

SCALE-4.2 コードシステムの導入・整備

— Sun ワークステーションへのインストール —

1994年6月

動力炉・核燃料開発事業団
東 海 事 業 所

この資料は、動燃事業団社内における検討を目的とする社内資料です。については複製、転載、引用等を行わないよう、また第三者への開示又は内容漏洩がないよう管理して下さい。また今回の開示目的以外のことには使用しないよう注意して下さい。

本資料についての問合せは下記に願います。

〒319-11 茨城県那珂郡東海村大字村松4-33

動力炉・核燃料開発事業団

東海事業所 技術開発推進部・技術管理室

SCALE-4.2コードシステムの導入・整備 - Sun ワークステーションへのインストール -

野尻 一郎^{*1}、深作 泰宏^{*2}
清水 義雄^{*1}、成田 倭^{*1}

要　　旨

核燃料サイクル施設の安全解析のために米国オークリッジ国立研究所で開発された計算コードシステムSCALEの最新版であるSCALE-4.2についてSun SPARC-2 UNIXワークステーション（以下SunEWSという）への導入・整備を実施した。

SCALE-4.2は、IBM RS/6000 UNIXワークステーション（以下IBMEWSという）をベースに開発されているため、SunEWSとはオペレーティング・システム（OS）、Fortran及びCコンパイラのバージョン等のシステム環境が異なるため、SunEWSに対応したオリジナルのソースプログラムの修正やスクリプトファイル（PC上でのバッチファイル、メインフレームでのJCLに相当するファイルをいう）の修正を行った。

また、Sun EWSによりサンプル計算を行い、システムに添付されているIBMEWSによる計算結果と比較し、SunEWS上でSCALE-4.2が正常に動作することを確認した。

*1 安全管理部 安全技術課

*2 原子力システム（株）

目 次

1. はじめに	1
2. SCALE-4.2の概要	2
2.1 SCALE-4.2の構成	2
2.2 SCALE-4.2の改良	2
2.2.1 ファンクション・モジュールの改良	2
2.2.2 コントロール・モジュールの改良	4
2.2.3 データライブラリの改良	4
3. インストール手順	6
3.1 SCALE-4.2の公開内容	6
3.2 インストールの基本的な手順	6
4. SCALE-4.2のデータ転送	8
4.1 DCからSunハードディスクへのデータ転送	8
4.2 tarファイルの展開	8
4.3 ディレクトリの概要	8
5. SCALE-4.2ロードモジュールの作成	10
5.1 COMMENTプログラム	10
5.1.1 COMMENTのロードモジュール作成	10
5.1.2 COMMENTプログラムの実行	10
5.1.3 対象ファイル	11
5.2 SCALE-4.2のロードモジュール作成	11
5.2.1 ソースプログラムの修正	11
5.3 ロードモジュール作成用スクリプトファイルの修正	13
5.4 ロードモジュールの作成	14
6. AMPXライブラリの作成	15
6.1 AMPXライブラリ作成プログラムのロードモジュール作成	15
6.1.1 clgampxスクリプトファイルの修正	15
6.1.2 ロードモジュールの作成	15

6.2 KENO-V.a albedo・weightライブラリの作成	15
6.2.1 malwgts.jcl・scale4スクリプトファイルの修正	16
6.2.2 malwgts.jclスクリプトファイルの実行	16
6.3 Standard Composition ライブラリの作成	16
6.3.1 stdcomp.jclスクリプトの修正	16
6.3.2 Standard Composition ライブラリの作成	17
6.4 ORIGEN-S ライブラリの作成	17
6.4.1 データライブラリの概要	17
6.4.2 mkinlib1・mkinlib2スクリプトファイルの修正	18
6.4.3 ORIGEN-S ライブラリの作成	18
 7. SCALE-4.2の実行	19
 8. サンプル計算	20
8.1 計算概要	20
8.2 計算体系	20
8.2.1 計算モデル	20
8.2.2 使用混合物	20
8.3 計算結果の相違	21
8.3.1 中性子増倍率	21
8.3.2 中性子スペクトル	22
8.4 検討	23
 9. あとがき	24
 10. 参考文献	25

添付資料

1. convert スクリプトファイル	28
2. h7lib.c ソースプログラム	29
3. jstime.c ソースプログラム	31
4. input.f ソースプログラム	33

5. getmdl.f ソースプログラム	35
6. rcover.f ソースプログラム	37
7. clgscale スクリプトファイル	39
8. fort スクリプトファイル	45
9. fortg スクリプトファイル	45
10. fortno スクリプトファイル	46
11. lked スクリプトファイル	47
12. load スクリプトファイル	47
13. clgampx スクリプトファイル	48
14. malwgts.jcl スクリプトファイル	50
15. stdcomp.jcl スクリプトファイル	51
16. mkbinlib スクリプトファイル	52
17. mkbinlib1 スクリプトファイル	53
18. mkbinlib2 スクリプトファイル	59
19. scale4 スクリプトファイル	62

1. は じ め に

SCALEコードシステム¹⁾(以下SCALEという)は、米国オークリッジ国立研究所(ORNL)で開発された核燃料サイクル施設の臨界安全、遮蔽解析及び伝熱解析の安全解析を行うためのコードシステムであり、Radiation Shielding Information Center(RSIC)を通じて公開されている。

安全技術課では、核燃料サイクル施設の安全解析コードの整備の一環として、SCALEの導入・整備を行い、MOX燃料加工施設を対象としたSCALE-4の臨界安全解析への適用性検討²⁾³⁾⁴⁾や中性子遮蔽パネル開発における遮蔽効果解析⁵⁾等といった種々の安全解析に適用している。

今回、SCALEの最新版であるSCALE-4.2⁶⁾が1993年12月に公開された⁷⁾ことから、安全技術課のSun EWS(Sun OS 4.1.2及びFortran1.4)の環境下で、このシステムを用いた安全解析が行えるよう導入・整備を行った。もともとIBM EWS環境下で動作していたものを、Sun EWSの環境で動作させるということで、オリジナルのソースプログラムへの修正やロードモジュール作成用のスクリプトファイルの修正等、そのインストール作業において様々な修正を行った作業内容について報告する。

第2章において、SCALE-4.2のシステム構成や改良点等の概要について述べる。

第3章から第7章では、今回実施したSCALE-4.2のインストール作業結果について述べる。第3章では、SCALE-4.2の公開データの内容やSunEWSへのインストール手順及び基本的な操作方法について述べ、第4章においては、SCALE-4.2をSunEWSのOS環境下で実行させるためのソースプログラムを変換することのできるCOMMENTプログラムについて、ロードモジュールの作成、実行操作方法を述べる。第5章では、オリジナルのソースプログラムをコンパイル・リンクし、その時発生したエラーを基にソースプログラムへ修正を加えた内容とスクリプトファイルに加えた修正内容及びSCALE-4.2のロードモジュール作成について述べる。第6章においては、SCALE-4.2で使用するデータライブラリの作成について、そのライブラリを作成するためのソースプログラムのロードモジュール作成やスクリプトファイルの実行方法について述べる。第7章では、SCALE-4.2の実行方法について述べる。

第8章では、第3章から第7章までのインストール手順に従って作成したロードモジュールの動作確認とその検討を行うためサンプル計算を行い、システムに添付されているIBM EWS上で計算された臨界安全解析モジュールのサンプル出力とSunEWSの出力結果との比較を行った内容について述べる。

2. SCALE-4.2の概要

2.1 SCALE-4.2の構成

SCALEは、使用済核燃料輸送容器等の許認可業務にかかる安全審査のための解析計算で用いる標準的手法を確立する目的で、米国原子力規制委員会の要請により、ORNLで開発されたコードシステムであり、臨界安全解析、遮蔽解析及び伝熱解析を相互に関連を持って、体系的に計算できるようになっている。

SCALEは初版の公開以降、種々の機能拡張を行っており1990年には、SCALE-4.0(SCALE-4)が公開されている。このコードシステムは、IBMのメインフレームをベースに開発され、これまでに既存のFortranソースプログラムを標準Fortran-77コンパイラで対応できるよう広範囲にわたってソースプログラムの書き換えが行われている。これは、追加される機能がFortran-77コンパイラに対応したプログラムで提供されるようになったためである。

SCALE-4の開発は主に、IBM/MVSとCray/CTSSの両方のマシンによって進められ、SCALE-4.1はIBM/MVS・Cray/UNICOSマシン上でテストが行われ公開されている。今回公開されたSCALE-4.2は、IBM/MVS3090とIBMEWSで開発され、他のメインフレーム及びワークステーション環境に適合できるようCOMMENTと呼ばれるプログラムが提供されている。

図1に、SCALE-4.2のシステム構成を表す。

2.2 SCALE-4.2の改良⁹⁾

今回新しく改良された点を以下に述べる。

2.2.1 ファンクション・モジュールの改良

(1) ORIGEN-S・COUPLE

ORIGEN-Sは、核種の崩壊・消滅解析コードORIGEN⁹⁾の拡張版であり、SCALE-4.2では多数の核種断面積が含まれている結合バイナリ形式のライブラリを読み込むプログラムの改良が行われている。また、バイナリ形式ライブラリの編集オプションが組み込まれ、さらに崩壊データ・核分裂生成物収率ライブラリ及び消滅ライブラリが含まれている結合バイナリ形式ライブラリの更新・拡張が行われている。

COUPLEは、ORIGEN-Sで使用される核種の燃焼又は崩壊サイクルに伴った断面積ライブラリを作成する計算コードであり、ORIGEN-Sのプログラムの改良、ライブラリの更新等が行われたことから、互換性を持たせるためにプログラムの改良が行われている。

(2) HEATING 7.2

HEATING7.2 (Heat Engineering and Transfer In Nine Geometries) は、1次元、2次元又は3次元の円筒あるいは球形状における多種の熱解析を行うための解析コードであり、従来用いられていたHEATING6.1をUNIX及びIBM/MVS両マシンのOSで動作するよう改進したものである。

