Tape-Archive Time-Capsule via SAILDART

by

Bruce Guenther Baumgart

Submitted to Github, published on saildart.org and sent to a short email list

in the year 2021, in partial fulfillment of the requirement

to explain a personal software folly to future generations

as well as to my dwindling circle of SAIL companions, while

at home in Los Gatos, California

twenty miles south of

LELAND STANFORD JUNIOR UNIVERSITY

June 2021

Author .....

Certified by .....

Accepted by .....

This document parallels Brian K. Zuzga's 1995 MIT Thesis on archiving the MIT A.I. Lab backup tapes. Hence this mock *thesis* cover page, academic TEX formating, and the tongue-in-cheek, patronizing comments which are based on my having enjoyed the over 25 years of progress since BKZ1995. See the original for MIT at tcfs-thesis.pdf and here at faux-thesis.pdf is the riposte for Stanford.

Tape-Archive "*Time-Capsule*" via SAILDART

Bruce Guenther Baumgart

2020 and 2021

Abstract

Stanford reseach has generated data since the founding of the university in 1885, and for A.I. computer vision even earlier in 1878, Eadweard Muybridge's research was funded by Leland Stanford and was done on Stanford's land. at Palo Alto, near the present CSD building. A century later, the first Stanford A.I. Lab generated 50 Gigabytes of data between 1972 and 1990. The knowledge needed to decipher that data is here in this document. Since physical offline media deteriorates, the SAILDART archive software was developed to preserve both the data and the information, for your use now and for your heirs and successors; way out there in the future. Specifically, SAILDART is not intended as a universal format. SAILDART software is licensed GPLv3. SAILDART documentation is licensed Creative Commons Attribution-Share Alike 4.0 International.

Using SAILDART, I believe I have preserved the data, the information and perhaps the knowledge and wisdom that John McCarthy asked me to rescue from the backup tapes written by DART (in two stages) during the twenty three years of SAIL-WAITS time sharing system operation at Stanford.

My Thesis Adviser was John McCarthy. My Thesis readers were Don Knuth, Alan Kay and Ken Colby. Title: Geometric Modeling for Computer Vision. The SAILDART preservation project was suggested to me by John McCarthy in 1998 as an extension to my hack of retrieving my thesis work from one (*um, very easy it was simply on the first one of the 229 reels*) of the DART tapes.

Acknowledgments

I acknowledge the encouragement of John McCarthy to do this project. I was helped by Marty Frost in locating the final set of nine-track DART tapes as well as a suitable tape drive. That tape drive was on a loading dock ready to be scraped when we rescued it. John Nagle and Tom Costello assisted with the reading of the 229 reels in 1998.

Les Earnest has expressed enthusiasm for having an institution adopt the SAILDART project, and has presented a paper **Visible Legacies for Y3K** (similar to this pair of papers) reviewing archive time-capsule problems and proposed solutions.

Frost, Hartwig, Earnest and I (Baumgart) moved the 229 reels of DART tape from Gates to Green, 26 April 2011. That is from the Computer Science Department, William Gates Building to special collections at the Green Library on the Stanford Campus.

Finally, adding to Zuzga's remark, I would include on Sesame Street the numbers $\mathbf{e}$ and $\mathbf{i}$ and π which can be taught to children as geometric concepts as they learn to count the integers. Here is one easy to understand You-tube video for $\mathbf{e}$ and $\mathbf{i}$ and π.

Contents

List of Figures

List of Tables

Chapter 1

Begin Here

"*History teaches that history teaches us nothing*"
- Georg Wilhelm Fredrich Hegel

The SAILDART project was suggested by John McCarthy to preserve data from the first Stanford Artificial Intelligence Lab. The project now (2021) consists of one volunteer, B.g.Baumgart. We copied approximately fifty gigabytes from nine track tape to SCSI disk drives in 1998.

This project, the SAILDART, did not start with the original seven track backup tapes, which were *'Zuzga'* rescued in 1990, to newer tape media, by Marty Frost (et al) using the tool named MCOPY, a modified DART. This project, the SAILDART, does not see a profusion of difficult formats of a vanished people. Call me Ismael, or call me Ishi. I see fewer than one million items, in primitive formats written by a few tens of programs, most of which I had used, or even wrote, when I was young. The value of the knowledge on the DART tapes, was published and monetized early and often. Like the gold of the Californian quartz layer; the tailings are only toxic piles of slight historical or personal value. Further legal issues of ownership, intellectual property theft, fraud and prior art are best left in escrow to 2100 or so. Further curating of stale email and the bulletin boards, continues for the amusement of some of the surviving SAIL principals who have access to this material. The old personnel files, budgets and administrative trivia are mostly kept in camera.

This document parallels Zuzga's 1995 MIT Thesis and so it too contains three sections, some of the material is interleaved out of order.

- General framework of the SAIL file data base.
- Specific issues concerning the DART tape input.
- Current 2020s software tools for hacking inside SAIL corpora.

Chapter 2

After capture of old tape – Hand off the digital baton !

