5. Ground Data System

5.1 JEM/SMILES products

5.2 Operational algorithm and performance

5.3 System design

5.4 Development schedule

5.5 Summary

JEM/SMILES data sets

Data Type	Description
RAW	Unprocessed mission data at binary packets
Level 0	Reconstructed, unprocessed mission data at binary packets
Level 1b	Calibrated instrument radiances and related data
Level 2	Derived geophysical variables at the same resolution and location as the Level 1 source data
Level 3	Variables mapped on uniform space-time grid scales, usually with some completeness and consistency

Processing

Products of level 2 data

■ **Standard Products :**

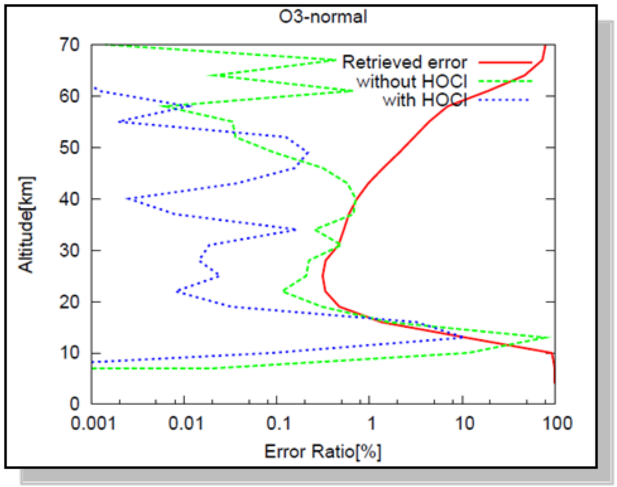
- Routine processing :O_3, HCl, ClO, CH_3CN, O_3 isotopes, HOCl, HNO_3
- Nonroutine Processing :HO_2, BrO

■ **Research Products :** (These products are outside of this system)

volcanic SO_2, H_2O_2, UTH, Cirrus Clouds

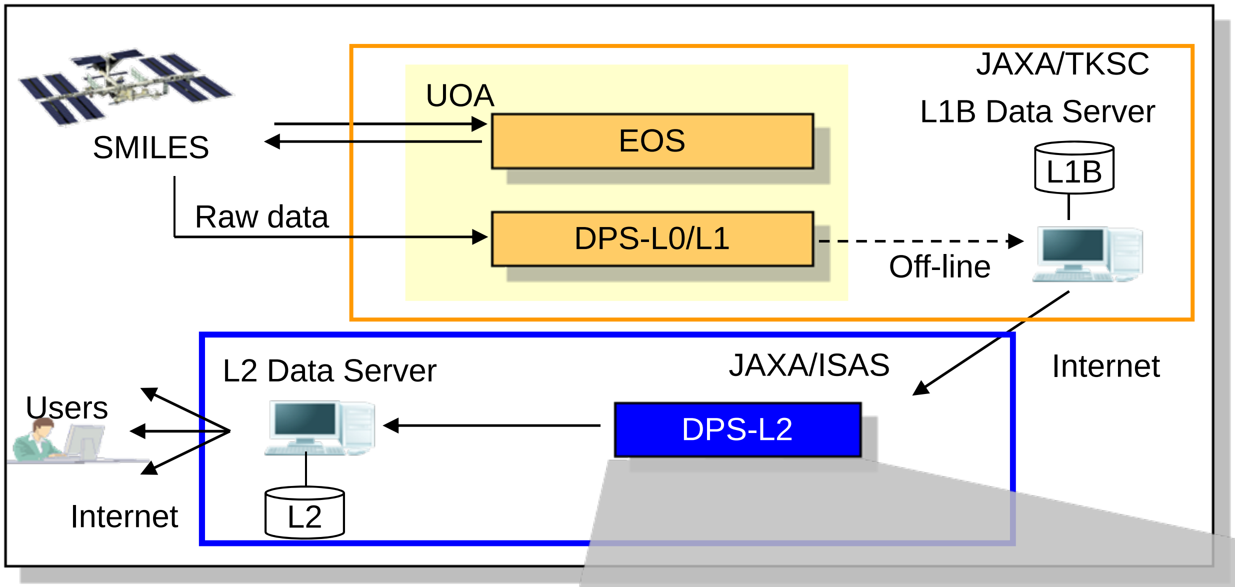
Type	Band A	Band B	Band C
Species retrieved from single-scan data	O_3 $H^{37}Cl$ ^{18}OOO HNO_3 CH_3CN HOCl $O^{17}OO$	O_3 $H^{35}Cl$ ^{18}OOO $O^{17}OO$ ClO	O_3 ^{18}OOO HNO_3
Species retrieved from multi-scan data (noisy products)	BrO	HO_2	BrO HO_2

