

This repository 13af9a0b Pull requests Issues Gist

accre / lstore Unwatch 11 Unstar 2 Fork 2

Code Issues 22 Pull requests 1 Wiki Pulse Graphs Settings

Merge 120ff4d into e7213e7

PerilousApricot committed 5 hours ago 2 parents e7213e7 + 120ff4d commit 808a15263d77d8166bda573509d58e44a9f1b2b7

Showing 21 changed files with 403 additions and 219 deletions. Unified Split

67 CMakeLists.txt View

```

@@ -4,6 +4,15 @@ project(LStore C)
4   4
5   5     set(LSTORE_PROJECT_VERSION "0.5.1-unknown_commit" CACHE STRING "Project version <major>.<minor>.<patch>[-<git-hash>[-<other-tags>]]")
6   6
7 +## Preliminary configuration. Need to find ccache to set the defaults for the
8 +## corresponding option
9 +find_program(FOUND_CCACHE ccache)
10 +if(FOUND_CCACHE)
11 +  set(ENABLE_CCACHE_DEFAULT ON)
12 +else()
13 +  set(ENABLE_CCACHE_DEFAULT OFF)
14 +endif()
15 +
16 # Options
17 option(BUILD_SHARED_LIBS "Build shared LStore libraries" ON)
18 option(BUILD_TESTS "Build tests" ON)
@@ -17,6 +26,7 @@ option(ENABLE_MSAN "Enable memory sanitizer" OFF)
17 26
18 27
19 28     option(ENABLE_TSAN "Enable thread sanitizer" OFF)
20 29     option(ENABLE_UBSAN "Enable undefined behaviour sanitizer" OFF)
21 30     option(ENABLE_FUZZ "Build with AFL fuzz support" OFF)
22 31     +option(ENABLE_CCACHE "Build with ccache" ${ENABLE_CCACHE_DEFAULT})
23 32
option(INSTALL_EXTERNALS "Install externals to \${CMAKE_INSTALL_PATH}" OFF)
option(INSTALL_YUM_RELEASE "Install yum lstore-release (may require root)" OFF)
@@ -37,7 +47,7 @@ set(CMAKE_INSTALL_PREFIX "\${CMAKE_BINARY_DIR}/local"
37 47
38 48     set(USE_SUPERBUILD ON)
39 49     set(CMAKE_MODULE_PATH "\${CMAKE_SOURCE_DIR}/cmake")
40 50     set(CMAKE_C_FLAGS "\${CMAKE_C_FLAGS} -std=c99 -fPIC -D_REENTRANT -D_GNU_SOURCE -D_LARGEFILE64_SOURCE -D_FILE_OFFSET_BITS=64 -fvisibility=hidden -Werror=implicit-function-declaration")
41 51     -set(CMAKE_C_FLAGS_DEBUG "\${CMAKE_C_FLAGS_DEBUG} -Wall -Wextra -Wno-unused-parameter -Wno-deprecated-declarations -Werror=format -Werror=implicit-function-declaration -Werror=incompatible-pointer-types")
42 52     +set(CMAKE_C_FLAGS_DEBUG "\${CMAKE_C_FLAGS_DEBUG} -Wall -Wextra -Wno-unused-parameter -Wno-deprecated-declarations -Werror=format -Werror=implicit-function-declaration")
43 53     set(CMAKE_MACOSX_RPATH ON)
if(INSTALL_EXTERNALS)
  set(EXTERNAL_INSTALL_DIR \${CMAKE_INSTALL_PREFIX})
@@ -77,6 +87,10 @@ set(CMAKE_C_FLAGS "\${CMAKE_C_FLAGS} \${LSTORE_SANITIZE_FLAGS}")
77 87
78 88     set(CMAKE_EXE_LINKER_FLAGS "\${CMAKE_EXE_LINKER_FLAGS} \${LSTORE_SANITIZE_LINK_FLAGS}")
79 89     set(CMAKE_SHARED_LINKER_FLAGS "\${CMAKE_SHARED_LINKER_FLAGS} \${LSTORE_SANITIZE_LINK_FLAGS}")
+if(ENABLE_FUZZ AND ENABLE_CCACHE)
+  message(FATAL "Don't have proper support for nested wrappers yet")
+endif()
+
if(ENABLE_FUZZ)
  if("\${CMAKE_C_COMPILER_ID}" STREQUAL "GNU")
    set(FUZZ_COMPILE_NAME "afl-gcc")
@@ -89,6 +103,40 @@ if(ENABLE_FUZZ)
  find_program(CMAKE_C_COMPILER \${FUZZ_COMPILE_NAME})