"*History will be kind to me, for I intend to write it myself.*"

–Winston Churchill

I too like this Churchillian bon mot. Digital archivist cliches include: this Churchill quote, that famous Santayana quote, the Rosetta Stone meme and the Library of Alexandria trope. I have visited the Rosetta Stone in London and I helped install some 900 computers for the Internet Archive at the Library of Alexandria in Egypt.

- B. g. Baumgart

And I implemented rounded-border sidebar-boxes in 1972, years before TEX or CSS.

2.1 DART Tape Inventory of SAIL files.

In a round number, there are eight hundred thousand SAIL files found on the final 229 reels of DART, of which some 3000 of those files hold the card-catalog database of which files were on the disk and which files were on tape, for nearly each week for 19 years.

The tape archive Bill-of-Lading is **DART . DAT [DMP, SYS]**
off line file names and tape numbers

The seen on the disk Letters-of-Transit are hidden
in the Piano at **ALLDIR . DAT [DMP, SYS]**
on line file names and perhaps a tape number

At the Stanford Artificial Intelligence Lab (SAIL), like at MIT's A.I. Lab (CSAIL), we too had rooms full of 7-track and 9-track magnetic tapes in various states of decay. These tapes were the incremental, full, and archival dumps of machines used for everyday work by students, faculty, staff and internet guests from 1972 to 1990. Parallel to Zuzga's effort at MIT, John McCarthy directed Marty Frost and staff in the Labs[1] in 1990, to consolidate almost 3000 of the old 7-Track 800 BPI failing tapes onto 229 new 9-Track 6250 BPI tapes. Those 9-Track tapes in turn were copied to SCSI disk drives in 1998. The physical 229 reels of tape are kept in the Green Library. The SAIL DART Archive project only deals with the 50 Gigabyte image of the final 9-track tapes.

Tape Inventory Table and Thesis motive.

year	reels	tapes	files	size	
1972	3	33	32,774	0.4 GB	D.C.Power
1973	13	205	135,134	2.4 GB	Lab
1974	12	191	134,976	2.3 GB	
1975	12	194	137,037	2.2 GB	
1976	12	196	129,178	2.2 GB	
1977	15	210	142,389	2.4 GB	
1978	16	266	165,106	3.2 GB	
1979	19	298	179,349	3.6 GB	
1980	15	229	125,504	2.8 GB	Margaret
1981	21	335	171,410	4.1 GB	Jacks
1982	18	208	117,491	3.1 GB	Hall
1983	12	132	79,264	2.3 GB	
1984	16	157	85,058	2.9 GB	
1985	8	71	43,403	1.5 GB	
1986	14	47	63,219	2.6 GB	
1987	11	51	44,350	2.2 GB	
1988	16	54	65,294	2.9 GB	
1989	8	48	27,224	1.6 GB	
1990	7	32	11,996	1.2 GB	
	248	2957	1890156	46. GB	naive totals
19 years		2984 tape numbers			canonical
	229 reels		50 Gigabytes		totals
		2,000,000 file tags			
		900,000 blob values			

[1] It is important to have one footnote. The first Stanford A.I. Lab was merged into CSD in 1979 when the buildings changed from DCP to MJH. The A.I. winter lasted 25 years. The second Stanford A.I. Lab was founded in 2005.

The collection of WAITS backup DART tapes contains approximately 50 gigabytes of data. There are only 229 reels; reels spanning two years are counted twice in the naive total. The maximum tape number used was 2984, the saga of lost and non-existent tapes is recited else where. Tapes 1 to 2746 were 7-Track 800 BPI. From September 1985, tapes 2747 to 2984 were 9-Track 6250 BPI.

2.1.1 "I Got My Thesis !"

I returned to the Lab in 1998 (*well but 'my Lab' had moved to Campus, the backup data had been re-copied once, details...*) and I asked Marty Frost if I could retrieve my thesis work and its accompanying code and its 3D data models, raster images, vector plot files and whatever else was saved on the DART backup tapes. It was easy to service this request. First, we did not bother looking for an index (there wasn't one yet); we started with the first reel of physical tape, which happened to be on-site. Then we found a suitable tape drive, conveniently located nearby on the loading dock at Allen; it was about to be taken away as junk. We pushed it across Sierra Street (which has since been renamed Jane Stanford Way) from Allen to Gates. The 1998 copying was done using the low level Unix (Solaris that year) utilities, dd and tar. Finally SMOP, Small Matter of Programming, I hacked into the records found on the tape using od, octal dump, and a bit of 'C' code named readtape.c which I expanded substantially and later re-named undart.c. I happened to have had hardcopy paper manuals for PDP-10, WAITS UUO and WAITS Monitor that I had kept from the 1970s. I also happened to recall a lot of WAITS details such as the SIXBIT encoding of filenames, and I was at the time fiddling with large data file systems at IBM Almaden Research as my day job. So this vast number of WAITS tapes proved easy for me to use. YMMV by quite a bit.