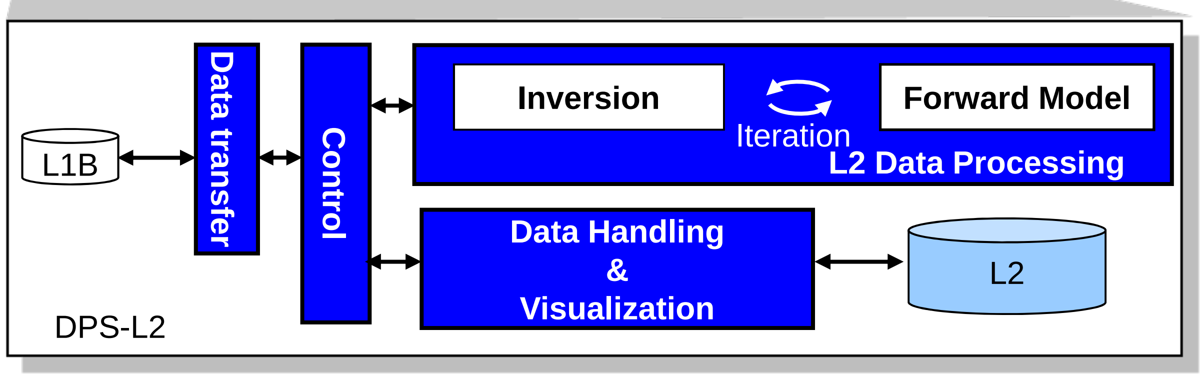
Effects from Other Molecules



JEM/SMILES data flow

- Downlinked raw data from the SMILES will be received by the DPS-L0/L1 at User Operation Area (UOA) on Tsukuba Space Center (TKSC).
- The DPS-L0/L1 processes the raw data consisting of house keeping (HK) data and mission data to brightness temperature (level 1B data) in near-real-time.
- The DPS-L2 produces the vertical profiles of target species called level 2 data in near real time and distributes the level 2 data to data users by a Web server.





UOA: User Operation Area,
EOS: Experiment Operations System,
DPS: Data Processing System

Level 2 data processing system

- **DPS-L2**

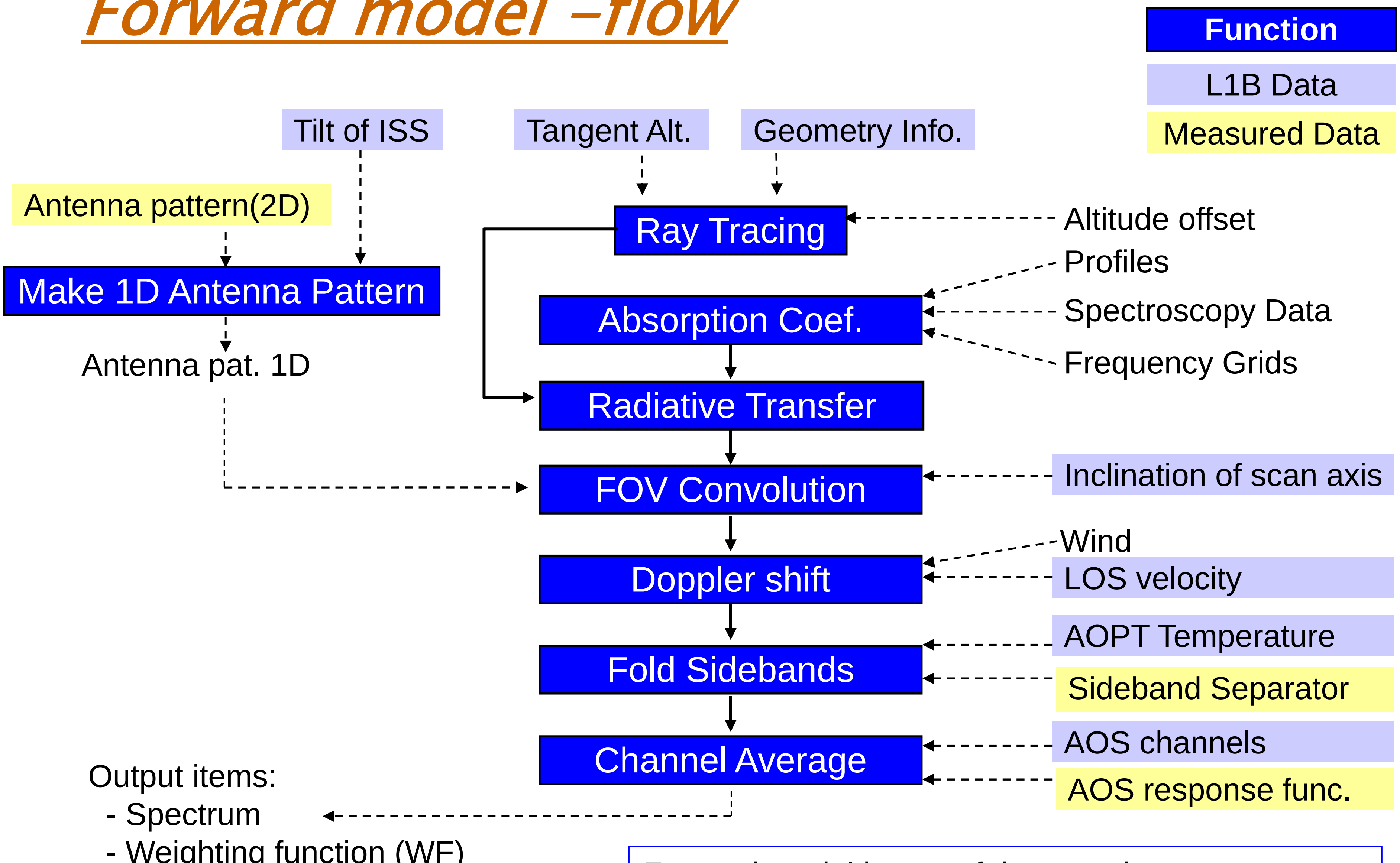
  The DPS-L2 produces distributions of the target species concentrations from calibrated spectra in near-real-time. The calculation process consists of a forward model process, which computes the simulated spectra, and an inverse model process, which deduces an atmospheric state.