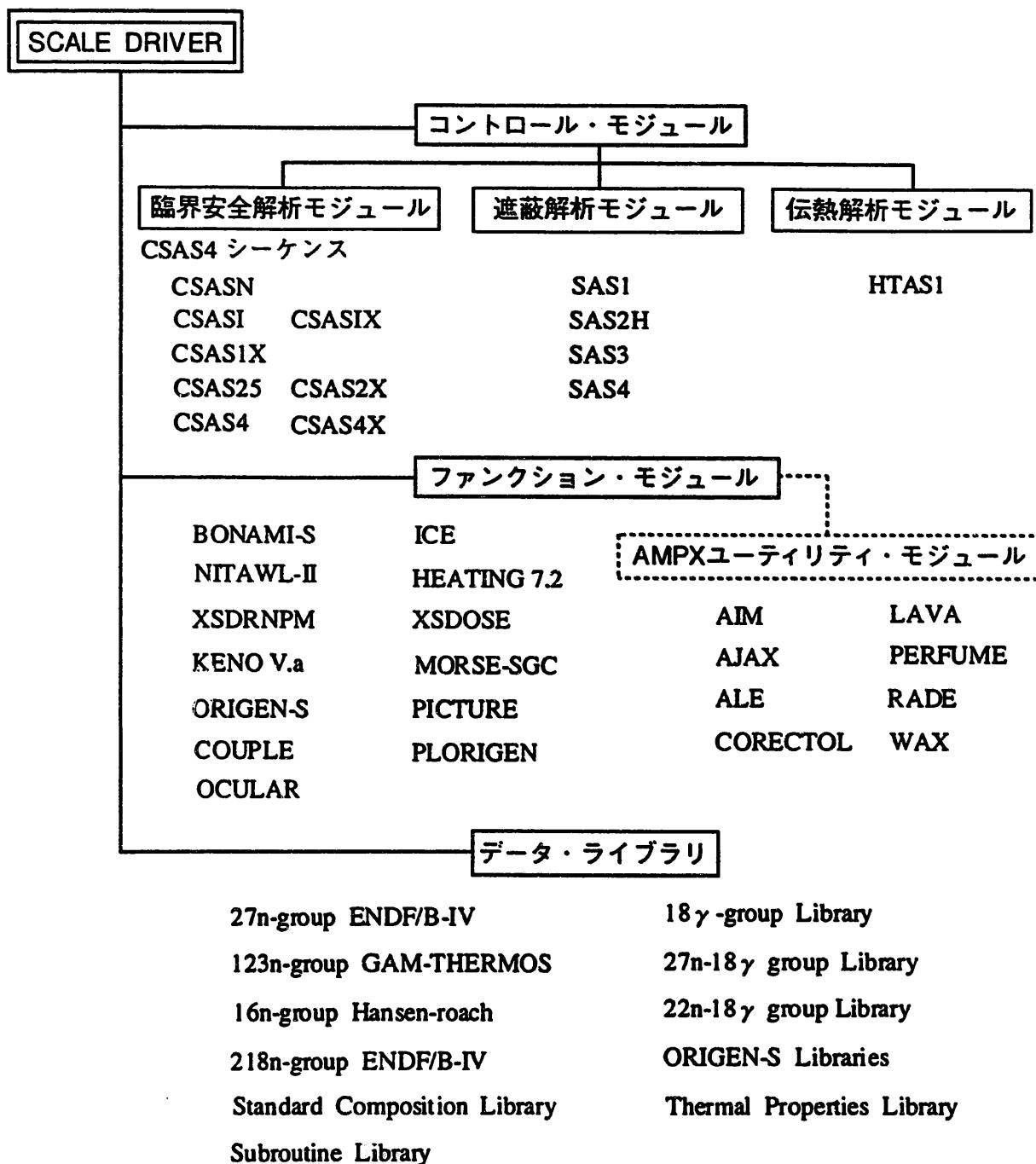


図1. SCALE-4.2コードシステムの構成

2.2.2 コントロール・モジュールの改良

(1) SAS4

SAS4では、点検出器評価計算の機能が改良され、収束性及び計算効率の向上のために点検出器で使用するバイアスシーケンスの自動化が行われている。

また、半径方向及び軸方向の線量計算においては、線源プロフィールを入力する機能が追加されており、この機能における線量計算の精度が向上されている。

(2) SAS2H

SAS2Hでは、ORIGEN-Sの崩壊データ及び核分裂生成物ライブラリの更新・拡張に対応するためプログラムの改良が行われている。また、SAS2Hの計算で生成されるバイナリ形式の消滅ライブラリが、単一のデータセットへ出力できるよう改良されている。この改良により、データセットの数・格納スペースの効率化が図られている。

2.2.3 データライブラリの改良

(1) ORIGEN-S データライブラリ

核種の崩壊・消滅解析に用いられる半減期、分岐崩壊分割、崩壊データライブラリの改良が行われている。評価対象核種数が、1609から1697核種になり、そのうちの1128核種についてはENDF/B-VIデータをベースに改訂が行われ、他の140核種についてはENSDFデータにより改訂が行われている。核分裂生成物生成データについては、これまでLWRライブラリのみENDF/B-Vデータが使用されていたが、SCALE-4.2では、従来ENDF/B-IVデータをベースに作成されていたそれ以外の炉型の核分裂生成物生成データについてもENDF/B-Vデータにより改良されている。

(2) Standard Composition ライブラリ

このライブラリは、SCALEを使用する上で混合物を簡単に定義することができるよう作成されているものであり、臨界安全及び遮蔽解析で一般的に使用される混合物と同位体元素から構成されている。

評価対象核種の中でENDF/B-Vデータにより共鳴データフラグの改訂が行われている。

(3) 27BURNLIB ライブラリの改良

27BURNLIBは、評価済核データライブラリENDF/B-IVを基に作成された中性子27群の断面積データライブラリである。従来の27BURNLIBのGd-155、Gd-157及びSm-147核種においてマイナーエラーが発見されたため、その修正が行われている。

(4) MIPLIBの改良

MIPLIB (Material Information Processor Library) は、計算に対応した断面積ライブラリを作成する機能である。Bondarenko方程式¹⁰⁾及びNordheim積分法¹¹⁾によって共鳴計算を行い、断面積データの補正が行われている。

従来のMIPLIBでは、小さな円筒系セル（0.3cmより小さい）の計算において、固有値が非安全側の結果を与える傾向にあったため、ダンコフ係数を算出するプログラムの改良が行われている。さらに、多領域平板系におけるダンコフ係数の算出においてもプログラムの改良が行われている。

3. インストール手順

3.1 SCALE-4.2の公開内容

SCALE-4.2はRSICから公開されており、データカートリッジ（DC）及びフロッピーディスク（FD）によって配布されている。DC及びFDには、以下に示すデータが格納されている。

(1) DC

収納形式：c00545irisc01.tar.Z

tarファイルの圧縮形式と呼ばれるファイル形式で格納されている。これを展開することによって以下のファイルが作成される。

- a). ソースプログラム
- b). データライブラリ
- c). 各種スクリプトファイル
- d). サンプル入力データ
- e). サンプル出力データ

(2) FD

FDには、インストールを行う上で必要な操作方法や注意事項が書かれたREADMEファイル並びに対話形式のSCALE入力データ作成プログラムOFFSCALE及びORIGEN-S入力データ作成プログラムORIGNATEが格納されている。なお、OFFSCALE及びORIGNATEは、IBM互換機PC上で動作するプログラムである。

3.2 インストールの基本的な手順

SCALE-4.2をSunEWSへインストールするための手順や操作方法について述べる。インストールからロードモジュール作成・実行までの作業手順を図2に示す。

インストールの基本的な手順は、インストール対象機種のハードディスク容量を確認し、DCからハードディスクへデータの転送を行う。

ロードモジュール作成では、SCALE-4.2の開発ベースがIBMEWSであることから、Sun EWSのOSやコンパイラ等が異なるため、Sun EWSに対応した自動変換プログラムCOMMENTによるプログラム変換及びプログラムの修正を必要としている。この作業の詳細については、第5章及び第6章で述べる。

データライブラリは、カードイメージ形式で格納されている。プログラム上では、バイナリ形式で扱われるため、データ形式の変換を必要とする。AMPX¹²⁾ライブラリ作成においては、システムで用意されている変換プログラムのロードモジュール作成とデータライブラリのデータ形式変換作業を行う。詳細は、第7章で述べる。

以上の手順によって、SCALE-4.2の一連の作業が終了し、実行環境が整う。

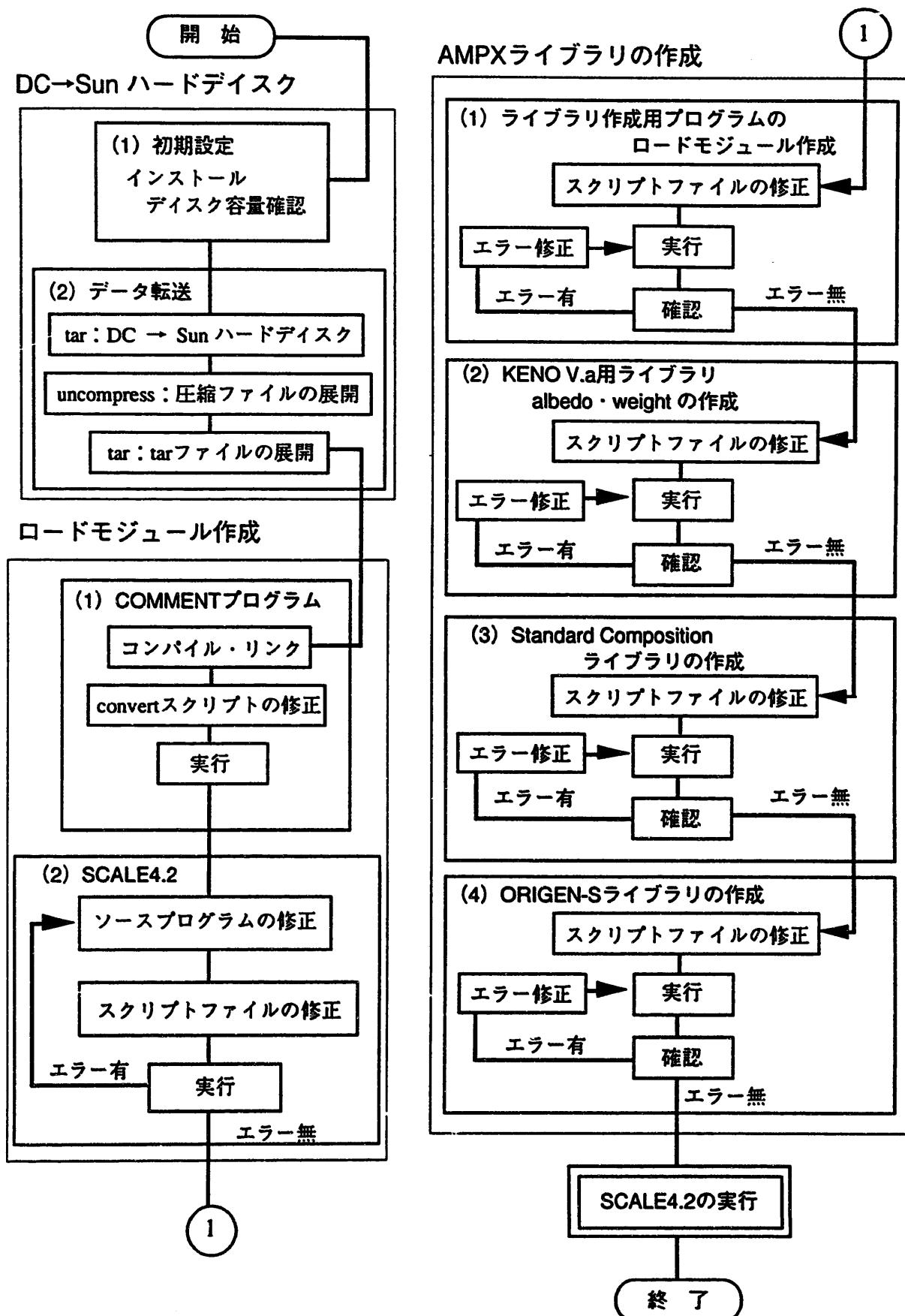


図2. SCALE-4.2のインストールから実行までの概要

4. SCALE-4.2のデータ転送

SCALE-4.2をSun EWSの/localディレクトリ配下へインストールするものとして以下にその整備内容について述べる。

4.1 DCからSunハードディスクへのデータ転送

DCからのデータ転送は、カレントディレクトリへ移動した後、tarコマンドによってデータの転送を行う。

```
% cd /local
% tar xvf /dev/rst0
```

4.2 tarファイルの展開

DCに収納されているc00545irisc01.tar.Zファイルは、tarファイルの圧縮形式で格納されているため、Sunハードディスクへデータ転送を行った後に、uncompress、tarコマンドによってファイルの展開を必要とする。

```
% uncompress c00545irisc01.tar.Z
% tar xvf c00545irisc01.tar
```

展開されたtarファイルは、以下のようなディレクトリ構成になる。

```
/local/scale/cmds
      /datalib
      /smplprbs
      /src
      /output
```

4.3 ディレクトリの概要

上記の展開された個々のディレクトリについて概要を述べる。

- cmds ----- ロードモジュール作成スクリプトファイル、核データライブラリ作成スクリプトファイル等のすべてのスクリプトファイルが格納されている。
- datalib ----- SCALE-4.2で使用される核データライブラリがカードイメージ形式で格納されている。さらに、カードイメージ形式データからバイナリ形式に変換されたデータファイルについても、このディレクトリ配下に格納される。
- smplprbs ----- 臨界安全、遮蔽解析及び伝熱解析を行うサンプル問題が格納されている。また、KENO-V.aやMORSE-SGC/S等の解析コー

ド单独で実行できるサンプル問題も用意されている。

- **src** ----- ソースプログラムが格納されている。
- **output** ----- IBMEWS上で実行したときの出力ファイルが格納されている。