2.1.2 Historical Value

Answering Zuzga, these days we lookup the etymology of "*Foobar*" and the chronology of "*Minsky K-lines*" using the search engines. Those search engines have constantly visited the SAILDART archive web site since it first appeared on the web in 2002. The earliest Archive.org Wayback Machine SAILDART crawl image is dated 11 October 2006; *"I had thought that many years would pass before the cybersapiens would read this archive, but Googlebot/2.1 was already chewing on it back in 2002."*

2.1.3 Intellectual Property and Patent Searches

Again. The valuable information and knowledge from the A.I. labs at MIT and Stanford has been published and monetized early and often. The idea of investigating intellectual property and priority of discovery issues does not appeal to me. *Do not awaken sleeping giants. Do not meddle in the affairs of Wizards, for they are subtle and quick to anger.* Avoid patent trolls and copyright mickey mouse.

Nevertheless the SAILDART has several controversial priority cases that may yet attract further investigation; if anyone still cares about the exact origins of Sun Microsystems and Cisco Systems; or possible prior art examples predating Adobe font technology patents. And Kodak patented the *digital camera* years after SAIL and others demonstrated digital imaging technology.

2.2 Five Practical Alternatives to a Universal Format.

After SAIL MCOPY copied the old tapes onto new tapes; and after SAILDART copied those new tapes onto SCSI disks; the five alternatives that Zuzga considered as impractical in 1995 were all implemented to some extent before 2020.

2.2.1 Preserving the Byte Vector

Reading 229 reels in 1998 generated 41,594 files most of which contained valid DART version #6 formated records. The header of each record has its length in PDP10 words. There were only 61 gaps where the record header count (and other fiducial markings) failed to appear where they were expected to be. So I coined a new "GAP" record type and have concatenated all the tape bytes into one file named **flat_DART_data8** with its MD5 hash value **3adbff17fd7f9f6eb9107755594ae0b9**.

2.2.2 Maintaining Vintage Systems

The Living Computer History Museum in Seattle has both a working KL10 and a working KA10.

2.2.3 Emulators

Software for PDP-10 emulators exist - see GITHUB. There is canonical specification for the PDP-10 instruction set, however the (hand drawn !) wiring diagrams exist. The PDP-10 at Stanford was serial number 32, and was modified, with two significant opcodes, XCTR and FIXX. Also the I/O devices included several crucial one of its kind implementation for the XGP, the III vector display, the Datadisc raster displays, the video crossbar switch, the audio crossbar switch, the color synthsizer, and four different robotic arms the TV input, and the mobile CART. The PDP-6 at Stanford was serial number 16 and had a CONS instruction which was never used.

2.2.4 Maintaining the Operating System and Relevant Applications

The MIT operating system was named ITS. The Stanford operating system was belatedly named WAITS. Prior to 1978 the operating system boot initialization

banner name was simply SYSTEM and the manuals often refered to it as the MONITOR, hence that iron clad boat image on the early monitor manuals.

2.2.5 Translating the Data into so called *Standard* Formats

One of the major improvements in computer software technology since the 1990s has been the universal adoption of the UTF8 character encoding. The SAIL 7-bit non-standard ASCII is easily converted into UTF8 text.

Chapter 3

Rosetta Stones and Fortune Cookies

The Rossetta Stone in the British Museum (stolen from France, who stole it from Egypt) has three incomplete copies of an edict issued 27 March 196 BCE (on the negative ISODATE -195-03-27 uh, was that on a Sunday ? Lots of priests were standing around who are listed by name. ISO 8601:2004 includes a year zero.) the lower copy on the stone is written in greek, the middle copy is in demonic and the upper copy is in Hieroglyphic. The Rossetta Stone provided crucial hints to reading ancient Egyptian Hieroglyphics. The Rosetta lesson is: "leave several copies in different formats". Similar to the rule: "Lots Of Copies Keeps Stuff Safe". Do reflect that the Rosetta Stone is inadequate for what you need to understand Egyptian Hieroglyphic text which should include a grammar (I started with Gardiner 3rd edition Oxford) a dictionary, a thesaurus and we might as well throw in the complete surviving corpus of ancient Egyptian text. Here, here and here. In the plural, generic rosetta stones pave the way to natural language understanding software; for example, the corpus of parallel Canadian French and Canadian English government documents is one such rosetta stone highway.

One of the early "chat bots" was the program named FORTUN which selected a line of text from the file FORTUN.TXT[2,2] which is the first reasonably long public text file on the SAILDART tapes. I like to consider the email messages as Private and the first version of the MAINT.TXT[2,2] file is only three lines of text and the first NOTICE.TXT[2,2]

Do read the Wikipedia article concerning the Rosetta Stone, as well as the article about Fortune Cookies.

3.1 Design Goal : enable the reader to read the data

The zero-information time-capsule bootstrap scenario depicted by Carl Sagan and others is unnecessary, but fun to think about now and then. English is the Greek of our day. UTF8 is the ASCII of this decade, the 2020s.

3.2 The Solution : Name Value data sets

Zuzga invented yet another name-value notation; while for SAILDART I have not yet moved off much from CSV, Comma delimited String Values. My bench top working dataset starts from DATA8 and/or OCTAL on whatever Linux file system is best supported and widely used in a given year. I was very fond of RieserFS prior to the uxoricide, but for 2021 SAILDART is on ext4 or xfs.

