Overview of MELCOR Code Development

Larry L. Humphries, Randall Gauntt (SNL)

and

Hossein Esmaili (NRC)

Second European MELCOR User Group

Prague, Czech Republic

March 1-2, 2010

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000.

Sandia National Laboratories

MELCOR Code Development

- **MELCOR is developed by:**
  - **US Nuclear Regulatory Commission**
  - **Office of Nuclear Regulatory Research**
  - **Division of Safety Analysis (DSA)**
- **MELCOR development is also strongly influenced by the participation of many International Partners through the US NRC Cooperative Severe Accident Research Program (CSARP and MCAP)**
  - **Development Contributions – New models**
  - **Development Recommendations**
  - **Validation**

Current MELCOR Development

MELCOR 1.8.6 Branch

- **MELCOR 1.8.6**
  - **Molten pool models**
  - **Core Package upgrade**
  - **Released Fall 2005**
  - **Code Maintenance Only (no new modeling)**
  - **Current Workhorse**
  - **Maintenance release (Jan 2010)**

MELCOR 2.X Branch

- **Current Version MELCOR 2.1**
  - **Source code in FORTRAN 95**
  - **New input format**
  - **All new development**
  - **2.0 beta version released Sept 2006**
  - **2.1 Release Sept 2008**
  - **2.1 Maintenance release (Jan 2010)**

MELCOR 3.0 Code Development Thrust Areas

Code reliability

- Validation
- QA
- Numerical stability

MELCOR Code Development

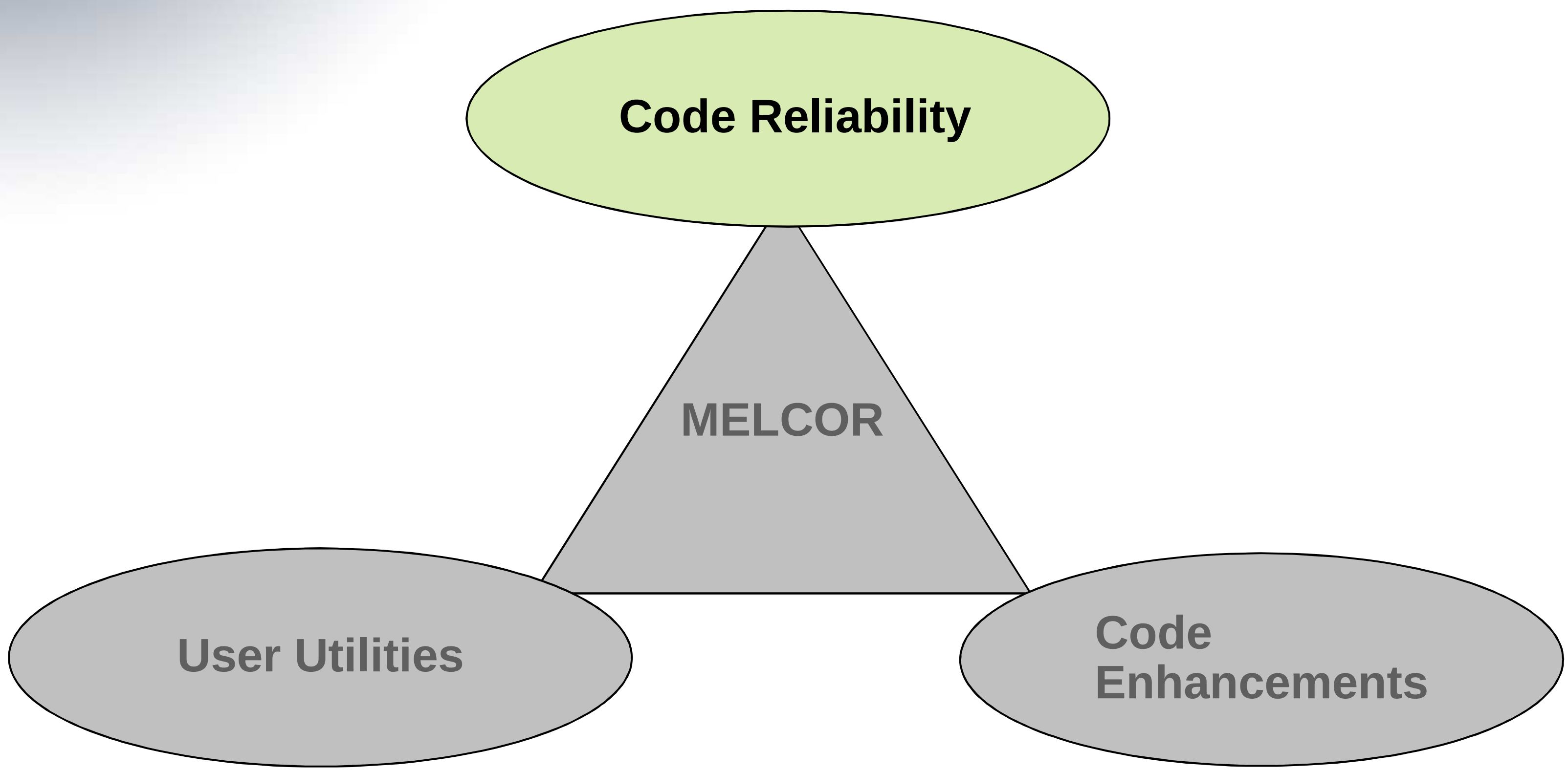
User Utilities

- Converter
- PTFREAD
- SNAP
- Uncertainty Engine

Code Enhancements

- New/improved modeling
- Code performance

Code Development: Code Reliability



Intel Visual FORTRAN Compiler

- Intel Visual FORTRAN has replaced Compaq Visual FORTRAN as our standard development platform
  - Did not want to make any changes until after 2.1 was released
  - Technical support for CVF no longer available
  - Problems with CVF rebuilding entire project
  - Problems with error checking
- Advantages to Intel Compiler
  - Able to build true 64-bit code for 64-bit operating systems
    - Performance improvements
  - Better support for F95 code
    - Improved error checking
  - More consistent support for Linux
  - Improvements to the programmer interface (Visual Studio.NET)
    - Automatic keyword completion
    - Integration with subversion
  - Capability to automatically convert CVF projects

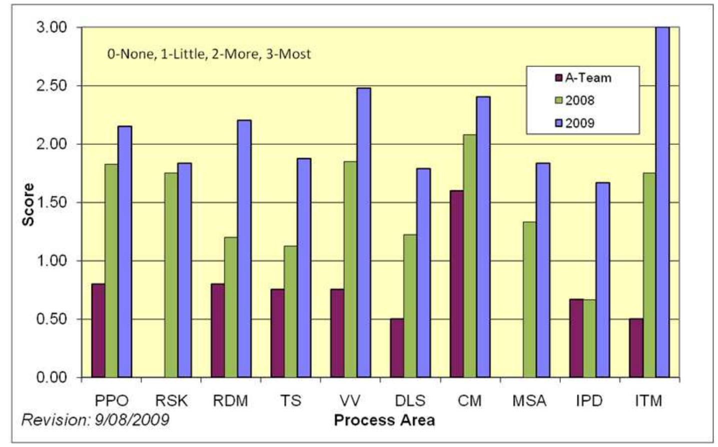
Software Quality Assurance

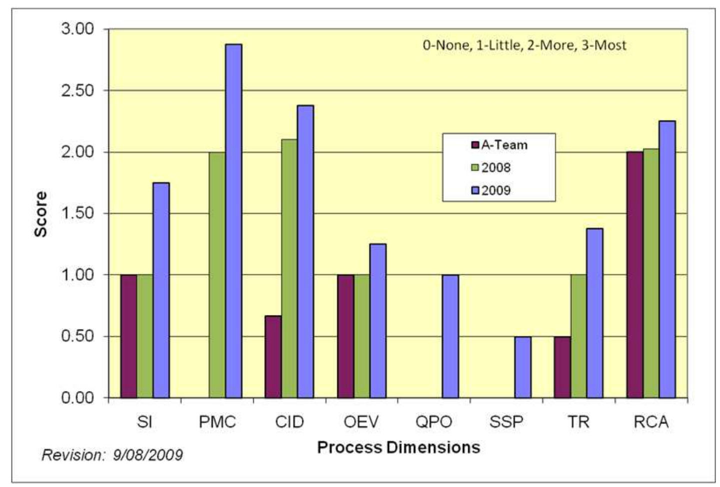
- **Sandia Corporate Process Requirement 001.3.6 (CPR 001.3.6)**
- **The software management framework adapted from two internationally recognized standards**
  - **the Capability Maturity Model Integration (CMMI) ®**
  - **and ISO 9001**
  - **These standards provide elements of traceability, repeatability, visibility, accountability, roles and responsibilities, and objective evaluation**

- **Areas showing improvement**
  - **Code Review (Code Collaborator)**
  - **Requirements Management**
  - **Configuration Management (Automated Build & Test)**
  - **Further Development of Internal Wiki**
  - **Development of SQA training materials**
- **Areas needing improvement**
  - **Improve debugging response time**
  - **Improve user access to documentation updates**
  - **Explore possibility of external Users' Wiki**
- **Sandia's commitment to SQA**