```

5/19/2016

Merge 120ff4dbc33db1b410bbbcc681f4fd53c125c1e1 into e7213e748c17a9904... · accre/lstore@808a152

```

90  104      endif()
91  105
92  106      +if(ENABLE_CCACHE)
93  107          + message(STATUS "Enabling ccache")
94  108          + set(LSTORE_COMPILER_WRAPPER "ccache ")
95  109          + set(LSTORE_COMPILER_WRAPPER_CMAKE "-DCMAKE_C_COMPILER=ccache"
96  110                  "-DCMAKE_CXX_COMPILER=ccache")
97  111          + if(CMAKE_C_COMPILER)
98  112              list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
99  113                  "-DCMAKE_C_COMPILER_ARG1=${CMAKE_C_COMPILER}")
100 114          + else()
101 115              list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
102 116                  "-DCMAKE_C_COMPILER_ARG1=cc")
103 117          + endif()
104 118          + if(CMAKE_CXX_COMPILER)
105 119              list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
106 120                  "-DCMAKE_CXX_COMPILER_ARG1=${CMAKE_CXX_COMPILER}")
107 121          + else()
108 122              list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
109 123                  "-DCMAKE_CXX_COMPILER_ARG1=c++")
110 124          + endif()
111 125          set_property(GLOBAL PROPERTY RULE_LAUNCH_COMPILE ccache)
112 126          set_property(GLOBAL PROPERTY RULE_LAUNCH_LINK ccache)
113 127      +else()
114 128          set(LSTORE_COMPILER_WRAPPER "")
115 129          set(LSTORE_COMPILER_WRAPPER_CMAKE "-DCMAKE_C_COMPILER=${CMAKE_C_COMPILER}"
116 130                  "-DCMAKE_CXX_COMPILER=${CMAKE_CXX_COMPILER}")
117 131      +endif()
118 132
119 133      ## Detect compiler flags
120 134      +include(CheckCCompilerFlag)
121 135      +check_c_compiler_flag(-Werror=incompatible-pointer-types HAS_INCOMPATIBLE_POINTER_WERROR)
122 136      +if(HAS_INCOMPATIBLE_POINTER_WERROR)
123 137          + set(CMAKE_C_FLAGS "${CMAKE_C_FLAGS} -Werror=incompatible-pointer-types")
124 138      +endif()
125 139
126 140      # Use gnu-style paths
127 141      include(GNUInstallDirs)
128 142
129 143      @@ -155,6 +203,23 @@ if(BUILD_TESTS)
130 144          endif()
131 145      endif()
132
133 146      ## Build documentation
134 147      +find_program(DOXYGEN_PATH doxygen
135 148          +      DOC "Path to doxygen executable")
136 149      +add_custom_target(docs COMMENT "Build documentation")
137 150      +add_custom_target(doxygen COMMAND "${DOXYGEN_PATH}"
138 151          +      WORKING_DIRECTORY "${CMAKE_SOURCE_DIR}/doc/"
139 152          +      COMMENT "Document code with doxygen")
140 153      +add_custom_target(sphinx-html COMMAND ${MAKE} "html"
141 154          +      WORKING_DIRECTORY "${CMAKE_SOURCE_DIR}/doc/"
142 155          +      COMMENT "Produce sphinx documentation (HTML)")
143 156      +add_custom_target(sphinx-man COMMAND ${MAKE} "man"
144 157          +      WORKING_DIRECTORY "${CMAKE_SOURCE_DIR}/doc/"
145 158          +      COMMENT "Produce sphinx documentation (Man Pages)")
146 159      +add_dependencies(sphinx-html doxygen)
147 160      +add_dependencies(sphinx-man doxygen)
148 161      +add_dependencies(docs sphinx-html sphinx-man)
149 162
150 163      # Import packaging
151 164      include(LStorePackaging)
152
153 165      @@ -90,7 +90,7 @@ compile_map['scan-build'] =
154 166          def scan_checks = "-enable-checker alpha.core.BoolAssignment -enable-checker alpha.core.CallAndMessageUnInitRefArg -enable-
155 167              dir('build') {
156 168                  sh "mkdir clang-static-analyzer"
157 169                  - sh "CCC_CC=clang cmake -DCMAKE_EXPORT_COMPILE_COMMANDS=ON -DBUILD_TESTS=on -DCMAKE_INSTALL_PREFIX=local/ .."
158 160
159 161
160 162
161 163
162 164
163 165
164 166
165 167
166 168
167 169
168 170
169 171
170 172
171 173
172 174
173 175
174 176
175 177
176 178
177 179
178 180
179 181
180 182
181 183
182 184
183 185
184 186
185 187
186 188
187 189
188 190
189 191
190 192
191 193
192 194
193 195
194 196
195 197
196 198
197 199
198 200
199 201
200 202
201 203
202 204
203 205
204 206
205 207
206 208
207 209
208 210
209 211
210 212
211 213
212 214
213 215
214 216
215 217
216 218
217 219
218 220
219 221
220 222
221 223
222 224
223 225
224 226
225 227
226 228
227 229
228 230
229 231
230 232
231 233
232 234
233 235
234 236
235 237
236 238
237 239
238 240
239 241
240 242
241 243
242 244
243 245
244 246
245 247
246 248
247 249
248 250
249 251
250 252
251 253
252 254
253 255
254 256
255 257
256 258
257 259
258 260
259 261
260 262
261 263
262 264
263 265
264 266
265 267
266 268
267 269
268 270
269 271
270 272
271 273
272 274
273 275
274 276
275 277
276 278
277 279
278 280
279 281
280 282
281 283
282 284
283 285
284 286
285 287
286 288
287 289
288 290
289 291
290 292
291 293
292 294
293 295
294 296
295 297
296 298
297 299
298 300
299 301
300 302
301 303
302 304
303 305
304 306
305 307
306 308
307 309
308 310
309 311
310 312
311 313
312 314
313 315
314 316
315 317
316 318
317 319
318 320
319 321
320 322
321 323
322 324
323 325
324 326
325 327
326 328
327 329
328 330
329 331
330 332
331 333
332 334
333 335
334 336
335 337
336 338
337 339
338 340
339 341
340 342
341 343
342 344
343 345
344 346
345 347
346 348
347 349
348 350
349 351
350 352
351 353
352 354
353 355
354 356
355 357
356 358
357 359
358 360
359 361
360 362
361 363
362 364
363 365
364 366
365 367
366 368
367 369
368 370
369 371
370 372
371 373
372 374
373 375
374 376
375 377
376 378
377 379
378 380
379 381
380 382
381 383
382 384
383 385
384 386
385 387
386 388
387 389
388 390
389 391
390 392
391 393
392 394
393 395
394 396
395 397
396 398
397 399
398 400
399 401
400 402
401 403
402 404
403 405
404 406
405 407
406 408
407 409
408 410
409 411
410 412
411 413
412 414
413 415
414 416
415 417
416 418
417 419
418 420
419 421
420 422
421 423
422 424
423 425
424 426
425 427
426 428
427 429
428 430
429 431
430 432
431 433
432 434
433 435
434 436
435 437
436 438
437 439
438 440
439 441
440 442
441 443
442 444
443 445
444 446
445 447
446 448
447 449
448 450
449 451
450 452
451 453
452 454
453 455
454 456
455 457
456 458
457 459
458 460
459 461
460 462
461 463
462 464
463 465
464 466
465 467
466 468
467 469
468 470
469 471
470 472
471 473
472 474
473 475
474 476
475 477
476 478
477 479
478 480
479 481
480 482
481 483
482 484
483 485
484 486
485 487
486 488
487 489
488 490
489 491
490 492
491 493
492 494
493 495
494 496
495 497
496 498
497 499
498 500
499 501
500 502
501 503
502 504
503 505
504 506
505 507
506 508
507 509
508 510
509 511
510 512
511 513
512 514
513 515
514 516
515 517
516 518
517 519
518 520
519 521
520 522
521 523
522 524
523 525
524 526
525 527
526 528
527 529
528 530
529 531
530 532
531 533
532 534
533 535
534 536
535 537
536 538
537 539
538 540
539 541
540 542
541 543
542 544
543 545
544 546
545 547
546 548
547 549
548 550
549 551
550 552
551 553
552 554
553 555
554 556
555 557
556 558
557 559
558 560
559 561
560 562
561 563
562 564
563 565
564 566
565 567
566 568
567 569
568 570
569 571
570 572
571 573
572 574
573 575
574 576
575 577
576 578
577 579
578 580
579 581
580 582
581 583
582 584
583 585
584 586
585 587
586 588
587 589
588 590
589 591
590 592
591 593
592 594
593 595
594 596
595 597
596 598
597 599
598 600
599 601
600 602
601 603
602 604
603 605
604 606
605 607
606 608
607 609
608 610
609 611
610 612
611 613
612 614
613 615
614 616
615 617
616 618
617 619
618 620
619 621
620 622
621 623
622 624
623 625
624 626
625 627
626 628
627 629
628 630
629 631
630 632
631 633
632 634
633 635
634 636
635 637
636 638
637 639
638 640
639 641
640 642
641 643
642 644
643 645
644 646
645 647
646 648
647 649
648 650
649 651
650 652
651 653
652 654
653 655
654 656
655 657
656 658
657 659
658 660
659 661
660 662
661 663
662 664
663 665
664 666
665 667
666 668
667 669
668 670
669 671
670 672
671 673
672 674
673 675
674 676
675 677
676 678
677 679
678 680
679 681
680 682
681 683
682 684
683 685
684 686
685 687
686 688
687 689
688 690
689 691
690 692
691 693
692 694
693 695
694 696
695 697
696 698
697 699
698 700
699 701
700 702
701 703
702 704
703 705
704 706
705 707
706 708
707 709
708 710
709 711
710 712
711 713
712 714
713 715
714 716
715 717
716 718
717 719
718 720
719 721
720 722
721 723
722 724
723 725
724 726
725 727
726 728
727 729
728 730
729 731
730 732
731 733
732 734
733 735
734 736
735 737
736 738
737 739
738 740
739 741
740 742
741 743
742 744
743 745
744 746
745 747
746 748
747 749
748 750
749 751
750 752
751 753
752 754
753 755
754 756
755 757
756 758
757 759
758 760
759 761
760 762
761 763
762 764
763 765
764 766
765 767
766 768
767 769
768 770
769 771
770 772
771 773
772 774
773 775
774 776
775 777
776 778
777 779
778 780
779 781
780 782
781 783
782 784
783 785
784 786
785 787
786 788
787 789
788 790
789 791
790 792
791 793
792 794
793 795
794 796
795 797
796 798
797 799
798 800
799 801
800 802
801 803
802 804
803 805
804 806
805 807
806 808
807 809
808 810
809 811
810 812
811 813
812 814
813 815
814 816
815 817
816 818
817 819
818 820
819 821
820 822
821 823
822 824
823 825
824 826
825 827
826 828
827 829
828 830
829 831
830 832
831 833
832 834
833 835
834 836
835 837
836 838
837 839
838 840
839 841
840 842
841 843
842 844
843 845
844 846
845 847
846 848
847 849
848 850
849 851
850 852
851 853
852 854
853 855
854 856
855 857
856 858
857 859
858 860
859 861
860 862
861 863
862 864
863 865
864 866
865 867
866 868
867 869
868 870
869 871
870 872
871 873
872 874
873 875
874 876
875 877
876 878
877 879
878 880
879 881
880 882
881 883
882 884
883 885
884 886
885 887
886 888
887 889
888 890
889 891
890 892
891 893
892 894
893 895
894 896
895 897
896 898
897 899
898 900
899 901
900 902
901 903
902 904
903 905
904 906
905 907
906 908
907 909
908 910
909 911
910 912
911 913
912 914
913 915
914 916
915 917
916 918
917 919
918 920
919 921
920 922
921 923
922 924
923 925
924 926
925 927
926 928
927 929
928 930
929 931
930 932
931 933
932 934
933 935
934 936
935 937
936 938
937 939
938 940
939 941
940 942
941 943
942 944
943 945
944 946
945 947
946 948
947 949
948 950
949 951
950 952
951 953
952 954
953 955
954 956
955 957
956 958
957 959
958 960
959 961
960 962
961 963
962 964
963 965
964 966
965 967
966 968
967 969
968 970
969 971
970 972
971 973
972 974
973 975
974 976
975 977
976 978
977 979
978 980
979 981
980 982
981 983
982 984
983 985
984 986
985 987
986 988
987 989
988 990
989 991
990 992
991 993
992 994
993 995
994 996
995 997
996 998
997 999
998 1000
999 1001
1000 1002
1001 1003
1002 1004
1003 1005
1004 1006
1005 1007
1006 1008
1007 1009
1008 1010
1009 1011
1010 1012
1011 1013
1012 1014
1013 1015
1014 1016
1015 1017
1016 1018
1017 1019
1018 1020
1019 1021
1020 1022
1021 1023
1022 1024
1023 1025
1024 1026
1025 1027
1026 1028
1027 1029
1028 1030
1029 1031
1030 1032
1031 1033
1032 1034
1033 1035
1034 1036
1035 1037
1036 1038
1037 1039
1038 1040
1039 1041
1040 1042
1041 1043
1042 1044
1043 1045
1044 1046
1045 1047
1046 1048
1047 1049
1048 1050
1049 1051
1050 1052
1051 1053
1052 1054
1053 1055
1054 1056
1055 1057
1056 1058
1057 1059
1058 1060
1059 1061
1060 1062
1061 1063
1062 1064
1063 1065
1064 1066
1065 1067
1066 1068
1067 1069
1068 1070
1069 1071
1070 1072
1071 1073
1072 1074
1073 1075
1074 1076
1075 1077
1076 1078
1077 1079
1078 1080
1079 1081
1080 1082
1081 1083
1082 1084
1083 1085
1084 1086
1085 1087
1086 1088
1087 1089
1088 1090
1089 1091
1090 1092
1091 1093
1092 1094
1093 1095
1094 1096
1095 1097
1096 1098
1097 1099
1098 1100
1099 1101
1100 1102
1101 1103
1102 1104
1103 1105
1104 1106
1105 1107
1106 1108
1107 1109
1108 1110
1109 1111
1110 1112
1111 1113
1112 1114
1113 1115
1114 1116
1115 1117
1116 1118
1117 1119
1118 1120
1119 1121
1120 1122
1121 1123
1122 1124
1123 1125
1124 1126
1125 1127
1126 1128
1127 1129
1128 1130
1129 1131
1130 1132
1131 1133
1132 1134
1133 1135
1134 1136
1135 1137
1136 1138
1137 1139
1138 1140
1139 1141
1140 1142
1141 1143
1142 1144
1143 1145
1144 1146
1145 1147
1146 1148
1147 1149
1148 1150
1149 1151
1150 1152
1151 1153
1152 1154
1153 1155
1154 1156
1155 1157
1156 1158
1157 1159
1158 1160
1159 1161
1160 1162
1161 1163
1162 1164
1163 1165
1164 1166
1165 1167
1166 1168
1167 1169
1168 1170
1169 1171
1170 1172
1171 1173
1172 1174
1173 1175
1174 1176
1175 1177
1176 1178
1177 1179
1178 1180
1179 1181
1180 1182
1181 1183
1182 1184
1183 1185
1184 1186
1185 1187
1186 1188
1187 1189
1188 1190
1189 1191
1190 1192
1191 1193
1192 1194
1193 1195
1194 1196
1195 1197
1196 1198
1197 1199
1198 1200
1199 1201
1200 1202
1201 1203
1202 1204
1203 1205
1204 1206
1205 1207
1206 1208
1207 1209
1208 1210
1209 1211
1210 1212
1211 1213
1212 1214
1213 1215
1214 1216
1215 1217
1216 1218
1217 1219
1218 1220
1219 1221
1220 1222
1221 1223
1222 1224
1223 1225
1224 1226
1225 1227
1226 1228
1227 1229
1228 1230
1229 1231
1230 1232
1231 1233
1232 1234
1233 1235
1234 1236
1235 1237
1236 1238
1237 1239
1238 1240
1239 1241
1240 1242
1241 1243
1242 1244
1243 1245
1244 1246
1245 1247
1246 1248
1247 1249
1248 1250
1249 1251
1250 1252
1251 1253
1252 1254
1253 1255
1254 1256
1255 1257
1256 1258
1257 1259
1258 1260
1259 1261
1260 1262
1261 1263
1262 1264
1263 1265
1264 1266
1265 1267
1266 1268
1267 1269
1268 1270
1269 1271
1270 1272
1271 1273
1272 1274
1273 1275
1274 1276
1275 1277
1276 1278
1277 1279
1278 1280
1279 1281
1280 1282
1281 1283
1282 1284
1283 1285
1284 1286
1285 1287
1286 1288
1287 1289
1288 1290
1289 1291
1290 1292
1291 1293
1292 1294
1293 1295
1294 1296
1295 1297
1296 1298
1297 1299
1298 1300
1299 1301
1300 1302
1301 1303
1302 1304
1303 1305
1304 1306
1305 1307
1306 1308
1307 1309
1308 1310
1309 1311
1310 1312
1311 1313
1312 1314
1313 1315
1314 1316
1315 1317
1316 1318
1317 1319
1318 1320
1319 1321
1320 1322
1321 1323
1322 1324
1323 1325
1324 1326
1325 1327
1326 1328
1327 1329
1328 1330
1329 1331
1330 1332
1331 1333
1332 1334
1333 133
```