5. SCALE-4.2ロードモジュールの作成

5.1 COMMENTプログラム

COMMENTプログラムは、インストールする計算機の機種に応じてプログラムを有効にするために用意されているプログラムであり、IBM（メインフレーム）、AIX（IBM EWS）、UNICOS（Cray機種）、ULTRIX（DEC EWS）、UNIX（AIX, ULTRIX、UNICOS選択時に適応）にそれぞれ対応している。

SunEWSへインストールする場合には、ULTRIXキーワードを用いる。

COMMENTプログラムは、/local/scale/src/commentディレクトリ配下のcomment.fプログラムをコンパイルすることによってロードモジュールが作成される。

5.1.1 COMMENTのロードモジュール作成

COMMENTのロードモジュールの作成では、/local/scale配下のexeディレクトリに'comment'として作成される。scaleディレクトリ配下にexeディレクトリが存在しない場合は、mkdirコマンドを用いてディレクトリを作成する必要がある。

操作方法は、以下のとおりである。

```
% mkdir /local/scale/exe
% cd /local/scale/exe
% f77 -o comment ../src/comment/comment.f
```

5.1.2 COMMENTプログラムの実行

(1) 実行スクリプトファイルの修正

COMMENTプログラムの実行においては、/local/scale/cmds配下のconvertスクリプトファイルを使用するが、実行前にconvertスクリプトファイル（添付資料-1 参照）を修正する必要がある。

この修正は、if文において\$TMPDIRで指定されたテンポラリファイルを検索し、存在しないとき、\$TMPDIRで指定するディレクトリにファイルを作成する実行文があるが、\$TMPDIRで指定するディレクトリ名を、if文の前のsetコマンドで設定する文を挿入するものである。ここで指定するテンポラリファイルは、ユーザが任意に設定することができる。

今回の作業において挿入したコマンド文を以下に示す。

```
set TMPDIR=/local/scale/tmp
```

さらに、COMMENTプログラムの存在するディレクトリの指定が異なるため、CPコマンドによりCOMMENTプログラムのロードモジュールをテンポラリファイルにコピーす

るコマンド文の修正を行う。

```
cp /local/scale/exe/comment $TMPDIR
```

(2) COMMENTプログラムの実行

COMMENTプログラムは、修正したconvertスクリプトファイルにSunEWSに対応したキーワードULTRIXを入力し、入力ファイル名及び出力ファイル名を与えることで自動的に実行される。

```
% cd /local/scale/cmds
% convert ultrix inputfile outfile
```

5.1.3 対象ファイル

COMMENTプログラムによって修正されるファイルを以下に示す。

```
/local/scale/src/driver/cntrlr.f
          /scale.f
/local/scale/src/sublib/opnfil.f
/local/scale/src/unixlib/eyenit.f
          /getfile.f
          /getmc.f
          /rndin.f
          /rndout.f
```

5.2 SCALE-4.2のロードモジュール作成

SCALE-4.2のロードモジュールの作成にあたっては、第5.1項で述べたソースプログラムの修正に加え、マシンの違いによって生ずる構文エラーに対応して、ソースプログラムの修正を直接行う必要がある。

これについては、RSICからの情報はなく、インストール作業の過程において発見したものである。この作業について行ったソースプログラム及びロードモジュール作成用スクリプトファイルの修正内容を以下に示す。

5.2.1 ソースプログラムの修正

SunEWSのシステム環境に合致しない構文が含まれるプログラムを以下に示す。

```
scale/src/unilib/h7lib.c
          jstime.c
/ocular/input.f
/driver/getmdl.f
```

```
/marslib/rcover.f
```

(1) h7lib.c (添付資料-2参照)

エラー内容は、h7lib.cで定義しているサブルーチンプログラムgetmem、chgmemの型宣言がSunEWSのコンパイラに合わないため、' syntax error' として処理される。

修正内容を以下に示す。

```
void getmem(number, size, array, addr, offset)
    int *number, *size, *array, *addr, *offset
```

```
void chgmem(number, size, array, addr, offset)
    int *number, *size, *array, *addr, *offset
```

h7lib.cで定義しているサブルーチンプログラムsecondの変数cpuにおいて、UNIXで用意されているinclude文がIBMとSunでは異なることから、オリジナルで記述されている変数CLK_TCKが未定義シンボルとして処理される。SunEWSでは、この変数が_SC_CLK_TCKと設定されており、以下のように修正する必要がある。

```
cpu=(double)now.tms_utime/_SC_CLK_TCK;
```

また、この変数の修正により、変数が格納されているディレクトリ名を追加する必要があり、以下のように、プログラムの先頭部分へ挿入する。

以下に挿入文を示す。

```
#include <sys/unistd.h>
```

(2) jstime.c (添付資料-3参照)

jstime.cプログラムのエラーは、変数time_leftの型宣言文がSunEWSの環境に合致しないため生じているもので、プログラム中で指定されていない変数として処理されている。

型宣言の記述方法を以下のように修正することで、正常に処理される。

```
extern void JSTIME(time_left)
    long *time_left
```

(3) input.f (添付資料-4参照)

input.fプログラムのエラーは、170から201行目のwrite文に対応したformat文の記述が複雑であるため、SunEWSでは対応できないために生ずるものである。このエラー部分は、出力リストにヘッダー印刷を行うもので、計算上は支障がないことから、プログラ

ム内のエラー部分にあたる箇所をコメント文として置き換える。これに伴い、170行にある行番号12を202行目の1カラム目に追加しなければならない。

(4) getmdl.f (添付資料-5参照)

getmdl.fプログラムでは、変数putenvを外部関数として記述している。このため、リンク処理のときにputenvシンボルが定義されていないものとして処理される。

このエラーを解消するため、getmdl.fの宣言文にputenvがC言語プログラムであるように模擬処理¹³⁾を行う必要がある。

追加宣言文を以下に示す。

```
external putenv    !$pragma c(putenv)
```

(5) rcover.f (添付資料-6参照)

rcover.fプログラムでは、34から42行目において' nrun' サブルーチンを呼び出しているが、' nrun' サブルーチンが存在しないため、SunEWSでは、"未定義シンボル"として処理され、リンク時にエラーとなる。

このプログラムでは、問題となる行に行番号160が記述されているが、この行番号の呼び出しプログラムが存在しないことから、プログラム上問題ないため、34から42行目のプログラムをコメント文とする。

5.3 ロードモジュール作成用スクリプトファイルの修正

ロードモジュール作成用スクリプトファイルには、clgscaleが用意されている。このスクリプトファイルには、コンパイル専用スクリプトファイル fort (添付資料-8 参照)、fortg (添付資料-9 参照)、fortno (添付資料-10 参照)、アーカイブ作成スクリプトファイル lked (添付資料-11 参照)、リンク専用スクリプトファイル load (添付資料-12 参照) がある。これらのスクリプトファイルの中で修正を必要とするファイルは、clgscale及びlkedであり、修正内容は以下のとおりである。

(1) lkedスクリプトファイルの修正

```
/usr/5bin/ar rvs ../lib/lib$dir.a $argv[*]
```

arは、ライブラリ・アーカイブ作成コマンドであり、修正、変更等が生じたときアーカイブ・シンボルテーブルを強制的に再作成してくれるオプションが用意されている。

Sun EWSでは、通常のSystemコマンドと機能の拡張されているSystemVコマンドがあり、arコマンドは両方に存在する。アーカイブ・シンボルテーブルを強制的に再作成し

てくれるオプション' s' であるが、このオプションを有効にするため、System Vコマンドを使用する必要があり、コマンドの前にSystemVコマンドが格納されているディレクトリ名の追加を行わなければならない。

(2) clgscaleスクリプトファイルの修正

```
set SCALE=/local/scale
```

clgscaleについては、\$SCALEで指定するディレクトリ名をシステムがインストールされたディレクトリ名に修正しなければならない。

また、55～61行目の記述内容は、第5.1項で述べたCOMMENTプログラムのロードモジュール作成内容であり、ここでは不要であるため削除する必要がある。

```
cc -c -O -DUNDERSCORE $SRCDIR/unixlib/*.c
```

さらに、68行目のCプログラムのコンパイル文を実行すると、C言語の場合は、自動的にシンボル名に' _' が付加されるため、Fortranプログラムとのリンク処理において異常終了する。

このエラーは、C言語コンパイルコマンドccの-Dオプションを用いて' _' を削除する設定を加えることで、C言語プログラムとFortranプログラムのリンク処理が正常に行えるようスクリプトファイルを修正しなければならない。

5.4 ロードモジュールの作成

ロードモジュールの作成は、修正を加えたclgscaleスクリプトファイルを以下のように実行することで作成される。

```
% clgscale
```

なお、clgscale スクリプトファイル実行時のエラーは、すべてスクリーンに表示されるため、コマンド・プロンプトの画面で実行した場合、処理結果が流れてしまいエラー確認ができなくなる。このため、OPENWINDOW環境下のコマンドツールを用いて実行することを推奨する。

6. AMPXライブラリの作成

SCALE-4.2では、KENO-V.aのライブラリであるalbedo, weightライブラリ、Standard Compositionライブラリ及びORIGEN-Sデータライブラリをカードイメージ形式からバイナリ形式へ変換する必要がある。

SCALE-4.2では、この変換プログラムが用意されており、また、これらのプログラムのロードモジュール作成が行えるようclgampxと呼ばれるスクリプトファイルが用意されている。

第6章では、AMPXライブラリ作成プログラムのロードモジュール作成を行うclgampxスクリプトファイル（添付資料-13 参照）、KENO-V.a用ライブラリ作成スクリプトファイルmalwgts.jcl（添付資料-14 参照）及びORIGEN-Sの光子、崩壊ライブラリ等を作成するmkbinlib（添付資料-16 参照）、mkbinlib1（添付資料-17 参照）及びmkbinlib2（添付資料-18 参照）スクリプトファイルについて、作業を行うにあたり必要な修正内容とロードモジュール実行手順について述べる。

6.1 AMPXライブラリ作成プログラムのロードモジュール作成

6.1.1 clgampxスクリプトファイルの修正

dgampxスクリプトファイルについては、SCALE-4.2をインストールしたディレクトリを絶対パスで指定するよう修正する必要がある。

```
set SCALE=/local/scale
```

6.1.2 ロードモジュールの作成

```
% clgampx
```

AMPXライブラリ作成プログラムのロードモジュールは、修正を加えたclgampxスクリプトファイルを実行することで作成される。

6.2 KENO-V.a albedo・weightライブラリの作成

第6.1項に示した手順によって作成されたロードモジュールを使用して、KENO-V.a用のalbedo・weightライブラリのバイナリ形式データへの変換を行う。

この変換は、malwgts.jclスクリプトファイルを使用して行うが、その前に、このスクリプトファイルの修正を行う必要がある。また、このスクリプトファイルによって呼び出されるscale4スクリプトファイル（添付資料-19 参照）についても修正の内容を述べる。

6.2.1 malwgts.jcl・scale4スクリプトファイルの修正

malwgts.jcl及びscale4スクリプトファイルは、SCALE-4.2をインストールしたディレクトリを絶対パスで指定するよう修正する必要がある。

```
set SCALE=/local/scale
```

さらに、if文において\$TMPDIRで指定されたテンポラリファイルを検索し、存在しないときに\$TMPDIRで指定するディレクトリにファイルを作成する実行文があるが、\$TMPDIRで指定されるディレクトリ名を、if文の前のTMPDIRに設定するコマンドを挿入する。ここで指定するテンポラリファイルは、ユーザが任意に設定することができる。

今回の作業において挿入したコマンド文を以下に示す。

```
set TMPDIR=/local/scale/tmp
```

6.2.2 malwgts.jclスクリプトファイルの実行

```
% malwgts.jcl
```

KENO-V.aのalbedo及びweightライブラリは、修正を加えたmalwgts.jclスクリプトファイルを実行することで自動的に作成される。

この実行後、/local/scale/datalib/kenova/binrylib配下にalbedo・weightライブラリが格納される。実行結果の出力ファイルは、カレントディレクトリにmalwgts.outファイルとして出力される。