Software Quality Assurance Annual Re-evaluation

- **Process areas**
  - **Project planning and oversight, PPO**
  - **Risk Management, RSK**
  - **Requirements Development and Management, RDM**
  - **Technical Solution, TS**
  - **Verification and Validation, VV**
  - **Development and Lifecycle Support, DLS**
  - **Configuration Management, CM**
  - **Measurement and Analysis, MSA**
  - **Integrated Product, IPD**
  - **Integrated Teaming, ITM**
- **Process Dimensions**
  - **Stakeholder Involvement , SI**
  - **Ongoing Process Monitoring and Control, PMC**
  - **Collected Improvement Data, CD**
  - **Objective Evaluations, OEV**
  - **Quantitative Objectives Defined for Processes, QPO**
  - **Stable Subprocess Performance, SSP**
  - **Training , TR**
  - **Problem Reporting & Corrective Action, RCA**

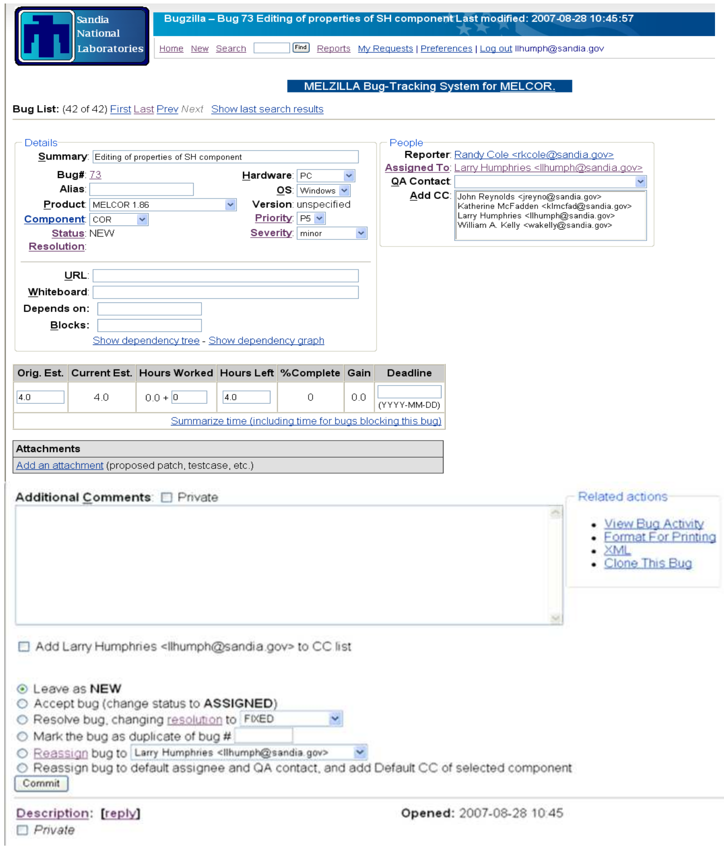




Bug Statistics

MELCOR 1.8.6 (since 7/07)	
Number of Bugs Reported	105
Number of Bugs Resolved:	72
Number of Bugs Unresolved:	33
Number of Bugs Reported (this year):	36
Number of Bugs Resolved (this year):	27
Number of Bugs Unresolved (this year):	9
Number of Unresolved User Bugs :	15

MELCOR 2.1 (since 9/07)	
Number of Bugs Reported	138
Number of Bugs Resolved:	98
Number of Bugs Unresolved:	40
Number of Bugs Reported (this year):	84
Number of Bugs Resolved (this year):	77
Number of Bugs Unresolved (this year):	7
Number of Unresolved User Bugs :	15



MELCOR Code Testing

- **Build Testing**
  - **Automated to perform Nightly Builds**
    - **MELCOR 1.8.6 Windows Compaq Visual FORTRAN (CVF)**
    - **MELCOR 2.X Windows CVF**
    - **MELCOR 2.X Windows Intel Visual FORTRAN (IVF)**
    - **MELCOR 1.8.6 Linux IVF**
    - **MELCOR 2.X Linux IVF**
  - **Using CMAKE to generate Makefiles for use on Unix variants to extend building on other platforms**
- **Code Testing**
  - **Performed Daily (at least frequently)**
    - **Dedicated machine for performing auto testing to**
  - **Standard test cases chosen for physics coverage ~14 test cases**
    - **New cases will be added as validation calculations are run**
    - **Debug & optimized versions tested**
    - **Unix versions not tested as frequently (will test more frequently in future)**
  - **Comparison of results**
    - **Windows system**
      - **Latest 1.8.6 (CVF) and YT, 2.X(CVF)**
      - **Latest 2.X (CVF) and Latest 1.8.6 (CVF)**
      - **Latest 2.X (IVF) and latest 2.X (CVF)**
    - **Linux System (not yet realized)**
      - **Latest 2.X (IVF Windows) Latest 2.x (IVF Linux)**
- **Optimization Testing (not yet realized)**
  - **Run automated optimization studies to determine effects of optimization**
    - **Many routines can be optimized with no impact on answers**
    - **Order of arithmetic will mean some routines should not be optimized**

Case	BUR	CAV	CF	COR	CVH	DCH	FCL	FDI	FL	HS	NCG	PAR	RN	SPR
M-8-1_NoMix			X		X				X	X	X			
M-8-1_SYM			X		X				X	X	X			
Lace7			X		X	X			X	X	X		X	
Lace8			X		X	X			X	X	X		X	
Vanam-M3			X		X				X	X	X		X	
Molten Salt			X	X	X				X	X	X			
PHEBUS-B9			X	X	X				X	X	X			
FPT1			X	X	X	X			X	X	X		X	
LOFT			X	X	X	X			X	X	X			
Test_lnew	X	X	X	X	X	X	X	X	X	X	X	X	X	X
SURRY (LBLOCA)	X	X	X	X	X	X	X	X	X	X	X		X	X
Zion (SBO)		X	X	X	X	X	X	X	X	X	X	X	X	X
PeachBottom (SBO)	X	X	X	X	X	X			X	X	X		X	X
Grand Gulf (SBO)	X	X	X	X	X	X		X	X	X	X		X	

Table 1-1: Physics Package Coverage

Automated Nightly Build and Test

Navigation

MELCOR Web Site
MELZILLA Web Site

External MELCOR Wiki Site
Build & Test History

Build Report Tables

- MELCOR 2.X
- MELCOR 1.8.6

Regression Reports

- M186
- M2.X(CVF)
- M2.X(IVF)

MELCOR Build and Test Report

Current Status: 9/5/2009

MELCOR 2.X

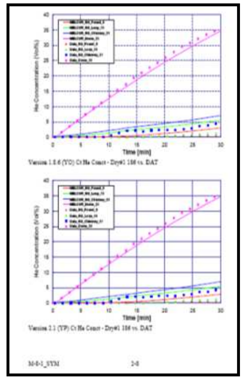
MELCOR 2.1 Automated Build History

Revision	Submitter	Date	Time	Compiler	Build Status	Test Status	Compiler2	Build Status	Test Status
1244	jjoonyu	9/4/2009	9:20:20 AM	IVF 11.0\072	Successful	Submitted	CVF 6.6C	Successful	Submitted
124			8:25:44 AM	IVF 11.0\072	Failed	Unk	CVF 6.6C	Failed	Unk
124			1:15:16 PM	IVF 11.0\072	Failed	Unk	CVF 6.6C	Failed	Unk
1241	jjoonyu	9/2/2009	4:58:07 PM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1239	jreyno	9/2/2009	2:02:46 PM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1238	jreyno	9/2/2009	1:14:03 PM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1236	llhumph	9/1/2009	3:41:57 PM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1235	llhumph	9/1/2009	1:59:16 PM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1234	llhumph	9/1/2009	12:56:54 PM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1233	llhumph	9/1/2009	8:00:10 AM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1232	llhumph	9/1/2009	7:50:36 AM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1231	llhumph	9/1/2009	7:13:29 AM	IVF 11.0\072	Successful	Unk	CVF 6.6C	Successful	Unk
1229	nastya	9/1/2009	5:33:15 AM	IVF 11.0\072	Failed	Unk	CVF 6.6C	Successful	Unk
1228	nastya	9/1/2009	5:31:39 AM	IVF 11.0\072	Failed	Unk	CVF 6.6C	Successful	Unk
1226	nastya	9/1/2009	1:54:37 AM	IVF 11.0\072	Failed	Unk	CVF 6.6C	Successful	Unk

Technical Details: Correct function usage
User Impact: NA
Unit test (include SVN link):

MELCOR Regression Report Page

- **Automated Regression Report Web page**
  - **Links to regression report with side-by side comparisons of trend plots**
  - **Table of variables sorted by variance**



Navigation

MELCOR Web Site
MELZILLA Web Site
External MELCOR Wiki Site
Build & Test History
Regression Reports
- M186
- M2.X(CVF)
- M2.X(IVF)

MELCOR Regression Report Page

MELCOR 2.1 (IVF compiler) Regression Results

Regression Reports

Following is a list of regression reports currently on file for MELCOR 2.1 (IVF compiler). The regression report for MELCOR 2.1 IVF represents regression results between the current test revision for the Intel FORTRAN code (IVF) against the most recent developmental version of MELCOR 2.1 compiled with the COMPAQ compiler (CVF).

