Sandia National Laboratories

Securing the future of Nuclear Energy

MELCOR

MELCOR/ORIGEN Integration

2025 European MELCOR Users' Group Meeting

April 7th-11th, 2025

SAND2025-04006PE

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Overview

Effort underway to integrate ORIGEN and MELCOR

- ORIGEN functionally replaces MELCOR Decay Heat package (DCH)
- Facilitated with ORNL-developed Melcor ORigen TidY (MORTY) interface

Review

- ORIGEN and MORTY
- Conventional DCH
- MELCOR/ORIGEN integration

Progress Report (since EMUG '24)

- MORTY interface
- COR demonstrations (using HPR and fixed fuel)
  - Initial and transient radionuclide mass and decay power
  - Point kinetics with XE feedback and irradiation (power/flux)
- CVH demonstrations (using MSRE and fluid fuel)
  - Initialization as aerosol form in pool
  - Single and multiple regions (advection)
  - Vaporization and independent atmosphere/pool tracking
- Future work

Summary

ORIGEN

ORIGEN does isotopic depletion analysis and enables computation of time-dependent concentrations, activities, and radiation source terms accounting for transmutation, fission, and decay

Arrange isotopic equations into a system with a solution that – spatially – applies "at a point" or that could be viewed as an "average over a volume"

Isotopic equations can account for "continuous feed/removal processes" couched as decay constants and/or elements of a source vector

$$\frac{dN_i}{dt} = \sum_{j \neq i} (l_{ij}\lambda_j + f_{ij}\sigma_j\Phi)N_j(t) - (\lambda_i + \sigma_i\Phi)N_i(t) + S_i(t),$$

where

N_i = amount of nuclide i (atoms),
λ_i = decay constant of nuclide i (1/s),
l_{ij} = fractional yield of nuclide i from decay of nuclide j,
σ_i = spectrum-averaged removal cross section for nuclide i (barn),
f_{ij} = fractional yield of nuclide i from neutron-induced removal of nuclide j,
Φ = angle- and energy-integrated time-dependent neutron flux (neutrons/cm^2-s), and
S_i = time-dependent source/feed term (atoms/s).

ORIGEN – MORTY

MORTY is a limited-scale C/Fortran ORIGEN API

- Developed with the expressed intent of linking ORIGEN directly into MELCOR
- Functionally, a "C++ shim" to a core set of ORIGEN capabilities
- Utilizes primitive data types (integers, floating point reals, strings)
  - MORTY handles the interfacing to native ORIGEN data types
  - Access to ORIGEN without in-depth knowledge of its methods and data structures

"Integration" entails a linking at MELCOR build time via shared libraries

- Use/call MORTY without the need to recompile SCALE/ORIGEN source
- Requires SCALE "data resources"

Basic decay and irradiation calculations on logical units called "materials"

- Materials initialized with isotope masses via JSON files typically
- Material inventories are subjected to decay and can be subjected to a scalar flux
- Materials can transfer isotopic masses between each other
- Nuclear data (decay, yields, reaction cross sections, etc.) from SCALE resources
- If irradiating, temperature/flux profiles and multi-group reaction resources

Decay Heat Package

DCH input determines initial DCH/RN1 class mass inventory by:

- Setting total initial mass for each element, and
- Grouping all elements into classes

DCH input gives specific decay power time tables [time vs W/kg] for each element

MELCOR transports radionuclide class mass and loads COR cells, CVs, HSs, and filters by applying specific decay power to local class (element) mass at given time

- DCH "decay" does not move mass between classes
- DCH "decay" cannot alter the fission product chemistry
- One DCH decay rule (table) per chemical element which applies to that element no matter where the class mass resides (COR, CVH, HS, filters)

Traditional DCH focuses on capturing decay heat in safety analyses

ORIGEN/MELCOR Integration

MELCOR and ORIGEN require corresponding spatial domains

- ORIGEN “materials”, each with 2000+ isotopes generally
- MELCOR “regions”, each with elements assigned to classes
- Regions further break down into COR, CV, HS, and FLT (filter)
  - Materials are agnostic to this characteristic
  - Regions “contain” MELCOR objects accordingly (core cells, HS surfaces, etc.)