6.3 Standard Compositionライブラリの作成

Standard Compositionライブラリの作成にあたっては、カードイメージ形式からバイナリ形式のライブラリへ変換を行うロードモジュールを実行するstdcomp.jclスクリプトファイル（添付資料-15 参照）が用意されている。このスクリプトファイルに修正を加え、実行することでStandard Compositionライブラリが構築される。

6.3.1 stdcomp.jclスクリプトの修正

stdcomp.jclスクリプトファイルは、SCALE-4.2をインストールしたディレクトリを絶対パスで指定するよう修正する必要がある。

```
set SCALE=/local/scale
```

さらに、if文において\$TMPDIRで指定されたテンポラリファイルを検索し、存在しないときに\$TMPDIRで指定するディレクトリにファイルを作成する実行文があるが、\$TMPDIRで指定されるディレクトリ名を、if文の前のTMPDIRに設定するコマンドを挿入する。ここで指定するテンポラリファイルは、ユーザが任意に設定することができる。

今回の作業において挿入したコマンド文を示す。

```
set TMPDIR=/local/scale/tmp
```

6.3.2 Standard Compositionライブラリの作成

```
% stdcomp.jcl
```

Standard Compositionライブラリは、修正を加えたstdcomp.jclスクリプトファイルを実行することで自動的に作成される。

実行後、/local/scale/datalib/ampx/binrylib配下にscale.rev04.sclibという名称で格納される。実行結果の出力ファイルは、カレントディレクトリにstdcomp.outファイルとして出力される。

6.4 ORIGEN-Sライブラリの作成

6.4.1 データライブラリの概要

ORIGEN-Sのライブラリ作成においても、Standard Compositionライブラリと同様カードイメージ形式からバイナリ形式へ変換するプログラムが用意されている。このプログラムにより、ORIGEN-Sを実行するために必要なバイナリ形式ライブラリが作成される。

作成されるライブラリの概要を以下に示す。

- maphnobr ----- バイナリ形式マスター光子ライブラリ
 - X線及びガンマ線の光子データ
 - 自発核分裂及び(α , n)反応によるガンマ線スペクトル光子データ
- maphh2ob ----- バイナリ形式マスター光子ライブラリ
`maphnobr`ライブラリデータに加え、
 - 水中の電子による制動放射光子ライブラリ
 - 水中の陽電子による制動放射光子ライブラリ
- baslmfbr ----- LMFBRのORIGEN-S用バイナリ形式ワーキングライブラリ
 - 構造材核種、アクチニド核種及び核分裂生成物等の核データ
 - (β , n)崩壊データ、光子生成データ
- maphuo2b ----- バイナリ形式マスター光子ライブラリ
`maphnobr`ライブラリデータに加え、
 - UO_2 の電子による制動放射光子ライブラリ
 - UO_2 の陽電子による制動放射光子ライブラリ
- prlimlwr ----- LWRのORIGEN-Sバイナリ形式ワーキングライブラリ

- ・ (β,n) 崩壊データ
- ・ 核分裂生成データ
- ・ pwr33gwd ----- 燃焼度 バイナリ形式ワーキングライブラリ
- ・ pwr33f71.sas1inp ----- sas1用のサンプルとしてまとめられた燃焼度データ

このバイナリ形式のライブラリ作成は、mkinlib、mkinlib1、mkinlib2のスクリプトファイル（添付資料-16、17、18 参照）が用意されている。

mkinlibは、mkinlib1・mkinlib2を動作させるスクリプトファイルとして用意され、mkinlib1は、mapnbr、maphh2ob、baslmfbr、mapuo2b及びprlmlwrのバイナリ形式ライブラリ作成スクリプトファイルとして用意されている。mkinlib2は、燃焼度ライブラリであるpwr33gwdバイナリ形式ライブラリ・pwr33f71.sas1inpの作成スクリプトファイルとして用意されている。

6.4.2 mkinlib1・mkinlib2スクリプトファイルの修正

ORIGEN-Sのライブラリ作成は、mkinlib1・mkinlib2スクリプトファイルについて修正する。

ORIGEN-Sのデータが格納されるディレクトリの設定及びその配下でカードイメージ形式が格納されるディレクトリの設定を修正する必要がある。

```
set ORDATA=/local/scale/datalib/origen
set ORCARD=/local/scale/datalib/origen/cardlib
```

さらに、if文において\$TMPDIRで指定されたテンポラリファイルを検索し、存在しないときに\$TMPDIRで指定するディレクトリにファイルを作成する実行文があるが、\$TMPDIRで指定されるディレクトリ名を、if文の前のTMPDIRに設定するコマンドを挿入する。ここで指定するテンポラリファイルは、ユーザが任意に設定することができる。

今回の作業において挿入したコマンド文を以下に示す。

```
set TMPDIR=/local/scale/tmp
```

6.4.3 ORIGEN-Sライブラリの作成

```
% mkinlib
```

ORIGEN-Sライブラリは、修正を加えたmkinlib1及びmkinlib2スクリプトファイルを連続的に実行するmkinlibスクリプトファイルを実行することで自動的に作成される。

実行後、/local/scale/datalib/origen/binrylib配下に第6.4.1項で示したファイル名が格納される。実行結果の出力ファイルは、カレントディレクトリにmkinlib1.out及びmkinlib2.outファイルとして出力される。

7. SCALE-4.2の実行

SCALE-4.2は、scale4スクリプトファイル（添付資料-19 参照）を用いて実行する。
実行形式を以下に述べる。

```
% scale4 inputfile outfile
```

scale4スクリプトファイルは、臨界安全、遮蔽解析及び熱解析とSCALE-4.2で行えるすべての安全解析を行える実行スクリプトファイルとして用意されている。

また、上記の実行形式では、UNIXによるシステムメッセージが出力されることから、そのメッセージを予めファイル出力する方法もある。実行形式を以下に示す。

```
% scale4 inputfile outfile > & inp.msg
```

8. サンプル計算

8.1 計算概要

SCALE-4.2の動作確認を行うにあたり、システムで用意されている多くのサンプル問題（臨界安全解析、遮蔽解析、熱解析及びファンクションモジュール単一のサンプル問題）の中から臨界安全解析モジュールCSAS25の問題を採用した。

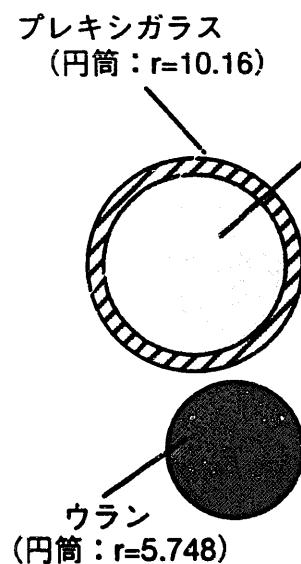
SCALE-4.2の動作確認は、SunEWSへインストールしたロードモジュールでサンプル問題を用いて計算し、SCALE-4.2に格納されているIB MEWSで計算されたサンプル出力と比較することにより行った。

8.2 計算体系

8.2.1 計算モデル

計算モデルの上面図を図3、側面図を図4にそれぞれ表す。

この問題は、4つの円筒形状高濃縮ウラン金属（93.2%）と円筒形状をしたプレキシガラス容器に硝酸ウラニル水溶液（415g U/l）が満たされた4つが図のように配列されている体系についてCSAS25を用いて計算した。



単位：cm

図3. 計算モデル図（上面図）

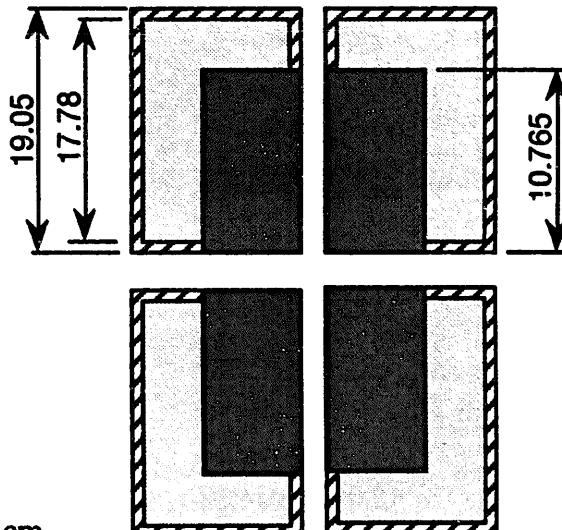


図4. 計算モデル図（側面図）

8.2.2 使用混合物

サンプル問題で用いられている混合物について、以下に示す。

(1) 高濃縮ウラン金属

- ・ ウラン同位体組成：U-235 - 93.2 wt%
- U-238 - 5.6 wt%

U-234 - 1.0 wt%

U-236 - 0.2 wt%

・密度：18.76 g/cc

(2) 硝酸ウラニル水溶液

・濃度：415 gU/l

・ウラン同位体組成：U-235 - 92.6 wt%

U-238 - 5.9 wt%

U-234 - 1.0 wt%

U-236 - 0.5 wt%

・比重：1.555

(3) プレキシガラス

・密度：1.18 g/cc

・組成： $C_5H_8O_2$

8.3 計算結果

SunEWS と IB MEWS の計算結果を以下に示す。比較項目は、最終的な結果を与えるK-ENO-V.aの中性子増倍率及び中性子スペクトルに着目した。

8.3.1 中性子増倍率

(1) SunEWS 計算結果

1kno v.a sample problem 19 4 aqueous 4 metal array of arrays (samp prob 12)		
0	average	avg k-eff
generation	k-effective	deviation
99	9.99247E-01	6.30209E-03
100	9.98489E-01	6.28339E-03
101	9.98235E-01	6.22477E-03
102	9.98198E-01	6.16232E-03
103	9.98551E-01	6.11121E-03

(2) IBMEWS 計算結果

```

1kno v.a sample problem 19 4 aqueous 4 metal array of arrays (samp prob 12)
      0           average       avg k-eff
    generation   k-effective     deviation
      99        1.00457E+00    5.50266E-03
     100        1.00378E+00    5.50431E-03
     101        1.00346E+00    5.45761E-03
     102        1.00379E+00    5.41262E-03
     103        1.00380E+00    5.35877E-03

```

8.3.2 中性子スペクトル

(1) SunEWS 計算結果

0fluxes for unit 1							
0group	region 1		region 2		region 3		deviation
	flux	percent	flux	percent	flux	percent	
1	3.281E-05	2.48	2.153E-05	3.29	1.934E-05	2.96	
2	6.033E-05	1.89	4.172E-05	2.01	3.792E-05	2.10	
3	2.835E-05	1.99	2.055E-05	2.62	1.994E-05	2.85	
4	4.121E-05	1.62	3.095E-05	2.14	2.989E-05	2.34	
5	4.141E-05	1.62	2.950E-05	2.11	2.595E-05	2.28	
6	2.855E-05	1.82	1.956E-05	2.25	1.264E-05	3.64	
7	2.192E-05	1.83	1.378E-05	2.76	9.445E-06	4.45	
8	1.897E-05	1.72	1.203E-05	2.89	8.073E-06	4.70	
9	1.780E-05	1.95	1.050E-05	3.10	6.600E-06	5.08	
10	1.109E-05	1.90	6.612E-06	3.71	4.330E-06	5.73	
11	9.336E-06	2.16	5.501E-06	4.29	3.366E-06	6.30	
12	9.238E-06	2.21	5.917E-06	4.16	3.619E-06	6.18	
13	8.245E-06	2.08	4.594E-06	4.88	2.710E-06	6.88	
14	6.224E-06	2.15	3.656E-06	4.42	2.286E-06	8.59	
15	1.149E-05	1.73	7.687E-06	4.10	4.594E-06	6.44	
16	9.534E-06	1.61	1.146E-05	4.16	6.109E-06	5.33	

(2) IBMEWS 計算結果

0fluxes for unit 1							
0group	region 1		region 2		region 3		deviation
	flux	percent	flux	percent	flux	percent	
1	3.222E-05	2.77	2.096E-05	3.06	1.882E-05	3.01	
2	6.176E-05	1.83	4.331E-05	1.90	3.838E-05	1.93	