- **Requirements for the DPS-L2**
  - Accuracy
    - Forward Model Parameter Error caused by the approximations

      < 0.01K（1% of the measurement error）
  - Processing speed
    - Near real time processing （1 scan: 53 s）
  - Portability
    - Work on a general linux system

System	Software
Operating System	Linux (kernel: up to 2.6)
Language	gcc (or Intel C++), Ruby, GNU Octave, java
Database	MySQL
Web Application	GFDnavi, ruby on rails

Forward model –flow



Forward model is one of the most important part to determine the accuracy and its computing cost

Doppler shift & Instrumental functions

Doppler shift

- Velocity of the ISS : 8 km/s
- Rotation of the earth : 460 m/s (on the equator)
- Wind : ≲100 m/s.

FOV Convolution

$$T_A(\nu, z_0) = \int_{z_{min}}^{z_{max}} P(z, z_0) \cdot T_p(\nu, z) dz,$$

Fold Sidebands (Single Sideband Separator)

$$T_{mix:i}(\nu, z_0) = \begin{bmatrix} K_{i,a}^{LSB}(\nu_{LO} - \nu_{if}, z_0) \\ K_{i,c}^{LSB}(\nu_{LO} - \nu_{if}) \end{bmatrix}^T \cdot \begin{bmatrix} T_A(\nu_{LO} - \nu_{if}, z_0) \\ T_c(\nu_{LO} - \nu_{if}) \end{bmatrix} + \begin{bmatrix} K_{i,a}^{USB}(\nu_{LO} + \nu_{if}) \\ K_{i,c}^{USB}(\nu_{LO} + \nu_{if}) \end{bmatrix}^T \cdot \begin{bmatrix} T_A(\nu_{LO} + \nu_{if}, z_0) \\ T_c(\nu_{LO} + \nu_{if}) \end{bmatrix}$$

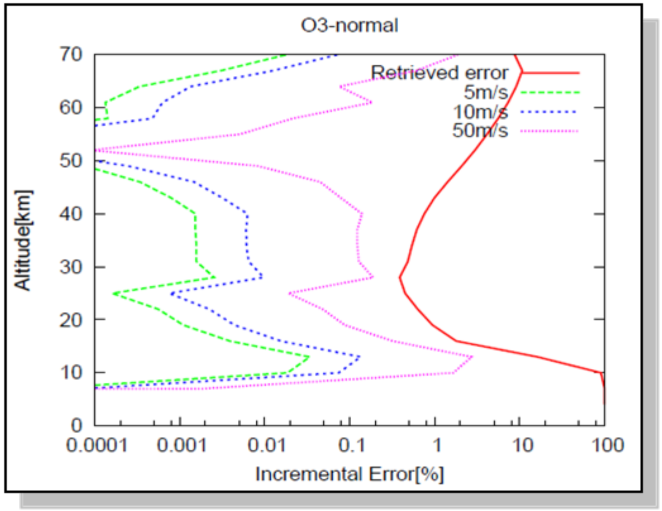
$$K_{i,j}^{LSB,USB}(\nu, T) = \frac{1 + \alpha(T)^2 + 2\alpha(T) cos\left(\frac{m\pi\nu}{\nu_0(T)}\right)}{4}.$$

Channel Average (AOS)

