Level 1B
Software Development Document

for the

Sounding of the Atmosphere using Broadband Emission Radiometry (SABER)

(Draft)

Submitted By: ____ ______________

G.M.Beaver, GATS Instrument Calibration Engineer Date

Concurred By: ____ ______________

L.L. Gordley, GATS CEO Date

G & A Technical Software
Hampton, Virginia

Contract No: NAS1-20472
GATS DOC No: GATS-SABER-L1BSDD-98-1.V1

CHANGE INFORMATION PAGE

List of Effective Pages			
Page Number		**Issue**	
Title		Original	
		Original	
		Original	
		Original	
		Original	
Document History			
Document Number	**Status/Issue**	**Publication Date**	**CCR Number**
	Original		

SABER Level 1B Software Development Document

1. Scope

This Software Design Document (SDD) describes an overall design plan for SABER Level 1B processing software. The software reads in Level 1A data, performs signal processing and correction, converts the signals from volts to radiance units, and removes instrument and spacecraft motion effects. The data are then geolocated and gridded to common angle & height scales and output as a daily NetCDF file (see Appendix A).

2 Referenced Documents

GIS
Level 1A Software Development Document: GATS-SABER-L1ASDD-98-1.V1

3 System Overview

The SABER Level 1B data flow is shown in Figure 1. The Level 1A file is ASCII format and is described in the Level 1A Format Document. It contains data merged into scan events, determined by the scan mode of the instrument.

