IRTS データアーカイブの再整備から学ぶ 科学データアーカイブ構築における留意事項

松崎恵一[1], 稲田久里子[1], 吉野彰[2], 山村一誠[1],
海老沢研[1], 篠原育[1] , 山本幸生[1]

[1] JAXA/ISAS, [2] NAOJ

1. はじめに

- ISASが科学衛星データベース「DARTS」が公開 (1997年) されてから、20年。あすか衛星以降、DARTS には、ほとんどの衛星のデータが収納されるに至った。

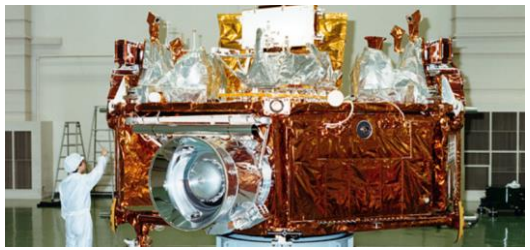
- 「宇宙科学観測から得られる成果を最大化するにはデータアーカイブの整備が重要である。」

  - と、論理的には思う。が、日本の宇宙科学プロジェクトにおいて、コンセンサスになっている兆候はない。
  - DARTS の維持は、ポスドク、派遣技術者、“兼業” のパーマネントスタッフで細々と続けられている。
  - 例外) かぐやプロジェクト [NASDA-ISAS の共同ミッションとして出発した経緯から、相当規模の資金、人的リソースでアーカイブ作成が行われた。]
  - 昨今の科学衛星プロジェクトでは、資金規模が限られる (小型科学衛星) ていたり、データアーカイブの経験がない ISAS 外にデータセンターを設ける方向で調整が進むなど、長期的に安定したデータの保全やサービスの提供に懸念が増えている。

1. はじめに

- 優れたアーカイブの一つとして、NASA JPL を中心に開発・維持が行われている、惑星科学分野のデータベース Planetary Data System (PDS) がある。これは、100年先もデータを使うことを目指しているが、PDS の準拠に必要な作業量は膨大であり、日本の宇宙科学プロジェクトにて対応するのは容易ではない。
- PDSの精神からミニマムな決まり事を抽出し、最大の効果がえられる留意事項は何か？

- チーム外の第3者にデータを提供する、長期間保存する、といった概念が知られていなかった時代に構築されたデータアーカイブ (IRTSデータベース) からデータを発掘した。
- この作業を通して抽出された問題点から、いま利用可能な技術の前提において、今後の長期間保存を目指したデータアーカイブの構築に役立つ、データ・プロセッシング上あるいはアーカイブ構築時のファイル管理のポイントが抽出したので、紹介する。



1.1. IRTS ミッション

(宇宙研の赤外線グループのホームページより)

あかりの先行ミッション

- IRTS (Infrared Telescope in Space) は、赤外線天体観測用としては、日本で初めて、地球を回る軌道に投入された観測器です。
- IRTSは、高感度の赤外線観測を可能にするために、超流動液体ヘリウムによって冷却された 赤外線望遠鏡です。IRTSの焦点面には、赤外線の全域をカバーする 4つの観測器 (NIRS, MIRS, FILM, FIRP) が搭載されていました。

小型科学衛星に近いプロジェクト規模！？

- IRTSは単独の衛星ではなく、多目的の宇宙実験用プラットフォームである SFU (Space Flyer Unit) に搭載されていました。IRTSは、宇宙開発事業団の新鋭機 HII ロケット によって、1995年3月18日に 打ち上げられました。
- その後3月30日にIRTSは 観測を開始し、超流動液体ヘリウムが消費された4月26日まで の間に、全天の7%にもわたる領域を、今までにない高感度で、サーベイ観測しました。この観測結果は、現在解析が進められており、太陽系内天体の研究から、銀河系の研究、そして 宇宙論の研究に至るまで、大変に有効な情報をもたらすと期待されています。
- SFU を搭載したIRTSは、スペースシャトルにより、1996年1月13日に 回収されました。
- IRTSの観測によって作成された、NIRS, MIRS点源天体カタログ、FILM, FIRP遠赤外線イメージマップが、2002年に宇宙研の 天文データアーカイブDARTS において公開されました。

今回再整備が行ったのでその事例報告 (過去から学ぼう)