Chapter 4

Translate Geek 1970 into Geek 2020

Convenient for SAILDART, there is only the one dataset which is the DART backup reels from 1990 which captured the earlier DART backup tapes. This chapter is a narrative on translating the DART data from PDP-10 WAITS 1970s formats into 2020 GNU/Linux platform formats which are internet compatible. Unlike Zuzga, I am not providing an example of *platform-independence* but merely discussing the conversion from the old-platform to a newer-platform, which is the the IPv4 and IPv6 internet. In the long view, the internet formats will change to communicate with the extra terrestials and with the future, beyond Y10K, for the platforms built and inhabited by our cybersapien successors. Unless a nearby supernova gamma ray event sterilizes our neighborhood.

4.1 Capture

The nasty chore of copying about 3000 old 7-track tapes into late 20th century 9-track tapes was done by Marty Frost, before my involvement with DART data preservation.

4.1.1 Design

The MCOPY 1990 capture software is available inside the SAILDART for your inspection. The SAILDART 1998 capture simply applied the unix command shell tools **dd** and **tar**.

4.1.2 Media Memory Management

Even the second capture process in 1998 left iron oxide on the tape heads and on the operator's hands. We cleaned the tape heads and tape path with isopropyl and Kimwipes, before reading each and every tape.

Our attitude in 1998 was that of informal hacking applied to securing a further sample of the DART data of unknown value or more likely of insignificant

value; rather than an attitude of careful professional expertise to recover high value data. I had been lucky - my personal data was on the first reel - going for all 229 reels, at 20 minutes each, was a bit of a chore as well as an extra commute; although I happened to own, and to be slinging around, quite a few 4GB Maxtor disks that year. Larry and Sergey's Duplo kludge was still sitting in the next room at Gates in 1998. My disk kludge "big-data" experiment was four disk drives in each of many pendflex folders, with tens of folders hanging in plastic milk crates; with bare motherboards and SCSI controllers.

4.1.3 Contemporaneous Logs

The 1990 months of MCOPY work left extensive log files, reports, summaries and email messaging. The 1998 weeks of DD_TAR have no logs; I did find a single sheet of paper with a few Post-it like notes left at the keyboard as we changed the input tapes or the output SCSI disk units:

```
5 March 1998
11:15 pm left tape #p3061 reading BgB
6 March 1998 4pm start #p3062
11:20 pm finished #p3085
Lower disks ARE Brand New
console break
NO DISMOUNT WAS POSSIBLE WITHOUT ROOT Permission. BgB 11:30pm

Replaced two disks
left tape #3086 reading in
2:30 pm Sat 7 March BgB

PAST HALF DONE !
Tapes read as marked
with Green Stickers
98.3.9 / 10pm
BgB
```

4.2 Translation

4.2.1 SAIL text into UTF8

4.2.2 MFD and UFD user codes

4.2.3 Binary into CSV, SVG, PNG and octal

4.2.4 Software: the PDP10 machine code into 'C' hack

4.3 Handoff the Baton - Output to SD chips

Chapter 5

Haystack Access, Search, Privacy

5.1 How Haystack Heaps work

5.2 What Z. Would Really Prefer

In the Brian Zuzga paper an adequate database of the contents of the MIT archival tapes is envisioned. Related to this paper, a database of the contents of the Stanford archival tapes has existed and been used for some twenty years; but it has always been easy to envision improvements.

5.3 Privacy and Permission

"It is easier to ask forgivenness, than for permission" John McCarthy told me; he further advised "do not be in a hurry to contact Stanford administrators".

Chapter 6

Road Map Atlas

6.1 File Classifier

- Envelopes and Wrappers
- Metadata that is in-band or out-of-band
- carrier, noise, signal and payload
- page attributes

6.1.1 Identify the PDP-10 Executable Files

- **by dot extension name .DAT**
- **by the JOBDAT 96. word header**
- **by the RUN command — can this dog run ?**

For SAILDART almost all the PDP-10 executable files are dump files with the DMP extension. Many dump files can be found renamed with the extension OLD. There are also executible files for secondary processors such as the SDS40, the PDP-11 and the IMLAC. PDP-10 executables dominate the SAILDART corpus. The PDP-10 DMP executibles must have a ninety-six word header that is called the Job Data Area, JOBDAT. The integer ninety-six in octal is written "0140" in 'C' or Javascript, "0o140" in python and horrible to relate it is "octal!140" in 'D'.

6.1.2 Distinguish Text vs. Binary

In retrospect, this test now seems to be a trivial problem. The dominate text editor was named E. (Enterprise, the aircraft carrier, is on the E manual cover. Enterprise, the starship, is on the GEOMED cover.) E maintained a table of contents at the front of each file; the first line is always like this "COMMENT

⊗ VALID 00008 PAGES". Only the numeric characters may vary to indicate the number of pages. The earlier text editors, STOPGAP and SOS (Son of Stopgap) had distinctive line number conventions so that looking at bit #35 for line numbers is also an easy test.

The full bore histogram of the SAIL 7-bit bytes for character frequency of a file also easily distinguish text from binary; except I initially mislead myself on the class of files that have printer hardware escape sequences XGP DVI POX and so on, and then are the files with a lot of markup text or really binary ? unfortunately the answer is NO, highly marked up text is NOT binary it is still text, but not what I would consider "natural text" or "abstract text" or "simple text" which might be a 1D vector or list of UTF8 characters.