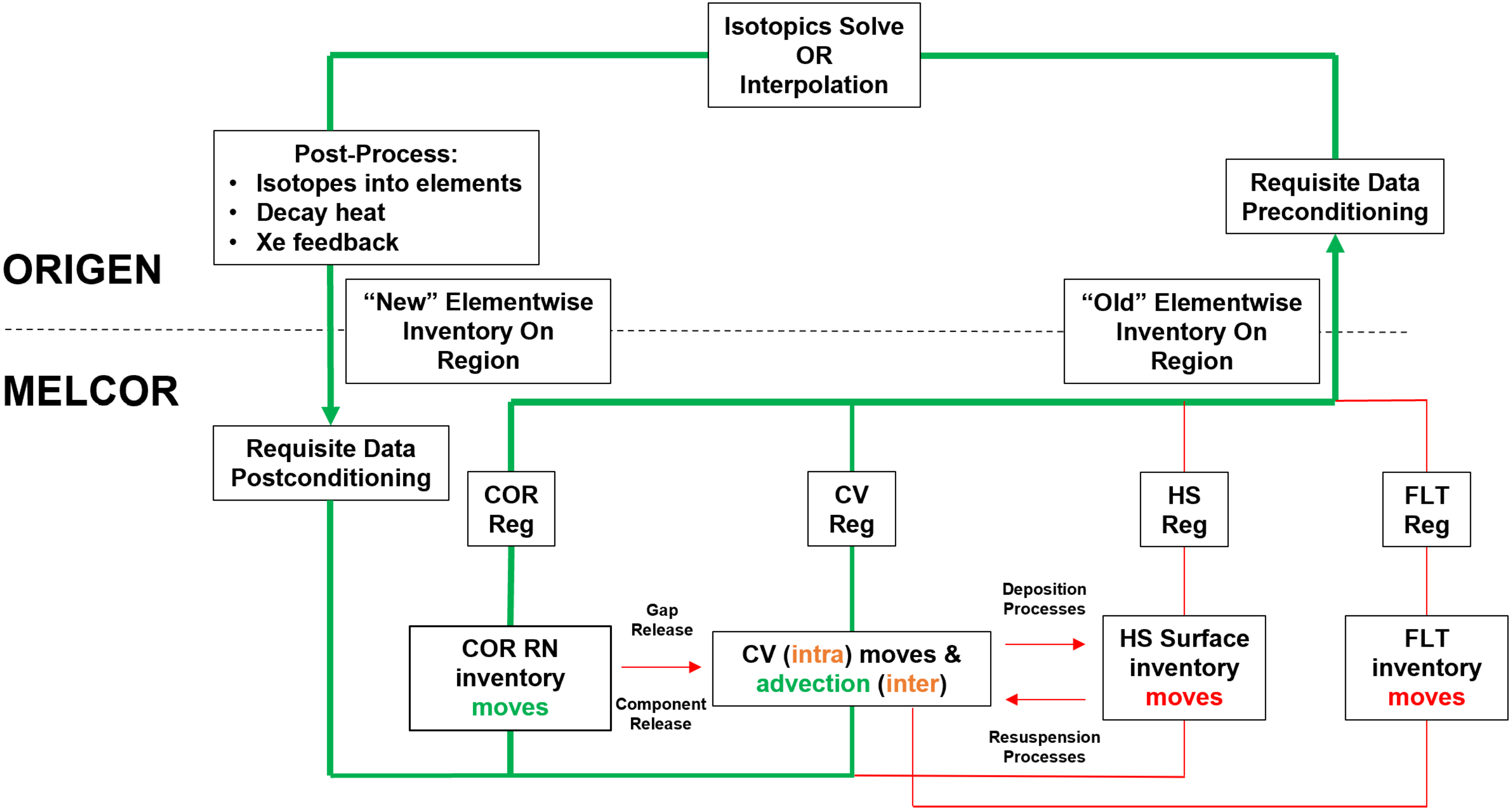
Initialization:

- Initialize materials with ORIGEN JSON file(s)
- Initialize regions according to corresponding materials
  - MORTY tells element-wise mass and decay heat totals
  - MELCOR input arranges elements into classes
  - MELCOR input specifies class mass apportioning (core cells, CV pool/atmosphere, etc.)