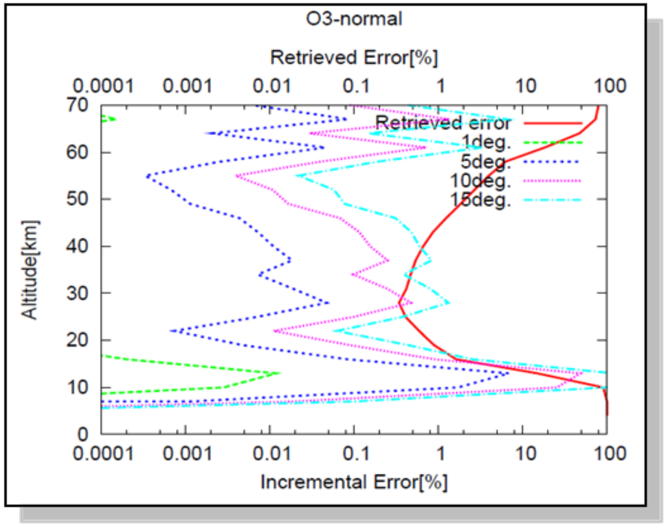
$$T_{AOS(l)}(\nu_j, z_0) = \frac{\int_{\nu_{min}}^{\nu_{max}} H_{AOS(l)}(\nu - \nu_j) \cdot T_{mix(k)}(\nu, z_0) d\nu}{\int_{\nu_{min}}^{\nu_{max}} H_{AOS(l)}(\nu - \nu_j) d\nu}.$$

$$H_{AOS(l)}(\nu - \nu_j) = \sum_{j-1}^{N_j} \frac{A_{i,j}}{w_{i,j}\sqrt{\pi/2}} \cdot exp\left(-2\frac{(\nu - \nu_j - xc_{i,j})^2}{w_{i,j}^2}\right).$$

Effects of wind



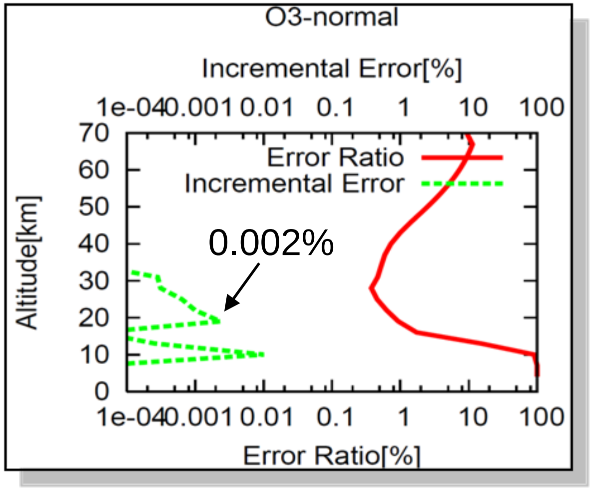
Effects of tilt of antenna scan axis



Fast algorithm

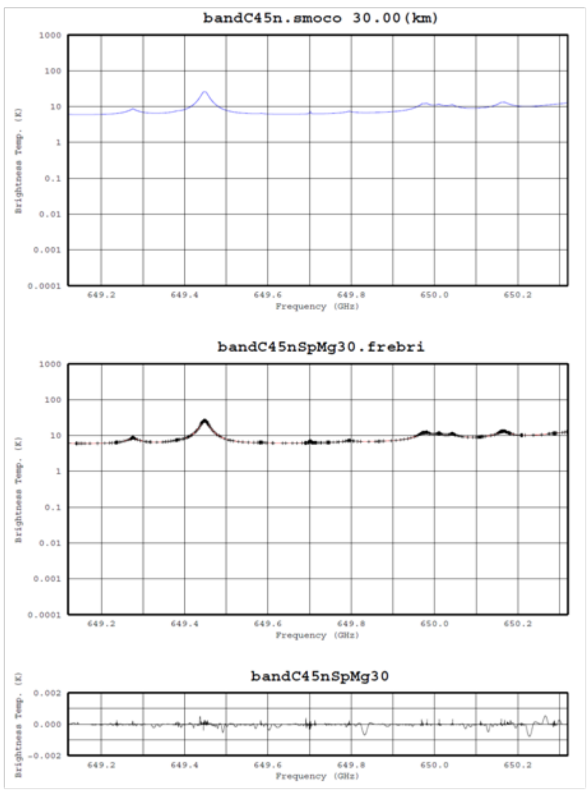
Selection of Absorption Lines

- The effect of the line selection. The incremental retrieved error by this line selection (Green line) is less than 0.002% in the stratosphere. It is enough small comparing the retrieved precision (Red line).