6.1.3 More File attributes

Public or Private

The obvious public areas on the SAIL-WAITS time sharing system start with the binary executibles in [1,3] and the documentation in [DOC,SYS] and even a lot of the source code in [CSP,SYS] was public.

Quality and Quantity

Component or Conglomerate

Datetime stamps and PRG code authors

The naive belief that the PRG codes correspond to proving the authorship (or ownership) of files on the 1970s backup tapes; is only exceeded by a naive belief in the datetime stamps of files. Most of the PRG codes indeed correspond to the author of the files, and almost all of the datetime stamps are indeed credible. However, my file **QUEENS.FAI** shows up first timestamped and PRG code DBA (Bruce Andersen). Back when the puzzle problem of how many ways can you arrange five queens to cover (attack) every square on the chess board, I was the only one to code my solution in machine code.

Even the matter of associating PRG codes with human names requires paying close attention to the in band SAILDART personnel records; and reconstructing which records were primary and which were derived in each year of the epoch. Some what starts off in Queenette Bauer's personal disk area gets expanded by Lester Earnest and automated by Ralph Gorin and finally is absorbed into the Stanford Computer Science Department as it growns in size well beyond the original SAIL as the world learns how to write and then how to run database software.

File Protection and File Mode

The SAIL file protection bits are NOT unix and changed slightly in meaning over the years.

Redaction and Damaged

By volume most of the file redaction is for redundant material, the processing is call dedupping.

6.1.4 File Type by dot ● Extension and Write Tool

File type identification by extension code and by the name of the program that wrote the file. Because of the short fixed length file names (one to six characters), the disk retrieval info block could log the name of the program that wrote the file in a single PDP-10 word. And all those disk Retrieval Info Blocks, RIB, are included in the DART metadata for every file fragment ! Again and again and again ad nauseum.

6.1.5 Classification Recapped

6.2 Concordance

A brute force concordance, which can be made using a short perl script, was easy to generate and somewhat useful in the early 2000 aughts. One initial surprise back then was that there were files in the tape archive that had almost every English word ! Sheesh, I had forgotten my associates who invented Spell Checkers, Word Hyphenators or worked on Natural Language software in the early 1970s. Also potentially useful were all the hits on human names, but that becomes tiresome when you begin to comprehend the extent that a 1970s university time sharing computer was just a huge Rolodex, as well as a recipe notebook and a dumpster full of newspaper clippings from the earliest experimental NYT and AP wire service computerized news.

Another catagory of easy classification, *like shooting fish in a barrel*, is email. At SAIL on WAITS the partial differential character ∂ is the prefix for the date of a message, and by some dumb fluke the ∂ glyph was used for very little else. So grep on ∂ and split and sort and load into your data base to see all the email / bulletin board / message traffic for two decades for a couple of hundred opinionated people who lived in the previous century.

6.3 Production Quality Software

6.4 Authenticity and Provenance

The SAIL accounting records are excellent. In the 1970s computer service was expensive and time sharing required accurate accounting on how the limited funding was shared to fairly cover the users' session time at a console, disk space, memory used and CPU execution cycles consumed. The three dimension were measured in units minutes of time logged in, Tracks-of-Disk space, and Kilo-Core-Ticks of memory x CPU-time. (Reservations at Stanford, followed

an Ivan Sutherland invention from MIT / Lincoln Labs of bidding in Whams and Bams on a paper sheet for hours of console time and machine time).

Due to the 2020 plague, I reread the historical novel *A Distant Mirror* by Barbara Tuchman 1978, set in the period of the Black Death 1348-50 with the protagonist Enguerrand de Couey VII the source of this history is the daily logs and accounting records kept by scribes employed by the noble gangster knight to keep track of all his dirty money transactions.

Chapter 7

Final Remark

"Truth is found in synthesis which reconciles thesis and anti thesis."
– Georg Friedrich Wilhelm Hegel (his dialectical method paraphrased)

In summary, Zuzga's thesis is that there should be one universal archival format, one ring to rule them all. Baumgart's antithesis is that digital archiving is a relay race, you carry the baton for whatever distance you can and hand it off to others translated into the most common open formats prevaling in your final days.

Chapter 8

Appendix A
DART Format Details

8.1 Narrative Description of the DART format.

Five bytes to the PDP-10 word

When writing to 9-track magnetic tape, PDP-10 words of 36-bits were transferred by the I/O hardware into bytes in big endian order with the final four bits in the low order of the fifth byte.

The thirty-six bit SAIL computer words were written to tape, big endian, in five octet bytes. The fifth byte of each word has four low order data bits and four bits of high order zero-bit padding. When the tape drives loses sync, octets can be lost or inserted, and the remaining words would be garbled, however by scanning in the serial byte stream for DART record landmarks, the position of the misaligned octets can be adjusted and the remaining words of the record properly aligned.