1.2. IRTSのデータベース特徴

- 高次に処理されたデータベース
- 比較的に短期間の観測

NIRS, MIRS点源天体カタログ

MIRS, FILM, FIRP遠赤外線イメージマップ

- ４つの観測器毎にプロセッシング担当が存在し、全データを処理後、アーカイブ担当に処理結果を渡した
- 処理はパイプライン化されることなく、手作業が多かった

1.3. IRTS データベース開発・維持の歴史

開発フェーズ

データセンタ (PLAIN センター, C-SODA) の作業

- 1999/4-2001/3: JST 研究費により、IRTS に関わったポスドク A が名大で開発 (単独WSで開発)
  - ISAS 納入前に WS がクラッシュ！→なんとか “体裁を整え” 納入！
- 2001/4: ポスドク B により宇宙研へセットアップ DARTS として公開

維持フェーズ

- 2002年: ポスドク C によりデータ追加

- 2007/5: 職員 D (?) リプレースに向けた準備作業 (?)
- 2008/7 : メーカE による計算機システムリプレースにおけるデータ移行作業
- 2009/12 職員 D によりリンク切れの “修復”

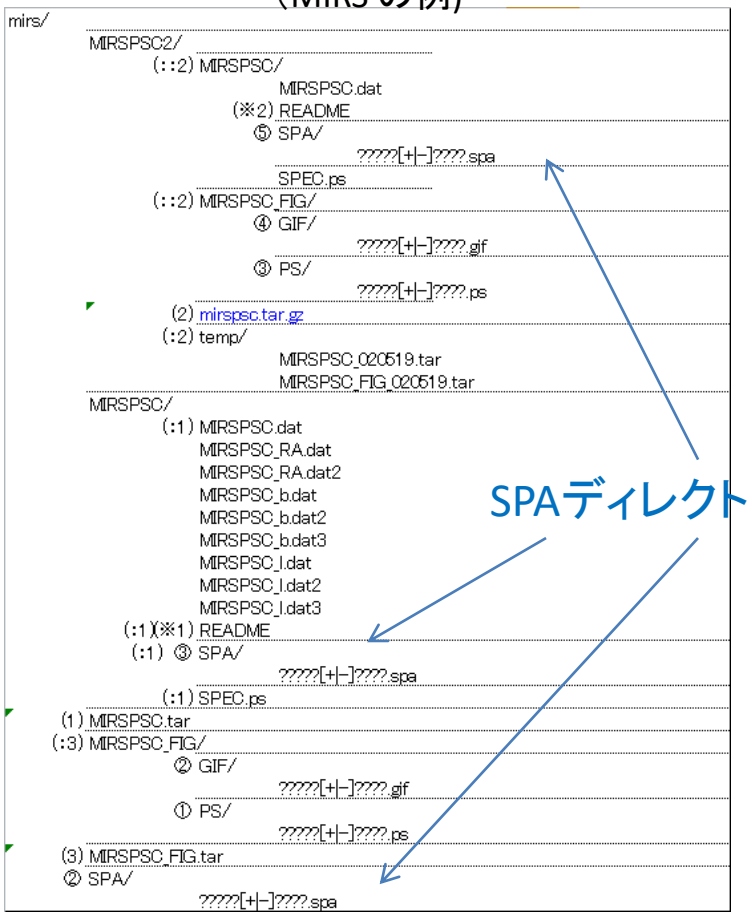
(2010/12 : ポスドク F による構成変更作業)
(2013/7 :メーカGによる計算機システムリプレースにおけるデータ移行作業)

- 2015/12 : 派遣 H (稲田) によるサルベージ作業 → 今回の報告!