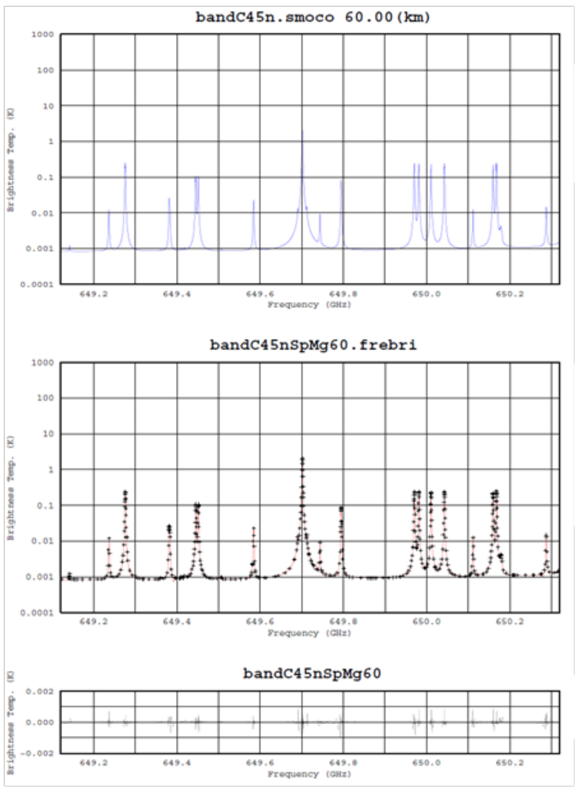


Selection of Frequency Grids

30km



60km



- 0.1MHz equally-spaced frequency grids (12000 points)
- Optimized frequency grids interpolated cubic spline (~900 points)
- Residual (1 – 2)

Inversion model

- Retrieval Algorithm: Optimal estimation method (OEM)
  - Levenberg-Marquardt Method

$$\mathbf{x}_{i+1} = \mathbf{x}_i + (\mathbf{S}_\mathbf{a}^{-1} + \mathbf{K}_i^\mathbf{T}\mathbf{S}_y^{-1}\mathbf{K}_i + \gamma\mathbf{S}_\mathbf{a})^{-1}\left\{\mathbf{K}_i^\mathbf{T}\mathbf{S}_y^{-1}\left[\mathbf{y} - \mathbf{F}(\mathbf{x}_i)\right] - \mathbf{S}_\mathbf{a}^{-1}\left[\mathbf{x}_i - \mathbf{x}_\mathbf{a}\right]\right\}$$

- Inversion model is implemented by GNU Octave
  - GNU Octave: High-level language, primarily intended for numerical computations. (Mostly compatible with Matlab)
  - Link ATLAS to speed up for matrices calculation.
  - Matrix size
    - n : about 200~300
      (number of elements for a state vector)
    - m: about 1500*70
      (number of elements for a measurement vector)
    - Use n-form of Levenberg-Marquardt Method for calculation speed.

Algorithm for noisy products

- To avoid the bias from a priori, we retrieve the multi-scan data simultaneously [Livesey,2004]. i.e. The observation data $\mathbf{y_i}$ (i=1~N), the weighting function $\mathbf{K_i}$, the reference spectra $\mathbf{f_i}$, and the covariance matrix of the measurements $\mathbf{S_{yi}}$ are represented by :

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} \mathbf{f}_{i1} \\ \mathbf{f}_2 \\ \vdots \\ \mathbf{f}_N \end{pmatrix}, \quad \mathbf{K} = \begin{pmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \\ \vdots \\ \mathbf{K}_N \end{pmatrix}, \quad \mathbf{S}_y = \begin{pmatrix} \mathbf{S}_{y1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{y1} & \mathbf{0} & \vdots \\ \vdots & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{S}_{y1} \end{pmatrix}$$

- To reduce the load of the system, we calculate below matrices and vectors, $\mathbf{K_i^T S_{yi} K_i}$ and $\mathbf{K_i^T S_{yi} (y_i - f_i)}$ foe each scan and save, because the size of these matrices and vectors are small.

$$\hat{\mathbf{x}} = \mathbf{a} + \left[\mathbf{S}_a^{-1} + \sum \mathbf{K}_i^T \mathbf{S}_{yi}^{-1} \mathbf{K}_i \right] \cdot \left[\sum \mathbf{K}_i^T \mathbf{S}_{yi}^{-1} [\mathbf{y}_i - \mathbf{f}_i(\mathbf{x}_0, \mathbf{b})] - \mathbf{K}_i^T \mathbf{S}_{yi}^{-1} \mathbf{K}_i [\mathbf{a} - \mathbf{x}_0]\right]$$

$$\sum \mathbf{K}_i^T \mathbf{S}_{yi}^{-1} \mathbf{K}_i = \left(\mathbf{K}_1^T \mathbf{S}_{y1}^{-1} \mathbf{K}_1 + \mathbf{K}_2^T \mathbf{S}_{y2}^{-1} \mathbf{K}_2 + \cdots + \mathbf{K}_N^T \mathbf{S}_{yN}^{-1} \mathbf{K}_N\right)$$