Seven bit SAIL text encoding

Most text at SAIL was encoded in a 7-bit ASCII where the thirty-one characters after zero 000 ASCII NUL were mapped into non-Standard non-ASCII glyphs. For example 001 was ↓ down arrow. Unless your are listening to this on ear buds while flying across the Pacific, *Look* at the SAIL-to-Unicode table and *Skip* reading the next paragraph.

Continuing as narration, I wish to recite the SAIL octal codes for the **arrow** glyphs as 001 down arrow, 027 double arrow horizontal, 031 right arrow, 136 up arrow and 137 left arrow; and which are *exegesized* into the Unicode hexadecimal codes U2193 , U2194, U2192, U2191 and U2190 respectively. The **Greek letter** glyphs at SAIL were octal code 002 for α alpha, 003 for β beta, 006 for ε epsilon, octal 010 for λ lambda, and 007 for π pi. The five Greek letters become Unicode u03b1, u03b2, u03b5, u03bb and u03c0. For **Logic and Math** the SAIL codes 004, 037, 024, 025, 005, 026, 016 and 017 represent

glyphs for boolean AND, boolean OR, for each, there exists, boolean NOT, XOR as a circle X, infinity as the lazy eight symbol and the partial differential operator. These codes respectively become Unicode u2227, u2228, u2200, u2203, u00AC, u2297, u221E and u2202. Then a few extra mathematical **relation and horseshoe** symbols encoded at SAIL as octal 033, 034, 035, 036 and 020, 021, 022, 023 which are again in order are $\neq \leq \geq \equiv$ and $\subset \supset \bigcap \bigcup$ that become Unicode u2260, u2264, u2265, u2261 and for the horse shoes u2282, u2283, u2229 and u222A.

The point here is that even if this Prolegomenon and its Exegesis are lost (or are not provided to an archive decoding test candidate) the DART gram is not hard to interpret after the idea of 7-bit characters that are nearly ASCII is re-discovered. Then the DART record metadata are merely irritating hiccups in a stream of text. Reading all that text provides the future Cyber-Sapien archivist (*long after the self destructive Homo-Stupids have disappeared*), with the actual software which wrote the DART message, late in its analysis the decoding expert will "see" the glyph shapes in the binary font files or in the TEX metafont files.

SIXBIT file name encoding

The SAIL-WAITS file system is primitive, it was a tool of pioneers working on the frontier. Filenames were one to six characters, optionally followed by dot and a one to three character extension. All filename alphabetic characters where uppercase. The character codes for A to Z were six bits wide as octal 041 to octal 074. The digits are octal 020 to octal 031. The blank is zero. Filename characters on the DART media (as well as internal to the Operating System) could have any of the 64 character values.

SAIL-WAITS file system

Each file belonged to a directory specified by left square bracket project code comma programmer code right square bracket. The project and the programmer codes where each one to three characters long.

Alien to the SAIL file system is the now familiar file system concept of having content blobs separate from directory entries. On GNU/Linux file systems, one or many file path names may be hard linked to one content blob, which was impossible in the SAIL-WAITS file system. At SAIL the early disk hardware was unreliable so that a seek command was not be fully trusted to get to the proper cylinder, head and sector of a disk drive. So the file name (directory entry), which SAIL called the Retrieval Information Block (or RIB), included the file name and was written into each data block (called a Track) that was needed to hold the file's data. And so too on the DART tapes within the sequence of FILE tape data blocks each tape block has a full copy for the Retrieval Information.

Segmentation into DART records

There are two original DART record types: Tape-Marker (Head or Tail) File-Data (Start or Continue). I have added a third record type named Gap, to passover the 61 segments of bytes which failed to decode as DART records. Previewing the data shows that all the Tape-Marker records are exactly sixty bytes long and each contains the tape reel number and a date-time stamp. The first word of each DART record has its record size. The record lengths segment the whole DART byte stream with only 63 defects, continuing after a defect requires scanning for the next sane record.

With the extreme precision that is available to latter day archival software, the DART segmentation goes as follows: the long byte vector, of exactly 56_446_334_821 bytes, contains exactly 2_937_291 short segments of which 5_486 are head-tail records, 1_886_472 are file-start records, 1_045_270 are file-continue records plus the 63 gaps. Or more colloquially, the fifty-six gigabytes of tape data have nearly three million short records which contain the data and the names of about one million old SAIL files.

Three further mechanisms need to be previewed here. First, it was the intentional DART backup policy to write two copies of each SAIL file that was deemed of permanent value to two different permanent backup reels of tape. A file found by the utility programs named DSKUSE and DART resident on the SAIL community commons SYS: disk system would be marked as archived once, then marked as archived for a second time, and then there after omitted from further archiving. So each SAIL file should appear in the dart record in two places in the tape records with the same identical content, name and date-time stamp. Second, a unique SAIL-WAITS filename will appear again (with yet two further copies each time) for each newer date-time stamped revision. Generally human edited files do not change very much between revisions. Third, it was the unintentional result of unreliable disk seeking mechanism that meant that file retrieval information including the file name was stored multiple times within the file "blocks" on the disk media. That meant that the SAIL-WAITS file system would contain multiple copies of exactly the same content of a file when a file was copied from one user directory into another. Other kinds of short files (the professional digital archivist term for these files is "turd" or "fart") are generated by common utility programs in many user directories with content of no value to the historical record aside from traffic analysis. The result is that the population of 1_886_472 SAIL files in the DART halves to fewer than 900_000 different content blobs, each content blob has one to many hundreds (and for a couple of blobs even thousands) of directory entry name tags (aka retrieval information) rows in the database table of the SAIL-WAITS file names.