5/19/2016

Merge 120ff4dbc33db1b410bbbcc681f4fd53c125c1e1 into e7213e748c17a9904... · accre/lstore@808a152

```

94    94    +    sh "CCC_CC=clang scan-build cmake -DCMAKE_EXPORT_COMPILE_COMMANDS=ON -DBUILD_TESTS=on -DCMAKE_INSTALL_PREFIX=local/ -DE
95    95    sh "CC=clang make externals -j2"
96    96    sh "CCC_CC=clang scan-build -o clang-static-analyzer -v -v ${scan_checks} --keep-empty make -j16"
97    97    sh "mv clang-static-analyzer/* ../clang-report"
98    98    @@ -105,15 +105,15 @@ for (int i = 0 ; i < distros.size(); ++i) {
99    105   compile_map["${x}"] = { node('docker') {
100   106     deleteDir()
101   107     unstash 'source'
102   108     -    sh "bash scripts/generate-docker-base.sh ${x}"
103   109     -    sh "bash scripts/build-docker-base.sh ${x}"
104   110     -    sh "bash scripts/package.sh ${x}"
105   111     -    sh "bash scripts/update-repo.sh ${x}"
106   108   +    sh """bash scripts/generate-docker-base.sh ${x}
107   109   +    bash scripts/build-docker-base.sh ${x}
108   110   +    bash scripts/package.sh ${x}"""
109   111   +    sh """bash scripts/update-repo.sh ${x}
110   112   +    bash scripts/test-repo.sh ${x}"""
111   112   archive 'build/package/**'
112   113   -    sh "bash scripts/test-repo.sh ${x}"
113   114   stash includes: 'build/package/**', name: "${x}-package"
114   115   dockerFingerprintFrom dockerfile: "scripts/docker/builder/${x}/Dockerfile", \
115   116   -    image: "lstore/builder:${x}"
116   116   +    image: "lstore/builder:${x}"
117   117   } }
118   118   }
119   119

```

9 cmake/LStoreExternals.cmake View

```

@@ -13,7 +13,7 @@ if(BUILD_APACHE OR (NOT APR-ACCRE_FOUND) OR (APR_LIBRARY MATCHES "^${EXTERNAL_INST
13   13      "--includedir=${EXTERNAL_INSTALL_DIR}/include/apr-ACCRE-1"
14   14      "--with-installbuilddir=${EXTERNAL_INSTALL_DIR}/lib/apr-ACCRE-1/bui
15   15      "--enable-static=yes" "--enable-shared=no" "--with-pic"
16   16      -    "CC=${CMAKE_C_COMPILER}"
16   16      +    "CC=${LSTORE_COMPILER_WRAPPER}${CMAKE_C_COMPILER}"
17   17      BUILD_COMMAND ${MAKE}
18   18      TEST_COMMAND ${MAKE} "test"
19   19      INSTALL_COMMAND ${MAKE} "install"
@@ -43,7 +43,7 @@ if(BUILD_APACHE OR (NOT APRUTIL-ACCRE_FOUND) OR (APU_LIBRARY MATCHES "^${EXTERNAL_I
43   43      "--includedir=${EXTERNAL_INSTALL_DIR}/include/apr-util-ACCRE-1"
44   44      "--with-installbuilddir=${EXTERNAL_INSTALL_DIR}/lib/apr-util-ACCRE-1
45   45      "--enable-static=yes" "--enable-shared=no" "--with-pic"
46   46      -    "CC=${CMAKE_C_COMPILER}"
46   46      +    "CC=${LSTORE_COMPILER_WRAPPER}${CMAKE_C_COMPILER}"
47   47      BUILD_COMMAND ${MAKE}
48   48      TEST_COMMAND ${MAKE} "test"
49   49      INSTALL_COMMAND ${MAKE} "install"
@@ -61,7 +61,7 @@ if(BUILD_JERASURE OR (NOT JERASURE_FOUND) OR (JERASURE_LIBRARY MATCHES "^${EXTERNAL_L
61   61      INSTALL_DIR "${EXTERNAL_INSTALL_DIR}"
62   62      CMAKE_ARGS -DCMAKE_INSTALL_PREFIX=${EXTERNAL_INSTALL_DIR}
63   63      -DWANT_SHARED:BOOL=OFF -DWANT_STATIC:BOOL=ON
64   64      -    "-DCMAKE_C_COMPILER=${CMAKE_C_COMPILER}"
64   64      +    "${LSTORE_COMPILER_WRAPPER_CMAKE}"
65   65      TEST_COMMAND ""
66   66      INSTALL_COMMAND ${MAKE} "install"
67   67      )
@@ -76,7 +76,8 @@ if((NOT CZMQ_FOUND) OR (CZMQ_LIBRARY MATCHES "^${EXTERNAL_INSTALL_DIR}"))
76   76      INSTALL_DIR "${EXTERNAL_INSTALL_DIR}"
77   77      CMAKE_ARGS -DCMAKE_INSTALL_PREFIX=${EXTERNAL_INSTALL_DIR}
78   78      -DBUILD_SHARED_LIBS:BOOL=OFF
79   79      -    "-DCMAKE_C_COMPILER=${CMAKE_C_COMPILER}"
79   79      +    "-DCMAKE_C_FLAGS=-fPIC"
80   80      +    "${LSTORE_COMPILER_WRAPPER_CMAKE}"
81   81      TEST_COMMAND ${MAKE} "test"
82   82      INSTALL_COMMAND ${MAKE} "install"
83   83      TEST_EXCLUDE_FROM_MAIN 1

```

5 cmake/LStorePackaging.cmake

```

@@ -48,7 +48,10 @@ set(CPACK_DEBIAN_PACKAGE_SHLIBDEPS ON)
48   48
49   49
50   50
51 -  ADD_CUSTOM_TARGET(rpm rpmbuild -ta ${CPACK_SOURCE_PACKAGE_FILE_NAME}.tar.gz)
51 +  ADD_CUSTOM_TARGET(rpm rpmbuild --verbose
52 +    -D "_rpmdir ${CMAKE_BINARY_DIR}/rpm_output"
53 +    -D "_srcrpmdir ${CMAKE_BINARY_DIR}/srpm_output"
54 +    -ta ${CPACK_SOURCE_PACKAGE_FILE_NAME}.tar.gz)
52   55
53   56
54   57
endif()

```

4 debian/rules

[View](#)

```

@@ -20,6 +20,6 @@ export DH_BUILD_DDEBS=1
20   20
21   21
22   22
23 -#override_dh_auto_configure:
24 -#  dh_auto_configure -- # -DCMAKE_LIBRARY_PATH=$(DEB_HOST_MULTIARCH)
23 +override_dh_auto_configure:
24 +  dh_auto_configure -- -DCMAKE_INSTALL_PREFIX=/usr # -DCMAKE_LIBRARY_PATH=$(DEB_HOST_MULTIARCH)
25   25

```

2 doc/Doxyfile

[View](#)

```

@@ -771,7 +771,7 @@ WARN_LOGFILE      =
771  771
772  772
773  773
774 -INPUT          = ../src/toolbox/chksum.h
774 +INPUT          = ../src/toolbox/tbx
775
776  # This tag can be used to specify the character encoding of the source files
777  # that doxygen parses. Internally doxygen uses the UTF-8 encoding. Doxygen uses

```

12 lstore.spec

[View](#)

```

@@ -17,17 +17,19 @@ Source: https://github.com/accre/lstore-release/archive/LStore-%{_version}.tar.g
17   17
18
19 %prep
20 -%setup -n LStore-%{_version}
20 +%setup -q -n LStore-%{_version}
21
22 %build
23 CFLAGS="-I%{_prefix}/include $RPM_OPT_FLAGS"
24 CMAKE_FLAGS="-DCMAKE_INSTALL_PREFIX=%{_prefix}"
25 CMAKE_FLAGS="$CMAKE_FLAGS -DINSTALL_YUM_RELEASE:BOOL=ON"
26 CMAKE_FLAGS="$CMAKE_FLAGS -DINSTALL_META:BOOL=ON"
27 +CMAKE_FLAGS="$CMAKE_FLAGS -DCMAKE_BUILD_TYPE:BOOL=RelWithDebInfo"
28 cmake $CMAKE_FLAGS .
28
29 -make
29 +make %{?_smp_mflags}
29
30 %install
31 +rm -rf $RPM_BUILD_ROOT
32 make DESTDIR=$RPM_BUILD_ROOT install
33
34 %clean
35
@@ -39,8 +41,8 @@ rm -rf $RPM_BUILD_ROOT
39   41
40   42
41   43
41 %files
41 %defattr(-,root,root,-)

```

5/19/2016

Merge 120ff4dbc33db1b410bbbcc681f4fd53c125c1e1 into e7213e748c17a9904... · accre/lstore@808a152

```

42      -/usr/bin
43      -/usr/lib64
44      +%{_bindir}/*
45      +%{_libdir}/*
46
47      %changelog
48      * Sat Apr 23 2016 Andrew Melo <andrew.m.melo@vanderbilt.edu> 0.5.1-1
49
50      @@ -53,7 +55,7 @@ Requires: lstore
51      %description devel
52      Development files for LStore
53      %files devel
54
55      -/usr/include
56      +%{_includedir}/*
57
58      %package meta
59      Summary: Default LStore configuration
60
61

```

2 scripts/generate-docker-base.sh

```

@@ -65,6 +65,7 @@ for DISTRO in "${DISTROS[@]}"; do
65      rpm)
66      ADDITIONAL_PACKAGES=(
67          autoconf
68      +
69          ccache
70          curl
71          createrepo
72          czmq-devel
73
74      @@ -85,6 +86,7 @@ for DISTRO in "${DISTROS[@]}"; do
75      ADDITIONAL_PACKAGES=(
76          autoconf
77          ca-certificates
78      +
79          ccache
80          cmake
81          curl
82          debhelper
83
84

```

120 scripts/package-internal-separate.sh

```

...
...
@@ -0,0 +1,120 @@
1  +#!/bin/bash
2  +
3  +#
4  +## package-internal.sh - Runs within docker container to package LStore
5  +##
6  +
7  +##
8  +## Preliminary bootstrapping
9  +##
10 +set -eu
11 +ABSOLUTE_PATH=$(cd `dirname "${BASH_SOURCE[0]}`` && pwd)
12 +source $ABSOLUTE_PATH/functions.sh
13 +PACKAGE_DISTRO=${1:-unknown_distro}
14 +PACKAGE_SUBDIR=$PACKAGE_DISTRO
15 +umask 0000
16 +
17 +case $PACKAGE_DISTRO in
18 +undefined)
19 +    # TODO Fail gracefully
20 +    ;;
21 +ubuntu-*|debian-*)
22 +    # switch to gdebi if automatic dependency resolution is needed
23 +    PACKAGE_INSTALL="dpkg -i"
24 +    PACKAGE_SUFFIX=deb
25 +    ;;
26 +centos-*)
27 +    PACKAGE_INSTALL="rpm -i"
28 +    PACKAGE_SUFFIX=rpm
29 +    ;;
30 +*)