- Autotest for M2.1 IVF Rev 1033.pdf
- Autotest for M2.1 IVF Rev 1040.pdf
- Autotest for M2.1 IVF Rev 1045.pdf
- Autotest for M2.1 IVF Rev 1056.pdf
- Autotest for M2.1 IVF Rev 1078.pdf
- Autotest for M2.1 IVF Rev 1093.pdf

Current Regression Test Summary (IVF 1093, CVF 1093)

The following table summarizes regression results by exposing and categorizing discrepancies for a select set of problem variables. The variance is calculated as the difference in the values calculated for the variables at the end of calculation for calculations performed with two executable versions. Time histories for all variables throughout the calculation for both executable versions are plotted side by side in the detailed regression reports (see above).

This does not represent a complete list of variables generated by MELCOR but only a subset of calculated variables that were selected to characterize the calculation or particular models that are represented by this test case.

Variables are sorted according to the calculated variance. Actual values calculated at the end of the calculation for each of these variables can be obtained by hovering the cursor over the variable name in the table. Generally, variances under 5% are considered acceptable and variances greater 25% represent a significant variation that must be explained. It is possible that modeling changes can be responsible for such variances.

This table was generated for a particular code SVN revision.

Regression Test Summary Table

Test Case	<1%	1% to 5%	5% to 10%	10% to 25%	25% to 50%	>50%
Falcon1.xls	CVH-TVAP.35	-	-	-	-	-
-	CVH-PPART.3.70	-	-	-	-	-
-	-	-	-	-	-	-
lace7.xls	CFVALU.2	-	-	-	-	-
-	-	-	-	-	-	-
LOFT_Plots.xls	CVH-P.210	COR-TCL.106	-	-	-	-
-	CFVALU.100	COR-DMH2-ZIRC	-	-	-	-
-	CVH-TLIQ.130	COR-DMH2-TOT	-	-	-	-
-	CVH-TSAT(P).130	-	-	-	-	-
-	COR-TCL.109	-	-	-	-	-
-	COR-TCL.112	-	-	-	-	-
-	COR-TCL.206	-	-	-	-	-
-	COR-TCL.207	-	-	-	-	-
-	-	-	-	-	-	-
NUPEC M-8-1_NoMix.xls	CVH-TVAP.8	-	-	-	-	-
-	CVH-TVAP.15	-	-	-	-	-
-	CVH-TVAP.21	-	-	-	-	-
-	CVH-TVAP.25	-	-	-	-	-
-	CFVALU.215	-	-	-		
-	CFVALU.221	-	-	-		
-	CFVALU.225	-	-	-		
-	-	-	-	-		
NUPEC M-8-1.xls	CVH-TVAP.8	-	-	-		
-	CVH-TVAP.15	-	-	-		
-	CVH-TVAP.21	-	-	-		
-	CFVALU.208	-	-	-		
-	CFVALU.221	-	-	-		
-	CFVALU.225	-	-	-		
-	-	-	-	-		
PHEBUS_B9+(LLH).xls	COR-TFU.307	-	-	-		
-	COR-TSVC.305	-	-	-		
-	COR-TSVC.205	-	-	-		
-	COR-ENERGYERR	-	-	-		
-	COR-ENERGY-TOT	-	-	-		
-	COR-ECNV-TOT	-	-	-		
-	FL-I-MFLOW.5.112	-	-	-		
-	FL-I-MFLOW.5.123	-	-	-		

MELCOR Code Regression Test Report

Autotest for M2.1 Rev 1093

Sandia National Laboratories

6/23/2009

This report compares results from the MELCOR regression tests for Autotest for M2.1 Rev 1093 against results obtained from Autotest for M2.1 IVF_Rev 1093, using the same test suite. For each member of the test suite, significant variables characterizing the calculation have been plotted for both variants and displayed side-by-side on the same page. These calculations were performed at SNL as part of our general V&V practices and were chosen to represent a wide, though not exhaustive, range of reactor safety phenomenon. Many of these cases are simulations of both separate effects tests as well as integral tests and many are

Changes to Documentation Published With Released Code Version

- **Word documents stored locally in Subversion Repository**
  - **Documents available to all developers for modification**
  - **Word compare is readily used to 'diff' two revisions of a document**
- **PDF Generated from Working versions of Word Documents**
  - **Marked as draft version and revision numbers clearly marked in document.**
  - **SVN revision log is inserted after title page.**
  - **All modified content is highlighted in yellow**
  - **Style & formatting changes ignored.**
  - **All user font coloring is disabled.**

Working Version (Dec/29/2009)
Revision 1237 - Should not be cited

Note: All modifications and additions are highlighted in yellow

MELCOR Computer

Code Manuals

Summary of all Revisions to Documents since Rev 646

Rev/Auth	Date	Comment	Files
1543 llhumph	2010-01-05	Changed TOC formatting	CVH_UG_M2
1509 rkcole	2009-12-16	Document SC4407(13), SVRMAX for bubble separation, and the extension to SC4407(1) for NQE	CVH_UG_M2
1505 rkcole	2009-12-16	Revise and clarify description of FORMULA CF and input requirements	CF_UG_M2
1442 llhumph	2009-11-24	Clarification of default behavior when the RN1_input record is absent.	RN_UG_M2
1439 jjoonyu	2009-11-24	Enhancement for bug 495. RN1 INPUT default value changed to 0 or Active to reflect the codes treatment of: RN1_INPUT == RN1_INPUT 0	RN_UG_M2
1423	2009-	added COR_RFA record, fixed blank in COR_ RFD	COR_UG_M2

4408 – Pressure Iteration Parameters

Several submodels in the flow solution may be turned off if the user so desires. This capability is intended primarily for use in debugging by the code developers. The first element of the sensitivity coefficient array, C4408(1), entered as a single real number, is interpreted as six packed decimal digits, each controlling a single model. Setting any digit $\neq 0$ disables the corresponding model. Any or all of the digits may be nonzero as desired.

Setting the 1,000,000s digit $\neq 0$ disables the revised treatment of non-equilibrium thermodynamics that prevents the collapse of a boiled-up pool in a volume with a very small atmosphere and the possible revaporization of a very small pool. Here "very small" is defined by VPHEQL, element 5 of sensitivity coefficient array 4411. The resulting behavior corresponds to the model (a switch to equilibrium thermodynamics) used in previous MELCOR versions through early releases of 1.8.6.

Extended Code Pedigree Information

- **MELCOR has always provided limited pedigree information**
  - **Available in output and on console**

    MELCOR BASE CODE VERSION
    2.1 DEC-16-2009
    DEVELOPMENT VERSION
    $Revision: 1660 $

- **MELCOR 2.1 now provides extended information**
  - **Includes compiler version, libraries used, compiler settings, and linker settings**
  - **Lists any files that were not optimized**
  - **Printed in output and also available from executable using command line**
    - **-CONFIG (command line argument)**

MELCOR BASE CODE VERSION
2.1 DEC-10-2009

DEVELOPMENT VERSION

BASE CODE YU $Revision: 1490 $

Configuation information for MELCOR

Subversion revision: 1495

Compiler: IVF 11.1\051

Compiler settings

/nologo /O2 /Oy- /fpp /I'C:\Program Files (x86)\Intel\Compiler\11.1\051\fortran
\include\ia32' /DLICENSING_INACTIVE /DWINDOWS /recursive /real_size:64 /Qauto /align:rec8byte /Qtrapuv /iface:cvf /module:'' /traceback /fp:source /Qprec-div /Qprec-sqrt /assume:protect-parens /check:none /libs:static /threads /dbglibs /c

Linker settings:

/Version:2.1 /INCREMENTAL:NO /MANIFEST /NOLOGO /SUBSYSTEM:CONSOLE /TSAWARE:NO kernel32.lib Ws2_32.lib user32.lib mkl_solver_sequential.lib mkl_intel_c.lib mkl_sequential.lib mkl_core.lib

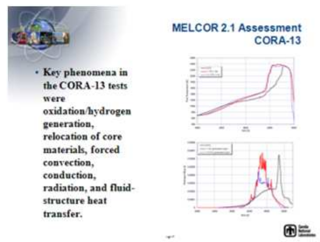
Files that were not optimized:

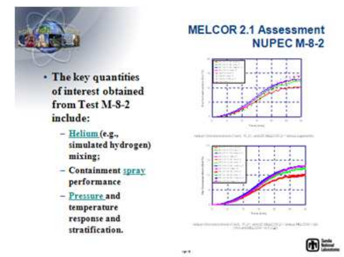
CORFZO, CORFZP, MXXDBD, POOL3_NSI, RES_NSI, CAVENR_NSI,

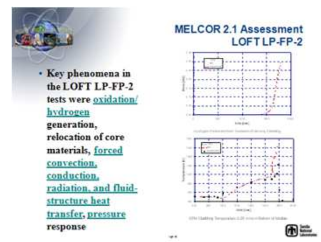
!!!LICENSING HAS BEEN DISABLED!!!