3	2.879E-05	2.17	2.028E-05	2.85	1.915E-05	3.01
4	4.322E-05	1.57	3.058E-05	2.31	2.911E-05	2.50
5	4.270E-05	1.59	2.955E-05	2.08	2.634E-05	2.25
6	2.969E-05	1.61	1.899E-05	2.70	1.339E-05	3.62
7	2.271E-05	1.67	1.445E-05	2.57	9.586E-06	3.72
8	2.011E-05	1.65	1.209E-05	3.32	7.725E-06	4.41
9	1.854E-05	1.97	1.089E-05	3.42	7.046E-06	5.18
10	1.155E-05	1.85	6.848E-06	3.75	4.042E-06	6.47
11	9.824E-06	2.11	5.714E-06	4.10	3.479E-06	6.49
12	9.875E-06	1.94	5.734E-06	3.84	3.618E-06	5.92
13	8.297E-06	1.86	4.974E-06	3.73	3.302E-06	6.97
14	6.278E-06	2.16	3.666E-06	5.16	2.448E-06	7.97
15	1.193E-05	1.78	7.760E-06	4.07	4.456E-06	5.63
16	9.866E-06	1.53	1.198E-05	4.21	6.659E-06	5.54

8.4 検討

中性子増倍率と中性子スペクトルを比較すると、中性子増倍率では1 %に満たない違いである。中性子スペクトルにおいても、1 %～3 %程度の違いであるためマシン環境（コンパイラやOS等）の違いによる計算誤差範囲内であると考える。

SunEWSにインストールしたSCALE-4.2のCSAS25の動作確認は、開発ベースのIBM-EWSと同様な結果を与えることが確認できた。

なお、今後は、SCALE-4.2の他のモジュールについて、サンプル計算との比較計算を行い、動作確認を実施する計画である。

9. あとがき

現在の情報処理技術は、EWS・PCの飛躍的進歩（計算精度の向上）やユーザニーズに対応したダウンサイジング傾向にあるのが世界的風潮である。これを背景に科学技術計算の分野でも開発ベースは、メインフレームからEWS・PCへと移行しつつある。しかし、開発側とユーザ側のマシンが異なる場合、その機種に合致しないケース（すべてではない）が発生するものと予想される。ユーザは、常に最適な対応ができるよう計算機に対する十分な知識と柔軟性が必要であろう。

最後に、SCALE-4.2の入手及びサポートにあたって多大なご協力御尽力頂いた東海事業所再処理工場技術課副主任研究員須藤俊幸氏にこの場を借りて厚く御礼申し上げます。

10. 参考文献

- 1) Bucholz J.A. (ed.) " SCALE : A Module Code System for Performing Standardized Computer Analyses for Licensing Evaluation" NUREG/CR-0200 (1980)
- 2). 野尻他、"SCALE-4コードシステムを用いた臨界ベンチマーク計算
～MOX粉末系及びPu粉末系を対象とした～"
日本原子力学会「1992年春の年会」、H-43
- 3). 野尻他、"SCALE-4コードシステムを用いた臨界安全データの計算
～MOX燃料加工施設を対象とした～"
日本原子力学会「1993年春の年会」、A-43
- 4). 清水他、"MOX粉末系ベンチマークにおけるMCNP4とKENO-V.aの比較計算"
日本原子力学会「1994年春の年会」、G-22
- 5). 野尻他、"透明中性子パネルの開発 (2) - 計算解析 -"
日本保健物理学会「第28回研究発表要旨集」、B-27
- 6). Bucholz J.A. (ed.) " SCALE4.2 : A Module Code System for Performing Standardized Computer Analyses for Licensing Evaluation" NUREG / CR-0200 Revision4 (1993)
- 7). RSIC Newsletter , No.349 ; December 1993
- 8). SCALE Newsletter , No.8 ; January 1994
- 9). Bell M.J. " ORIGEN : The ORNL Isotope Generation and Depletion Code" ORNL-4628 (May 1973)
- 10). Bondarenko I.I. , Ed., Group Constants for Nuclear Calculations , Consultants Bureau , New York(1964).
- 11). Nordheim L.W. " The Theory of Resonance Absorption , " Proceedings of

**Symposia in Applied Mathematics , Vol.XI, 58, G.Birkhoff and E.P. Wigner,
Eds.,Am.Math.Soc.(1961)**

- 12). **Greene N.M. , Lucius J.L. , Petrie L.M. , Ford W.E. , White J.E. , and Wright
R.Q. " AMPX-A Modular Code System to Generate Multigroup Neutron-Gamma
Cross Section From ENDF/B"**

ORNL/TM-3706(1976)

- 13). **Sunマイクロシステムズ、" Sun FORTRAN1.4マニュアルセット"**

N-825-1005-10

添 付 資 料

1. convert スクリプトファイル
格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
set RTNDIR=`pwd`
set TMPDIR=/local/scale/tmp
if (!(-e $TMPDIR)) mkdir $TMPDIR
set tmpdir=$TMPDIR
cp $2 $TMPDIR/INPUT
cp /local/scale/exe/comment $TMPDIR
cd $TMPDIR
rm OUTPUT
rm SYSPRINT
rm SYSIN
cat > SYSIN <<EOF
$1
EOF
comment
cd $RTNDIR
cp $TMPDIR/OUTPUT $3
```

2. h7lib.cソースプログラム

格納ディレクトリ : /local/scale/src/unixlib

```

#include <sys/types.h>
#include <sys/times.h>
#include <time.h>
#include <sys/utsname.h>
#include <sys/unistd.h>

extern struct tm *usr_time();
extern time_t time();
struct tm *tm;
time_t Clock;

void datimz(string)
char *string;
{
    Clock = time(0);
    tm = usr_time(&Clock);
    memcpy(string, asctime(tm), 24);
}

void getmem(number,size,array,addr,offset)
    int *number, *size, *array, *addr, *offset;
{ int *malloc();
    int *addrp;
    addrp = malloc((*number)*(*size));
    *offset = (int)(addrp-array)*sizeof(int)/(*size);
    *addr = (int)addrp;
}

void chgmem(number,size,array,addr,offset)
    int *number, *size, *array, *addr, *offset;
{ int *realloc();
    int *addrp;

```

```
addrp = realloc(*addr, (*number)*(*size));
*offset = (int)(addrp-array)*sizeof(int)/(*size);
*addr = (int)addrp;
}

double second()
{
    double cpu;
    struct tms now;
    times(&now);
    cpu = (double)now.tms_utime/_SC_CLK_TCK;
    return (cpu);
}
```

3. jstime.cソースプログラム

格納ディレクトリ：local/scale/src/unixlib

```
#####
#
```

Audit Trail Information

Date the module was last permanently updated:	93/07/22
Time the module was last permanently updated:	07:38:55
Programmer name:	L.M.PETRIE
Module name:	ULJSTIME
Current archiving level number:	00001
Current number of permanent updates:	00001
Date of last access by librarian:	93/07/22
Dataset name:	X4S.SCALE4.MASTER

```
#####
#
```

*/

#include <sys/times.h>

```
#ifdef UNDERSCORE
#define JSTIME jstime_
#else
#define JSTIME jstime
#endif
```

```
extern void JSTIME (time_left)
    long *time_left;
```

```
{
    struct tms buf,*buffer;
    double time_used;
    long time_gms;
```

```
buffer = &buf;
time_gms = (long)times(buffer);
if (time_gms != -1)
{
    time_used = (buffer->tms_utime + buffer->tms_stime +
                  buffer->tms_cutime + buffer->tms_cstime)/60.;
*time_left = 100.* (20000000.-time_used);
return;
}
else
{
    *time_left = 0L;
return;
}
```

4. input.fソースプログラム

格納ディレクトリ : /local/scale/src/ocular

c

```

subroutine input(rrin,rrot,thlt,thrt,zzbk,zzfr,rf,thf,zf,rg,thg,
*      zg,r,th,z,re,the,ze,a,arg,val,noreg,mats,matl,its,ngens,
*      nin,not,nlt,nrt,nbk,nfr,mat,matnam,nemint,nrrin,nrrot,
*      nthlt,nthrt,nzzbk,nzzfr,nenclt,nenbct,nsrf,nsurft,enrrin,
*      enrrot,enthlt,enthrt,enzzbk,enzzfr,nensrt,nemist,
*      nemvyt,emis,nemr,nemth,nemz,nemir,nemith,
*      nemiz,nanalt,ntab,ntbprs,nparm,ndrg,ndthg,ndzg,
*      nerrin,nerrot,nethlt,nethrt,nezzbk,nezzfr,
*      hival,loval,lstop)

```

***** 省略 *****

```

c----- fukasaku changed 94/2/ -----  

c      write(io,900)  

c900  format(1hl,/////////////22x,10('o'),4x,10('c'),4x,2('u'),6x,2('u'),
c      .4x,2('l'),14x,6('a'),6x,8('r')/  

c      .22x,10('o'),4x,10('c'),4x,2('u'),6x,2('u'),4x,2('l'),
c      .13x,8('a'),5x,9('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r'))  

c      write(io,901)  

c901  format(22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,
c      .2('l'),12x,10('a'),4x,9('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,10('a'),4x,8('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),5x,2('r')/  

c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),

```

```

c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/
c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/
c      .22x,2('o'),6x,2('o'),4x,2('c'),12x,2('u'),6x,2('u'),4x,2('l'),
c      .12x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/
c      .22x,10('o'),4x,10('c'),4x,10('u'),4x,10('l'),
c      .4x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/
c      .22x,10('o'),4x,10('c'),4x,10('u'),4x,10('l'),
c      .4x,2('a'),6x,2('a'),4x,2('r'),6x,2('r')/)

12    write(io,902) versun(2)

c-----

```

***** 省 略 *****

```

      return

902 format(/// ',22x,'release date: ',a8,18x,' fortran 77 - first ',
. 'edition')

1000 format('1',// code: ',a8,1x,a8,// date: ',a8,// time: ',a8,
* // jobnam: ',a8,// computer: ',a8)

1001 format(/2x,'job description -- ',18a4)

1002 format('1***** job description *****',
* 18a4,// ***** the input data for this case have been read and',
* ' processed',// ***** number of errors encountered in the',
* ' input data for this case was --',i5,// ***** number of warnin',
* 'gs encountered in the input data for this case was --',i5)

      end

```

5. getmdl.fソースプログラム

格納ディレクトリ : /local/scale/src/driver

```

subroutine getmdl ( n5,n6,n7,module,i15 )
c----- fukasaku changed 94/2/ -----
      external putenv !$pragma c( putenv )
c-----
      integer putenv
      c   external putenv
      character module*8,card*80,equal*1,pound*1,blank*1,end*3
      character sclprm*11,parmfl*82
      common /params/ parmfl
      logical lmodl
      data equal/'=/,pound/'#/,:blank/' ',end/'end'/
      data sclprm/'scaleprms='/
      save /params/
      lmodl = .false.