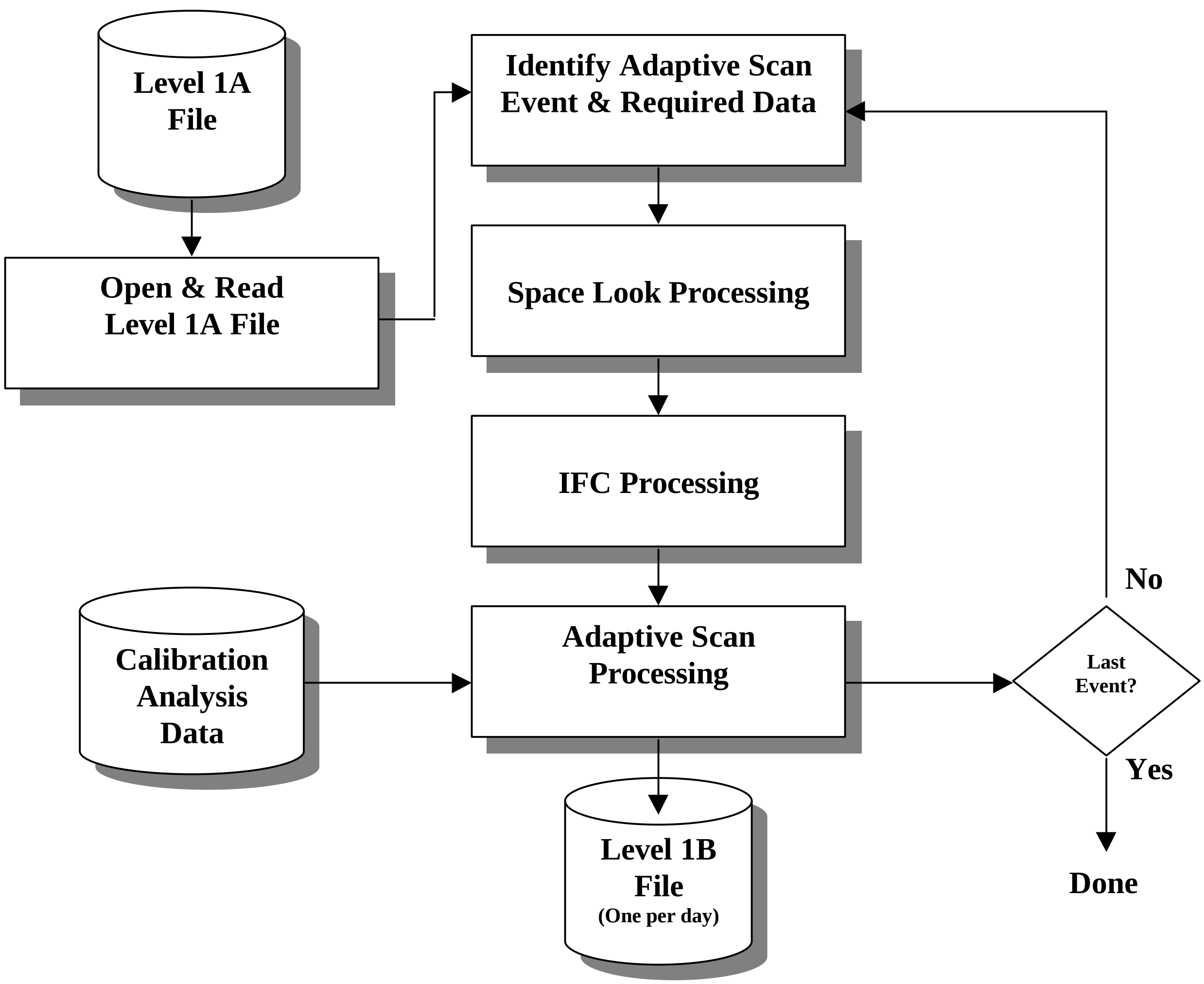

Figure 1: SABER Level 1B dataflow.

4 System Requirements

The Level 1B software must be able to run on Linux workstations. It must be able to read and write files across the Internet via NFS.

5 Computer Software Configuration Items

The requirements for each CSCI discussed in Figure 1 are presented below.

5.1 CSCI: Open & Read Level 1A Files

This CSCI opens and reads data from the SABER Level 1A file. The data are read into event classes, utilizing software from Level 1A processing with additional methods for reading.

5.1.1 Requirements

The CSCI must open a Level 1A file, determine the scan event mode from the scan mode flag in the event header, and load the data into specific classes depending on event type. Classes defined in Level1A processing will be reused by adding methods for reading in scan event data. Quality flags from Level 1A processing are tracked and passed with the data.

5.1.2 Testing

Each scan event class will have a method to dump data read in for comparison with Level 1A file.

5.2 CSCI: Identify Adaptive Scan Event & Required Data

This CSCI searches through the adaptive scan class, finds the next adaptive scan event, and identifies the nearest bracketing Space Look and IFC events.

5.2.1 Requirements

The CSCI must read through the adaptive scan classes and find the next down and up pair of scans. The nearest space look and IFC events that bracket these data are then loaded for calibration processing.

5.2.2 Testing

Test data will be generated using data taken during calibration that places the instrument in all scan modes. These data will be run through Level 1A processing to generate a Level 1A file. This ASCII file will then be examined to determine the each adaptive scan event and correct bracketing Space Look and IFC events are identified for each Adaptive Scan Event.

5.3 CSCI: Space Look Processing

This CSCI processes data taken by SABER when the mirror is staring at cold space to get a zero offset value for each channel.

5.3.1 Requirements

The CSCI must calculate the mean voltage for each channel for the duration of the Space Look events that bracket the current adaptive scan profile.

5.3.2 Testing

Measurements from calibration will be used to simulate Space Look events. These data will be evaluated for mean voltages for each channel. The mean voltages will be compared to the CSCI calculation.

5.4 CSCI: IFC Processing

This CSCI processes data from looks at the In-Flight Calibrator in order to get coefficients for converting volts to radiance units for each channel. The known radiance S(T) is measured on the ground by transferring Blackbody calibration measurements. The on-orbit calibration coefficients are then computed by C = V/S(T), where T is known from IFC temperature measurements.

5.4.1 Requirements
The CSCI must read detector voltages and IFC temperatures for the duration of the IFC events that bracket the Adaptive Scan event. The IFC Source function (S(T(t)) where T(t) is the time-dependent IFC temperature) must be calculated using the calibration transfer measured on the ground. The calibration coefficient (C(t) = V(t)/S(T,t)) for each sample in time, must be computed and averaged over the time duration of the bracketing IFC events to give the calibration coefficient for each channel.

5.4.2 Testing
Simulated data from ground calibration will be used to generate IFC events in test Level 1A files. These data will be evaluated for mean calibration coefficients that will be compared to the coefficients calculated by the CSCI.