$$\sum \mathbf{K}_i^T \mathbf{S}_{yi}^{-1} [\mathbf{y}_i - \mathbf{f}_i(\mathbf{x}_0, \mathbf{b})] = \left(\mathbf{K}_1^T \mathbf{S}_{y1}^{-1} [\mathbf{y}_1 - \mathbf{f}_1(\mathbf{x}_0, \mathbf{b})] + \cdots + \mathbf{K}_N^T \mathbf{S}_{yN}^{-1} [\mathbf{y}_N - \mathbf{f}_N(\mathbf{x}_0, \mathbf{b})]\right)$$

Reference：N J. Livesey and W. Van Snyder, EOS MLS Retrieval Processes Algorithm Theoretical Basis, 2004

Performance

- Test of the processing time by our prototype program
  - Dual-Core Xeon 3.6 GHz (Use 1 core),
  - Intel C++ compiler
  - 64 bit Linux (Debian)

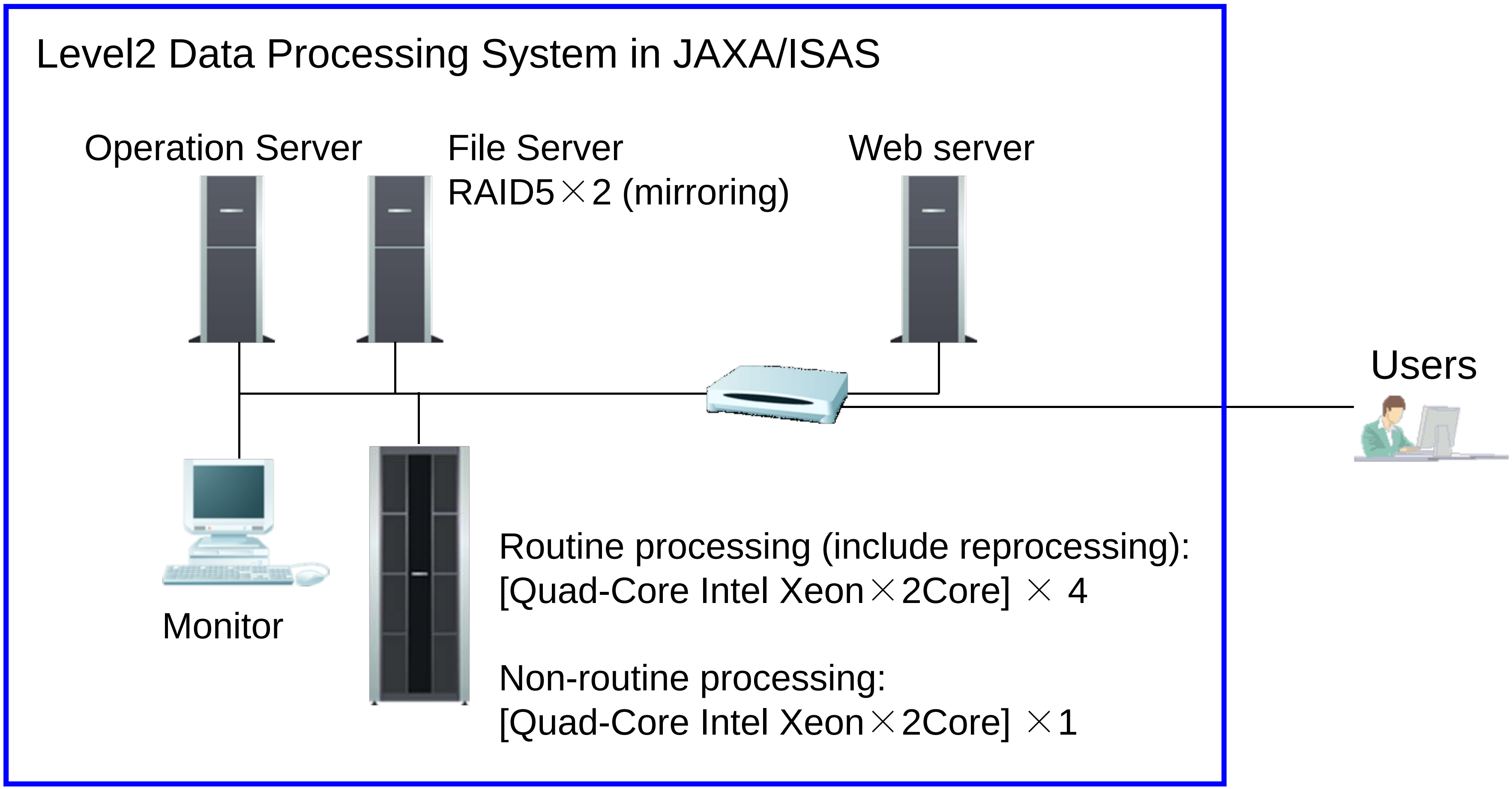
■ Processing time for one scan data of band A[*1]

• Forward Model (Spectrum and 11 WFs) [*2]	: 46 s
• Inversion	: 10 s
Total (no iteration)	: **56 s**

*1: Band A needs the longest processing time, but processing time of band B and band C are about 30% less than that of band A.

