Addressing Precision Limitations in the Seismic Analysis Code (SAC) File Header and Data Format

SAC format is 20+ years old. Precision in timing and (perhaps) distance has improved beyond the SAC single-precision capability.

Authors: Brian Savage (Geosciences, University of Rhode Island);
J. Arthur Snoke (Geosciences, Virginia Tech)

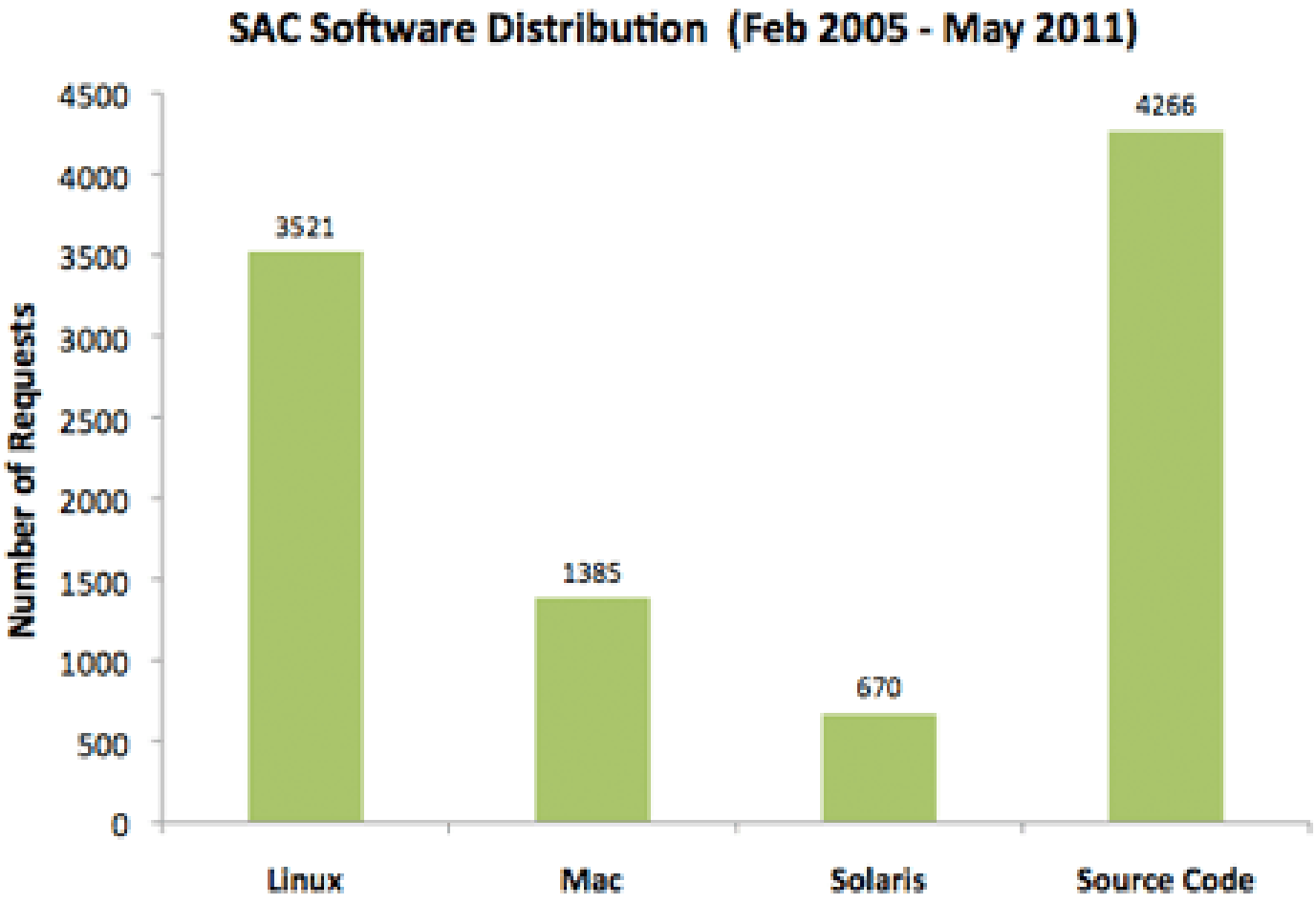
Eastern Section SSA Meeting
Blacksburg, VA
30 October 2012

Outline

- What is SAC?
- Current (Version 6) SAC Header
- Precision Primer
- Which variables to promote
  - timing? distances and geographic variables? amplitude?
  - blackboard variables? and/or internal calculations? (separate from SAC format)
- Implementation Considerations: Backward compatibility; Coordination with other programs that use SAC format; I/O routines
- How do we proceed?

Seismic Analysis Code (SAC)

- Developed in 1980s by LLNL as a seismic data file format and processing package
- Until 1994, source code available through LLNL
- 1994–2005 only binary distributions available from IRIS for a few operating systems (Vax, Sun, IRIX, Mac, ...
- Since 2005, Source and/or binaries available through IRIS DMC. Brian Savage and Arthur Snoke have overseen development and the five updates.
- SAC files are used by many other programs, and SAC I/O allow researchers to use SAC in own programs
- Programs gsac (Bob Herrmann) and MacSac (George Hellfrich) parallel packages



Ten years ago mostly Sun Solaris (big endian), now mostly Linux (little endian). Mac has switched from big endian to little endian.