10 continue
      read(n5,'(a80)',end=20) card
      if (card(1:1).eq.equal .or. card(1:1).eq.pound) then
         if (lmodl) then
            write(n6,'(a)') '***** the preceding data stream does '//*
                           'not terminate with an end card'
            close(unit=n7)
            i15 = 1
            return
         else
            open(unit=n7,status='unknown',form='formatted',file='input')
            lmodl = .true.
            module = card(2:9)
            ilast = max(11,lnblk(card))
            parmfl = sclprm//card(11:ilast)//char(0)
            ierr = putenv(parmfl)
            write(n6,'(a)') '- module //module// will be called'
         end if
      else
         if (lmodl) then
            if ((card(1:3).eq.end) .and. ((card(4:4).ne.blank) .or.
*               (card(5:5).eq.blank))) then

```

```
close(unit=n7)
i15    = 0
return
else
  write(n7,'(a80)') card
  write(n6,'(6x,a80)') card
end if
else
  write(n6,'(a)') '0 the following data cards precede an - card'
  i15    = 1
  return
end if
end if
go to 10
20 continue
if ( lmodl ) then
  write(n6,'(a)') '***** the preceding data stream does not'//
*           ' terminate with an end card *****'
end if
i15    = -1
return
end
```

6. rcover.fソースプログラム

格納ディレクトリ : /local/scale/src/marslib

```

subroutine rcover(d)

c   this routine normalizes results based on previous batch
c   when a fatal geometry error occurs during morse-sgc run
c   this requires altering number batches and number particles
common/input/iadjm,nstrt,nmost,nits,nquit,ncoltp,istat,nsplit,
* nkill,npast,noleak,iebias,nkcalc,normf, media,nmix,medalb,mxreg,
* mfistp,nnga,ngga,nngtp, nopt,iggopt,ndsn,ndsg,ncoef,nsct,maxgp,
* irdsg,istr, ifmu,imom,iprin,ipun,ixtape,jxtape,io6r,igqpt, isour,
* ngpfs,isbias,nsour, nd,nne,ne,nt,na,nresp,nex,nexnd,iflag(20),
* tmax,tcut,wtstrt,agstrt,xstrt,ystrt,zstrt, uinp,vinp,winp,nxpm,
* nhistr,nhismx,nbind(36),ncolls(13), ng,iftg,igg,nnuc,idt,nrp,nlm,
* n2m,nsgps,title(20),dat(8) ,jftg,kftg,lftg
common /point/lfp1,lfp2,lfp3,lfp4,lfp5,lfp6,lfp7,lfp8,lfp9, lfp10,
* lfp11,lfp12,lfp13,lfp14,lfp15,lfp16,lfp17,lfp18,lfp19, lfp20,
* lfp21,lfp22,lfp23,lfp24,lap1,lap2,lap3,lap4,lap5, lap6,lap7,lsp1,
* lsp2,lsp3, lfp25,lfp26,lfp27,lfp28,lfp29,lfp30,lfp31,lfp32, lx1,
* lx2,lx3,lx4,lt1,lt9
common /perm/ inn,iout,nlft,lci,n12,n14,n16,n17,n81,n90,n91,n92
1 ,n95,n96,n97,n98
common/qdet/locrsp,locxd,locib,locco,loci,locud,locsrd, locqe,
* locqt,locqte,locqae,lmax,efirst,egtop
common/apoll/dff,ddf,deadwt(5),eta,etath,etausd,xtra(10), iters,
* itime,itstr,maxtim,mgpreg,nalb,ndead(5),newnm,ngeom, nlast,nmem,
* nmgp,nmtg, npscl(13),nsigl,nxtra(10)
common/fisbnk/nfisbn,nfish,ftotl,fwate,watef,awate,npart
dimension d(*)
npart = npart-nmem
nbatch=iters-1
irun=nquit
nits = nbatch
write(iout,10100)nbatch
10100 format('0results are for',i6,' batches.')
if(nbbatch.eq.0) stop
c----- fukasaku changed 94/2/ -----c
c 160 call nrun(d,nits,irun,d(locud+nresp*nd),d(locud+2*nresp*nd),

```

```
c      * d(locsd+nresp*nd),d(locsd+2*nresp*nd),d(locqe+ne*nd),
c      * d(locqe+2*ne*nd),d(locqte+nt*ne*nd),d(locqte+2*nt*ne*nd),
c      * d(locqae+na*ne*nd),d(locqae+2*na*ne*nd),d(locxd+5*nd),
c      * d(locib+2*ne),d(loct+nd*nt),d(locqae),d(locco),d(lapl),
c      * d(locsd+3*nresp*nd),d(locib+ne),d(locqe+3*ne*nd),
c      * d(locqt+3*nt*nd*nresp),d(loct),d(locqte+3*nt*ne*nd),
c      * d(locqae+3*na*ne*nd),d(locqt+nt*nd*nresp),d(locqt+2*nt*nd*nresp),
c      * d(locxd+6*nd),d(locxd+4*nd),nd,ne,nt,na,nresp,nexnd)
c-----
```

stop 6135

7. clgscale スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh -xv
# shell script to compile and link scale modules
set SCALE=/local/scale
setenv OBJDIR $SCALE/obj
setenv CMDDIR $SCALE/cmds
setenv SRCDIR $SCALE/src
cd $SCALE
mkdir obj
mkdir exe
cd obj
mkdir aim
mkdir ajax
mkdir ale
mkdir alpo
mkdir bonami
mkdir compost
mkdir corectol
mkdir couple
mkdir csas
mkdir driver
mkdir h7map
mkdir h7monitor
mkdir h7tecplot
mkdir heating
mkdir htasl
mkdir ice
mkdir kenova
mkdir lava
mkdir lib
mkdir mal
mkdir marslib
mkdir miplib
mkdir modify
mkdir morse
mkdir nitawl
mkdir ocular
```

```
mkdir origen
mkdir osbico
mkdir osbire
mkdir perfume
mkdir picture
mkdir plorigen
mkdir rade
mkdir sasl
mkdir sas2
mkdir sas3
mkdir sas4
mkdir sublib
mkdir unixlib
mkdir wax
mkdir wgt
mkdir xsdose
mkdir xsdrn
#
#  create sublib
cd $OBJDIR/sublib
$CMDDIR/fort "*.f"
$CMDDIR/lked sub *.o
cd $OBJDIR/unixlib
$CMDDIR/fort "*.f"
cc -c -O -DUNDERSCORE $SRCDIR/unixlib/*.c
$CMDDIR/lked sub *.o
#  create miplib
cd $OBJDIR/miplib
$CMDDIR/fort "*.f"
$CMDDIR/lked mip *.o
#  create marslib
cd $OBJDIR/marslib
$CMDDIR/fort "*.f"
$CMDDIR/lked mars *.o
#
#  create scale functional modules
#
cd $OBJDIR/bonami
$CMDDIR/fort "*.f"
```

```
$CMDDIR/lked bonami *.o
$CMDDIR/load bonami bonami
$CMDDIR/load o0o008 bonami
#
cd $OBJDIR/couple
$CMDDIR/fort "*.f"
$CMDDIR/lked couple *.o
$CMDDIR/load couple couple
$CMDDIR/load o0o005 couple
#
cd $OBJDIR/driver
$CMDDIR/fort "*.f"
$CMDDIR/lked driver *.o
$CMDDIR/load scale driver
#
cd $OBJDIR/heating
$CMDDIR/fort "*.f"
$CMDDIR/lked heating *.o
$CMDDIR/load heating heating
$CMDDIR/load htng72 heating
#
cd $OBJDIR/h7map
$CMDDIR/fort "*.f"
$CMDDIR/lked h7map *.o
$CMDDIR/load h7map h7map
$CMDDIR/load h7maprz h7map
#
cd $OBJDIR/htas1
$CMDDIR/fort "*.f"
$CMDDIR/lked htas1 *.o
$CMDDIR/load htas1 htas1
#
cd $OBJDIR/h7tecplot
$CMDDIR/fort "*.f"
$CMDDIR/lked h7tecplot *.o
$CMDDIR/load h7tecplot h7tecplot
#
cd $OBJDIR/h7monitor
$CMDDIR/fort "*.f"
```

```
$CMDDIR/lked h7monitor *.o
$CMDDIR/load h7monitor h7monitor
#
cd $OBJDIR/ice
$CMDDIR/fort "*.f"
$CMDDIR/lked ice *.o
$CMDDIR/load ice ice
$CMDDIR/load o0o007 ice
#
cd $OBJDIR/kenova
$CMDDIR/fort "*.f"
$CMDDIR/lked kenova *.o
$CMDDIR/load kenova kenova
$CMDDIR/load o0o009 kenova
#
cd $OBJDIR/csas
$CMDDIR/fort "*.f"
$CMDDIR/lked csas *.o
$CMDDIR/load csas csas kenova mip
#
cd $OBJDIR/modify
$CMDDIR/fort "*.f"
$CMDDIR/lked modify *.o
$CMDDIR/load modify modify kenova
#
cd $OBJDIR/morse
$CMDDIR/fort "*.f"
$CMDDIR/lked morse *.o
$CMDDIR/load morse morse mars
$CMDDIR/load o0o006 morse mars
$CMDDIR/load o0o106 morse mars
#
cd $OBJDIR/nitawl
$CMDDIR/fort "*.f"
$CMDDIR/lked nitawl *.o
$CMDDIR/load nitawl nitawl
$CMDDIR/load o0o002 nitawl
#
cd $OBJDIR/ocular
```

```
$CMDDIR/fort "*.f"
$CMDDIR/lked ocular *.o
$CMDDIR/load ocular ocular
$CMDDIR/load ocular72 ocular
#
cd $OBJDIR/origen
$CMDDIR/fort "*.f"
$CMDDIR/fortno "nvrp.f"
$CMDDIR/lked origen *.o
$CMDDIR/load origins origen
$CMDDIR/load o0o004 origen
#
cd $OBJDIR/osbico
$CMDDIR/fort "*.f"
$CMDDIR/lked osbico *.o
$CMDDIR/load osbico osbico
#
cd $OBJDIR/osbire
$CMDDIR/fort "*.f"
$CMDDIR/lked osbire *.o
$CMDDIR/load osbire osbire
#
cd $OBJDIR/picture
$CMDDIR/fort "*.f"
$CMDDIR/lked picture *.o
$CMDDIR/load main picture mars
mv $SCALE/exe/main $SCALE/exe/picture
#
cd $OBJDIR/sas1
$CMDDIR/fort "*.f"
$CMDDIR/lked sas1 *.o
$CMDDIR/load sas1 sas1 mip
#
cd $OBJDIR/sas2
$CMDDIR/fort "*.f"
$CMDDIR/lked sas2 *.o
$CMDDIR/load sas2 sas2 mip
#
cd $OBJDIR/sas3
```

```
$CMDDIR/fort "*.f"
$CMDDIR/lked sas3 *.o
$CMDDIR/load sas3 sas3 mip mars
#
cd $OBJDIR/sas4
$CMDDIR/fort "*.f"
$CMDDIR/lked sas4 *.o
$CMDDIR/load sas4 sas4 mip mars
#
cd $OBJDIR/xsdose
$CMDDIR/fort "*.f"
$CMDDIR/lked xsdose *.o
$CMDDIR/load xsdose xsdose
$CMDDIR/load o0o102 xsdose
#
cd $OBJDIR/xsdrn
$CMDDIR/fort "*.f"
$CMDDIR/fortno "plsnt.f"
$CMDDIR/lked xsdrn *.o
$CMDDIR/load xsdrn xsdrn
$CMDDIR/load o0o001 xsdrn
#
```

8. fort スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
# shell script to compile a deck
if ( ! $#argv ) then
    echo "Usage: $0 decks"
    echo ""
    exit 0
endif
set src=`echo $cwd | sed s/obj/src/`
set list=""
while ( $#argv >= 1 )
    set list = "$list $src/$1"
    shift
end
f77 -c -O $list
```

9. fortgスクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
# shell script to compile a deck
if ( ! $#argv ) then
    echo "Usage: $0 decks"
    echo ""
    exit 0
endif
set src=`echo $cwd | sed s/lib/src/`
set list=""
while ( $#argv >= 1 )
    set list = "$list $src/$1"
    shift
end
f77 -c -g $list
```

10. fortnoスクリプトファイル
格納ディレクトリ：/local/scale/cmds

```
#!/bin/csh
# shell script to compile a deck without optimization.
if ( ! $#argv ) then
    echo "Usage: $0 decks"
    echo "
    exit 0
endif
set src=`echo $cwd | sed s/obj/src/`
set list=""
while ( $#argv >= 1 )
    set list = "$list $src/$1"
    shift
end
f77 -c $list
```

11. lkedスクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
# shell command to link modules
set dir=$argv[1]
shift
/usr/5bin/ar rvs ..../lib/lib$dir.a $argv[*]
```

12. load スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
# shell command to link modules
cd ../
set prg=$argv[1]
shift
set dir=$argv[1]
shift
set onames="..../exe/$prg $dir/$prg.o"
set libs="-L./lib -l$dir"
foreach ds ($argv[*])
  if ($ds == *.o) then
    set onames="$onames $ds"
  else
    set libs="$libs -l$ds"
  endif
end
f77 -o $onames $libs -lsub
```