*2: Spectrum and 11 WFs are calculated simultaneously to avoid overlaps.

System configuration

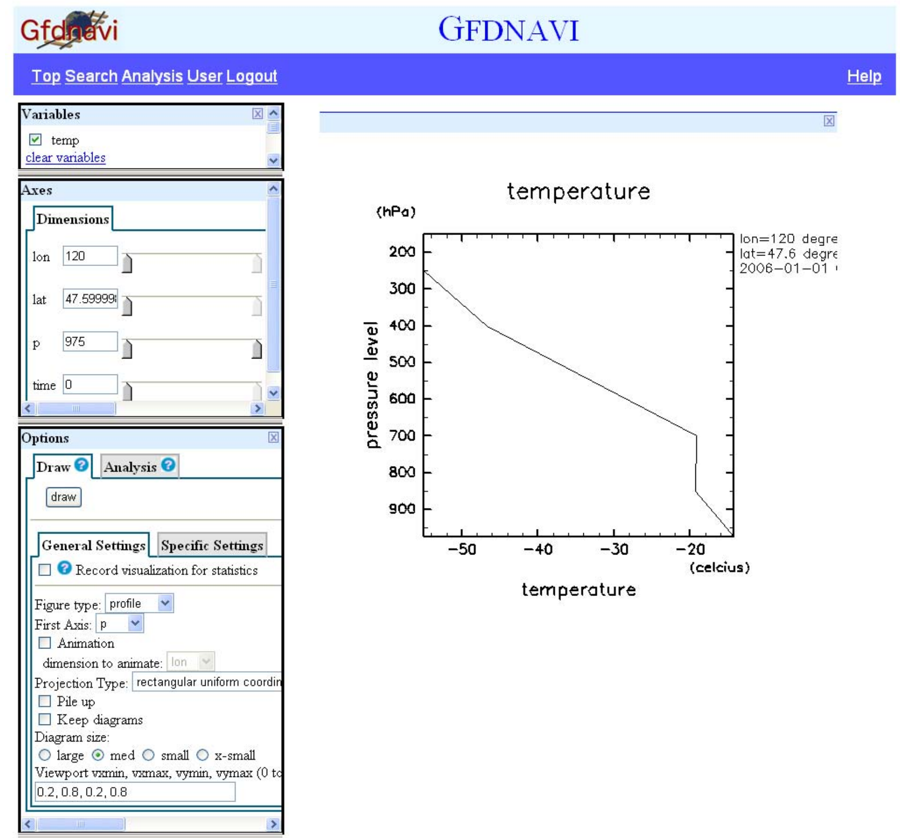


A Priori / first guess

Object		Reference data
Tracer constituents	O_3	LBLRTM[*1], Aura/MLS[*2], CCSR/NIES[*3]
	O_3 isotopes	*O_3 references with isotopic abundances for HITRAN*
	HCl	LBLRTM[*1], Aura/MLS[*2], CCSR/NIES[*3]
	ClO	LBLRTM[*1], Aura/MLS[*2], CCSR/NIES[*3]
	CH_3CN	UARS/MLS[*4] (, Aura/MLS[*2])
	HOCl	LBLRTM[*1], Aura/MLS[*2], CCSR/NIES[*3]
	HNO_3	LBLRTM[*1], Aura/MLS[*2], CCSR/NIES[*3]
	HO_2	Aura/MLS[*2], CCSR/NIES[*3]
	BrO	Aura/MLS[*2], CCSR/NIES[*3]
Dynamics	Temperature H_2O Wind Pressure	NASA/GMAO[*5]

*1: Reference profiles used in Line-by-Line Radiative Transfer Model
*2: Monthly climatology based on measurements by EOAS-Aura/MLS
*3: Monthly climatology based on the CCMVal-REF2 run for 2001-2010 by CCSR/NIES CCM
*4: CH_3CN reference profiles based on measurements by UARS/MLS
*5: Near-realtime analyses produced by NASA/GMAO’s GEOS-5 DAS

Level 2 data distribution system



- Level 2 Data Format
  - HDF-EOS (version 5)