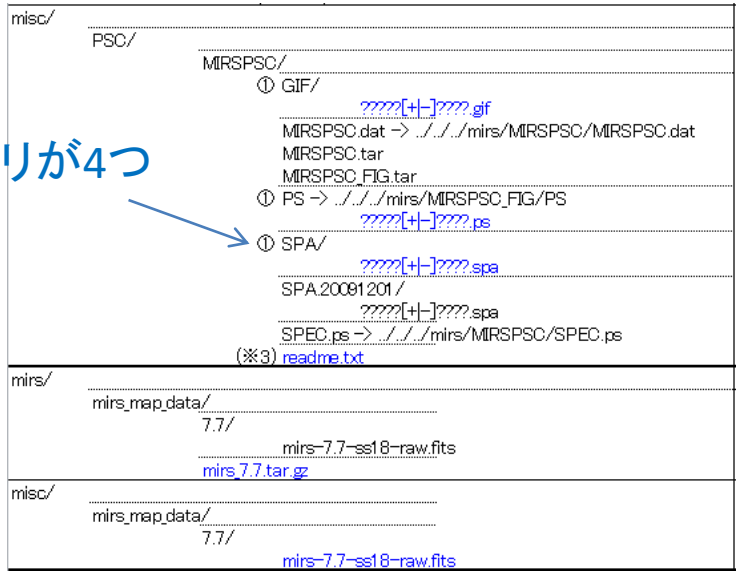
Typical な日本の宇宙科学データアーカイブ

2. 実施した分析 – 現状

DARTSで公開されているディレクトリの現状
（MIRS の例)