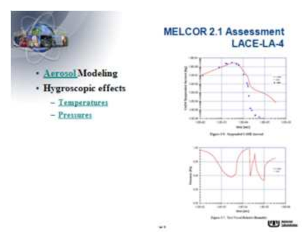
Continuous Assessments

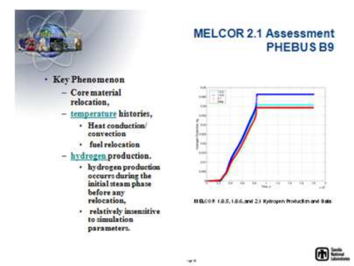
- **Continuous reassessment of many ISPs to assure consistency between code versions and effect of code improvements**
  - **LOFT**
  - **Cora-13**
  - **Phebus-B9**
  - **Lace**
  - **Vanam**
  - **SURC-2**
  - **DEHBI**
  - **NUPEC (M-8-1 & M-8-2)**
  - **Falcon (FAL-ISP 1)**
  - **LOFT (LP-FP-2)**





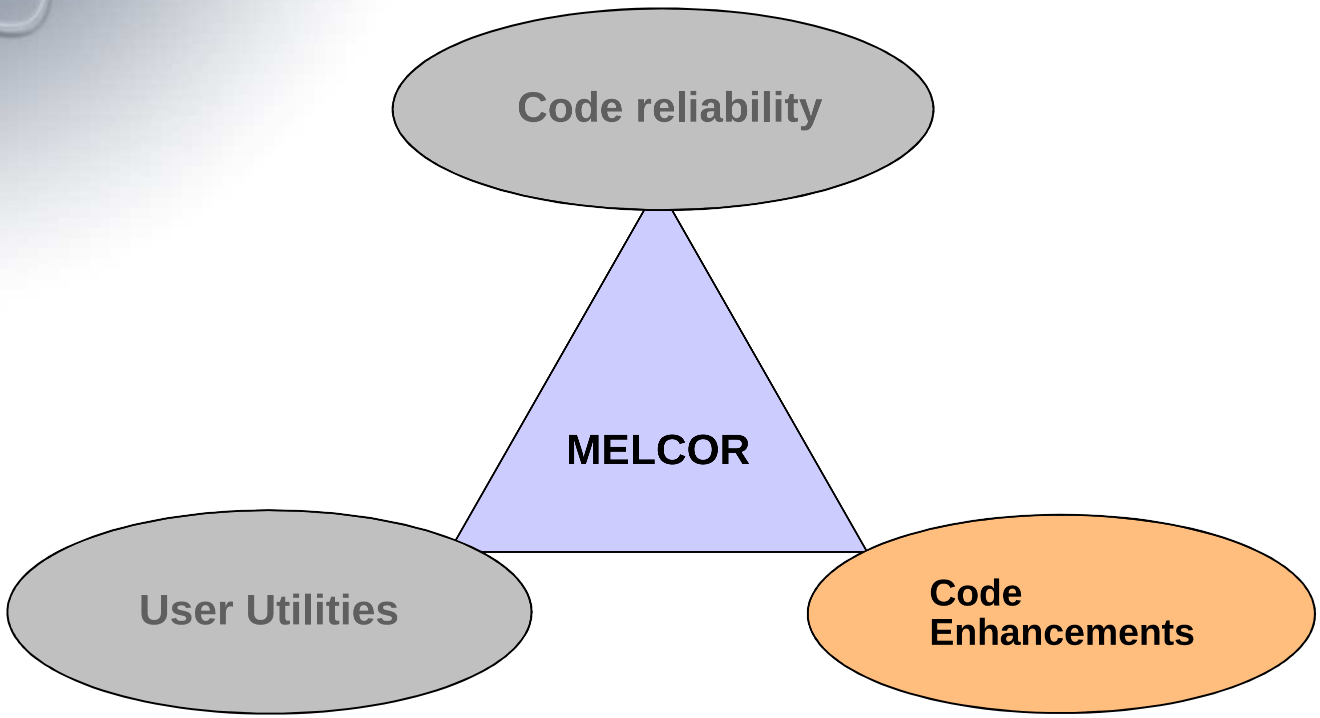






- **Additional assessments to be performed.**
  - **Intel allows us to test code coverage**
  - **Traceability matrix to enhance our code coverage**

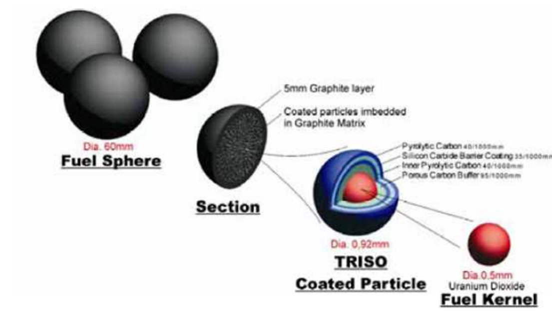
Code Development: Code Enhancements

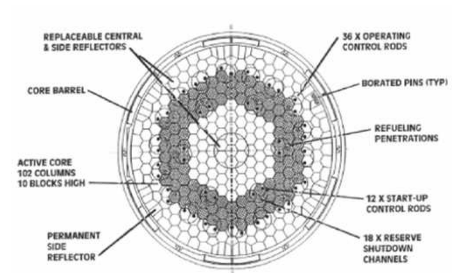


- **All new code development will be performed in MELCOR 2.1**
- **MELCOR 1.8.6 – Code Maintenance only**

Code Development
HTGR - COR Package Updates

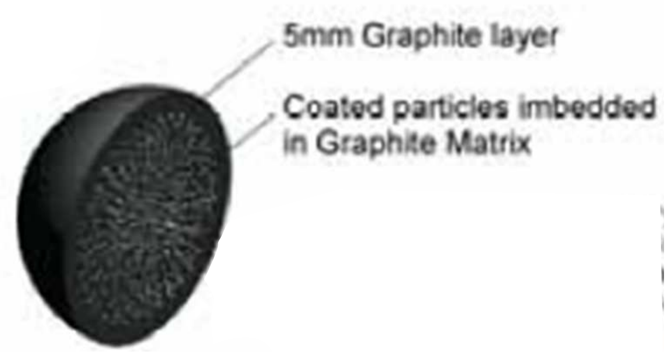
- **New reactor models (like present PWR, BWR)**
  - **PBR (pebble bed)**
    - **New components**
      - **Pebble fuel**
    - **New heat transfer coefficients**
    - **Effective thermal conductivity**
      - **radiation/convection/conduction**
    - **Coolant friction loss**
  - **PBR (prismatic)**
    - **New components**
      - **Reflectors**
      - **Hexagonal graphite blocks**
    - **cell-cell conductive/radiative heat transfer**
- **Graphite oxidation models**
- **Point Kinetics**
- **On-going work**
  - **Fission product release and transport from HTGR fuel**
  - **Plant demonstration calculations**





PBR COR Model

- **Pebble represented using fuel and clad components**
  - **Fuel radial temperature profile for sphere**
    - **Provides peak and surface pebble temperature**
  - **Fueled part of pebble (*Coated particles embedded in Graphite Matrix*) is fuel component**
    - **UO_2 with extra COR material as graphite**
  - **Unfueled shell (*5mmGraphite layer*) is clad component**
    - **Clad material becomes graphite**
  - **Example Input:**
    - `! graphite pebble`
    - `!        RFUEL RCLAD DRGAP PITCH`
    - `COR_GP   0.025 0.03  0.0   0.0143`

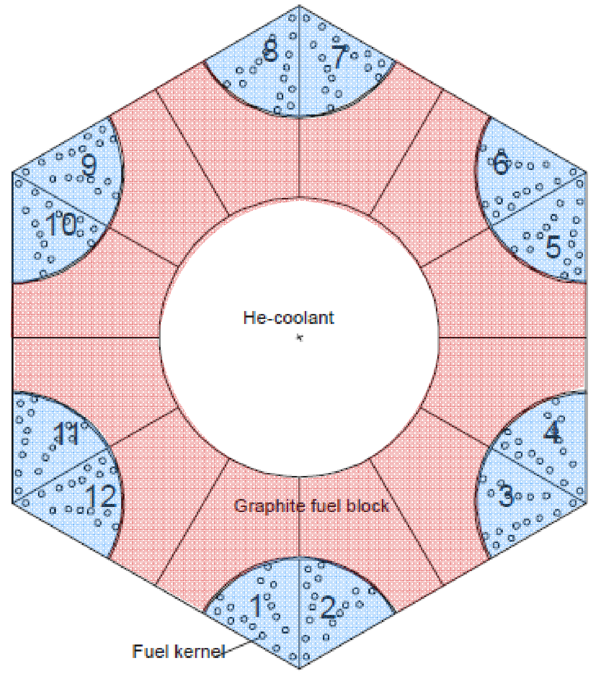


PBR COR Model contd

- Radial cell-cell conductive/radiative heat transfer added
  - Uses effective bed conductivity (Zehner-Schlunder with thermal radiation)
  - New input record: COR_TKE (optional for PBR)
- Axial cell-cell conduction modified to use effective bed conductivity
- Coolant friction factor is for pebble bed with PBR model
  - Ergun correlation is default
  - KTA Rule (Achenbach) correlation can be used by changing SC1413
  - Pebble bed friction factor specified with options PBR-A or PBR-R on COR_BLK record
- Coolant heat transfer uses pebble bed heat transfer coefficients
  - KTA Rule correlation can be entered with SC1231