Calculation:

- ORIGEN stands in for DCH, updates inventories and decay heat on all materials
- MELCOR does class-wise radionuclide transport on objects that constitute regions
- On MELCOR side, elements transfer as their corresponding classes transfer
- Any MELCOR-side class-wise region-wise transfers must be reported to ORIGEN

ORIGEN/MELCOR Implementation



DCH_ORG BURNFUEL
DCH_ORGCOR 1

!	REGION NAME	JSON NAME	RESPONSE NAME	MTU SCALAR
1	'ALLCORCELLS'	'JSON_NAME'	'RESP_NAME'	1.0

CF_RANGE 'ALLCORCELLS' CELLS 10
CONSTRUCT 1

1	ALL

Progress Report - MORTY

Functionality explored in practice (MELCOR/ORIGEN) and in stand-alone unit tests

- 18 unit tests for coverage on Morty functions
- 2 public-facing demonstration calculations and other demo problems used for development

Morty ~ "the ORIGEN implementation"

- Initialize and load ORIGEN material isotopes from JSON(s) during MELGEN execution
- Advance time, get updated decay heat and mass by element by ORIGEN material
- Update ORIGEN material isotopics after MELCOR-side ORIGEN region transfers
- Irradiation
  - Use energy/flux shape factors and various multi-group libraries
  - Update scalar flux (or power) with MELCOR and inform ORIGEN via MORTY
- Reactivity feedback (Xe-135)
- ORIGEN interpolation and time control
- SCALE data and resource access

Bugs found at shake-down are addressed quickly

Documentation under the SCALE banner (ORNL) – SCALE 7.0 Beta User Manual

Progress Report – COR

DCH-only vs ORIGEN JSON, cycle 0, decay power

		DCH		ORIGEN	
Class Number	Class Name	Class Specific Power	Class Power	Class Specific Power	Class Power
1	XE	2.27736E+04	3.01334E+04	2.27736E+04	3.01333E+04
2	CS	4.54254E+04	5.37956E+04	4.54254E+04	5.37956E+04
3	BA	5.34472E+04	3.85400E+04	5.34472E+04	3.85400E+04
4	I	6.90674E+05	3.63647E+04	6.90674E+05	3.63647E+04
5	TE	1.11599E+05	1.54345E+04	1.11599E+05	1.54344E+04
6	RU	2.56859E+03	1.95167E+03	2.56859E+03	1.95167E+03
7	MO	3.61492E+04	4.43293E+04	3.61492E+04	4.43293E+04
8	CE	3.85053E+03	2.60087E+04	3.85053E+03	2.60088E+04
9	LA	3.05700E+04	6.99030E+04	3.05700E+04	6.99030E+04
10	UO2	1.252713E+00	5.70672E+03	1.25271E+00	5.70670E+03
11	CD	1.14154E+06	1.23664E+04	1.14154E+06	1.23664E+04
12	AG	2.71994E+05	4.61502E+03	2.71994E+05	4.61502E+03
13	BO2	0	0	1.45216E+04	4.19175E-09
14	H2O	0	0	0	0
15	CON	0	0	0	0
16	CSI	3.60597E+05	0	0	0
17	CSM	4.29653E+04	0	0	0
TOTAL			3.39149E+05		3.39149E+05

DCH-only vs ORIGEN JSON, cycle 0, RN1 class masses in core cells

For cell IA=2, IR=3, IC=1 (IFU), the DCH route yields:

```
IR IA KCMP   CLASS MASS(KG)

 2  3  1 3.754E-03 3.360E-03 2.046E-03 1.494E-04 3.924E-04 2.156E-03 3.480E-03
         1.917E-02 6.488E-03 1.293E+01 3.074E-05 4.814E-05 0.000E+00 0.000E+00
         0.000E+00 0.000E+00 0.000E+00
         DECAY HEAT =   9.6233E+02 WATTS
```

Computed as classes from element mass on DCH_EL using RN1_FPN

For the same cell and component, the ORIGEN route yields:

```
IR IA KCMP   CLASS MASS(KG)

 2  3  1 3.754E-03 3.360E-03 2.046E-03 1.494E-04 3.924E-04 2.156E-03 3.480E-03
         1.917E-02 6.488E-03 1.293E+01 3.074E-05 4.814E-05 8.191E-16 0.000E+00
         0.000E+00 0.000E+00 0.000E+00
         DECAY HEAT =   9.6233E+02 WATTS
```

Computed as sums over elements in classes according to DCH_CL, with element mass from JSON using RN1_FPN