SPAディレクトリが4つ

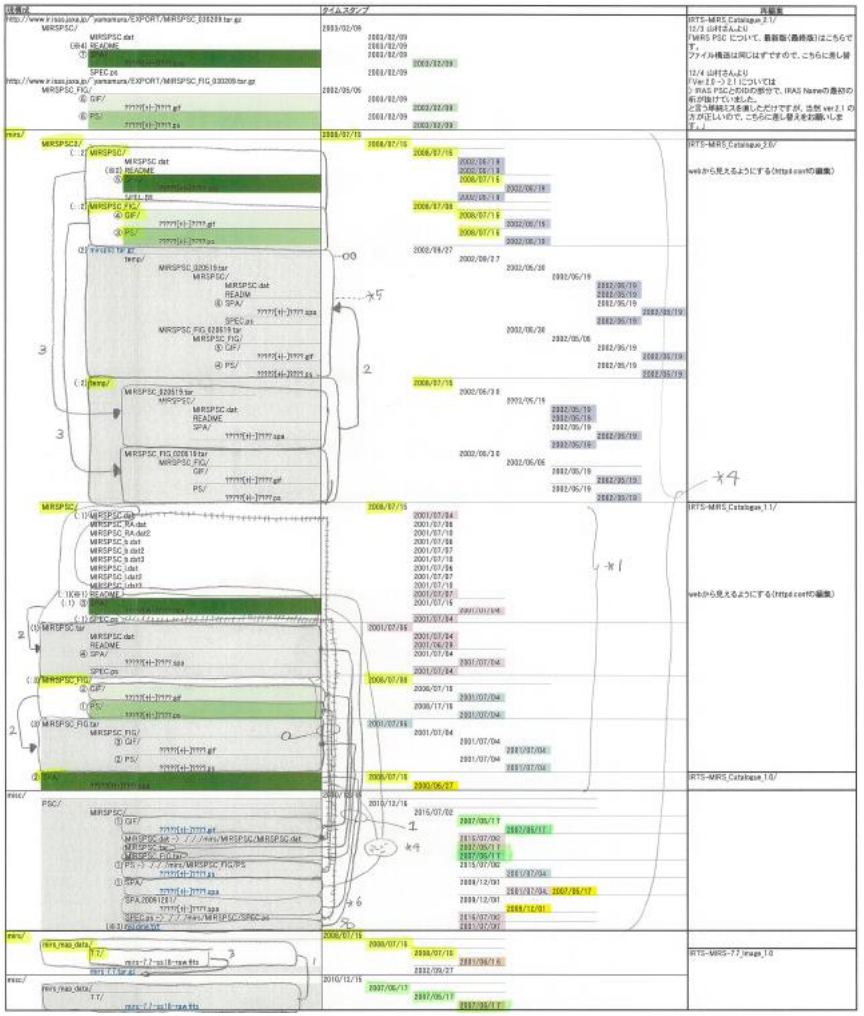


- 重複したファイルがみられる
- バージョンがぱっと見では分からない

2. 実施した分析 – 分析作業

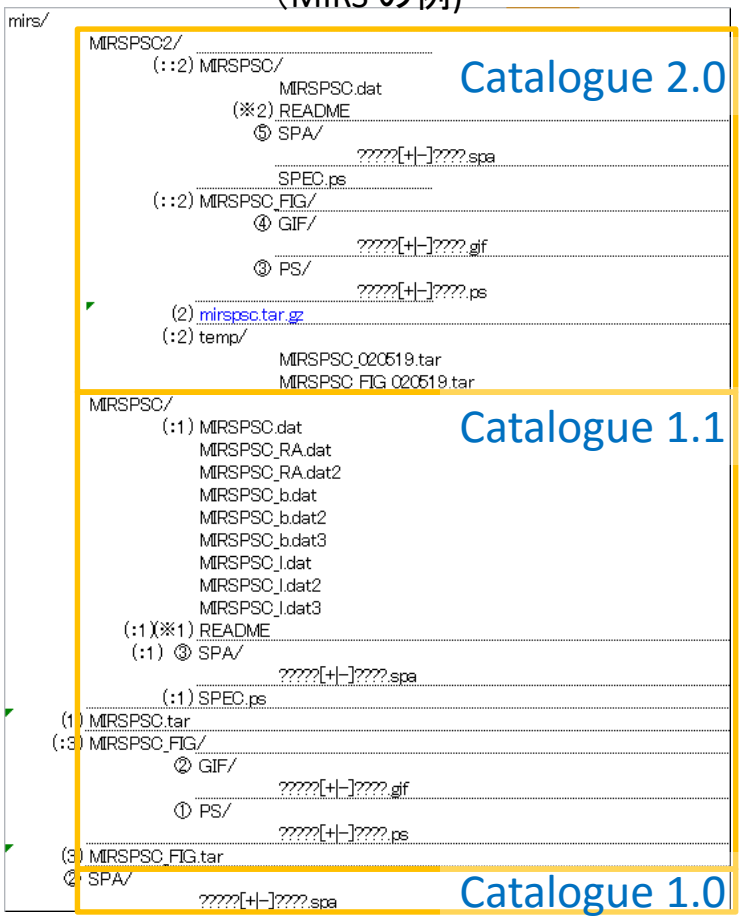
- tar(.gz) ファイルは何が収納されているか不明であったので、展開し確認した。
- リリースノートとファイルのタイムスタンプを比較し、バージョンを推定した。タイムスタンプが失われている場合など、ファイルの中身も比較した。
- 判断不能な点については、当時の関係者にヒアリングを行い、データセットと版を同定した。
- リーバース開発は、フォワード開発に比べてとても大変！
- (問い合わせた結果、DARTS に収蔵されているよりも新しい版があることもわかった)

（MIRS の例)