```

```

31 +     fatal "Unexpected distro name $DISTRO_NAME"
32 +   ;;
33 +esac
34 +
35 +# todo could probe this from docker variables
36 +PACKAGE_BASE="/tmp/lstore-release"
37 +REPO_BASE=$LSTORE_RELEASE_BASE/build/package/$PACKAGE_SUBDIR
38 +# Here and elsewhere, we need to set the umask when we write to the host-mounted
39 +# paths. Otherwise users outside the container can't read/write files. But,
40 +# we don't want to just blindly set umask 0000, in case there's some
41 +# weirdness with how the installer works.
42 +
43 +  umask 000
44 +  mkdir -p $PACKAGE_BASE/{build,cmake} $REPO_BASE
45 +  # That being said, umask appears to not work right...
46 +  # FIXME: Why is umask so sad?
47 +  chmod 777 $PACKAGE_BASE $PACKAGE_BASE/{build,cmake} $REPO_BASE
48 +
49 +check_cmake $PACKAGE_BASE/cmake
50 +note "Beginning packaging at $(date) for $PACKAGE_SUBDIR"
51 +
52 +
53 +# Get weird. Build each package, then install the RPM. If this is
54 +# configurable in the future, the order of packages matters.
55 +
56 +cd $PACKAGE_BASE/build
57 +for PACKAGE in apr-accre apr-util-accre jerasure czmq \
58 +    toolbox gop ibp lio release meta; do
59 +#for PACKAGE in meta; do
60 +  SOURCE_BASE=$(get_repo_source_path ${PACKAGE})
61 +  if [ "$PACKAGE" == "czmq" ];then
62 +    if (ldconfig -p | grep -q libczmq); then
63 +      echo "libczmq.so is available: skipping czmq package build.";
64 +      continue
65 +    else
66 +      echo "libczmq.so not found, building a czmq package...";
67 +    fi
68 +  fi
69 +
70 +
71 +  (
72 +    umask 000
73 +    mkdir -p $PACKAGE
74 +    # See above.
75 +    chmod 777 $PACKAGE
76 +  )
77 +  pushd $PACKAGE
78 +  # NOTE: Can't do this with --git-dir. Git 1.8.3.1 considers the tag "dirty"
79 +  #       if the CURRENT working directory is dirty...
80 +  # NOTE: The git update-index is needed since the host and container git
81 +  #       versions might be different.
82 +  TAG_NAME=$(cd $SOURCE_BASE &&
83 +            ( git update-index -q --refresh >/dev/null || true ) && \
84 +            git describe --abbrev=32 --dirty="-dev" --candidates=100 \
85 +            --match 'v*' | sed 's,^v,,')
86 +  if [ -z "$TAG_NAME" ]; then
87 +    TAG_NAME="0.0.0-$(cd $SOURCE_BASE &&
88 +              ( git update-index -q --refresh >/dev/null || true ) && \
89 +              git describe --abbrev=32 --dirty="-dev" --candidates=100 \
90 +              --match ROOT --always)"
91 +  fi
92 +
93 +  TAG_NAME=${TAG_NAME:-"0.0.0-undefined-tag"}
94 +  (cd $SOURCE_BASE && note "$(git status)")
95 +  PACKAGE_REPO=$REPO_BASE/$PACKAGE/$TAG_NAME
96 +  if [ ! -e $PACKAGE_REPO ]; then
97 +    set -x
98 +
99 +    build_lstore_package $PACKAGE $SOURCE_BASE $TAG_NAME \
100 +                           $PACKAGE_DISTRO
101 +    mkdir -p $PACKAGE_REPO
102 +
103 +    (
104 +      umask 000
105 +      cp *.${PACKAGE_SUFFIX} $PACKAGE_REPO
106 +      chmod 666 $PACKAGE_REPO/*

```

```

105 +         # Update lstore-release if we built it
106 +         if test -n "$(shopt -s nullglob; set +u; echo lstore-release*.rpm)"; then
107 +             cp lstore-release*.rpm $REPO_BASE/lstore-release.rpm
108 +         fi
109 +     )
110 +     set +x
111 + else
112 +     note "Tag $TAG_NAME of $PACKAGE already exists in:"
113 +     note "  ${PACKAGE_REPO}"
114 +     note "  Instead of building, we are installing from there."
115 +     fi
116 +     $PACKAGE_INSTALL $REPO_BASE/$PACKAGE/$TAG_NAME/*.${PACKAGE_SUFFIX}
117 +     popd
118 + done
119 +
120 +note "Done! The new packages can be found in ./package/$PACKAGE_SUBDIR"

```

122 scripts/package-internal.sh

[View](#)

```

@@ -7,6 +7,7 @@
7   7   #
8   8   # Preliminary bootstrapping
9   9   #
10  +
11  11  set -eu
12  12  ABSOLUTE_PATH=$(cd `dirname "${BASH_SOURCE[0]}"` && pwd)
13  13  source $ABSOLUTE_PATH/functions.sh
@@ -22,100 +23,65 @@ ubuntu-*|debian-*)
22  23      # switch to gdebi if automatic dependency resolution is needed
23  24      PACKAGE_INSTALL="dpkg -i"
24  25      PACKAGE_SUFFIX=deb
26  +
27  27      CMAKE_ARGS="-DINSTALL_DEB_RELEASE=ON"
28  28      ;;
29  29      centos-*)
30  30      PACKAGE_INSTALL="rpm -i"
31  31      PACKAGE_SUFFIX=rpm
32  +
33  33      CMAKE_ARGS="-DINSTALL_YUM_RELEASE=ON"
34  34      ;;
35  35      *)
36  36          fatal "Unexpected distro name $DISTRO_NAME"
37  37          ;;
38  38      esac
39  39      # todo could probe this from docker variables
40  -PACKAGE_BASE="/tmp/lstore-release"
41  40      REPO_BASE=$LSTORE_RELEASE_BASE/build/package/$PACKAGE_SUBDIR
42  41      #-# Here and elsewhere, we need to set the umask when we write to the host-mounted
43  42      #-# paths. Otherwise users outside the container can't read/write files. But,
44  43      #-# we don't want to just blindly set umask 0000, in case there's some
45  44      #-# weirdness with how the installer works.
46  45      -(
47  46          umask 000
48  47          mkdir -p $PACKAGE_BASE/{build,cmake} $REPO_BASE
49  48          # That being said, umask appears to not work right...
50  49          # FIXME: Why is umask so sad?
51  50          chmod 777 $PACKAGE_BASE $PACKAGE_BASE/{build,cmake} $REPO_BASE
52  +
53  51      -check_cmake $PACKAGE_BASE/cmake
54  +PACKAGE_BASE=/tmp/lstore-package
55  +
56  50      note "Beginning packaging at $(date) for $PACKAGE_SUBDIR"
57  52      -#
58  53      -# Get weird. Build each package, then install the RPM. If this is
59  54      -# configurable in the future, the order of packages matters.
60  55      -#
61  56      -cd $PACKAGE_BASE/build
62  57      -for PACKAGE in apr-accre apr-util-accre jerasure czmq \
63  58          toolbox gop ibp lio release meta; do
64  59          -#for PACKAGE in meta; do
65  60              SOURCE_BASE=$(get_repo_source_path ${PACKAGE})
66  61              if [ "$PACKAGE" == "czmq" ];then
67  62                  if (ldconfig -p | grep -q libczmq); then

```

```

63          echo "libczmq.so is available: skipping czmq package build.";
64          continue
65      else
66          echo "libczmq.so not found, building a czmq package...";
67      fi
68  fi

44  +TAG_NAME=$(cd $LSTORE_RELEASE_BASE &&
45  +          ( git update-index -q --refresh >/dev/null || true ) && \
46  +          git describe --abbrev=32 --dirty="-dev" --candidates=100 \
47  +          --match 'v*' | sed 's,v,,,')
48  +if [ -z "$TAG_NAME" ]; then
49  +    TAG_NAME="0.0.0-$(cd $LSTORE_RELEASE_BASE &&
50  +          ( git update-index -q --refresh >/dev/null || true ) && \
51  +          git describe --abbrev=32 --dirty="-dev" --candidates=100 \
52  +          --match ROOT --always)"
53  +fi

69  54
70  (
71      umask 000
72      mkdir -p $PACKAGE
73      # See above.
74      chmod 777 $PACKAGE
75  )
76  pushd $PACKAGE
77  # NOTE: Can't do this with --git-dir. Git 1.8.3.1 considers the tag "dirty"
78  #       if the CURRENT working directory is dirty...
79  # NOTE: The git update-index is needed since the host and container git
80  #       versions might be different.
81  TAG_NAME=$(cd $SOURCE_BASE &&
82  +          ( git update-index -q --refresh >/dev/null || true ) && \
83  +          git describe --abbrev=32 --dirty="-dev" --candidates=100 \
84  +          --match 'v*' | sed 's,^v,,,')
85  if [ -z "$TAG_NAME" ]; then
86  +    TAG_NAME="0.0.0-$(cd $SOURCE_BASE &&
87  +          ( git update-index -q --refresh >/dev/null || true ) && \
88  +          git describe --abbrev=32 --dirty="-dev" --candidates=100 \
89  +          --match ROOT --always)"
90  +fi

55  +TAG_NAME=${TAG_NAME:-"0.0.0-undefined-tag"}
56  +
57  +(cd $LSTORE_RELEASE_BASE && note "$(git status)")
58  +PACKAGE_REPO=$REPO_BASE/$TAG_NAME
59  +
60  +set -x
61  +mkdir -p $PACKAGE_BASE/build
62  +cp -r ${LSTORE_RELEASE_BASE}/{scripts,src,vendor,doc,debian,test,cmake,CMakeLists.txt,lstore.spec} \
63  +      $PACKAGE_BASE
64  +
65  +if [[ $PACKAGE_SUFFIX == deb ]]; then
66  +    cd $PACKAGE_BASE
67  +    cmake .
68  +    make $PACKAGE_SUFFIX VERBOSE=1
69  +else
70  +    cd $PACKAGE_BASE/build
71  +    cmake ..
72  +    make $PACKAGE_SUFFIX VERBOSE=1
73  +fi

91  74
92  - TAG_NAME=${TAG_NAME:-"0.0.0-undefined-tag"}
93  - (cd $SOURCE_BASE && note "$(git status)")
94  - PACKAGE_REPO=$REPO_BASE/$PACKAGE/$TAG_NAME
95  - if [ ! -e $PACKAGE_REPO ]; then
96  -     set -x
97  -     build_lstore_package $PACKAGE $SOURCE_BASE $TAG_NAME \
98  -                           $PACKAGE_DISTRO
99  -     mkdir -p $PACKAGE_REPO
100 -     # Need to figure out what to do with these eventually.
101 -     rm *.source.rpm || true
102 -     (
103 -         umask 000
104 -         cp *.${PACKAGE_SUFFIX} $PACKAGE_REPO
105 -         chmod 666 $PACKAGE_REPO/*
106 -         # Update lstore-release if we built it
107 -         if test -n "$($shopt -s nullglob; set +u; echo lstore-release*.rpm)"; then

```

```

108      cp lstore-release*.rpm $REPO_BASE/lstore-release.rpm
109      fi
110      )
111      set +x
112      else
113          note "Tag $TAG_NAME of $PACKAGE already exists in:"
114          note "${PACKAGE_REPO}"
115          note "Instead of building, we are installing from there."
75
76      umask 000
77      mkdir -p $PACKAGE_REPO
78      cp -r ${s}rpm_output/ $PACKAGE_REPO
79      chmod -R u=rwX,g=rwX,o=rwX $PACKAGE_REPO/*
80      # Update lstore-release if we built it
81      if test -n "$(shopt -s nullglob; set +u; echo lstore-release*.rpm)"; then
82          cp lstore-release*.rpm $REPO_BASE/lstore-release.rpm
116      fi
117      $PACKAGE_INSTALL $REPO_BASE/$PACKAGE/$TAG_NAME/*.${PACKAGE_SUFFIX}
118      popd
119      -done
84
85      +)
+set +x
120      86
121      87      note "Done! The new packages can be found in ./package/$PACKAGE_SUBDIR"