PMR COR Models

- More "rod-like", requires fewer changes to COR
- Fuel compacts represented as fuel component (blue)
- Part of hex block associated with a fuel channel is "thick" clad component (red)
  - Temperature gradient in clad as well as fuel
  - Important change is inclusion of "clad" resistance in HTC
    - Can have significant effect on heat transfer to coolant
- Reflectors are new Reflector component

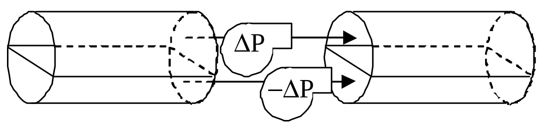


PMR Models

- **Radial cell-cell conductive/radiative heat transfer used, specified on COR_TKE record**
  - **TKECF – optional temperature-dep TF, usually ‘NO’**
  - **PORCHAN – effective block porosity for Tanaka-Chisaka conductivity model (see COR RM)**
    - **“Porosity” is something like (coolant channel volume per block)/(block volume)**
    - **Conductivity in “porosity” is coolant conductivity**
  - **DBLK –block characteristic length**
  - **BLKGAP – gap width between blocks**
    - **Conductance across gap using coolant conductivity and radiation**

Counter-Current Stratified Flow Modeling

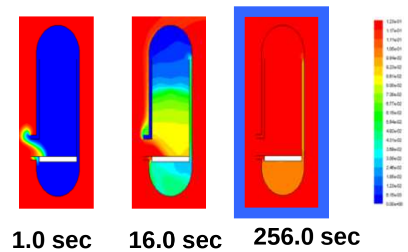
- **Counter-Current Stratified Flow Applications**
  - **Developed for HTGR air-ingress application**
  - **Modeling of natural circulation between the steam generator and the upper plenum of PWRs**
  - **Containment modeling**
- **Old concept used in MELCOR for many years**
  - **Split path in two to allow counter-current flow**
  - **Account for momentum exchange in flow equations**
    - **Implement as "pumps" with ΔP calculated from relative velocities**

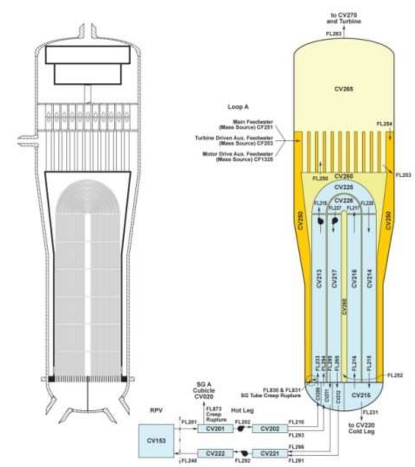


$$\Delta P = -\frac{2fL}{D_h}\bar{\rho}(v_1 - v_2)|v_1 - v_2| = -C\bar{\rho}(v_1 - v_2)|v_1 - v_2|$$

- **New approach where ΔP is "internalized"**
  - **We have coded an appropriate ΔP calculation**
    - **Specific input was added to couple two flow paths**
    - **Form of ΔP has been generalized to better match published correlations**
    - **Terms added directly to flow equations**
      - **Increased stability because of implicit numerics**

Snap Shots of Simulation (Air Mole Fraction) from "NRC/INL Meeting on Methods for VHTRs", E. Kim, C. Oh, R. Schultz, INL (2008)

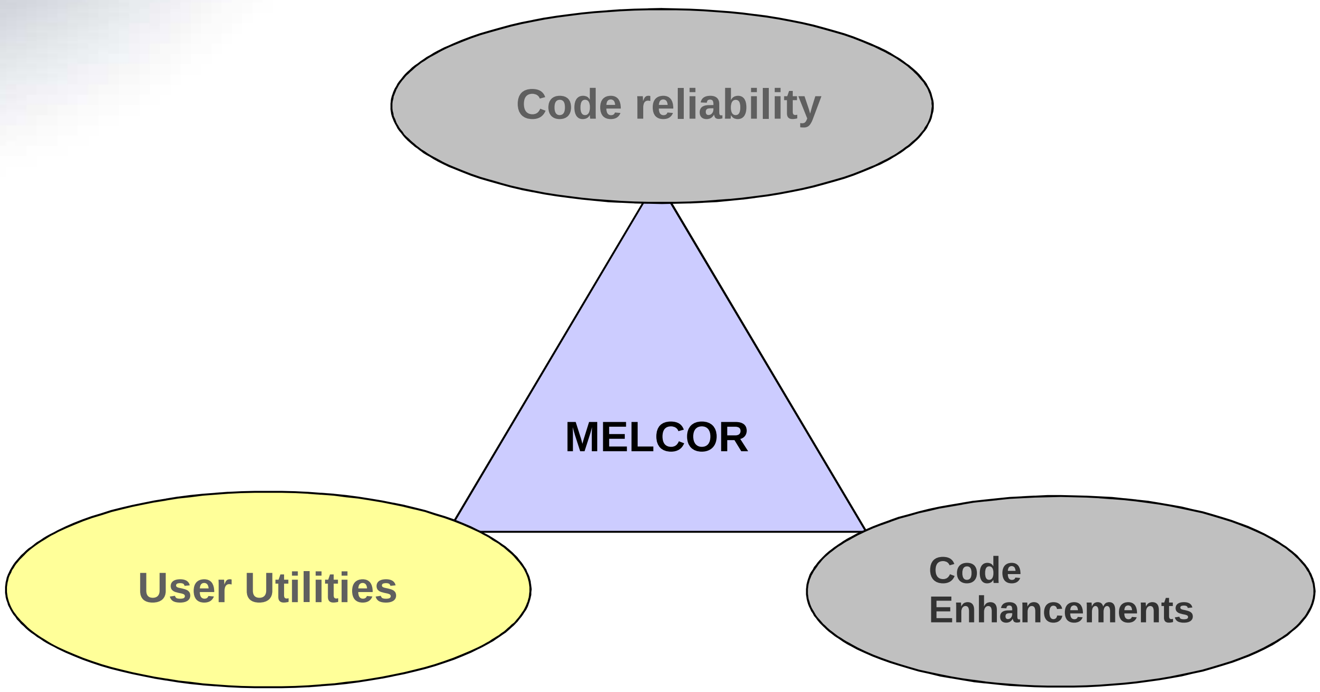




Ongoing or Recently Completed Code Model Development

- **Accumulator Model**
  - Replaces current control function approach to improve numerics and reduce errors and modeling effort of code user
- **Use a model to track activities.**
  - Existing MELCOR output doesn't allow tracking of radionuclides and user has to perform calculations to estimate activities.
- **Heat and Mass Transfer Correlations**
  - MELCOR code should be able to model the CONTAIN correlations by default
  - Modify the MELCOR film tracking model and default model parameters based on the CONTAIN parity
- **Engineered Safety Features (ESF) Enhancements**
  - heat exchanger models
  - fan cooler models
- **Improvement of SPARC Models**
  - Review the SPARC98 model for possible improvements over the earlier SPARC90 model
- **Improvement of VANESA Models**
  - improvements for ex-vessel fission product release. Specifically, the modeling of Ru and Mo)releases
- Others…

Code Development: User Utilities



NUCLEAR ENERGY & GLOBAL SECURITY TECHNOLOGIES

Improvements to Converted Input

September 2008

- HS_ID 'HS31014-UP' 31014 ! 186 name: UP
- …
- !* 103 - Next HS data ***
- HS_ID 'HS32001-LOWERPLE' 32001 ! 186 name: LOWERPLENUM
- HS_GD CYLINDRICAL YES ! Type of geometry, Steady-state initialization
- HS_EOD 0.1445 1.0 ! HS Elevation and Orientation Data
- HS_SRC NO ! No power source
- HS_ND 4 ! NXVALU NI XVALUE TEMPIN MATNAM
- 1 1 0.22403000E+01 0.56083727E+03 'STAINLESS-STEEL'
- 2 2 0.22874000E+01 0.55612846E+03 'STAINLESS-STEEL'
- 3 3 0.23345000E+01 0.55149002E+03 'STAINLESS-STEEL'
- 4 4 0.23816000E+01 0.54692239E+03
- ! Type CV name Mass transfer
- HS_LB CalcCoefHS 'CV320-LOWER-PLEN' YES
- HS_LBP INT 0.50000000E+00 0.50000000E+00
- HS_LBT 0.10000000E+03 0.10000000E+03 0.10000000E+01 0.10000000E+01
- HS_LBS 1.0 0.701 0.5230
- ! Type CF/TF CV name Mass transfer
- HS_RB CoefTempTF 'TF10-PIPE INSULA' 'CV1-CAVITY' YES
- HS_RBP EXT 0.50000000E+00 0.50000000E+00
- HS_RBT 0.10000000E+03 0.10000000E+03 0.10000000E+01 0.10000000E+01
- HS_RBS 7.8262 1.0 0.5230
- HS_FT OFF
- …
- HS_ID 'HS32002-LOWERPLE' 32002 ! 186 name: LOWERPLENUM