13. clgampx スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh -xv

# shell script to compile and link ampx modules
set SCALE=/local/scale
setenv OBJDIR $SCALE/obj
setenv CMDDIR $SCALE/cmds
setenv SRCDIR $SCALE/src
#
# create ampx utilities
#
cd $OBJDIR/aim
$CMDDIR/fort "*.f"
$CMDDIR/lked aim *.o
$CMDDIR/load aim aim
#
cd $OBJDIR/ajax
$CMDDIR/fort "*.f"
$CMDDIR/lked ajax *.o
$CMDDIR/load ajax ajax
#
cd $OBJDIR/ale
$CMDDIR/fort "*.f"
$CMDDIR/lked ale *.o
$CMDDIR/load main ale
mv $SCALE/exe/main $SCALE/exe/ale
#
cd $OBJDIR/alpo
$CMDDIR/fort "*.f"
$CMDDIR/lked alpo *.o
$CMDDIR/load main alpo
mv $SCALE/exe/main $SCALE/exe/alpo
#
cd $OBJDIR/compoz
$CMDDIR/fort "*.f"
$CMDDIR/lked compoz *.o
$CMDDIR/load compoz compoz
#
```

```
cd $OBJDIR/corectol
$CMDDIR/fort "*.f"
$CMDDIR/lked corectol *.o
$CMDDIR/load crctol corectol
mv $SCALE/exe/crctol $SCALE/exe/corectol
#
cd $OBJDIR/lava
$CMDDIR/fort "*.f"
$CMDDIR/lked lava *.o
$CMDDIR/load lava lava
#
cd $OBJDIR/mal
$CMDDIR/fort "*.f"
$CMDDIR/lked mal *.o
$CMDDIR/load mal mal
#
cd $OBJDIR/perfume
$CMDDIR/fort "*.f"
$CMDDIR/lked perfume *.o
$CMDDIR/load perfum perfume
mv $SCALE/exe/perfum $SCALE/exe/perfume
#
cd $OBJDIR/rade
$CMDDIR/fort "*.f"
$CMDDIR/lked rade *.o
$CMDDIR/load rade rade
#
cd $OBJDIR/wax
$CMDDIR/fort "*.f"
$CMDDIR/lked wax *.o
$CMDDIR/load wax wax
#
cd $OBJDIR/wgt
$CMDDIR/fort "*.f"
$CMDDIR/lked wgt *.o
$CMDDIR/load weight wgt
mv $SCALE/exe/weight $SCALE/exe/wgts
#
```

14. malwgts.jcl スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
#shell script to run mal and wgts to make albedos & weights libs for KENO
# this script runs mal and wgts stand-alone (without the SCALE driver.)
set RTNDIR=`pwd`
set SCALE=/local/scale
setenv TMPDIR $SCALE/tmp
setenv DATA_DIR $SCALE/datalib
setenv DATA_BCD $DATA_DIR/keno/cardlib
setenv DATA_BIN $DATA_DIR/keno/binrylib
setenv PGM_DIR $SCALE/exe
if (!(-e $TMPDIR)) mkdir $TMPDIR
set tmpdir=$TMPDIR
cp $DATA_BCD/albedos.data $tmpdir/fort.35
cp $DATA_BCD/weights.data $tmpdir/fort.36
cd $tmpdir
if (!(-e mal)) ln -s $PGM_DIR/mal mal
if (!(-e wgts)) ln -s $PGM_DIR/wgts wgts
mal <$DATA_BCD/mal.in > output
wgts <$DATA_BCD/weight.in >> output
cd $RTNDIR
cp $tmpdir/fort.79 $DATA_BIN/albedos
cp $tmpdir/fort.80 $DATA_BIN/weights
cp $tmpdir/output malwgts.out
```

15. stdcomp.jcl スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```

#!/bin/csh -xv

#shell script to run compoz to create standard composition library
set RTNDIR=`pwd`
set SCALE=/local/scale
setenv TMPDIR $SCALE/tmp
setenv PGM_DIR $SCALE/exe
setenv DATA_DIR $SCALE/datalib
setenv DATA_BCD $SCALE/datalib/ampx/cardlib
setenv DATA_BIN $SCALE/datalib/ampx/binrylib
if (!(-e $TMPDIR)) mkdir $TMPDIR
set tmpdir=$TMPDIR
cp $DATA_BCD/scale.rev04.sclib.data $tmpdir/sysin
cd $tmpdir
if (!(-e scale)) ln -s $PGM_DIR/scale scale
if (!(-e compoz)) ln -s $PGM_DIR/compoz compoz
if (!(-e qatable)) ln -s $DATA_DIR/qatable qatable
if (!(-e aliases)) ln -s $DATA_DIR/aliases aliases
rm print msg _prt* _out*
scale >& msg
cd $RTNDIR
cp $tmpdir/ft89f001 $DATA_BIN/scale.rev04.sclib
cat $tmpdir/msg > stdcomp.out
cat $tmpdir/print >> stdcomp.out
cat $tmpdir/_prt* >> stdcomp.out
cat $tmpdir/_out* >> stdcomp.out

```

16. mkbmilib スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```
#!/bin/csh
# csh script with command name: mkbmilib      (no additional arguments
#                                         required)
#
# this script creates all new origin-s binary library datasets
#
# it uses the 2 separate scripts:  mkbmilib1 and mkbmilib2
#
# inputs for the 2 cases are in files:  mkbmilib1.inp and mkbmilib2.inp
#
# outputs for the 2 cases are in files:  mkbmilib1.out and mkbmilib2.out
#
# invoke the first script:
#
#          mkbmilib1
#
# invoke the second script:
#
#          mkbmilib2
#
# this completes the production of the new origin-s binary library
#                                         datasets
#
```

17. mkbinlib1 スクリプトファイル

格納ディレクトリ：/local/scale/cmds

```

#!/bin/csh -xv

# csh script with command name: mkbinlib1
# no additional parameters are required to execute.

#
# this script creates the new origin-s binary library datasets:
#
#      $ORDATA/binrylib/maphuo2b
#      $ORDATA/binrylib/maphh2ob
#      $ORDATA/binrylib/maphnbr
#      $ORDATA/binrylib/prlimlwr
#      $ORDATA/binrylib/baslmfbr

#
#
#
# ****
#
# *          one of the major additions to this library preparation *
# *  case is the preliminary ("presas2") binary library made for   *
# *  input to sas2 (lwr) cases.  all of the individual nuclides    *
# *  (approximately 224) on the scale "burnup" cross section        *
# *  library are updated with a typical pwr flux spectrum.  when   *
# *  used in sas2 cases, any nuclides not updated by the case     *
# *  will at least have these generated presas-values as an       *
# *  improvement over the basic values of the card-image library. *
# *
#
# ****
#
set TMPDIR=/local/scale/tmp
set ORDATA=/local/scale/datalib/origin
set ORCARD=/local/scale/datalib/origin/cardlib

#
if (!(-e $TMPDIR))  mkdir $TMPDIR
set tmpdir=$TMPDIR
#
# get the card-image libraries used for this case:
#
# these are the master photon data concatenated:

```

```

#
cp $ORCARD/ft60f001 $TMPDIR
cp $ORCARD/ft61f001 $TMPDIR
cp $ORCARD/ft62f001 $TMPDIR
# cat $ORCARD/mpdkxgam $ORCARD/mpsfangm $ORCARD/mpbru* > $TMPDIR/ft60f001
# cat $ORCARD/mpdkxgam $ORCARD/mpsfangm $ORCARD/mpbrh* > $TMPDIR/ft61f001
# cat $ORCARD/mpdkxgam $ORCARD/mpsfangm > $TMPDIR/ft62f001
#
rm $TMPDIR/ft21f001
touch $TMPDIR/ft21f001
rm $TMPDIR/ft23f001
touch $TMPDIR/ft23f001
rm $TMPDIR/ft24f001
touch $TMPDIR/ft24f001
rm $TMPDIR/ft25f001
touch $TMPDIR/ft25f001
rm $TMPDIR/ft26f001
touch $TMPDIR/ft26f001
rm $TMPDIR/ft33f001
touch $TMPDIR/ft33f001
rm $TMPDIR/ft71f001
touch $TMPDIR/ft71f001
#
#
#-----#
#
test.scale4 $ORCARD/mkbinlibl.inp mkbinlibl.out >& mkbinlibl.msg
#
#-----#
#
***** * *****
#
# run origen-s, couple, and sas2h modules to generate *
# origen-s binary libraries *
#
***** * *****
#
# the first origen-s case creates a binary master photon library
# containing bremsstrahlung from a uo2 matrix.

```

```
#  
# the second origin-s case creates a binary master photon library  
# containing bremsstrahlung from h2o.  
#  
# the third origin-s case creates a binary master photon library  
# containing no bremsstrahlung.  
#  
# see section m6.5, /master photon data bases/, of /origin-s data  
# libraries/, in the scale documentation for a description of the  
# master photon libraries.  
#  
# the first couple case (all following the first /-couple/)  
# and the fourth origin-s case create a pwr origin-s binary  
# working library.  
#  
# the second couple case (all following the second /-couple/)  
# and the fifth origin-s case create the basic lmfbr origin-s  
# binary working library.  
#  
# the sas2h case makes a preliminary origin-s binary working library  
# with cross sections updated for all nuclides in the scale burnup  
# library using a pwr fuel assembly in the case description.  
#  
# see sections m6.7.3, /generation of some useful libraries/ and  
# m6.8, /origin-s libraries at ornl/, of /origin-s data libraries/,  
# in the scale documentation, for a description of the origin-s  
# binary working libraries generated by the couple, origin-s, and  
# sas2h cases.  
#  
# it should be noted that the unit numbers for the various origin-s  
# libraries given in the sample problems in /origin-s data libraries/  
# do not always correspond to the unit numbers used in the couple  
# and origin-s cases here. the unit numbers had to be changed for  
# the cases here to run all in one batch.  
#  
# a scratch dataset:  
#  
# ft21f001 is the unit for an lwr origin-s binary working  
# library created by origin-s by adding photon data to the library
```

```
# on unit ft42f001. after its creation, this library is read
# as input for the sas2h run which generates the origin-s binary
# working library with updated cross sections referred to as the
# preliminary lwr (prlimlwr) library.
#
#
# $ORDATA/binrylib/pwr33gwd:
#
#
# ft22f001 is the unit on which the origin-s binary library output of the
# 2nd sas2h run for a typical pwr is to be stored (on binrylib/pwr33gwd);
# but, because sas2h will write the library only on unit 33, the unit
# ft22f001 here is not used.
#
#
# $ORDATA/binrylib/maphnbr:
#
#
# ft23f001 is the unit on which the origin-s binary master photon
# library containing no bremsstrahlung is created by the third
# origin-s case, which converts the card-image master photon library
# on unit ft62f001.
#
#
# $ORDATA/binrylib/maphh2ob:
#
#
# ft24f001 is the unit on which the origin-s binary master photon
# library containing bremsstrahlung from h2o is created by the
# second origin-s case, which converts the card-image master photon
# library on unit ft61f001.
#
#
# $ORDATA/binrylib/baslmfbr:
#
#
# ft25f001 is the unit for the basic lmfbr origin-s binary working
# library created by origin-s by adding photon data to the library
# on unit ft43f001
#
#
# $ORDATA/binrylib/maphuo2b:
#
#
# ft26f001 is the unit on which the origin-s binary master photon
# library containing bremsstrahlung from a uo2 matrix is created by
# the first origin-s case, which converts the card-image master
# photon library on unit ft60f001. after its creation, this library
# is read by (a) the origin-s cases remaking the photon group data of
```

```
# the lwr & lmfbr binary libraries, and (b) the sas2h case which
# generates the updated origin-s binary working library for a
# typical pwr.

#
#
#
# ft89f001 is the scale standard composition library

#
#
# ft90f001 is the unit for the sas2h master interface

#
#
# ft93f001 is the unit for couple input data generated by sas2h

#
#
# ft94f001 is the unit for origin-s input data generated by sas2h

#
#
# ft96f001 is the unit for bonami-s input data generated by sas2h

#
#
# ft97f001 is the unit for nitawl-s input data generated by sas2h

#
#
# ft98f001 is the unit for xsdrnpm-s input data generated by sas2h

#
#
# $ORDATA/binrylib/prlimlwr:

#
#
# ft33f001 is the updated pwr binary working library with
# data which includes updated cross-section data for all nuclides
# of the scale burnup library, generated by the sas2h run.

#
#
# ft42f001 is the unit for the origin-s binary working library
# created by the couple code when converting the lwr origin-s
# card-image libraries to binary format.

#
#
# ft43f001 is the unit for the origin-s binary working library
# created by the couple code when converting the lmfbr origin-s
# card-image libraries to binary format. it is used during the
# creation of the origin-s basic lmfbr binary working library.

#
#
# ft53f001 contains xsdrnpm-s scalar flux data used by sas2h

#
#
# ft55f001 contains scale driver halt data for sas2h

#
#
# ft60f001 is the card-image master photon library with uo2
```

```
# bremsstrahlung
#
# ft61f001 is the card-image master photon library with h2o
# bremsstrahlung
#
# ft62f001 is the card-image master photon library with no
# bremsstrahlung
#
# ft70f001 is the unit for the sas2h auxiliary interface
#
# a scratch dataset:
#
# ft71f001 contains nuclide concentrations and all other data
# for the plorigen code and is made by origen-s during a sas2h run.
#
# note: ft71f001 is not saved from this case.
#
# ft72f001 contains a card-image version of the data on ft71f001
#
# ft74f001 contains neutron and gamma sources generated during a
# sas2h run.
#
# now store the new libraries and sas1 source file:
#
cp $TMPDIR/ft23f001 $ORDATA/binrylib/maphnbr
cp $TMPDIR/ft24f001 $ORDATA/binrylib/maphh2ob
cp $TMPDIR/ft25f001 $ORDATA/binrylib/baslmfbt
cp $TMPDIR/ft26f001 $ORDATA/binrylib/maphuo2b
cp $TMPDIR/ft33f001 $ORDATA/binrylib/prlimlwr
#
```