Progress Report – COR

DCH-only vs ORIGEN JSON, cycle 0, component-wise cell decay power

IR	IA	FUEL	MATRIX	REFLCTOR	FORMER	SUP-STR	NONS-STR	P-DEB	P-DEB-BY
2	14	0.	0.	0.	0.	0.	0.	0.	0.
2	13	0.	0.	0.	0.	0.	0.	0.	0.
2	12	822.1	0.	0.	0.	0.	0.	0.	0.
2	11	1041.	0.	0.	0.	0.	0.	0.	0.
2	10	1265.	0.	0.	0.	0.	0.	0.	0.
2	9	1457.	0.	0.	0.	0.	0.	0.	0.
2	8	1579.	0.	0.	0.	0.	0.	0.	0.
2	7	1594.	0.	0.	0.	0.	0.	0.	0.
2	6	1492.	0.	0.	0.	0.	0.	0.	0.
2	5	1294.	0.	0.	0.	0.	0.	0.	0.
2	4	1072.	0.	0.	0.	0.	0.	0.	0.
2	3	962.3	0.	0.	0.	0.	0.	0.	0.
2	2	-	-	-	-	0.	0.	0.	
2	1	-	-	-	-	0.	0.	0.	

IR	IA	FUEL	MATRIX	REFLCTOR	FORMER	SUP-STR	NONS-STR	P-DEB	P-DEB-BY
2	14	0.	0.	0.	0.	0.	0.	0.	0.
2	13	0.	0.	0.	0.	0.	0.	0.	0.
2	12	822.1	0.	0.	0.	0.	0.	0.	0.
2	11	1041.	0.	0.	0.	0.	0.	0.	0.
2	10	1265.	0.	0.	0.	0.	0.	0.	0.
2	9	1457.	0.	0.	0.	0.	0.	0.	0.
2	8	1579.	0.	0.	0.	0.	0.	0.	0.
2	7	1594.	0.	0.	0.	0.	0.	0.	0.
2	6	1492.	0.	0.	0.	0.	0.	0.	0.
2	5	1294.	0.	0.	0.	0.	0.	0.	0.
2	4	1072.	0.	0.	0.	0.	0.	0.	0.
2	3	962.3	0.	0.	0.	0.	0.	0.	0.
2	2	-	-	-	-	0.	0.	0.	
2	1	-	-	-	-	0.	0.	0.	

IR	IA	FUEL	MATRIX	REFLCTOR	FORMER	SUP-STR	NONS-STR	P-DEB	P-DEB-BY
15	14	0.	0.	0.	0.	0.	0.	0.	0.
15	13	0.	0.	0.	0.	0.	0.	0.	0.
15	12	1883.	0.	0.	0.	0.	0.	0.	0.
15	11	2385.	0.	0.	0.	0.	0.	0.	0.
15	10	2896.	0.	0.	0.	0.	0.	0.	0.
15	9	3338.	0.	0.	0.	0.	0.	0.	0.
15	8	3616.	0.	0.	0.	0.	0.	0.	0.
15	7	3652.	0.	0.	0.	0.	0.	0.	0.
15	6	3417.	0.	0.	0.	0.	0.	0.	0.
15	5	2964.	0.	0.	0.	0.	0.	0.	0.
15	4	2456.	0.	0.	0.	0.	0.	0.	0.
15	3	2204.	0.	0.	0.	0.	0.	0.	0.
15	2	-	-	-	-	0.	0.	0.	
15	1	-	-	-	-	0.	0.	0.	

IR	IA	FUEL	MATRIX	REFLCTOR	FORMER	SUP-STR	NONS-STR	P-DEB	P-DEB-BY
15	14	0.	0.	0.	0.	0.	0.	0.	0.
15	13	0.	0.	0.	0.	0.	0.	0.	0.
15	12	1883.	0.	0.	0.	0.	0.	0.	0.
15	11	2385.	0.	0.	0.	0.	0.	0.	0.
15	10	2896.	0.	0.	0.	0.	0.	0.	0.
15	9	3338.	0.	0.	0.	0.	0.	0.	0.
15	8	3616.	0.	0.	0.	0.	0.	0.	0.
15	7	3652.	0.	0.	0.	0.	0.	0.	0.
15	6	3417.	0.	0.	0.	0.	0.	0.	0.
15	5	2964.	0.	0.	0.	0.	0.	0.	0.
15	4	2456.	0.	0.	0.	0.	0.	0.	0.
15	3	2204.	0.	0.	0.	0.	0.	0.	0.
15	2	-	-	-	-	0.	0.	0.	
15	1	-	-	-	-	0.	0.	0.	

```
=================================
EDIT OF TOTAL POWER INPUT TO CORE
=================================

DECAY POWER      = 3.39149E+05 W
FISSION POWER    = 0.00000E+00 W
TOTAL POWER      = 3.39149E+05 W
```