5.5 CSCI: Adaptive Scan Processing
This CSCI processes (down-up) pairs of adaptive scans through the atmosphere.

5.5.1 Requirements
The CSCI must perform the following processing for each adaptive scan through the atmosphere:

-Deconvolve the electronic filter function from the data.
-Co-Align channels.
-Convert volts to radiance units.
-Estimate and remove any detectable spacecraft motion.
-Geolocate each sample and grid to uniform angle spacing.
-Write out each event profile in NetCDF format.

5.5.2 Testing
A simulated SABER Level 1A file will be generated from LIMS and SABER engineering calibration data. The test file will contain each scan mode identified in the SABER Level 1A SDD, as well as the simulated IFC and space-look events described in CSCI's above. Specific tests for each requirement are described below:

Electronic Filter deconvolution: One adaptive scan event will have scans that consist of the Butterworth filter response to an input delta function. The module will successfully remove the electronics filter if a delta function is the output.

Co-Align Channels: Knife-edge data from engineering calibration will be used to validate the module's ability to co-align each detector's offset from the center of the focal plane.

Convert volts to radiance units: IFC data from engineering calibration will be used to validate the radiance output by the module based on the IFC and Jones Source Temperatures.

Estimate and remove any detectable spacecraft motion: Simulated PVAT data will be used which have attitudes perturbed which will cause known rotations to the simulated data. The module will be validated if it successfully removes these perturbations.

Geolocate each sample and grid to uniform angle spacing: Irregularly spaced data with known piece-wise linear values will be input to the gridding routines. The output data will have values that can be checked by verification.

Write out each event profile in NetCDF format: NetCDF reader software (i.e. IDL, Xmgr) will be used to validate the output NetCDF file.

6 Acronym List

APID	Application Identifier
APL	Applied Physics Laboratory
CCSDS	Consultative Committee for Space Data Systems
CSCI	Computer Software Configuration Item
CVT	Current Value Table
DLL	Dynamic Link Library
FTP	File Transfer Protocol
GATS	Gordley and Associates Technical Software
GSE	Ground Support Equipment
HALOE	HALogen Occultation Experiment
H, S & P	Health, Safety and Performance
H/W	Hardware
ICD	Interface Control Document
IDL	Interactive Development Language
JHAPL	Johns Hopkins Applied Physics Laboratory
LIMS	Limb Infrared Monitor of the Stratosphere
MDC	Mission Data Center
MOC	Mission Operation Center
NetCDF	Network Common Data Format
POC	Payload Operations Center
PVAT	Position, Velocity, Attitude and Time
RT	Real Time
SABER	Sounding of the Atmosphere using Broadband Emission Radiometry
SAGE	Stratospheric Aerosol and Gas Experiment
S/C	Spacecraft
SDD	Software Development Document
SDL	Space Dynamics Laboratory
S/W	Software
TCP/IP	Transmission Control Protocol over Internet Protocol
TIMED	Thermosphere, Ionosphere, Mesosphere, Energetics, Dynamics

7 Appendix A: SABER Level 1B File Format Description

PURPOSE
The purpose of this appendix is to define the content and format of the SABER Level 1B file. This file will be a product produced by the SABER Level 1B processing software. It will contain calibrated radiance profiles for each SABER channel and all information required to retrieve constituents in Level 2 processing. The Level 1B file will be in the NetCDF format.

BACKGROUND
The Level 1B file will be the output from Level 1B processing, which reads in a Level 1A file, removes instrument and spacecraft effects, co-aligns channels, converts to radiance units, geolocates each sample and grids to uniform angle spacing.

REQUIREMENTS

The Level 1B file will contain radiance profiles as a function of angle, height, latitude, longitude and time. Other derived requirements include:

- Event number for current day
- Spacecraft latitude, longitude and altitude
- Day/Night flag for tangent point
- Orbit Ascending/Descending flag
- Solar Zenith Angle at Tangent point
- NMC profile (P,Z,T) near the 60 km tangent point location
- Solar Ap, Kp, F10.7 and sunspot number

FORMAT
Following the Common Data form Language (CDL) definition of a NetCDF file, the Level 1B will contain:

- **Dimensions**

```
event = 2200;                                  // Scan event number for current day (~2200)
elevation = 800;                               // Elevation angle relative to horizon
                                               //  (22.7 Hz x 33 sec rounded up)
pressure_nmc = 64;                             // NMC pressure levels
```

- **Variables**

```
short event(event)                             // Event number for current day
long date(event)                               // Date [YYYYDDD]
double elevation(elevation)                    // Elevation Angle (SABER centric)[milliradians]
long time(event, elevation)                    // Msec since midnight
char mode(event)                               // 0=Down 1=Up
float sclatitude(event, elevation)             // Spacecraft latitude (degrees)
float sclongitude(event, elevation)            // Spacecraft longitude (degrees)
float scaltitude(event, elevation)             // Spacecraft altitude (km)
float latitude(event, elevation)               // Tangent point latitude
float longitude(event, elevation)              // Tangent Point longitude
char tpDN(event)                               // 0=Day 1=Night
char scAD(event)                               // 0=Ascending 1=Descinding
float tpSolarZen(event)                        // Tangent point solar zenith angle (degrees)
float tpSolarLT(event)                         // Tangent point local solar time (msec since midnight UT)
float channel_1(event, elevation)              // Channel 1 radiance (watts/cm2/sr)
float channel_2(event, elevation)              // Channel 2 radiance (watts/cm2/sr)
float channel_3(event, elevation)              // Channel 3 radiance (watts/cm2/sr)
float channel_4(event, elevation)              // Channel 4 radiance (watts/cm2/sr)
float channel_5(event, elevation)              // Channel 5 radiance (watts/cm2/sr)
float channel_6(event, elevation)              // Channel 6 radiance (watts/cm2/sr)
float channel_7(event, elevation)              // Channel 7 radiance (watts/cm2/sr)
float channel_8(event, elevation)              // Channel 8 radiance (watts/cm2/sr)
float channel_9(event, elevation)              // Channel 9 radiance (watts/cm2/sr)
float channel_10(event, elevation)             // Channel 10 radiance (watts/cm2/sr)
float pressure_nmc(event, pressure_nmc)        // NMC pressure at TP (mbar)
float temperature_nmc(event, pressure_nmc)     // NMC temperature at TP (K)
float altitude_nmc(event, pressure_nmc)                 // NMC altitude at TP (km)
short solKP(event)                             // Solar Kp index
short solAP(event)                             // Solar Ap index
float solf10p7Daily(event)                     // F10.7 flux (daily)
float solF10p781dAvg(event)                    // F10.7 flux (81-day average)
short solSpotNo(event)                         // Zurich Sunspot Number
```

- Data
  Event (1-2500)
  Elevation (TBD)

- Attributes

ESTIMATED FILE SIZE

1-D Event data:	**Bytes**
short event(event)	2
long date(event)	4
char mode(event)	1
char tpDN(event)	1
char scAD(event)	1
float tpSolarZen(event)	4
float tpSolarLT(event)	4
short solKP(event)	2
short solAP(event)	2
float solf10p7Daily(event)	4
float solF10p781dAvg(event)	4
short solSpotNo(event)	2
Total (x 2200 events)	**68200**

2-D Event X Elevation data:	**Bytes**
float sclatitude(event, elevation)	4
float sclongitude(event, elevation)	4
float scaltitude(event, elevation)	4
float latitude(event, elevation)	4
float longitude(event, elevation)	4
float channel_1(event, elevation)	4
float channel_2(event, elevation)	4
float channel_3(event, elevation)	4
float channel_4(event, elevation)	4
float channel_5(event, elevation)	4
float channel_6(event, elevation)	4
float channel_7(event, elevation)	4
float channel_8(event, elevation)	4
float channel_9(event, elevation)	4
float channel_10(event, elevation)	4
Total (x 2200 events x 800 elevation samples)	**105600000**

2-D Event X NMC data	**Bytes**
float pressure_nmc(event, pressure_nmc)	4
float temperature_nmc(event, pressure_nmc)	4

float altitude_nmc(event, pressure_nmc)	4
float geopotential height	4
Total (x 2200 events x 64 NMC pressure levels)	**2252800**

Total Estimated file size (Mbytes): 0.068200 + 105.6 + 2.2528 = 108 Mbytes