2. 実施した分析

DARTSで公開されているディレクトリの現状
（MIRS の例）

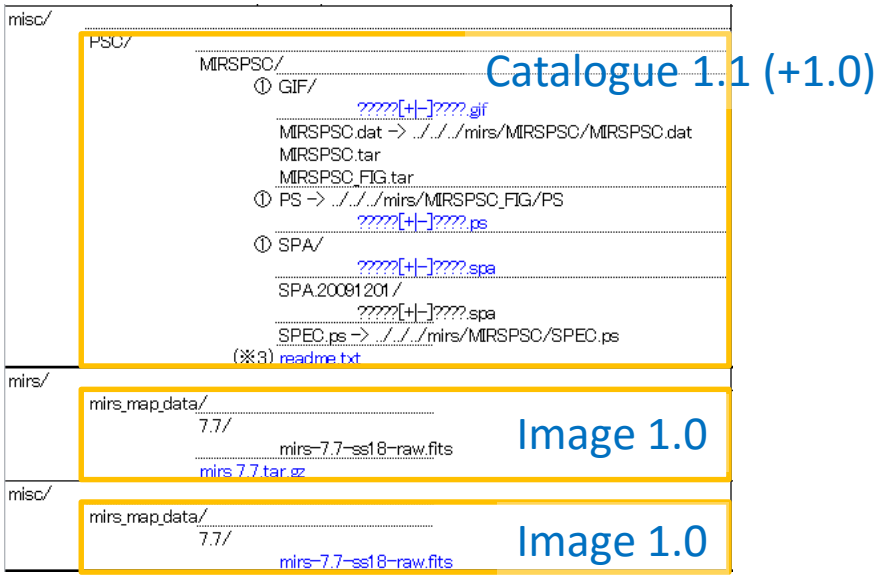


以下を同定

- Catalogue 3(+1) バージョン
- Image 1 バージョン

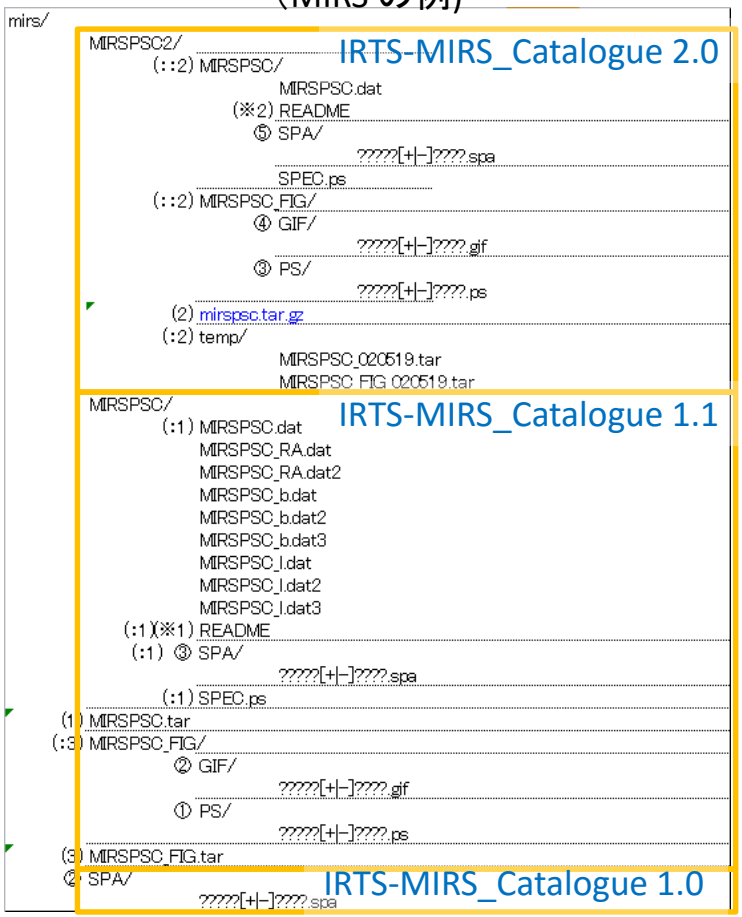
開発当初は以下の想定だと推定

- mirs はワーキングディレクトリ
- miscは公開用に再整備したディレクトリ



3. 開発フェーズの問題と対策(1)

DARTSで公開されているディレクトリの現状
（MIRS の例）



問題点

- リリースノート、ユーザが手にするアーカイブとしてのデータ構造を反映していない。
- 当初、データプロセスにおいてバージョンが管理されておらず、名前を見てもどのバージョンか分からないディレクトリや tar(.gz)ファイルが混在している。
- ディレクトリ名に new, old などの名前がはいっているが、後からみて版が分からない。
- ファイル名の変更などの微修正をアーカイブ担当が実施した可能性がある。