FILE-START and FILE-CONTINUE tape records File-Start and File-Continue records are identical in format and in content of their file metadata. The File-Start is marked type -3 in the left half of word 0. and the constant sixbit/*FILE*/ in word 19. The File-Continue record is marked type 0 in the

left half of word 0 and sixbit/*CONT*/ in word 19. So describing them both as FILE blocks they have 36. words of prefix, then up to 10240. words of data payload, then a 23. word postfix which is most often completely zero except when a few bits are tinked pursuant to observations of error conditions in the reading of the low density tapes.

The FILE metadata is sixbit/FILNAM/ sixbit/EXT/ sixbit/PRJPRG/ the length of the file in words, SAIL-WAITS protection bits, mode that the file was written, and a date-time stamp.

The file data block records seen on the high density tapes are surprisingly fat considering the computer poverty of the prior 18 year period. The explanation is that the DART data format version #3 was a final revision done to handle the massive MCOPY of the 3000 old tapes into the newer higher density ones, the format was over ambitious and had allocated many bytes of space that were never used.

HEAD and TAIL tape records All the tape HEAD and TAIL records are exactly 60. bytes long. Each contains 12. PDP-10 words. Seven of the twelve words have a fixed constant value, making the HEAD-TAIL records easy to find in a byte string, the other five words carry a date-time stamp, a checksum for 10. words of the HEAD-TAIL record, the tape reel number and the tape position in feet from the tape load point which is irrelevant to the SAIL-WAIT file system but it is amusing to know where the low density tape reel images fall within the high density tapes.

There are 41_594 tape records from the higher density tapes, which each in turn contain 1 to 100 or so small records from the lower density tapes. In total there are 2_934_700 of the small records plus the 63 gaps.

The 229 reels of high density DART tape are labeled P3000 to P3229, as mentioned earlier, the reels still exist and are kept in the Stanford University Digital Archive housed in the Green Library building on the campus in Palo Alto, California. Each tape contains high density (6250 bpi) records. Each high density record is a concatenation of records from the lower density (800 bpi) tapes which were label P1 to P2984. The letter 'P' indicated Permanent backup tape as oppose to the incremental ones which were marked 'T' for Temporary. The final reel of Permanent Tape was written 16 August 1990 and that reel of tape was copied to disk in March 1998, however the earliest file I have from that reel is time stamped 17 June 1998. The rescue of the high density tapes to disk was not a well documented process, the quantity of old tape in the basement of the CSD building was overwhelming, the speed of the tape drive was slow, the working hours were 2nd and 3rd shift, the disks drives were nine Gigabytes each and were taken off site to copy into several other systems since there was not enough disk space available to us on a single system.

The low density reels were written over a period of nearly 18 years. The HEAD of tape #P1 is time stamped 1972-11-05T11:59 and its TAIL is marked 1972-11-05T12:23 which implies that first tape took 24 minutes to be written on a quiet Sunday in November around lunch time. Richard Nixon wins re-

election to a second term as president of the United States on the following Tuesday 7 November 1972.

The high density reels were written over a period of nearly 31 months. The HEAD of the first high-density DART tape #P3000 is time stamped 1988-02-01T17:17 The TAIL record on the final high density tape #P3228 is dated 1990-08-16T22:55 so at nearly 11 PM on Thursday in mid August the DART record ends. Iraq had annexed Kuwait during the first week of August 1990.

The final lower density tape #P2984 is time stamped 1990-08-17T16:43 which overlaps the time period in which the final high density tape is written.

GAPS

The data found in the 63 gaps, is assigned its MD5 blob serial number and tagged with a unique SAIL file label and included in the SAILDART collection as allowed by KISS design authority (Keep It Simple Stupid) principle and the Brewster Kahle archiving principle of keep everything you can but don't fret the details. Working at the Internet Archive we would boost that we were going for Quantity first, not Quality; the SAILDART data of 1998 is a pleasant past time since its Fixed Quantity becomes a lot easier to manage with each passing year.

8.2 DART formats Illustrated.

Six frames of 7-track tape supply a 36-bit PDP10 word

frame 1	frame 2	frame 3	frame 4	frame 5	frame 6
A A A A A A	B B B B B B	C C C C C C	D D D D D D	E E E E E E	F F F F F F
Bits 0 to 5	bits 6 to 11	bits 12 to 17	bits 18 to 23	bits 24 to 29	bits 30 to 35

Table 8.1: 7-track PDP10 word

Five frames of 9-track tape supply a 36-bit PDP10 word

frame 1	frame 2	frame 3	frame 4	frame 5
A A A A A A B B	B B B B C C C C	C C D D D D D D	E E E E E E F F	0 0 0 0 F F F F
Bits 0 to 7	bits 8 to 15	bits 16 to 23	bits 24 to 31	bits 32 to 35