```

27 ████ █ scripts/package.sh

View

	@@ -32,18 +32,43 @@ for PROXY in http_proxy HTTPS_PROXY; do
32	32 fi
33	33 done
34	34
	+## Set up ccache to speed multiple compilations
35	+if [[-z "\${CCACHE_DIR:-}"]]; then
36	+ CCACHE_DEFAULT=1
37	+ CCACHE_DIR=/tmp/source/build/ccache
38	+ mkdir -p \$CCACHE_DIR
39	+fi
40	+
41	if [[! -z "\${HOST_VOLUME_PATH:-}" && ! -z "\${CONTAINER_VOLUME_PATH:-}"]]; then
42	LSTORE_RELEASE_RELATIVE="\${HOST_VOLUME_PATH}/\$(realpath \$(pwd) --relative-to \${CONTAINER_VOLUME_PATH})"
43	+ CCACHE_DIR_RELATIVE="\${HOST_VOLUME_PATH}/\$(realpath "\$CCACHE_DIR" --relative-to \${CONTAINER_VOLUME_PATH})"
44	else
45	LSTORE_RELEASE_RELATIVE="\$LSTORE_RELEASE_BASE"
46	CCACHE_DIR_RELATIVE="\$CCACHE_DIR"
47	+fi
48	+
49	+note "ccache default: \${CCACHE_DEFAULT:-} ccache_dir \$CCACHE_DIR"
50	+if [[-z "\${CCACHE_DEFAULT:-}" && -d "\$CCACHE_DIR"]]; then
51	+ EXTRA_ARGS="\$EXTRA_ARGS -e CCACHE_DIR=/tmp/ccache"
52	+ EXTRA_ARGS="\$EXTRA_ARGS -v \$CCACHE_DIR_RELATIVE:/tmp/ccache"
53	+else
54	+ EXTRA_ARGS="\$EXTRA_ARGS -e CCACHE_DIR=\$CCACHE_DIR"
55	fi
56	for DISTRO in "\${DISTROS[@]}"; do
57	note "Starting docker container to package \$DISTRO"
58	case \$DISTRO in
59	centos*)
60	INTERNAL_CMD="/tmp/source/scripts/package-internal.sh \$DISTRO"
61	;;
62	debian* ubuntu*)
63	INTERNAL_CMD="/tmp/source/scripts/package-internal.sh \$DISTRO"
64	;;
65	esac
66	set -x
67	docker run --rm=true -v \$LSTORE_RELEASE_RELATIVE:/tmp/source \ \$EXTRA_ARGS \ lstore/builder:\$DISTRO \
68	/tmp/source/scripts/package-internal.sh \$DISTRO
69	\$INTERNAL_CMD
70	set +x
71	done

2 scripts/test-repo-internal.sh

23	23	@@ -23,7 +23,7 @@ gpgcheck=0
24	24	protect=1
25	25	EOF
26	-	yum install -y epel-release
26	+	yum install -y accre-lio which
27	27	yum clean all
28	28	;;
29	29	ubuntu-* debian-*)

12 src/ibp/ibp_config.c

147	147	@@ -147,7 +147,7 @@ int ibp_rw_submit_coalesce(tbx_stack_t *stack, tbx_stack_ele_t *ele)
148	148	op_generic_t *gop = (op_generic_t *)tbx_get_stack_ele_data(ele);
149	149	ibp_op_t *iop = ibp_get_iop(gop);
149	149	ibp_context_t *ic = iop->ic;
150	-	ibp_op_rw_t *cmd = &(iop->rw_op);
150	+	+ ibp_op_rw_t *cmd = &(iop->ops.rw_op);
151	151	rw_coalesce_t *rwc;
152	152	tbx_pch_t pch;
153	153	
195	195	@@ -195,7 +195,7 @@ int ibp_rw_coalesce(op_generic_t *gop1)
196	196	ibp_op_t *iop1 = ibp_get_iop(gop1);
197	197	ibp_op_t *iop2;
197	197	ibp_context_t *ic = iop1->ic;
198	-	ibp_op_rw_t *cmd1 = &(iop1->rw_op);
198	+	+ ibp_op_rw_t *cmd1 = &(iop1->ops.rw_op);
199	199	ibp_op_rw_t *cmd2;
200	200	op_generic_t *gop2;
201	201	ibp_rw_buf_t **rdbuf;
266	266	@@ -266,7 +266,7 @@ int ibp_rw_coalesce(op_generic_t *gop1)
267	267	do {
268	268	 gop2 = (op_generic_t *)tbx_get_stack_ele_data(ele);
268	268	 iop2 = ibp_get_iop(gop2);
269	-	 cmd2 = &(iop2->rw_op);
269	+	+ cmd2 = &(iop2->ops.rw_op);
270	270	
271	271	 /** Unlink the element if needed
272	272	 if (gop2 != gop1) {
272	272	@@ -308,7 +308,7 @@ int ibp_rw_coalesce(op_generic_t *gop1)
308	308	 gop2 = (op_generic_t *)tbx_get_stack_ele_data(ele);
309	309	 if (gop2 == gop1) {
310	310	 iop2 = ibp_get_iop(gop2);
311	-	 cmd2 = &(iop2->rw_op);
311	+	+ cmd2 = &(iop2->ops.rw_op);
312	312	 rdbuf[n] = &(cmd2->buf_single);
313	313	 iov_sum += cmd2->n_tbx_iovec_total;
314	314	 workload += cmd2->size;
374	374	@@ -374,8 +374,8 @@ void _ibp_op_free(op_generic_t *gop, int mode)
375	375	 if (gop->op->cmd.coalesced_ops != NULL) { /** Coalesced R/W op so free the stack
376	376	 log_printf(15, "gid=%d Freeing rdbuf\n", gop_id(gop));
376	376	 iop = ibp_get_iop(gop);
377	-	 tbx_pch_release(iop->ic->coalesced_gop_stacks, &(iop->rw_op.rwcg_pch));
378	-	 free(iop->rw_op.rdbuf);
377	+	+ tbx_pch_release(iop->ic->coalesced_gop_stacks, &(iop->ops.rw_op.rwcg_pch));
378	+	+ free(iop->ops.rw_op.rdbuf);
379	379	 gop->op->cmd.coalesced_ops = NULL;
380	380	}
381	381	 gop_generic_free(gop, OP_FINALIZE); /** I free the actual op

134 src/ibp/ibp_op.c

431	431	@@ -431,10 +431,10 @@ void set_ibp_vec_read_op(ibp_op_t *op, ibp_cap_t *cap, int n_vec, ibp_tbx_iovec_
432	432	 op_generic_t *gop = ibp_get_gop(op);

```

433 433     set_ibp_rw_op(op, IBP_READ, cap, 0, buffer, boff, len, timeout);
434 -     op->rw_op.n_ops = 1;
435 -     op->rw_op.n_tbx_iovec_total = n_vec;
436 -     op->rw_op.buf_single.n_iovec = n_vec;
437 -     op->rw_op.buf_single.iovec = vec;
438 438
439 439         gop->op->cmd.send_command = vec_read_command;
440 440     }
441 @@ -466,7 +466,7 @@ op_status_t vec_read_command(op_generic_t *gop, tbx_ns_t *ns)
442     ibp_rw_buf_t *rdbuf;
443     op_status_t err;
444
445 -     cmd = &(op->rw_op);
446 +     cmd = &(op->ops.rw_op);
447
448     used = 0;
449
450 @@ -522,7 +522,7 @@ op_status_t read_command(op_generic_t *gop, tbx_ns_t *ns)
451     op_status_t err;
452     ibp_op_rw_t *cmd;
453
454 -     cmd = &(op->rw_op);
455 +     cmd = &(op->ops.rw_op);
456
457     if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {
458 //     tbx_ns_chksum_reset(&(op->ncs));
459
460 @@ -599,7 +599,7 @@ op_status_t read_recv(op_generic_t *gop, tbx_ns_t *ns)
461     double swait;
462     ibp_rw_buf_t *rdbuf;
463
464 -     cmd = &(op->rw_op);
465 +     cmd = &(op->ops.rw_op);
466
467     /* Need to read the depot status info
468      log_printf(15, "read_recv: ns=%d starting command size=%" OT "\n", tbx_ns_getid(ns), cmd->size);
469
470 @@ -687,7 +687,7 @@ op_generic_t *new_ibp_write_op(ibp_context_t *ic, ibp_cap_t *cap, ibp_off_t offs
471     op_generic_t *gop = new_ibp_rw_op(ic, IBP_WRITE, cap, offset, buffer, bpos, len, timeout);
472 //     ibp_op_t *iop = ibp_get_iop(gop);
473
474 -// log_printf(15, "new_ibp_write_op: gid=%d next_block=%p\n", gop_id(gop), iop->rw_op.next_block); tbx_flush_log();
475 +// log_printf(15, "new_ibp_write_op: gid=%d next_block=%p\n", gop_id(gop), iop->ops.rw_op.next_block); tbx_flush_log();
476
477     return(gop);
478
479
480 @@ -700,10 +700,10 @@ void set_ibp_vec_write_op(ibp_op_t *op, ibp_cap_t *cap, int n_iovec, ibp_tbx_iov
481     op_generic_t *gop = ibp_get_gop(op);
482
483     set_ibp_rw_op(op, IBP_WRITE, cap, 0, buffer, bpos, len, timeout);
484
485 -     op->rw_op.n_ops = 1;
486 -     op->rw_op.n_tbx_iovec_total = n_iovec;
487 -     op->rw_op.buf_single.n_iovec = n_iovec;
488 -     op->rw_op.buf_single.iovec = iovec;
489
490 +     op->ops.rw_op.n_ops = 1;
491 +     op->ops.rw_op.n_tbx_iovec_total = n_iovec;
492 +     op->ops.rw_op.buf_single.n_iovec = n_iovec;
493 +     op->ops.rw_op.buf_single.iovec = iovec;
494
495     gop->op->cmd.send_command = vec_write_command;
496
497
498 @@ -736,7 +736,7 @@ op_status_t vec_write_command(op_generic_t *gop, tbx_ns_t *ns)
499     ibp_op_rw_t *cmd;
500     ibp_rw_buf_t *rdbuf;
501
502 -     cmd = &(op->rw_op);
503 +     cmd = &(op->ops.rw_op);
504
505     used = 0;