18. mkbmilib2 スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```

#!/bin/csh -xv

# csh script with command name: mkbmilib2

#
#      this mkbmilib2 script is called by the script "mkbmilib".
#
#      it executes the second sas2h case for making binary libraries,
#      producing a 3-cycle library for the typical pwr burnup of 33 gwd/mtu.
#      this library is copied from ft33f001 to $ORDATA/binrylib/pwr33gwd.
#
#      this script creates the new origin-s binary library dataset:
#
#      $ORDATA/binrylib/pwr33gwd
#
#      ...and sources for the sasl control module samples on:
#
#      $ORDATA/pwr33f71.saslinp
#
set TMPDIR=/local/scale/tmp
set ORDATA=/local/scale/datalib/origin
#
if (!(-e $TMPDIR))  mkdir $TMPDIR
set tmpdir=$TMPDIR
#
rm $TMPDIR/ft21f001
touch $TMPDIR/ft21f001
rm $TMPDIR/ft33f001
touch $TMPDIR/ft33f001
rm $TMPDIR/ft71f001
touch $TMPDIR/ft71f001
#
#-----
#
scale4 $ORDATA/cardlib/mkbmilib2.inp mkbmilib2.out >& mkbmilib2.msg
#
#-----
#*****

```

```
#          *
# run sas2h module to generate a pwr (33 gwd/mtu)      *
# origen-s binary library                                *
#          *
*****  
# ft2lf001 is the sas2 input origen-s binary working library  
#
# ft89f001 is the scale standard composition library  
#
# ft90f001 is the unit for the sas2h master interface  
#
# ft93f001 is the unit for couple input data generated by sas2h  
#
# ft94f001 is the unit for origen-s input data generated by sas2h  
#
# ft96f001 is the unit for bonami-s input data generated by sas2h  
#
# ft97f001 is the unit for nitawl-s input data generated by sas2h  
#
# ft98f001 is the unit for xsdrnpm-s input data generated by sas2h  
# $ORDATA/binrylib/pwr33gwd:  
#
# ft33f001 is the updated origen-s pwr binary working library with  
# data which includes updated cross-section data for all three  
# cycles for a burnup of 33 gwd/mtu -- generated by the sas2h run.  
#
# ft53f001 contains xsdrnpm-s scalar flux data used by sas2h  
#
# ft55f001 contains scale driver halt data for sas2h  
#
# ft70f001 is the unit for the sas2h auxiliary interface  
#
# $ORDATA/pwr33f71.saslinp:  
#
# ft71f001 contains nuclide concentrations and all other data  
# for the plorigen code and is made by origen-s during a sas2h run.  
# n o t e: ft71f001 is saved from this case because it is to be the  
# the input data set for photon and neutron source spectra to sample  
# cases for the sas1 control module of scale.
```

```
#  
# ft72f001 contains a card-image version of the data on ft71f001  
#  
# ft74f001 contains neutron and gamma sources generated during a  
# sas2h run.  
#  
# now store the new libraries and sasl source file:  
cp $TMPDIR/ft33f001 $ORDATA/binrylib/pwr33gwd  
cp $TMPDIR/ft71f001 $ORDATA/binrylib/pwr33f71.saslinp
```

19. scale4 スクリプトファイル

格納ディレクトリ : /local/scale/cmds

```

#!/bin/csh

# shell script to execute a scale job

set RTNDIR=`pwd`
set SCALE=/local/scale
setenv TMPDIR $SCALE/tmp
setenv DATA_DIR $SCALE/datalib
setenv DATA_BIN $SCALE/datalib/ampx/binrylib
setenv DATA_KENO $SCALE/datalib/keno/binrylib
setenv ORDATA_DIR $SCALE/datalib/origen
setenv PGM_DIR $SCALE/exe

if (!(-e $1)) then
    echo -----
    echo "input file you specified does not exist"
    echo -----
    exit
endif

if (!(-e $TMPDIR)) mkdir $TMPDIR
set tmpdir=$TMPDIR
cp $1 $tmpdir/sysin
cd $tmpdir

if (!(-e scale))    ln -s $PGM_DIR/scale   scale
if (!(-e csas))     ln -s $PGM_DIR/csas    csas
if (!(-e bonami))   ln -s $PGM_DIR/bonami  bonami
if (!(-e nitawl))   ln -s $PGM_DIR/nitawl  nitawl
if (!(-e xsdrn))    ln -s $PGM_DIR/xsdrn   xsdrn
if (!(-e htng72))   ln -s $PGM_DIR/htng72  htng72
if (!(-e heating))  ln -s $PGM_DIR/heating  heating
if (!(-e htas1))    ln -s $PGM_DIR/htas1   htas1
if (!(-e h7maprz))  ln -s $PGM_DIR/h7maprz h7maprz
if (!(-e ice))      ln -s $PGM_DIR/ice     ice
if (!(-e kenova))   ln -s $PGM_DIR/kenova  kenova
if (!(-e ocular))   ln -s $PGM_DIR/ocular  ocular
if (!(-e oculr72))  ln -s $PGM_DIR/oculr72 oculr72
if (!(-e modify))   ln -s $PGM_DIR/modify  modify
if (!(-e couple))   ln -s $PGM_DIR/couple  couple
if (!(-e origns))   ln -s $PGM_DIR/origns origns

```

```

if (!(-e o0o001)) ln -s $PGM_DIR/o0o001 o0o001
if (!(-e o0o002)) ln -s $PGM_DIR/o0o002 o0o002
if (!(-e o0o004)) ln -s $PGM_DIR/o0o004 o0o004
if (!(-e o0o005)) ln -s $PGM_DIR/o0o005 o0o005
if (!(-e o0o007)) ln -s $PGM_DIR/o0o007 o0o007
if (!(-e o0o008)) ln -s $PGM_DIR/o0o008 o0o008
if (!(-e o0o009)) ln -s $PGM_DIR/o0o009 o0o009
if (!(-e wax)) ln -s $PGM_DIR/wax wax
if (!(-e aim)) ln -s $PGM_DIR/aim aim
if (!(-e ale)) ln -s $PGM_DIR/ale ale
if (!(-e ajax)) ln -s $PGM_DIR/ajax ajax
if (!(-e lava)) ln -s $PGM_DIR/lava lava
if (!(-e rade)) ln -s $PGM_DIR/rade rade
if (!(-e perfume)) ln -s $PGM_DIR/perfume perfume
if (!(-e corectol)) ln -s $PGM_DIR/corectol corectol
if (!(-e compost)) ln -s $PGM_DIR/compost compost
if (!(-e picture)) ln -s $PGM_DIR/picture picture
if (!(-e morse)) ln -s $PGM_DIR/morse morse
if (!(-e o0o006)) ln -s $PGM_DIR/o0o006 o0o006
if (!(-e o0o106)) ln -s $PGM_DIR/o0o106 o0o106
if (!(-e sasl)) ln -s $PGM_DIR/sasl sasl
if (!(-e xsdose)) ln -s $PGM_DIR/xsdose xsdose
if (!(-e o0o102)) ln -s $PGM_DIR/o0o102 o0o102
if (!(-e sas2)) ln -s $PGM_DIR/sas2 sas2
if (!(-e sas3)) ln -s $PGM_DIR/sas3 sas3
if (!(-e sas4)) ln -s $PGM_DIR/sas4 sas4
if (!(-e alpo)) ln -s $PGM_DIR/alpo alpo
if (!(-e ft21f001)) ln -s $ORDATA_DIR/binrylib/prlimlwr ft21f001
if (!(-e ft22f001)) ln -s $ORDATA_DIR/binrylib/pwr33gwd ft22f001
if (!(-e ft23f001)) ln -s $ORDATA_DIR/binrylib/maphnbr ft23f001
if (!(-e ft24f001)) ln -s $ORDATA_DIR/binrylib/maphh2ob ft24f001
if (!(-e ft25f001)) ln -s $ORDATA_DIR/binrylib/baslmfb ft25f001
if (!(-e ft26f001)) ln -s $ORDATA_DIR/binrylib/maphuo2b ft26f001
if (!(-e ft27f001)) ln -s $ORDATA_DIR/cardlib/end6dec ft27f001
if (!(-e ft28f001)) ln -s $ORDATA_DIR/cardlib/xsectpho ft28f001
#if (!(-e ft70f001)) ln -s $DATA_DIR/scale.rev02.xn238 ft70f001
if (!(-e ft79f001)) ln -s $DATA_KENO/albedos ft79f001
if (!(-e ft80f001)) ln -s $DATA_KENO/weights ft80f001
if (!(-e ft81f001)) ln -s $DATA_BIN/scale.rev02.xn16 ft81f001

```

```
if (!(-e ft82f001)) ln -s $DATA_BIN/scale.rev03.xn27    ft82f001
if (!(-e ft83f001)) ln -s $DATA_BIN/scale.rev02.xn123    ft83f001
if (!(-e ft84f001)) ln -s $DATA_BIN/scale.rev03.xn218    ft84f001
if (!(-e ft85f001)) ln -s $DATA_BIN/scale.rev02.xn22g18  ft85f001
if (!(-e ft86f001)) ln -s $DATA_BIN/scale.rev02.xg18     ft86f001
if (!(-e ft87f001)) ln -s $DATA_BIN/scale.rev03.xn27burn ft87f001
if (!(-e ft88f001)) ln -s $DATA_BIN/scale.rev03.xn27g18  ft88f001
if (!(-e ft89f001)) ln -s $DATA_BIN/scale.rev04.sclib    ft89f001
if (!(-e h7matlib)) ln -s $DATA_DIR/h7matlib           h7matlib
if (!(-e qatable))  ln -s $DATA_DIR/qatable            qatable
if (!(-e aliases)) ln -s $DATA_DIR/aliases             aliases
rm print _prt* _out*
scale
cd $RTNDIR
cat $tmpdir/print > $2
cat $tmpdir/_prt* >> $2
cat $tmpdir/_out* >> $2
#
#rm -r $SCALE/tmp
#
```