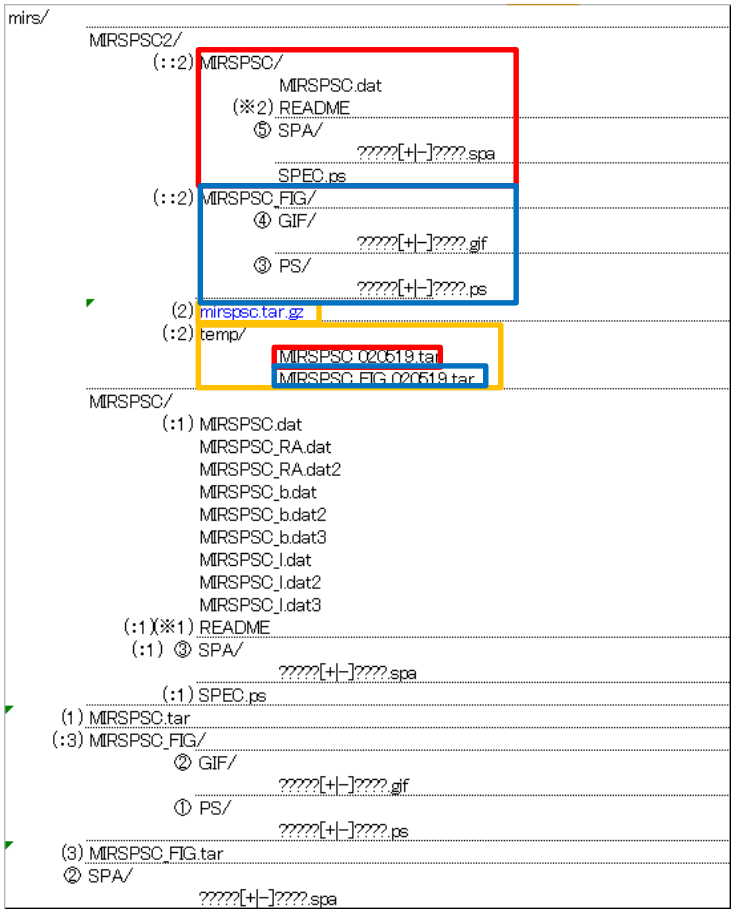
推奨事項

（ なにはともかくリリースノートを残すこと ）

データセットの版を管理する - プロジェクトを通じて、データプロダクトを管理する担当者を設ける、バージョン情報を集約する体制を整える

- データプロセス・アーカイブ担当が一体となり、ユーザ向けのリリースノートを用意すること。
- 一つのデータセット・版のデータとリリースノートなどの文書は一つのディレクトリに収納する。このディレクトリはトップディレクトリの直下に配置する。
- ディレクトリや tar(.gz)ファイルは、自己記述的な名称とする。具体的には「データセット名-バージョン」とすると良い。
  例: IRTS-MIRS_Catalogute_2.0

3. 開発フェーズの問題と対策(2)



- 配付用に tar(.gz) ファイルを用意する設計とした。

問題点

- 他方で、tar(.gz) を解凍して得られるものも公開しており、冗長である。tar.(gz) の範囲が自明でなく、何のデータが収納されているか分からない。
- tar(.gz) ファイルのディレクトリ構造が十分検討されていない。ディレクトリ名が temp など謎である。tar(.gz) ファイルが含まれている。

推奨事項

- 圧縮ファイルは、内容物をリリースノートに記述 and/or データセットのディレクトリ全体を固める and/or 作成しない

3. 開発フェーズの問題と対策(3)

当時の状況の推測

- 納期が来てしまったので整理は終わっていなかったが開発を終了した。
- 公開用ディレクトリにデータを集約する方向で開発を進めた。各種のデータは一旦Workingディレクトリに置いて作業をしていたが、最終的には公開用ディレクトリのみに整理することをあきらめ、Working ディレクトリも公開することとした。

問題点

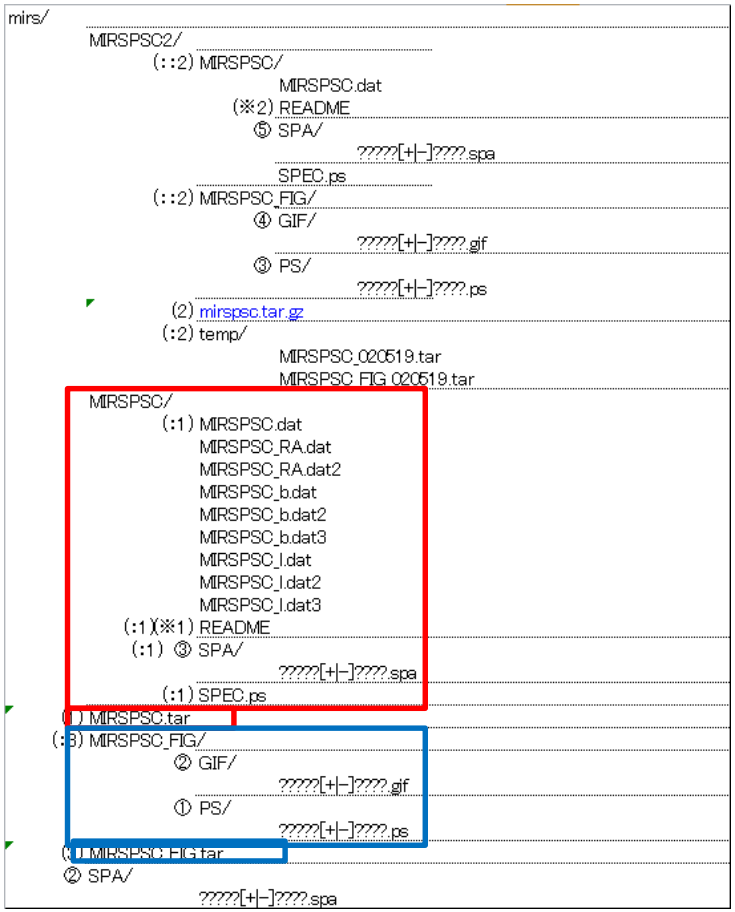
- その結果、どのように HTML ページを見せるか方針が不統一となってしまった。
  - HTML から Working ディレクトリを直接参照することとした。
  - 公開用ディレクトリに Working ディレクトリへのシンボリックリンクを配置した。
  - 公開用ディレクトリに Working ディレクトリのデータをコピーした。双方公開されることとなったので、データを冗長に持つこととなった。