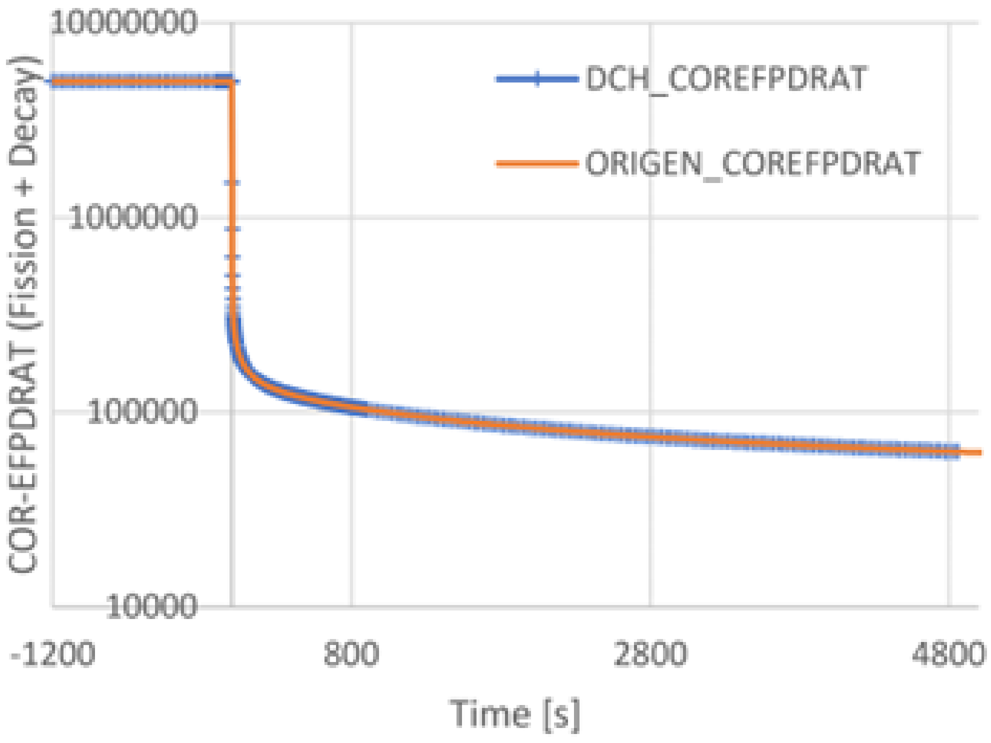
```
=================================
EDIT OF TOTAL POWER INPUT TO CORE
=================================

DECAY POWER      = 3.39148E+05 W
FISSION POWER    = 0.00000E+00 W
TOTAL POWER      = 3.39148E+05 W
```

Progress Report – COR

DCH-only vs ORIGEN JSON, Total Fission and Decay Power

- Time trace shows excellent agreement b/t SCALE-supplied W/kg tables and ORIGEN
- Point kinetics transient with scram and negative reactivity insertion at time zero



Progress Report – COR and CVH Regions

Latest on COR regions uses standard point kinetics to:

- Inform Morty/ORIGEN on irradiation with MELCOR PKM power level predictions
- Enforce Morty/ORIGEN Xe-135 reactivity feedback on MELCOR PKM predictions
- Demonstrated on/around gHPR NRC public demonstration
  - Modernized COR and modernized PKM
  - XE feedback effect on power provided reactivity coefficients furnished
  - Scalar flux (or power) modulation by MELCOR PKM (provided nuclear data furnished)
  - Combined decay, irradiation on COR region (no fuel failure or gap release)
  - Also works with conventional COR/PKM

Latest on CVH regions:

- Advection fully installed (multiple volumes in one region, multiple regions, etc.)
- Multiple CVH regions allowed (region-wise advection mass transfers)
- Each user-defined CVH region actually consists of two (for atmosphere and pool)
- Demonstration – aerosols in pool and vapors in atmosphere
  - MSBR fuel cycle (July '24)
  - Pool release of Xe as noble gas to atmosphere (generated by I decay in pool)
  - Subsequent formation of Cs in atmosphere by decay tracked as an independent inventory
  - Two independent Cs inventories in drain tank, with majority of Cs being in-atmosphere

Future Work

Modify CVH regions according to generalized RN

Modify/extend COR-to-CVH region-wise transfers via core degradation pending generalized RN and modernized COR

Add FLT region functionality

Add HS surface (COR surface?) region functionality

Continue evaluating:

- MORTY interface functionality
- Windows/Mac/Linux builds

Summary

Reviewed Morty, ORIGEN, and conventional DCH in MELCOR

Reviewed the MELCOR/ORIGEN integration strategy

Gave a progress report and outlined future work