

# Jitrino OPT bbp

Back-Branch Polling is for yield points insertion at back branches of certain kinds of loops to enable safe thread suspension. Several approaches are implemented there:

- 0 - the feature is turned off
- 1 - insert interruptable subgraph at all backedges except those going from initiallyInterruptable nodes (initialInterruptability is not being propagated to the successors. This is the feature of version 5)
- 2 - path analysis based on searching of pairs [isOnThreadInterruptablePath]->[!isOnThreadInterruptablePath]
- 3 - recursive version of "2"
- 4 - "1" + suspension flag addr [TLS base + offset] is calculated before the loop header
- 5 - like "1" but some backedges are not patched (if all paths through it are uninteruptable)
- 6 - "4" + "5"

They can be switched using the following: [jit.arg.bbp=\[0-6\]](#)

## Possible improvement

It may have sence to try BBPolling at earlier stages - in HLO. To clarify one need to implemet HLO pass for this and find the best position (from the performance POV) for it in the HLO path.

implementation file: <http://svn.apache.org/viewvc/harmony/enhanced/drlvm/trunk/vm/jitrino/src/codegenerator/ia32/la32BBPolling.cpp>