推奨事項

- View (HTML) とデータは分離し、データは、データセットのディレクトリをそのまま参照する。

(やむを得ない場合、削除するよりは as is で放置する)

3. 開発フェーズの問題と対策(4)



問題点

- データプロセス担当から入手した時点 (?) のファイルと展開後ディレクトリ構造の双方が残っている。どちらが源泉なのか分からない。
- 作業の途上で作成したコピーが残ってしまい、冗長になった (?)

推奨事項

- 入手ファイルは 、入手ファイル専用のディレクトリに配置する。
- 中間ファイルは作業の完了までに削除する。
- 履歴管理ソフトで扱えるデータ量の範囲においては履歴を記録しながら開発を進めるのが良い。

4. 維持フェーズの問題と対策(1)

問題点？

- ディレクトリ・ファイルのタイムスタンプが更新された (システム構成変更作業中の2007/5に発生)
- ディレクトリのタイムスタンプが更新された(ファイルのタイムスタンプは維持された;リプレース準備中の2008/7に発生)

推奨事項

- (長期的な保存を考慮すると) ファイルシステムの属性には依存しない設計がよい
- が、ファイルシステムの属性も (重要な情報なので) 保存するように作業時に注意する
  - cp, scp の –p オプション
  - rsync の -a オプション
  - データ移行の際に使用するツールの仕様に注意する。

4. 維持フェーズの問題と対策(2)

問題点 (MIRS Catalogue において発生)

- Working ディレクトリには、2.0, 1.1, 1.0 のデータが存在していた。作り置きの HTML からは、基本的に 1.1 のデータへのリンクが張られていた。が、公開ディレクトリに置かれたデータの一部は1.0のままであり、リンク切れが発生していた。
- 2009/12/1 に、このリンク切れを修復する作業が行われた。この際に、オリジナルのディレクトリ名に日付を付与したバックアップディレクトリが作成された(なお、バックアップの際にタイムスタンプが失われた)。新しいディレクトリは、1.1 のデータ置かれた。が、その後、1.0 のデータも配置され、1.1と1.0で座標が一致していたものについては 1.0 のデータで上書きされた。
- データをまとめた tar.gz ファイルについては、2.0版への直リンクになっている。

→ 結果的に、HTMLからは 1.0, 1.1, 2.0 のデータへのリンクが混在している。

データセットの版が管理されていれば、このような作業は必要なかったし、ミスが発生する確率も下げられたものと思われる。が、ヒューマンエラーの発生は根本的には防げないものと考えられる。

推奨事項

- チェック・レビューを行う独立な担当者を設ける

5. 推奨事項サマリ

データセットの作成 … 衛星プロジェクトやデータ処理プロジェクトチームなどが実施

… プロジェクトを通じて、データプロダクトを管理する担当者を設ける、バージョン情報を集約する体制を整える

- データプロセス・アーカイブ担当が一体となり、ユーザ向けのリリースノートを用意すること。
- 一つのデータセット・版のデータとリリースノートなどの文書は一つのディレクトリに収納する。
- ディレクトリや tar(.gz) ファイルは、自己記述的な名称とする。具体的には「データセット名-バージョン」とすると良い。
- 圧縮ファイルは、内容物をリリースノートに記述 and/or データセットのディレクトリ全体を固める and/or 作成しない

データ提供サービスとの関係 … データセットの作成とは別のチームで良い / 今後のC-SODAは主にこちら

- データセットのディレクトリはトップディレクトリの直下に配置する。
- View (HTML) とデータは分離し、データは、データセットのディレクトリをそのまま参照するとよい。

作業上の注意

- チェック・レビューを行う独立な担当者を設ける
- 入手ファイルは、入手ファイル専用のディレクトリに配置する。中間ファイルは作業の完了までに削除する。履歴管理ソフトで扱えるデータ量の範囲においては履歴を記録しながら開発を進めるのが良い。
- ファイルシステムの属性には依存しない設計がよいが、ファイルシステムの属性も保存するように作業時に注意する

6. まとめ

- IRTS 衛星のデータアーカイブをサルベージした
- 今後のデータプロセス・アーカイブ作成に対し、実践的な推奨事項をまとめた