- Records positioned as in original input
- Steady State temperatures to be calculated
- More readable formatting

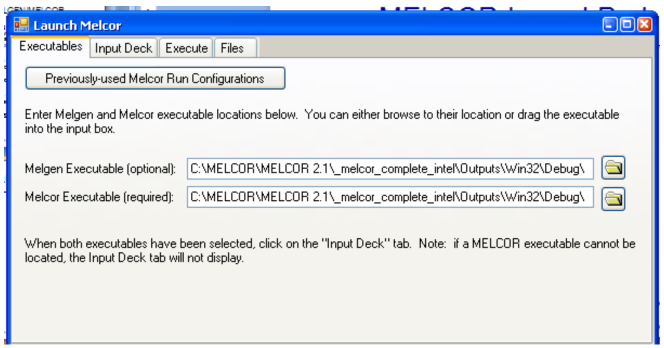
September 2009

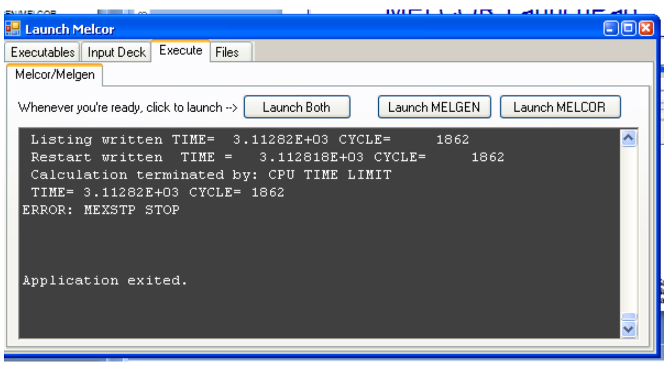
- ! * Lower Vessel Wall - Core level 1
- ! *
- ! * Description from S/R5 Model HS 5062
- ! * Inner Radius = 2.2403 m
- ! * Outer Radius = 2.3816 m
- ! * Inner Dhyd = 0.701 m
- ! * DZ = 0.523
- ! *
- ! * NODES CYL INPUT W4(I)
- HS_ID 'HS32001-LOWERPLE' 32001 ! 186 name: LOWERPLENUM
- HS_GD CYLINDRICAL SS ! Type of geometry, Steady-stat initialization
- HS_EOD 0.1445 1.0 ! HS Elevation and Orientation Data
- HS_SRC NO ! No power source
- HS_ND 4 ! NXVALU NI XVALUE TEMPIN MATNAM
- 1 1 2.2403 - 'STAINLESS-STEEL'
- 2 2 2.2874 - 'STAINLESS-STEEL'
- 3 3 2.3345 - 'STAINLESS-STEEL'
- 4 4 2.3816 -
- HS_LB CalcCoefHS 'CV320-LOWER-PLEN' YES
- HS_LBP INT 5.E-01 5.E-01
- HS_LBT 100.0 100.0 1.0 1.0
- HS_LBS 1.0 0.701 0.5230
- HS_RB CoefTempTF 'TF10-PIPE INSULA' 'CV1-CAVITY' YES
- HS_RBP EXT 5.E-01 5.E-01
- HS_RBT 100.0 100.0 1.0 1.0
- HS_RBS 7.8262 1.0 0.5230
- HS_FT OFF

- Comments from original input

MELCOR LaunchPad

- **Input files can be selected from multiple methods**
  - **Directly typed or pasted**
  - **Navigation through file manager**
  - **‘Drag and Drop’**
  - **Selected from recent history**
- **Executables can be selected in a similar manner**
- **Execution of MELGEN, MELCOR, or the Converter from one pane**
- **MELGEN/MELCOR input/output files can be accessed from hyperlinks**

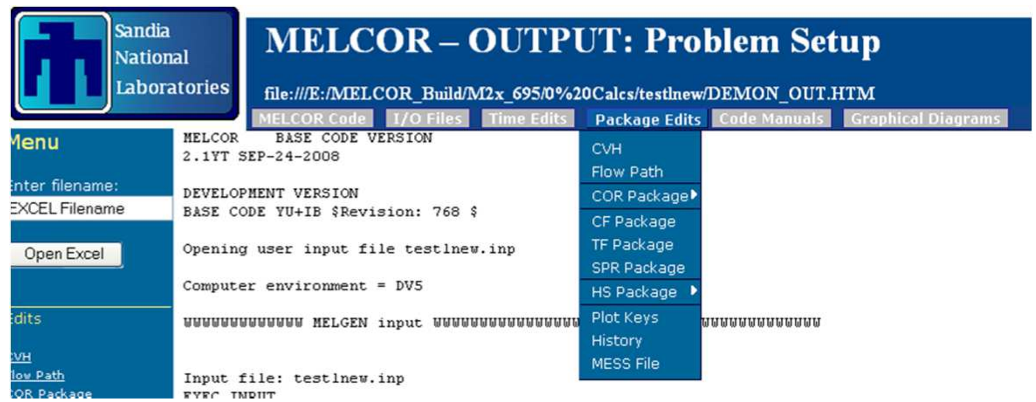


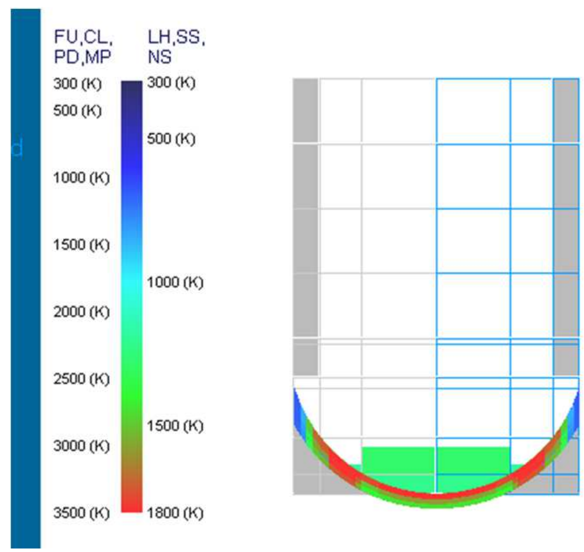


Sandia National Laboratories

New MELCOR Output Format HTML

- **HTML Time Edits**
  - Specified with global input
    - MEL_HTMLFILE ‘DEMON_Out.htm’
  - File for each time edit
  - Links to other time edits
  - Links to package edits/tables
- **Other Links**
  - SNL/Bugzilla
  - I/O files
  - Code Manuals
  - Graphical Diagrams
    - Node Diagram
    - Temperature contours





Named Comment Blocks (1)

- **This feature was added to allow the user to include blocks of input records that can be 'activated' or skipped by a variety of input methods.**
- **User can encapsulate multiple versions of a test case within a single input deck.**
  - **Sensitivity calculations**
  - **Variations for a standard plant deck**
- **Analogous to the *EOR* capability for MELCOR 1.8.6**
  - **More flexible and powerful to use**

Named Comment Blocks (2)

- MELCOR comment blocks were added as part of the code conversion project. In MELCOR 2.1, a block of input can be commented out by enclosing that block of input within a set of triple parenthesis.

```
(((
These input records are ignored
)))
```

- Named comment blocks are either ignored or processed depending on user input.
  - The example to the right demonstrates the input instructing MELGEN to read a block of records required for a SBO calculation.
    - Only the SBO comment block is read and the LBLOCA block is ignored.
    - Alternatively, if the second field on the CommentBlock record were LBLOCA, the LBLOCA records would be read.
  - The CommentBlock record can also contain more than one field so that multiple comment blocks can be read, i.e.,
    - CommentBlock SBO LBLOCA.

```
CommentBlock  SBO
...
PROGRAM MELGEN
...
(((SBO  !Additional comments placed here
...These input records are not ignored
)))
...
(((LBLOCA   !This is a large break LOCA scenario
...These input records are ignored
)))
```

Named Comment Blocks (3)

- There can be multiple comment blocks with the same name
  - Three comment blocks are given the name 'SBO' at right.
  - useful if you have many test case options with variations in multiple locations in the input deck
- Furthermore, the named comment block construct allows multiple names or pseudonyms for a comment block.
  - This would allow some commonalities between a subset of your multiple test options.
- Named Comment blocks can also be nested
  - Comment block 'Nested' can only be processed when both 'LBLOCA' and 'Nested' fields are present on the CommentBlock record.

```
CommentBlock  SBO
...
PROGRAM MELGEN
...
(((SBO  !Additional comments placed here
...These input records are not ignored
)))
...
(((LBLOCA   !This is a large break LOCA scenario
...These input records are ignored
(((Nested  !This is an example of a nested block
)))
)))
(((SBO
...These input records are not ignored
)))
(((SBO+LBLOCA
...These input records are not ignored for either SBO or LBLOCA cases
)))
```

