

User Problems

***2019 RELAP5 International Users Seminar
April 18-19, 2019
Idaho Falls, ID***

www.inl.gov

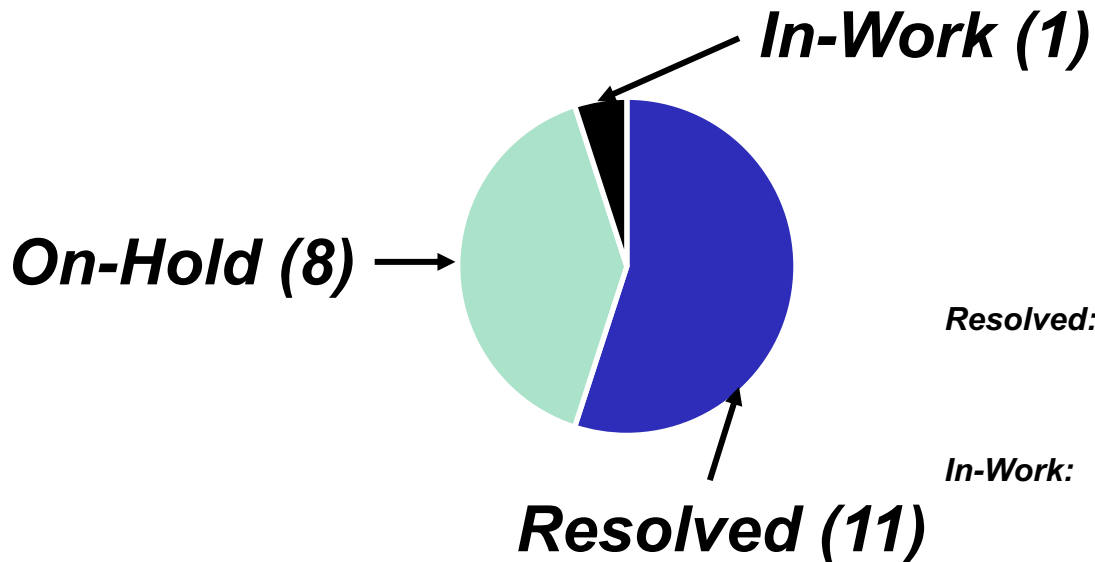


User Problem History

February 2018 – March 2019

9 reported

8 additional older problems were resolved



Resolved: Work has been completed on user problem, update has been submitted, and update is in latest developmental version

In-Work: Work on user problem is currently being carried out

On-Hold: Work has not begun on user problem, or work was begun on user problem but work is now stopped on user problem

Selected User Problems

- ***UP# 18001***
- ***UP# 18003***
- ***UP# 18004***
- ***UP# 18007***
- ***UP# 18012***

Problem 18001

- **Pressurizing an initially helium-filled volume with nitrogen results in a large mass error and reduction in both noncondensable quality and vapor temperature, apparently when the pressure reaches the helium critical pressure.**
- ***Root Cause: Coding error.***
- ***Resolution: Resolved.*** This issue was corrected by modifying two 'if tests' in `ijprop.F` and `jprop.F`. These tests previously checked the volume pressure against the critical pressure, but were modified to test the partial pressure against the critical pressure. This change corrected the observed issues.

Problem 18003

- ***Symptom:*** It was reported that using Word 9 of the CCC0401 Card for the turbine component resulted in an error indicating that the Card had too many words on it.
- ***Root Cause:*** *Array incorrectly sized.*
- ***Status: Resolved.*** Found that variable 'l3m' in subroutine RTURB.F was only allowing 8 words to be used for this card. Modified the variable so that it would allow 9 words and the problem now runs.

Problem 18004

- ***Symptom:*** An input deck generates an input error associated with control variable 1233, which was not entered correctly, but the problem runs anyway on Version 420, probably because nothing references this control variable.
- ***Root Cause:*** *Fail flag not set for error.*
- ***Status:*** *Resolved.* Found a few issues in the setting of variable 'cnverr' in subroutine RCONVR.F. In case(12) (real number integer power control variable) 'cnverr' was set to false instead of true for l3i(6) .le. 0, in case(23) (inverse kinetics) 'cnverr' is set to true when l3z(6) .gt. 0 – which should be a valid case, so 'cnverr' was changed so that it would be true when l3z(6) .le. 0 like the other cases. Last found that variable 'fail' was never set to true if 'cnverr' was true (which was the original cause of the problem), so after the 'end select' in the subroutine added the line – "if (cnverr) fail = .true."

Problem 18007

- ***Symptom:*** The Gnielinski correlation is selected through input (heat transfer package 160), but does not appear to be used if axial conduction is turned on for that structure.
- ***Root Cause:*** Variable 'h_{topta}' not set.
- ***Status: Resolved.*** Found that variable 'h_{topta}' was not set for 2D conduction cases. Added a line to set this variable in subroutine htrc2.F. After setting this variable, the problem correctly used the Gnielinski heat transfer correlation.

Problem 18012

- ***Symptom:*** It was found that a restart problem was getting different results than a base input deck. The base test problem failed, but when the problem was restarted it ran to completion. The results should have been identical.
- ***Root Cause: Restart differences.***
- ***Status: Resolved.*** The problem is restarted at 3750.0 seconds. Found that the problem run time variable 'timehy' = 3749.99999... When the code goes through the timestep read routine rtsc.F it checks whether timehy is greater than the timestep card end value which is 3750.0 seconds. Because it is slightly less than the restart time it uses the timestep size from the previous card and cuts the timestep to the minimum size to try and reach the endtime value of 3750.0 s. After the small timestep the code switches to the next timestep value and writes an extra set of edits. Modified the coding in rtsc.F so that the check is if the endtime is greater than timehy + dtmin, but the results still differed between the runs. Found that dtstep.F performs a similar check the first time it is called which caused the difference. Modified the dtstep.F check similarly and the results between the two runs now match.

Questions?