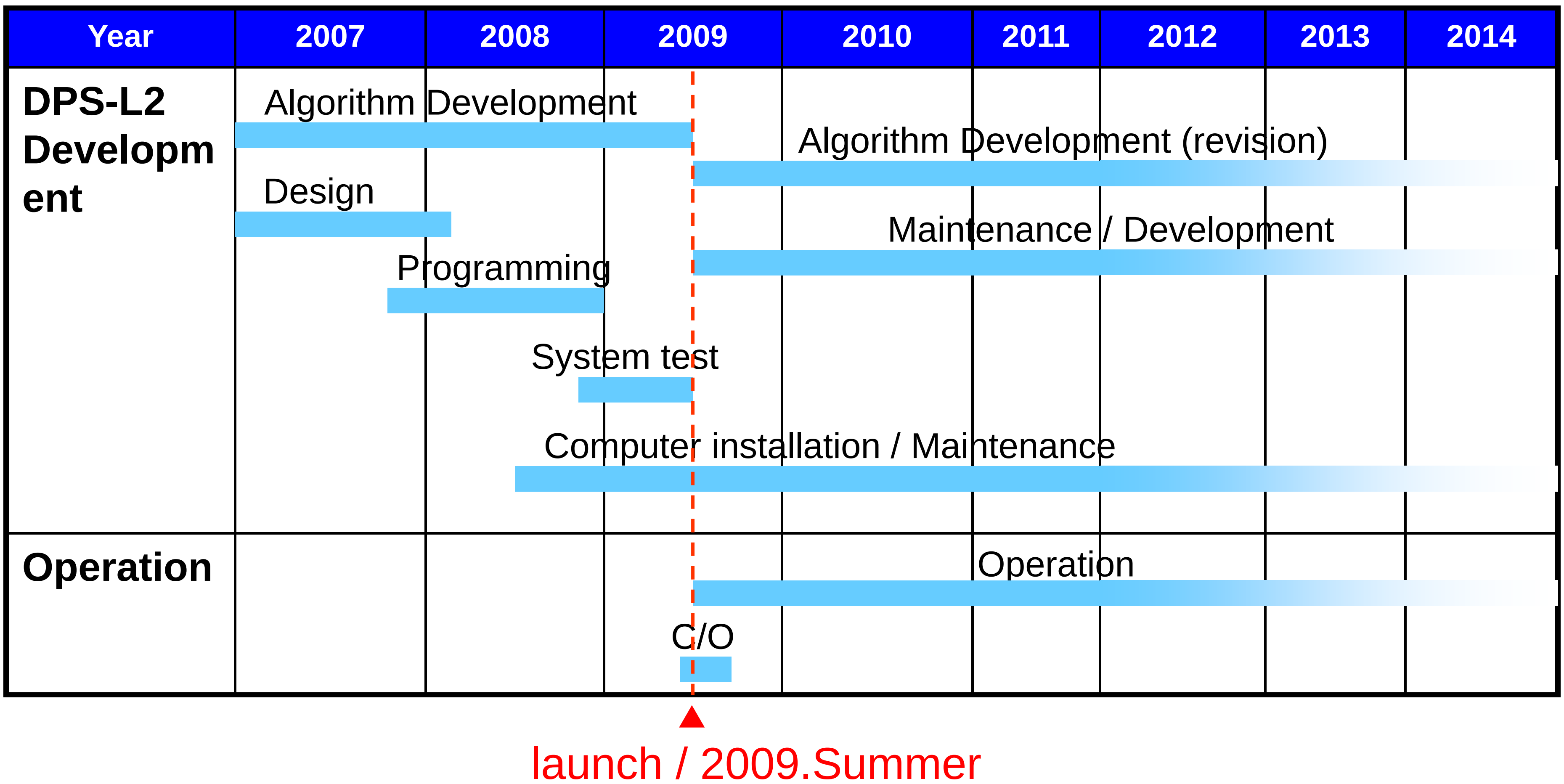
    → Compatible format with Aura/MLS
- File components（size: 11MB/day)
  - 1 file : 1day data for each molecule
  - Retrieved value / Precision / Quality / Geolocation Info. / Time etc.
- Level 2 data distribution system
  - Built on Gfdnavi (Geophysical fluid data navigator),　which is based on Ruby on Rails.

    Gfdnavi: http://www.gfd-dennou.org/arch/davis/gfdnavi/

    - a suite of software that facilitates databasing of geophysical fluid data, their analysis, and visualization

    Ruby on Rails: http://www.rubyonrails.org/
  - Implementation of this sub system has been finished.

Development Schedule



Summary

- The launch ready operational algorithm has been developed, and its performance meets the system requirements such as accuracy, processing speed and portability.
- The hardware system for the DPS-L2 is compact and consists of five linux servers with two CPU corresponding to Quad-Core Xeon 3.6GHz.
- The system development is due to be completed in 2009 early summer.