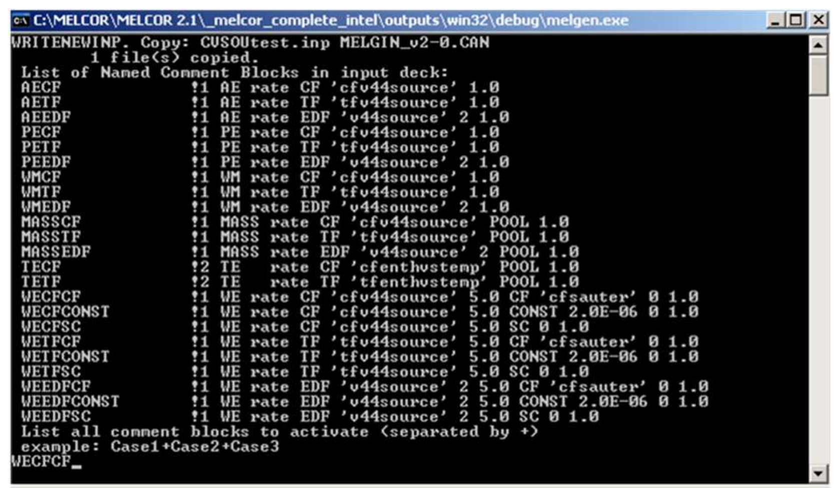
Named Comment Blocks (4)

- **Comment Blocks can be 'activated' through the following input:**
  - **CommentBlock Record**

    CommentBlock case1 case2 case3
  - **Command line arguments**

    melgen i=inputfile c=case1+case2+case3
  - **Interactive Method**
    - **Comments are echoed with comment block names**
    - **Multiple blocks can be specified by delimiting names with '+'**
- **DefaultNamedCommentBlock record**
  - **Placed in the global variable section**
  - **When present, if the user does not select any comment blocks, those listed on the DefaultNamedCommentBlock record are used**

◆**DefaultNamedCommentBlock 2HSNODE+BIGDT**



Variable Input Fields (1)

- **This feature allows the user to insert variable input fields into records which can then be updated through user input at the time of execution.**
- **This can be useful for testing ranges of input fields, managing variant assessment calculations, and could be used to aid interfacing MELCOR to other applications.**
- **Users can provide variable values through a variety of methods.**

Variable Input Fields (2)

- Input fields are tagged so that MELCOR will preprocess the input file, replacing the tagged variables with a value specified by the user.
- These tagged variables can occur multiple places in the input deck and all instances are replaced by the same value.
- The tagged placeholder on an input record includes the variable name and the default value enclosed in a set of triple curly parenthesis, {{{}}}, i.e.,

```
FL_SEG 1   !              SAREA            SLEN             SHYD         SRGH     LAMFLG    SLAM/CFNAME
           1       {{{SAREA=1.0}}}  {{{SLEN=0.2}}}  0.24000E-01  0.5E-04  CONST     0.00
```

- In this example, two variables, SAREA and SLEN, are specified on the flow segment record. Default values are 1.0 and 0.2 respectively are provided.

Variable Input Fields (3)

- **Input Values for the Variable fields can be provided as follows:**
  - **Variable Input File**
    - **A file can be specified in the global specifications, containing values for all or some of the variables found in the input deck.**

      ◆**DefineVariablesFile   Variables.dat !Values read from Variables.dat file**

    - **If some variable values are missing from this file, the user will be asked interactively to provide values.**
    - **If this file is missing, the user will be asked to interactively provide values**

      ◆**{{{Var1=1.0}}}   !Additional comment**
      ◆**{{{Var2='Wetwell'}}} !Additional comment**
      ◆**{{{Var3=3}}}   !Additional comment**

  - **VariableValue record**
    - **Values are provided on the VariableValue record located in the global variable section of the input.**
    - **Any data type can be provided but must be consistent with the requirements of the associated input record to avoid MELGEN errors.**
    - **The supplied value effectively replaces the {{{var=nnn}}} variable field.**
  - **Interactive Method**
    - **A list of all variables is provided and the user is queried for the required value.**
    - **If the user does not specify a value, then the default value will be used unless it was specified by any of the previous methods.**

◆**VariableValue   {{{Var1=1.00}}} {{{Var2='Wetwell'}}} {{{Var3=3}}} !These 3 variables are evaluated**

Capability to Print Default SC Values

- **PrintDefaultSC – Print Sensitivity Coefficient Default Values**
  - **Optional**
  - **Default values for all sensitivity coefficients in active packages are printed to the file named on this record. The default values printed are those defaults _after‘ all global or package default records are processed, i.e., EXEC_GLOBAL_DFT, COR_DFT, etc. This report is printed during MELGEN and does not include user specification from XXX_SC input records. When this record is absent, no report is printed.**
- **PrintCurrentSC – Print Sensitivity Coefficient Default Values**
  - **Optional**
  - **Current values for all sensitivity coefficients in active packages are printed to the file named on this record. This report is printed just after the MELCOR input is read and includes all user specifications from XXX_SC input records after and any default set specifications (EXEC_GLOBAL_DFT, COR_DFT, etc.). When this record is absent, no report is printed.**

Other Improved User Capabilities

- **Smart Restart Capability**
  - Allow user to change CF and TF parameters from the restart without adding new control or tabular functions.
- **Formula Control Functions**
  - Allow user to change CF and TF parameters from the restart without adding new control or tabular functions.

```
! ************** Formula version *****************
CF_ID    'sxjg'    801  Formula
CF_FORMULA  3  SQRT(MAX(ATM_FR*ATM_VEL,C0))
!            ArgName   Value
        1  ATM_FR    FL-VOID('flood')
        2  ATM_VEL   FL-VEL('flood',ATM)
        3  C0        0.0    ! bound positive
```

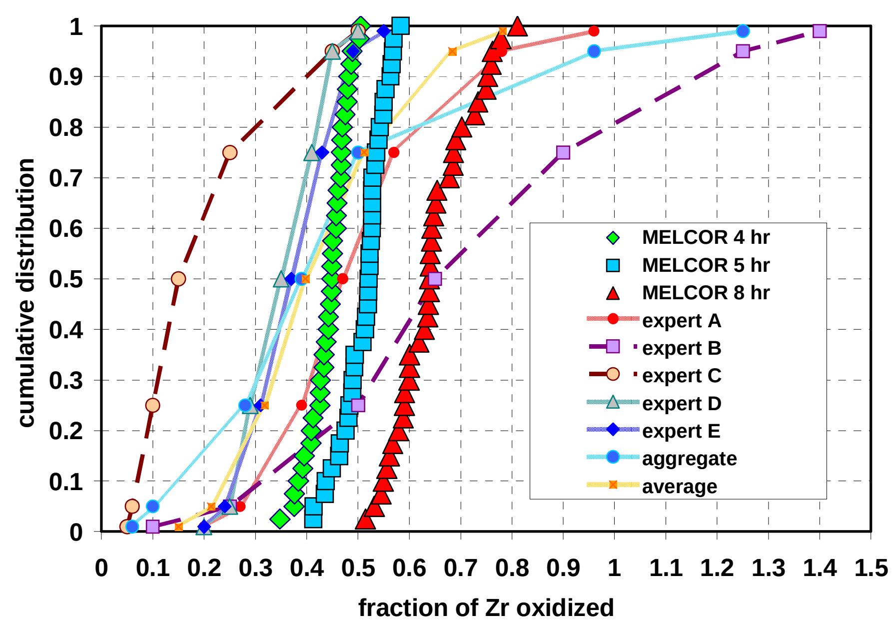
```
! ************** Classic version *****************
CF_ID 'JG'    501  MULTIPLY   ! JG = atm_fr*atm_vel
CF_SAI 1.0  0.0
CF_ARG    2
        1    FL-VOID('flood')      1.0     0.0
        2    FL-VEL('flood',ATM)   1.0     0.0
CF_ID 'XJG'   601  EQUALS     ! XJG = MAX(JG,0)
CF_SAI 1.0  0.0
CF_ULB     LW   0.0   ! Impose lower bound of 0.0
CF_ARG    1
        1    CF-VALU('JG')       1.0     0.0
CF_ID 'XJG'  701  SQRT       ! SXJG = SQRT(MAX(JG,0))
CF_SAI 1.0  0.0  !
CF_ARG    1
        1    CF-VALU('XJG')      1.0     0.0
```