```

```

@@ -797,7 +797,7 @@ op_status_t write_command(op_generic_t *gop, tbx_ns_t *ns)
797   797     op_status_t err;
798   798     ibp_op_rw_t *cmd;
799   799
800   - cmd = &(op->rw_op);
800   + cmd = &(op->ops.rw_op);
801   801     // log_printf(10, "write_command: gid=%d next_block=%p\n", gop_id(gop), cmd->next_block);
802   802
803   803
@@ -873,7 +873,7 @@ op_status_t write_send(op_generic_t *gop, tbx_ns_t *ns)
873   873     ibp_op_t *iop = ibp_get_iop(gop);
874   874     int i;
875   875     op_status_t err;
876   - ibp_op_rw_t *cmd = &(iop->rw_op);
876   + ibp_op_rw_t *cmd = &(iop->ops.rw_op);
877   877     ibp_rw_buf_t *rdbuf;
878   878
879   879     // log_printf(10, "write_send: START gid=%d next_block=%p\n", gop_id(gop), cmd->next_block);
@@ -928,7 +928,7 @@ op_status_t write_recv(op_generic_t *gop, tbx_ns_t *ns)
928   928
929   929     log_printf(15, "write_recv: Start!!! ns=%d\n", tbx_ns_getid(ns));
930   930
931   - cmd = &(op->rw_op);
931   + cmd = &(op->ops.rw_op);
932   932     tbx_ns_chksum_read_disable(ns);
933   933
934   934
@@ -986,7 +986,7 @@ op_status_t append_command(op_generic_t *gop, tbx_ns_t *ns)
986   986     op_status_t err;
987   987     ibp_op_rw_t *cmd;
988   988
989   - cmd = &(op->rw_op);
989   + cmd = &(op->ops.rw_op);
990   990
991   991     if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {
992   //     tbx_ns_chksum_reset(&(op->ncs));
@@ -1053,7 +1053,7 @@ void set_ibp_rw_op(ibp_op_t *op, int rw_type, ibp_cap_t *cap, ibp_off_t offset,
1053  1053     ibp_op_rw_t *cmd;
1054  1054     ibp_rw_buf_t *rdbuf;
1055  1055
1056   - cmd = &(op->rw_op);
1056   + cmd = &(op->ops.rw_op);
1057  1057
1058  1058     init_ibp_base_op(op, "rw", timeout, op->ic->rw_new_command + len, NULL, len, rw_type, IBP_NOP);
1059  1059     op_generic_t *gop = ibp_get_gop(op);
@@ -1124,7 +1124,7 @@ op_status_t validate_chksum_command(op_generic_t *gop, tbx_ns_t *ns)
1124  1124     ibp_op_t *op = ibp_get_iop(gop);
1125  1125     char buffer[1024];
1126  1126     op_status_t err;
1127   - ibp_op_validate_chksum_t *cmd = &(op->validate_op);
1127   + ibp_op_validate_chksum_t *cmd = &(op->ops.validate_op);
1128  1128
1129  1129     snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d\n",
1130  1130             IBPv040, IBP_VALIDATE_CHKSUM, cmd->key, cmd->typekey, cmd->correct_errors,
@@ -1149,7 +1149,7 @@ op_status_t validate_chksum_recv(op_generic_t *gop, tbx_ns_t *ns)
1149  1149     op_status_t err;
1150  1150     int status, fin;
1151  1151     ibp_off_t nerrors;
1152   - ibp_op_validate_chksum_t *cmd = &(op->validate_op);
1152   + ibp_op_validate_chksum_t *cmd = &(op->ops.validate_op);
1153  1153     char *bstate;
1154  1154     double swait;
1155  1155
@@ -1187,13 +1187,13 @@ void set_ibp_validate_chksum_op(ibp_op_t *op, ibp_cap_t *mcap, int correct_error
1187  1187     init_ibp_base_op(op, "validate_chksum", timeout, op->ic->other_new_command, NULL, 1, IBP_VALIDATE_CHKSUM, IBP_NOP);
1188  1188     op_generic_t *gop = ibp_get_gop(op);
1189  1189
1190   - cmd = &(op->validate_op);
1190   + cmd = &(op->ops.validate_op);
1191  1191
1192  1192     parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);

```

```

1193 1193      set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_VALIDATE_CHKSUM]));
1194 1194      op->dop.cmd.hostport = strdup(hoststr);
1195 1195
1196 -     cmd = &(op->validate_op);
1196 +     cmd = &(op->ops.validate_op);
1197 1197     cmd->correct_errors = correct_errors;
1198 1198     cmd->n_bad_blocks = n_bad_blocks;
1199 1199
✖ @@ -1227,7 +1227,7 @@ op_status_t get_chksum_command(op_generic_t *gop, tbx_ns_t *ns)
1227 1227     ibp_op_t *op = ibp_get_iop(gop);
1228 1228     char buffer[1024];
1229 1229     op_status_t err;
1230 -     ibp_op_get_chksum_t *cmd = &(op->get_chksum_op);
1230 +     ibp_op_get_chksum_t *cmd = &(op->ops.get_chksum_op);
1231 1231     sprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d\n",
1232 1232         IBPv040, IBP_GET_CHKSUM, cmd->key, cmd->typekey, cmd->checksum_info_only,
1233 1233
✖ @@ -1251,7 +1251,7 @@ op_status_t get_chksum_recv(op_generic_t *gop, tbx_ns_t *ns)
1251 1251     char buffer[1024];
1252 1252     op_status_t err;
1253 1253     int status, fin;
1254 -     ibp_op_get_chksum_t *cmd = &(op->get_chksum_op);
1254 +     ibp_op_get_chksum_t *cmd = &(op->ops.get_chksum_op);
1255 1255     char *bstate;
1256 1256     tbx_tbuf_t buf;
1257 1257
✖ @@ -1324,14 +1324,14 @@ void set_ibp_get_chksum_op(ibp_op_t *op, ibp_cap_t *mcap, int checksum_info_only,
1324 1324     init_ibp_base_op(op, "get_chksum", timeout, op->ic->other_new_command, NULL, 1, IBP_VALIDATE_CHKSUM, IBP_NOP);
1325 1325     op_generic_t *gop = ibp_get_gop(op);
1326 1326
1327 -     cmd = &(op->get_chksum_op);
1327 +     cmd = &(op->ops.get_chksum_op);
1328 1328
1329 1329     parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);
1330 1330     set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_VALIDATE_CHKSUM]));
1331 1331     op->dop.cmd.hostport = strdup(hoststr);
1332 1332
1333 1333     cmd->cap = mcap;
1334 -     cmd = &(op->get_chksum_op);
1334 +     cmd = &(op->ops.get_chksum_op);
1335 1335     cmd->checksum_info_only = checksum_info_only;
1336 1336     cmd->cs_type = cs_type;
1337 1337     cmd->cs_size = cs_size;
✖ @@ -1371,7 +1371,7 @@ op_status_t allocate_command(op_generic_t *gop, tbx_ns_t *ns)
1371 1371     char buffer[1024];
1372 1372     // apr_time_t atime, now;
1373 1373     op_status_t err;
1374 -     ibp_op_alloc_t *cmd = &(op->alloc_op);
1374 +     ibp_op_alloc_t *cmd = &(op->ops.alloc_op);
1375 1375
1376 1376     log_printf(10, "allocate_command: cs_type=%d\n", cmd->disk_checksum_type);
1377 1377
✖ @@ -1409,7 +1409,7 @@ op_status_t split_allocate_command(op_generic_t *gop, tbx_ns_t *ns)
1409 1409     ibp_op_t *op = ibp_get_iop(gop);
1410 1410     char buffer[1024];
1411 1411     op_status_t err;
1412 -     ibp_op_alloc_t *cmd = &(op->alloc_op);
1412 +     ibp_op_alloc_t *cmd = &(op->ops.alloc_op);
1413 1413
1414 1414     if (cmd->disk_checksum_type == CHKSUM_DEFAULT) { /** Normal split allocation
1415 1415         sprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d " I64T " %d\n",
✖ @@ -1445,7 +1445,7 @@ op_status_t allocate_recv(op_generic_t *gop, tbx_ns_t *ns)
1445 1445     char rcap[1025], wcap[1025], mcap[1025];
1446 1446     char *bstate;
1447 1447     op_status_t err;
1448 -     ibp_op_alloc_t *cmd = &(op->alloc_op);
1448 +     ibp_op_alloc_t *cmd = &(op->ops.alloc_op);
1449 1449
1450 1450     /** Need to read the depot status info
1451 1451     log_printf(15, "allocate_recv: ns=%d Start\n", tbx_ns_getid(ns));
✖ @@ -1516,7 +1516,7 @@ void set_ibp_alloc_op(ibp_op_t *op, ibp_capset_t *caps, ibp_off_t size, ibp_depo

```

5/19/2016

Merge 120ff4dbc33db1b410bbbcc681f4fd53c125c1e1 into e7213e748c17a9904... · accre/lstore@808a152

```

1516 1516     init_ibp_base_op(op, "alloc", timeout, op->ic->other_new_command, strdup(hoststr), 1, IBP_ALLOCATE, IBP_NOP);
1517 1517 //log_printf(15, "set_ibp_alloc_op: after init_ibp_base_op\n");
1518 1518
1519 -   cmd = &(op->alloc_op);
1519 +   cmd = &(op->ops.alloc_op);
1520 1520     cmd->caps = caps;
1521 1521     cmd->depot = depot;
1522 1522     cmd->attr = attr;
✖ @@ -1562,13 +1562,13 @@ void set_ibp_split_alloc_op(ibp_op_t *op, ibp_cap_t *mcap, ibp_capset_t *caps, i
1562 1562     init_ibp_base_op(op, "split_allocate", timeout, op->ic->other_new_command, NULL, 1, IBP_SPLIT_ALLOCATE, IBP_NOP);
1563 1563
1564 1564
1565 -   cmd = &(op->alloc_op);
1565 +   cmd = &(op->ops.alloc_op);
1566 1566     parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);
1567 1567     set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_SPLIT_ALLOCATE]));
1568 1568     op->dop.cmd.hostport = strdup(hoststr);
1569 1569
1570 1570
1571 -   cmd = &(op->alloc_op);
1571 +   cmd = &(op->ops.alloc_op);
1572 1572     cmd->caps = caps;
1573 1573     cmd->attr = attr;
1574 1574
✖ @@ -1610,7 +1610,7 @@ op_status_t rename_command(op_generic_t *gop, tbx_ns_t *ns)
1610 1610     ibp_op_t *op = ibp_get_iop(gop);
1611 1611     char buffer[1024];
1612 1612     op_status_t err;
1613 -   ibp_op_alloc_t *cmd = &(op->alloc_op);
1613 +   ibp_op_alloc_t *cmd = &(op->ops.alloc_op);
1614 1614     snprintf(buffer, sizeof(buffer), "%d %d %s %s %d\n",
1615 1615         IBPv040, IBP_RENAME, cmd->key, cmd->typekey, (int)apr_time_sec(gop->op->cmd.timeout));
1616 1616
✖ @@ -1640,7 +1640,7 @@ void set_ibp_rename_op(ibp_op_t *op, ibp_capset_t *caps, ibp_cap_t *mcap, int ti
1640 1640     init_ibp_base_op(op, "rename", timeout, op->ic->other_new_command, NULL, 1, IBP_RENAME, IBP_NOP);
1641 1641
1642 1642
1643 -   cmd = &(op->alloc_op);
1643 +   cmd = &(op->ops.alloc_op);
1644 1644     parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);
1645 1645     set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_RENAME]));
1646 1646
✖ @@ -1678,7 +1678,7 @@ op_status_t merge_command(op_generic_t *gop, tbx_ns_t *ns)
1678 1678     ibp_op_t *op = ibp_get_iop(gop);
1679 1679     char buffer[1024];
1680 1680     op_status_t err;
1681 -   ibp_op_merge_alloc_t *cmd = &(op->merge_op);
1681 +   ibp_op_merge_alloc_t *cmd = &(op->ops.merge_op);
1682 1682     snprintf(buffer, sizeof(buffer), "%d %d %s %s %s %d\n",
1683 1683         IBPv040, IBP_MERGE_ALLOCATE, cmd->mkey, cmd->mtypekey, cmd->ckey, cmd->ctypekey, (int)apr_time_sec(gop->op->cmd.timeout));
1684 1684
✖ @@ -1709,7 +1709,7 @@ void set_ibp_merge_alloc_op(ibp_op_t *op, ibp_cap_t *mcap, ibp_cap_t *ccap, int
1709 1709     init_ibp_base_op(op, "rename", timeout, op->ic->other_new_command, NULL, 1, IBP_RENAME, IBP_NOP);
1710 1710
1711 1711
1712 -   cmd = &(op->merge_op);
1712 +   cmd = &(op->ops.merge_op);
1713 1713     parse_cap(op->ic, mcap, host, &port, cmd->mkey, cmd->mtypekey);
1714 1714     set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_MERGE_ALLOCATE]));
1715 1715
✖ @@ -1747,7 +1747,7 @@ op_status_t alias_allocate_command(op_generic_t *gop, tbx_ns_t *ns)
1747 1747     ibp_op_t *op = ibp_get_iop(gop);
1748 1748     char buffer[1024];
1749 1749     op_status_t err;
1750 -   ibp_op_alloc_t *cmd = &(op->alloc_op);
1750 +   ibp_op_alloc_t *cmd = &(op->ops.alloc_op);
1751 1751     snprintf(buffer, sizeof(buffer), "%d %d %s %s " I64T " " I64T " %d %d\n",
1752 1752         IBPv040, IBP_ALIAS_ALLOCATE, cmd->key, cmd->typekey, cmd->offset, cmd->size, cmd->duration, (int)apr_time_sec(gop->op->cmd.timeout));
1753 1753
✖ @@ -1778,7 +1778,7 @@ void set_ibp_alias_alloc_op(ibp_op_t *op, ibp_capset_t *caps, ibp_cap_t *mcap, i

