, is a leader in digital archiving. Founded in 2013, our system is now used for continuous web site archiving at government agencies, at universities and businesses.

For further information about this white paper, contact Archiwwwe architects Matias Vangsnes and Mariya Lysenkova at **research@archiwwwe.se**.

For information about how Archiwwwe can help implement incremental archiving for your organisation, please contact **info@archiwwwe.se**.