Current (v6) SAC Header

0-7	delta	depmin	depmax	scale	odelta	b	e	o
8-15	a	fmt	t0	t1	t2	t3	t4	t5
16-23	t6	t7	t8	t9	f	resp0	resp1	resp2
24-31	resp3	resp4	resp5	resp6	resp7	resp8	resp9	stla
32-39	stlo	stel	stdp	evla	evlo	evel	evdp	mag
40-47	user0	user1	user2	user3	user4	user5	user6	user7
48-55	user8	user9	dist	az	baz	gcarc	sb	sdelta
56-63	depmen	cmpaz	cmpinc	xmin	xmax	ymin	ymax	unused
64-71	unused	unused	unused	unused	unused	unused	nzyear	nzjday
72-79	nzhour	nzmin	nzsec	nzmsec	nvhdr	norid	nevid	npts
80-87	nsnpts	nwfid	nxsize	nysize	unused	iftype	idep	iztype
88-95	unused	iinst	istreg	ievreg	ievtyp	iqual	isynth	imagtyp
96-103	imagsrc	unused	unused	unused	unused	unused	unused	unused
104-111	unused	leven	lpsol	lovrok	lcalda	unused	kstnm	
112-119	kevnm				khole		ko	
120-127	ka		kt0		kt1		kt2	
128-135	kt3		kt4		kt5		kt6	
136-143	kt7		kt8		kt9		kf	
144-151	kuser0		kuser1		kuser2		kcmpnm	
152-159	knetwk		kdatrd		kinst			

single
real*4
4 bytes

int
integer
4 bytes

Character
Strings

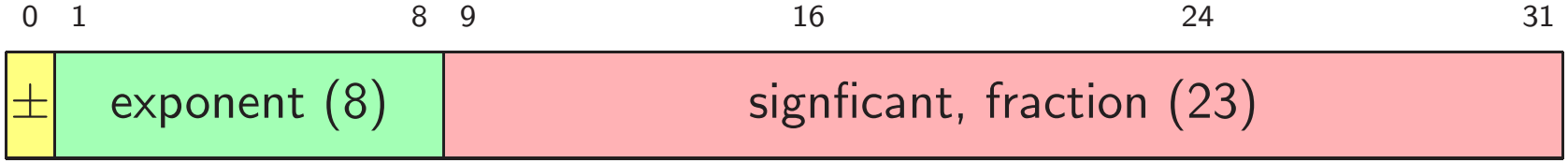
Size: $158 \text{ words} * 4 \text{ bytes/word} = 632 \text{ bytes}$

Variables labeled "unused" are inaccessible.

SAC Header

- If we promote some variables to double precision, where should they be placed?
- **Header variable nvhdr indicates the byte-order of the file**
- Several variables are not used within SAC but are available to be used. 18 4-byte values are inaccessible.
- Version 6 header organized by variable type:
  - 70 4-byte reals — last 7 are unused;
  - 40 4-byte integer in groups including 11 unused;
  - 192 bytes containing 22 8-byte character strings and 1 16-byte character string.

Precision Primer for Single Precision



- All reals in header/data are single-precision (32 bits; 4 bytes)
- For single-precision numbers, 23 bits define the maximum precision. All numbers between 2^{n-1} and 2^n have precision, DELTA, given by

$$\mathrm{DELTA} = \frac{2^{n-1}}{2^{23}}$$

  - ☛ If n $= 14$, $2^n = 16,384$, and DELTA $= 0.000976$.
  - ☛ If n $= 17$, $2^n = 131,872$, and DELTA $= 0.00781$.

Timing Precision

20 years ago, a sampling rate of 0.01 s was probably sufficient. Now thanks to GPS and other advances, many experiments require sampling rates of 0.001 s.

- For DELTA to be less than 0.01 s, the total time must be less than 131,872 s = 1.5 days (n = 17)
- For DELTA to be less than 0.001 s, the total time must be less than 16,384 s = 4.5 hours (n = 14)

Times within SAC are defined relative to a reference time. Timing variables are b, e, o, a, f, t0–t9 — 15 variables. If one is promoted, all must be. The smallest time increment within SAC I/O is 0.001 s (header variable nzmsec is milliseconds).

Distance and Geographic Variables Precision

It is now possible to measure locations to within meters.

- To the nearest meter, the maximum longitude is 179.9999. **The precision for latitudes/longitudes is ≤ 0.2 m.** That is marginally okay for one-meter precision.
- Half the circumference of the Earth is about 20,000 km. The precision for this distance is about 2 m, but a bigger limitation for large distances is the precision of the reference ellipsoid.
- Header variables eval, evlo, stla, and stlo are the prime candidates for promotion; Variables dist, az, baz, and gcarc would not need to be promoted.
- Relative positions require less precision than absolute position within SAC.

Amplitude Precision

- 20 years ago, 16 bits was the standard for data.
- Today 24 bits is standard. $2^{24} = 16,777,216$.
  - Precision of about 1.0
- **With current instrumentation, higher precision is not required.**

Maximum Number of Points

- Integer header variable npts has 4 bytes (32 bits).
  - Maximum is 2,147,483,647 (unsigned 4,294,967,296)
  - **If sampling rate is 0.001 s, maximum record length for 32 bits is 24.8 days.**

Developers' Current Thoughts on SAC Format

- Promotion of select values to double precision
  - Timing values **[Needed]**
  - Geographic values **[Marginally adequate]**
  - Amplitude values **[Not needed now]**
  - User values **[Probably some]**
  - Internal SAC **[Do all calculations in double precision]**
- Promotion of all header variables to double precision? **[No]**
- Header organization — still under discussion, but nvhdr offset must remain the same
- Promote blackboard variables to double precision

Conclusions

- Current SAC format limitations:
  - For high sampling rates and long record lengths, timing precision in SAC is inadequate
  - Geographic values are marginally adequate
  - Amplitude values are probably okay for now
- Promoting timing values addresses most immediate problems
- Increasing precision will change SAC header which has implications on existing programs that read/write SAC files as well as SAC I/O routines
- Developers are seeking feedback on these topics as well as how to handle backwards compatibility