```

```

1778 1778
1779 1779
1780 1780
1781 - cmd = &(op->alloc_op);
1781 + cmd = &(op->ops.alloc_op);
1782 1782
1783 1783
1784 1784
    parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);
    set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_ALIAS_ALLOCATE]));
  @@ -1829,7 +1829,7 @@ op_status_t modify_count_command(op_generic_t *gop, tbx_ns_t *ns)
1829 1829
1830 1830
1831 1831
1832 - cmd = &(op->probe_op);
1832 + cmd = &(op->ops.probe_op);
1833 1833
1834 1834
1835 1835
    sprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d\n",
            IBPv040, cmd->cmd, cmd->key, cmd->mode, cmd->captyle, (int)apr_time_sec(gop->op->cmd.timeout));
  @@ -1851,7 +1851,7 @@ op_status_t alias_modify_count_command(op_generic_t *gop, tbx_ns_t *ns)
1851 1851
1852 1852
1853 1853
1854 - cmd = &(op->probe_op);
1854 + cmd = &(op->ops.probe_op);
1855 1855
1856 1856
1857 1857
    sprintf(buffer, sizeof(buffer), "%d %d %s %s %d %s %s %d\n",
            IBPv040, cmd->cmd, cmd->key, cmd->mode, cmd->captyle, cmd->mkey, cmd->mtypekey, (int)apr_time_sec(gop->
  @@ -1917,7 +1917,7 @@ void set_ibp_generic_modify_count_op(int command, ibp_op_t *op, ibp_cap_t *cap,
1917 1917
1918 1918
1919 1919
1920 - cmd = &(op->probe_op);
1920 + cmd = &(op->ops.probe_op);
1921 1921
1922 1922
1923 1923
    parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
    set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[command]));
  @@ -1993,7 +1993,7 @@ op_status_t modify_alloc_command(op_generic_t *gop, tbx_ns_t *ns)
1993 1993
1994 1994
1995 1995
1996 - cmd = &(op->mod_alloc_op);
1996 + cmd = &(op->ops.mod_alloc_op);
1997 1997
1998 1998
1999 1999
    atime = cmd->duration - time(NULL); /** This is in sec NOT APR time
    if (atime < 0) atime = cmd->duration;
  @@ -2029,7 +2029,7 @@ void set_ibp_modify_alloc_op(ibp_op_t *op, ibp_cap_t *cap, ibp_off_t size, int d
2029 2029
2030 2030
2031 2031
    init_ibp_base_op(op, "modify_alloc", timeout, op->ic->other_new_command, NULL, 1, IBP_MANAGE, IBP_CHNG);
2032 - cmd = &(op->mod_alloc_op);
2032 + cmd = &(op->ops.mod_alloc_op);
2033 2033
2034 2034
2035 2035
    parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
    set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_MANAGE]));
  @@ -2072,7 +2072,7 @@ op_status_t alias_modify_alloc_command(op_generic_t *gop, tbx_ns_t *ns)
2072 2072
2073 2073
2074 2074
    int atime;
    ibp_op_modify_alloc_t *cmd;
2075 - cmd = &(op->mod_alloc_op);
2075 + cmd = &(op->ops.mod_alloc_op);
2076 2076
2077 2077
2078 2078
    atime = cmd->duration - time(NULL); /** This is in sec NOT APR time
    if (atime < 0) atime = cmd->duration;
  @@ -2108,7 +2108,7 @@ void set_ibp_alias_modify_alloc_op(ibp_op_t *op, ibp_cap_t *cap, ibp_cap_t *mcap
2108 2108
2109 2109
2110 2110
    init_ibp_base_op(op, "alias_modify_alloc", timeout, op->ic->other_new_command, NULL, 1, IBP_ALIAS_MANAGE, IBP_CHNG);
2111 - cmd = &(op->mod_alloc_op);
2111 + cmd = &(op->ops.mod_alloc_op);
2112 2112
2113 2113
2114 2114
    parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
    set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_ALIAS_MANAGE]));

```

```

@@ -2152,7 +2152,7 @@ op_status_t truncate_command(op_generic_t *gop, tbx_ns_t *ns)
    op_status_t err;
    ibp_op_modify_alloc_t *cmd;

2155 - cmd = &(op->mod_alloc_op);
+ cmd = &(op->ops.mod_alloc_op);

2156 2156
2157 2157     sprintf(buffer, sizeof(buffer), "%d %d %s %s %d" I64T "%d\n",
2158      IBPv040, IBP_MANAGE, cmd->key, cmd->typekey, IBP_TRUNCATE, cmd->size, (int)apr_time_sec(gop->op->cmd.timeout));
@@ -2183,7 +2183,7 @@ void set_ibp_truncate_op(ibp_op_t *op, ibp_cap_t *cap, ibp_off_t size, int timeo
2183 2183     init_ibp_base_op(op, "truncate_alloc", timeout, op->ic->other_new_command, NULL, 1, IBP_MANAGE, IBP_CHNG);
2184 2184
2185 2185 - cmd = &(op->mod_alloc_op);
+ cmd = &(op->ops.mod_alloc_op);

2186 2186     parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
2187 2187     set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_MANAGE]));
@@ -2263,7 +2263,7 @@ op_status_t probe_command(op_generic_t *gop, tbx_ns_t *ns)
2263 2263     op_status_t err;
2264 2264     ibp_op_probe_t *cmd;

2266 - cmd = &(op->probe_op);
+ cmd = &(op->ops.probe_op);

2267 2267     sprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d 0 0 0 %d \n",
2268 2268      IBPv040, IBP_MANAGE, cmd->key, cmd->typekey, IBP_PROBE, IBP_MANAGECAP, (int)apr_time_sec(gop->op->cmd.timeout));
@@ -2301,7 +2301,7 @@ op_status_t probe_recv(op_generic_t *gop, tbx_ns_t *ns)
2301 2301     status = atoi(tbx_stk_string_token(buffer, " ", &bstate, &fin));
2302 2302     if ((status == IBP_OK) && (fin == 0)) {
2303 2303         p = op->probe_op.probe;
2304 - p = op->ops.probe_op.probe;
+ p = op->ops.probe_op.probe;
2305 2305         log_printf(15, "probe_recv: p=%p QWERT\n", p);
2306 2306         p->readRefCount = atoi(tbx_stk_string_token(NULL, " ", &bstate, &fin));
2307 2307         @@ -2337,7 +2337,7 @@ void set_ibp_probe_op(ibp_op_t *op, ibp_cap_t *cap, ibp_capstatus_t *probe, int
2337 2337         log_printf(15, "AFTER ctype=%d\n", op->ic->cc[IBP_MANAGE].type);
2338 2338         tbx_flush_log();
2339 2339
2340 - cmd = &(op->probe_op);
+ cmd = &(op->ops.probe_op);

2341 2341     parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
2342 2342     set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_MANAGE]));
@@ -2380,7 +2380,7 @@ op_status_t alias_probe_command(op_generic_t *gop, tbx_ns_t *ns)
2380 2380     op_status_t err;
2381 2381     ibp_op_probe_t *cmd;

2383 - cmd = &(op->probe_op);
+ cmd = &(op->ops.probe_op);

2384 2384     sprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d \n",
2385 2385      IBPv040, IBP_ALIAS_MANAGE, cmd->key, cmd->typekey, IBP_PROBE, IBP_MANAGECAP, (int)apr_time_sec(gop->op->cmd.timeout));
@@ -2418,7 +2418,7 @@ op_status_t alias_probe_recv(op_generic_t *gop, tbx_ns_t *ns)
2418 2418     status = atoi(tbx_stk_string_token(buffer, " ", &bstate, &fin));
2419 2419     if ((status == IBP_OK) && (fin == 0)) {
2420 2420         p = op->probe_op.alias_probe;
2421 - p = op->ops.probe_op.alias_probe;
+ p = op->ops.probe_op.alias_probe;

2422 2422         p->readRefCount = atoi(tbx_stk_string_token(NULL, " ", &bstate, &fin));
2423 2423         p->writeRefCount = atoi(tbx_stk_string_token(NULL, " ", &bstate, &fin));
2424 2424         p->offset = atol(tbx_stk_string_token(NULL, " ", &bstate, &fin));
@@ -2445,7 +2445,7 @@ void set_ibp_alias_probe_op(ibp_op_t *op, ibp_cap_t *cap, ibp_alias_capstatus_t
2445 2445     init_ibp_base_op(op, "alias_probe", timeout, op->ic->other_new_command, NULL, 1, IBP_ALIAS_MANAGE, IBP_PROBE);
2446 2446
2447 2447 - cmd = &(op->probe_op);
+ cmd = &(op->ops.probe_op);

2449 2449     parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);