Table 8.2: 9-track PDP10 word

7-bit SAIL ASCII to Unicode and UTF-8 table

	0	1	2	3	4	5	6	7
000	null	↓	α	β	⋀	¬	ε	π
010	λ	\t	\n	\v	\f	\r	∞	∂
020	⊂	⊃	⋂	⋃	∀	∃	⊗	↔
030	_	→	~	≠	≤	≥	≡	⋁
040	␣	!	"	#	$	%	&	’
050	(	)	*	+	,	-	.	/
060	0	1	2	3	4	5	6	7
070	8	9	:	;	<	=	>	?
100	@	A	B	C	D	E	F	G
110	H	I	J	K	L	M	N	O
120	P	Q	R	S	T	U	V	W
130	X	Y	Z	[	\	]	↑	←
140	‘	a	b	c	d	e	f	g
150	h	i	j	k	l	m	n	o
160	p	q	r	s	t	u	v	w
170	x	y	z	{	|	ALT	}	BS

Table 8.3: 7-bit SAIL ASCII code

6-bit ASCII minus 040 code table

	0	1	2	3	4	5	6	7
00	␣	!	"	#	$	%	&	’
10	(	)	*	+	,	-	.	/
20	0	1	2	3	4	5	6	7
30	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G
50	H	I	J	K	L	M	N	O
60	P	Q	R	S	T	U	V	W
70	X	Y	Z	[	\	]	^	_

Table 8.4: 6-bit SIXBIT code

DART tape HEAD and TAIL record format.

word	name	value	description
0.	Type_Size	000006_000013	type#6 11.words
1.	_DART_	444162_640000	`sixbit/DART␣␣/`
2.	BOT_EOT	125045_414412 126441_515412	`sixbit/*HEAD*/` `sixbit/*TAIL*/`
3.	date_time		
4.	ppn	445560_637163 005543_637163	`sixbit/DMPSYS/` `sixbit/ MCSYS/`
5.	Class2 Tape	XWD 2,Tape#	
6.	Rel_Abs		
7.	feet		
8.	word8	0	`0`
9.	minus1	777777_777777	`-1`
10.	word10	0	`0`
11.	checksum		`Rotated`

Table 8.5: DART tape HEAD and TAIL record format

DART file START and CONTINUE record format.

The MCOPY version#6 DART file (start and continue) record format has five parts.

	words	name	description
I	2.	TypeSize	DART record Type and Size
II	16.	RIB	WAITS File System Retrieval-Info-Block
III	18.	Leader	MCOPY extra baggage
IV	$0 \leq N \leq$ 10240. $-$ 61.	Payload	portion of the actual file data
V	23.	PRMERR	Previous Media Errors

Table 8.6: DART file START and CONTINUE record format

Diagram of parts I and II

octal	decimal	symbolic	value	comment
...	0	type_size	(-3 or 0)„size	size is 2 short of record length
...	1	dsk_or_error	'DSK'	constant
000	2	DDNAM	'filnam'	file name
001	3	DDEXT	XWD 'ext',Date	create (c)Date
002	4	DDPRO	prot, mode, time, date	write (m)Date Time
003	5	DDPPN	XWD 'prj','prg'	project programmer
004	6	DDLOC	track#	disk track
005	7	DDLNG	file length	PDP10 words
006	8	DREFTM	reference date time	(a)Date Time
007	9	DDMPTM	(T or P)dump date	(d)Date
010	10	DGRP1R	=1	first group
011	11	DNXTGP	=0	next group
012	12	DSATID	03164236 then 'RSK' or 'TSK' or 0	Storage Allocation Table ID
013	13	DQINFO	=0	defective 154 times
014	14	zero14	=0	defective 32 times
015	15	wrtool	'progrm'	write program name
016	16	DDWPPN	XWD 'prj','prg'	write project programmer
017	17	DDOFFS	=1	

Table 8.7: DART file record details I and II.

Diagram of parts III, IV and V

octal	decimal	symbolic	comment
022	18	_DART_	sixbit/DART␣␣/
023	19	File_Con	sixbit/*FILE*/ or sixbit/CON␣␣#/
024	20	date-time	when MCOPY reel written
025	21	MC_SYS	sixbit/␣MCSYS/
026	22	two_reel	XWD class=2 and MCOPY reel#
027	23	one_one	XWD 1 and 1
030	24	Feet	MCOPY reel position
031	25	0	
032	26	-1	
033	27	0	
034	28	Words_To_Go	payload words remaining in file
035	29	0	
036	30	0	
037	31	0	
040	32	0	
041	33	0	
042	34	0	
043	35	0	
044	36...	file blob data payload	
000	-23	PRMERR	0
001	-22	“	0
002	-21	“	0
	...	“	0
024	-3	“	0
025	-2	’$PEND$’	046045_564404
026	-1	checksum	XOR

Table 8.8: DART file record details III, IV and V

Chapter 9

Appendix B
REMIX Format Details

REMIX unpacks the `flat_DART_data8` file into six csv files named attributes, coordinates, dates, ribs, tags and tapedex. The first line of each file suggests database field names.