Sandia National Laboratories

Supporting Applications

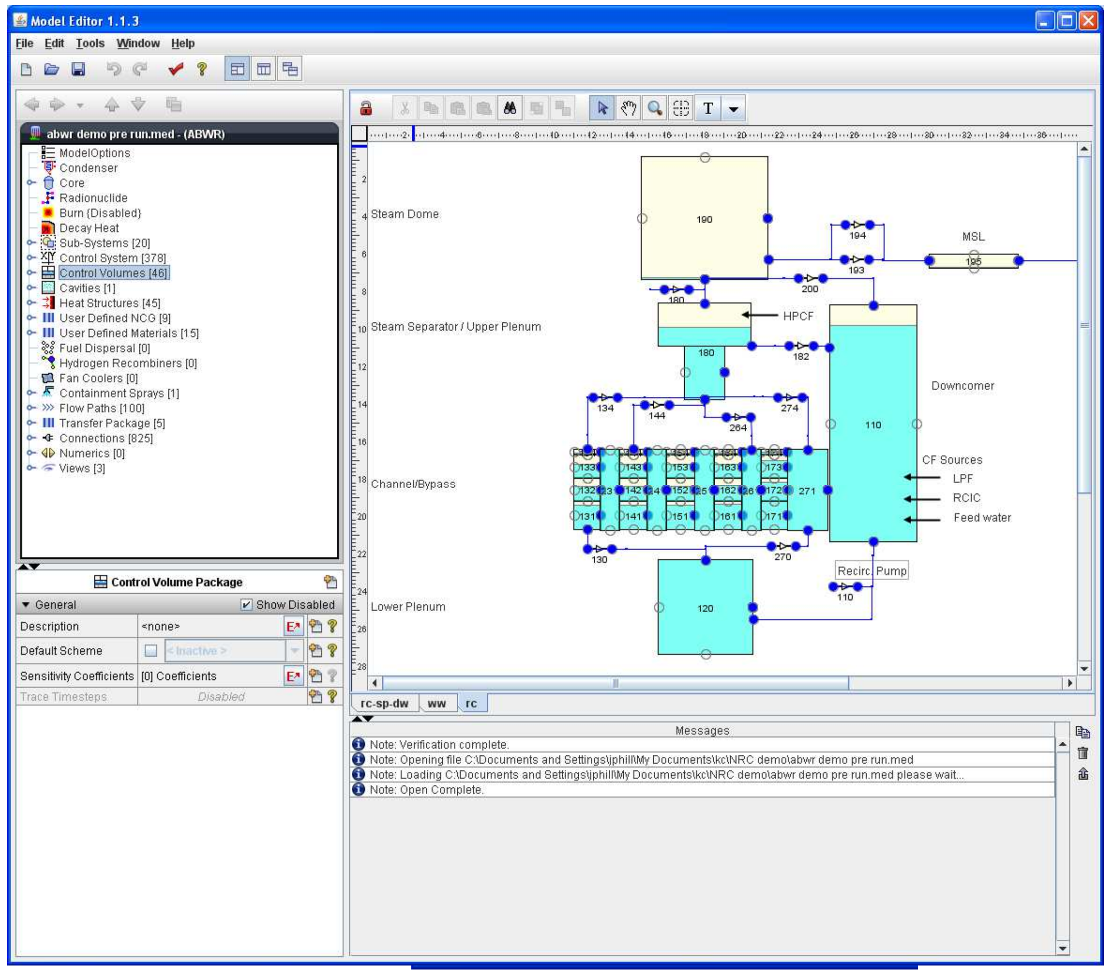
- **[SNAP](#)**
  - **Symbolic Nuclear Analysis Package develped by API – MELCOR Plug-in**
- **PTFREAD**
  - **EXCEL add-in for generating plots, analyzing data, creating AVI's, generation of regression reports**
- **MELCOR 2.1 GUI & Converter**
  - **Utility for generating MELCOR 2.1 input decks and converting existing MELCOR 1.8.6 decks to new format**
- **Uncertainty Software**
  - **Suite of tools for running MELCOR in batch, Monte Carlo sampling of variables and analyzing statistics**

- ***Best Estimate with Uncertainty Quantification***
- ***Powerful tool for risk-informing regulations***



SNAP Development

- **MELCOR 2.1 Plugin**
  - **Version 1.0.0 - Released 7/17/09**
  - **Will convert a 1.8.6 input deck to 2.x**
  - **Sandia is working with SNAP developers to recommend enhancements for MELCOR plug-in**
- **A future workshop will focus on the use of SNAP**
- **Model Editor -Components**
  - **Tree Structure organization**
  - **Arranged according to MELCOR package**
  - **ASCII view of object available**
  - **DIFF capability for components**
- **Object Properties Window**
  - **User Aids**
    - **User Guide Information**
    - **Field units**
  - **Tabular input**
    - **Can paste data from spreadsheet**
- **Views**
  - **Trend plots**
  - **Custom animations**
  - **Others**



Code Performance Improvement Work(1)

- **Code Optimization**
  - **Goal is to achieve maximum optimization of code, maintaining reproducibility of results, and consistency with debug code.**
- **New capability to test code optimization on a subroutine level**
  - **A number of compiler settings must be modified to achieve parity (/fp:source /Qprec-div /Qprec-sqrt /assume:protect-parens)**
  - **Only six routines remain un-optimimized to give parity**
    - **"CORFZO", "CORFZP", "MXXDBD", "POOL3", "RES", "CAVENR"**
  - **Often code that does not optimize well has bugs.**
    - **Review existing routines and test future code for optimization issues.**
- **Performance gains dependent on problem**
  - **BWR Plant decks**
    - **LBLOCA**
      - **GrandGulf LBLOCA (1.43)**
    - **SBO**
      - **GrandGulfSBO (1.56), Peach Bottom (1.24)**
  - **PWR Plant decks**
    - **LBLOCA**
      - **Surry LBLOCA (1.55 – 1.78)**
    - **SBO**
      - **Zion SBO (1.82)**
  - **Test Decks**
    - **Quench 6 (2.2), CORA-13 (1.4), FALCON (1.85), LOFT (2.1), NUPEC (1.57) , VANAM (1.7), LACE 7 (1.33)**

Code Performance Improvement Work (2)

- **Solver Improvements**
  - **The sparse matrix storage format changed to CSR format**
    - **A standardized format that will allow testing multiple solvers**
  - **Pardiso solver from Intel MKL has been tested**
- **Parallelization (OpenMP)**
  - **Currently focusing on CVH package**
  - **Restructuring of routines –**
    - **Eliminating of all the internal subroutines by removing them.**
    - **Treatment of shared data access**
      - **locally used variables removed from modules and passed through argument lists**
      - **Special treatment of accumulative variables**
  - **Implementation of OpenMP commands**
  - **Development of performance metrics**

Code Performance Improvement Work (3)

- **Developed as part of HTGR model development**
  - **Steady state initialization for HTGR may require lengthy computational time**
    - **Large heat capacity of core components**
    - **HTGR initialization can take 4 weeks CPU time without acceleration**
    - **Initialization reduced to less than a day**
  - **Woks by reducing the heat capacity of solid HS and COR components**
  - **Care must be taken to ensure no melting of components or structures**
  - **User activates/de-activates model from input**

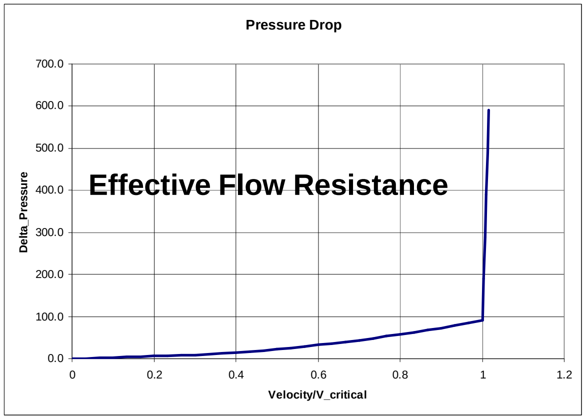
	TstartSS	TendSS	Rfactor
EXEC_SS	-6000.0	0.0	0.01

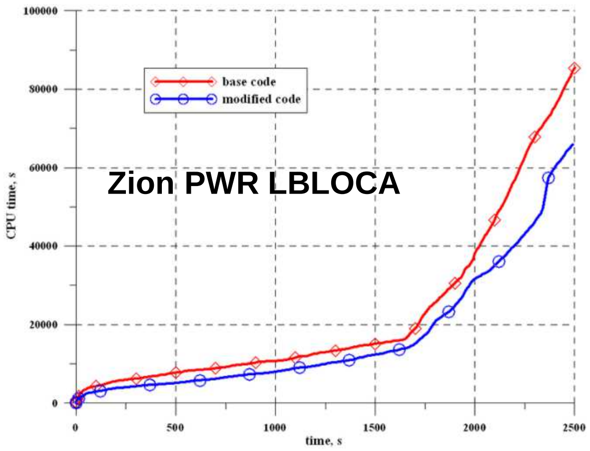
TstartSS is the time to start model,
TendSS time to end model,
Rfactor is the CP reduction factor.

- **Application to water reactors**

Code Performance Improvement Work (4)

- **Critical Flow Modeling**
  - **Current Modeling**
    - **Hydro packages solve the flow equations without consideration of the critical flow limit.**
    - **Once the overall flow solution has converged, each flow is compared to the critical value for the flow path (from the correlations).**
    - **If one or more flows exceed the critical value, the entire equation set is re-solved, with those flows imposed**
  - **Improved Modeling**
    - **Include imposition of critical within the initial solution.**
    - **MELCOR hydrodynamics treats a flow resistance that is a non-linear in velocity by iterating until each calculated flow is in good agreement with the velocity that was used to calculate the flow resistance.**
      - **Choked flow can be captured by making the flow resistance very large for flows greater than critical.**





Summary

Code reliability

- Validation
- QA
- Numerical stability

MELCOR Code Development

User Utilities

- Converter Improvements
- HTML Output
- Etc.

Code Enhancements

- New/improved modeling
- Code performance