```

```

2451 | 2451 | set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_ALIAS_MANAGE]));
2452 | @@ -2491,7 +2491,7 @@ op_status_t copyappend_command(op_generic_t *gop, tbx_ns_t *ns)
2453 |     op_status_t err;
2454 |     ibp_op_copy_t *cmd;
2455 | 
2456 |     cmd = &(op->copy_op);
2457 | 
2458 |     cmd = &(op->ops.copy_op);
2459 | 
2460 |     if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {
2461 |         sprintf(buffer, sizeof(buffer), "%d %d %s %s %s " I64T " " I64T " %d %d %d\n",
2462 |             @@ -2527,7 +2527,7 @@ op_status_t copy_recv(op_generic_t *gop, tbx_ns_t *ns)
2463 |             ibp_op_copy_t *cmd;
2464 |             double swait;
2465 | 
2466 |             cmd = &(op->copy_op);
2467 | 
2468 |             cmd = &(op->ops.copy_op);
2469 | 
2470 |             /** Need to read the depot status info
2471 |             log_printf(15, "copy_recv: ns=%d Start\n", tbx_ns_getid(ns));
2472 | 
2473 |             @@ -2566,7 +2566,7 @@ void set_ibp_copyappend_op(ibp_op_t *op, int ns_type, char *path, ibp_cap_t *src
2474 |                 init_ibp_base_op(op, "copyappend", src_timeout, op->ic->rw_new_command + size, NULL, size, IBP_SEND, IBP_NOP);
2475 | 
2476 |             cmd = &(op->copy_op);
2477 | 
2478 |             cmd = &(op->ops.copy_op);
2479 | 
2480 |             parse_cap(op->ic, srccap, host, &port, cmd->src_key, cmd->src_typekey);
2481 |             set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_SEND]));
2482 | 
2483 |             @@ -2631,7 +2631,7 @@ op_status_t pushpull_command(op_generic_t *gop, tbx_ns_t *ns)
2484 |                 op_status_t err;
2485 |                 ibp_op_copy_t *cmd;
2486 | 
2487 |                 cmd = &(op->copy_op);
2488 | 
2489 |                 cmd = &(op->ops.copy_op);
2490 | 
2491 |                 if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {
2492 |                     sprintf(buffer, sizeof(buffer), "%d %d %d %s %s %s %s " I64T " " I64T " %d %d %d\n",
2493 |                         @@ -2673,7 +2673,7 @@ void set_ibp_copy_op(ibp_op_t *op, int mode, int ns_type, char *path, ibp_cap_t
2494 |                             init_ibp_base_op(op, "copy", src_timeout, op->ic->rw_new_command + size, NULL, size, IBP_SEND, IBP_NOP);
2495 | 
2496 |                             cmd = &(op->copy_op);
2497 | 
2498 |                             cmd = &(op->ops.copy_op);
2499 | 
2500 |                             parse_cap(op->ic, srccap, host, &port, cmd->src_key, cmd->src_typekey);
2501 |                             set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_SEND]));
2502 | 
2503 |                             @@ -2738,7 +2738,7 @@ op_status_t depot_modify_command(op_generic_t *gop, tbx_ns_t *ns)
2504 |                                 op_status_t err;
2505 |                                 ibp_op_depot_modify_t *cmd;
2506 | 
2507 |                                 cmd = &(op->depot_modify_op);
2508 | 
2509 |                                 cmd = &(op->ops.depot_modify_op);
2510 | 
2511 |                                 sprintf(buffer, sizeof(buffer), "%d %d %s %d %s %d\n" I64T " " I64T " " TT "\n",
2512 |                                     IBPV040, IBP_STATUS, cmd->depot->rid.name, IBP_ST_CHANGE, cmd->password, (int)apr_time_sec(gop->op->cmd.timeout),
2513 | 
2514 |                                 @@ -2761,7 +2761,7 @@ op_status_t depot_modify_command(op_generic_t *gop, tbx_ns_t *ns)
2515 |                                     void set_ibp_depot_modify_op(ibp_op_t *op, ibp_depot_t *depot, char *password, ibp_off_t hard, ibp_off_t soft,
2516 |                                         int duration, int timeout)
2517 |                                     {
2518 |                                         ibp_op_depot_modify_t *cmd = &(op->depot_modify_op);
2519 | 
2520 |                                         ibp_op_depot_modify_t *cmd = &(op->ops.depot_modify_op);
2521 | 
2522 |                                         init_ibp_base_op(op, "depot_modify", timeout, op->ic->other_new_command, NULL,
2523 |                                             op->ic->other_new_command, IBP_STATUS, IBP_ST_CHANGE);
2524 | 
2525 |                                         @@ -2802,7 +2802,7 @@ op_status_t depot_inq_command(op_generic_t *gop, tbx_ns_t *ns)
2526 |                                             op_status_t err;
2527 |                                             ibp_op_depot_inq_t *cmd;
2528 | 
2529 |                                             cmd = &(op->depot_inq_op);
2530 | 
2531 |                                             cmd = &(op->ops.depot_inq_op);

```

```

2806 2806
2807 2807      tbx_ns_chksun_write_clear(ns);
2808 2808

✖ @@ -2881,7 +2881,7 @@ op_status_t depot_inq_recv(op_generic_t *gop, tbx_ns_t *ns)
2881 2881      char *bstate;
2882 2882      ibp_op_depot_inq_t *cmd;
2883 2883
2884 -    cmd = &(op->depot_inq_op);
2884 +    cmd = &(op->ops.depot_inq_op);
2885 2885
2886 2886      tbx_ns_chksun_read_clear(ns);
2887 2887

✖ @@ -2928,7 +2928,7 @@ void set_ibp_depot_inq_op(ibp_op_t *op, ibp_depot_t *depot, char *password, ibp_
2928 2928 {
2929 2929     char hoststr[MAX_HOST_SIZE];
2930 2930     char pchost[MAX_HOST_SIZE];
2931 -    ibp_op_depot_inq_t *cmd = &(op->depot_inq_op);
2931 +    ibp_op_depot_inq_t *cmd = &(op->ops.depot_inq_op);
2932 2932
2933 2933     ibppc_form_host(op->ic, pchost, sizeof(pchost), depot->host, depot->rid);
2934 2934     set_hostport(hoststr, sizeof(hoststr), pchost, depot->port, &(op->ic->cc[IBP_STATUS]));
✖ @@ -2989,7 +2989,7 @@ op_status_t depot_version_recv(op_generic_t *gop, tbx_ns_t *ns)
2989 2989     int pos, nmax, status, fin;
2990 2990     ibp_op_version_t *cmd;
2991 2991
2992 -    cmd = &(op->ver_op);
2992 +    cmd = &(op->ops.ver_op);
2993 2993
2994 2994     err = ibp_success_status;
2995 2995     pos = 0;
✖ @@ -3035,7 +3035,7 @@ void set_ibp_version_op(ibp_op_t *op, ibp_depot_t *depot, char *buffer, int buff
3035 3035 {
3036 3036     char hoststr[MAX_HOST_SIZE];
3037 3037     char pchoststr[MAX_HOST_SIZE];
3038 -    ibp_op_version_t *cmd = &(op->ver_op);
3038 +    ibp_op_version_t *cmd = &(op->ops.ver_op);
3039 3039
3040 3040     ibppc_form_host(op->ic, pchoststr, sizeof(pchoststr), depot->host, depot->rid);
3041 3041     set_hostport(hoststr, sizeof(hoststr), pchoststr, depot->port, &(op->ic->cc[IBP_STATUS]));
✖ @@ -3097,7 +3097,7 @@ op_status_t query_res_recv(op_generic_t *gop, tbx_ns_t *ns)
3097 3097     op_status_t err;
3098 3098     int fin, n, i, status;
3099 3099     char *p, *bstate;
3100 -    ibp_op_rid_inq_t *cmd = &(op->rid_op);
3100 +    ibp_op_rid_inq_t *cmd = &(op->ops.rid_op);
3101 3101
3102 3102     err = ibp_success_status;
3103 3103
✖ @@ -3142,7 +3142,7 @@ void set_ibp_query_resources_op(ibp_op_t *op, ibp_depot_t *depot, ibp_ridlist_t
3142 3142 {
3143 3143     char hoststr[MAX_HOST_SIZE];
3144 3144     char pchoststr[MAX_HOST_SIZE];
3145 -    ibp_op_rid_inq_t *cmd = &(op->rid_op);
3145 +    ibp_op_rid_inq_t *cmd = &(op->ops.rid_op);
3146 3146
3147 3147     ibppc_form_host(op->ic, pchoststr, sizeof(pchoststr), depot->host, depot->rid);
3148 3148     set_hostport(hoststr, sizeof(hoststr), pchoststr, depot->port, &(op->ic->cc[IBP_STATUS]));
✖

```

42 ████ src/ibp/ibp_op.h

[View](#)

```

✖ @@ -231,27 +231,27 @@ ibp_ridlist_t *rlist,
231 231     } ibp_op_rid_inq_t;
232 232
233 233     typedef struct _ibp_op_s { /* Individual IO operation
234 -ibp_context_t *ic;
235 -op_generic_t gop;
236 -op_data_t dop;
237 -tbx_stack_t *hp_parent; /* Only used for RW coalescing
238 -int primary_cmd; /* Primary sync IBP command family

```

```

239     -int sub_cmd;      /** sub command, if applicable
240     -tbx_ns_cksum_t ncs;  /** checksum associated with the command
241     -union {           /** Holds the individual commands options
242         -ibp_op_validate_cksum_t validate_op;
243         -ibp_op_get_cksum_t     get_cksum_op;
244         -ibp_op_alloc_t        alloc_op;
245         -ibp_op_merge_alloc_t  merge_op;
246         -ibp_op_probe_t        probe_op;
247         -ibp_op_rw_t           rw_op;
248         -ibp_op_copy_t         copy_op;
249         -ibp_op_depot_modify_t depot_modify_op;
250         -ibp_op_depot_inq_t   depot_inq_op;
251         -ibp_op_modify_alloc_t mod_alloc_op;
252         -ibp_op_rid_inq_t    rid_op;
253         -ibp_op_version_t    ver_op;
254     -};
254
234     + ibp_context_t *ic;
235     + op_generic_t gop;
236     + op_data_t dop;
237     + tbx_stack_t *hp_parent; /* Only used for RW coalescing
238     + int primary_cmd;/** Primary sync IBP command family
239     + int sub_cmd;      /** sub command, if applicable
240     + tbx_ns_cksum_t ncs;  /** checksum associated with the command
241     + union {           /** Holds the individual commands options
242         + ibp_op_validate_cksum_t validate_op;
243         + ibp_op_get_cksum_t     get_cksum_op;
244         + ibp_op_alloc_t        alloc_op;
245         + ibp_op_merge_alloc_t  merge_op;
246         + ibp_op_probe_t        probe_op;
247         + ibp_op_rw_t           rw_op;
248         + ibp_op_copy_t         copy_op;
249         + ibp_op_depot_modify_t depot_modify_op;
250         + ibp_op_depot_inq_t   depot_inq_op;
251         + ibp_op_modify_alloc_t mod_alloc_op;
252         + ibp_op_rid_inq_t    rid_op;
253         + ibp_op_version_t    ver_op;
254     + } ops;
255
255     } ibp_op_t;
256
256
257
257

```

4 src/lio/ds_ibp.c View

	<code>@@ -577,7 +577,7 @@ op_generic_t *ds_ibp_res_inquire(data_service_fn_t *dsf, char *res, data_attr_t</code>
577 577	<code>ds_ibp_alloc_op_t *cmd;</code>
578 578	
579 579	<code>ds_ibp_op_t *iop = ds_ibp_op_create(ds, attr);</code>
580 -	<code>cmd = &(iop->alloc);</code>
580 +	<code>cmd = &(iop->ops.alloc);</code>
581 581	
582 582	<code>/* Fill in the depot structure</code>
583 583	<code>res2ibp(res, &(cmd->depot));</code>
	<code>@@ -601,7 +601,7 @@ op_generic_t *ds_ibp_allocate(data_service_fn_t *dsf, char *res, data_attr_t *da</code>
601 601	<code>ds_ibp_alloc_op_t *cmd;</code>
602 602	
603 603	<code>ds_ibp_op_t *iop = ds_ibp_op_create(ds, attr);</code>
604 -	<code>cmd = &(iop->alloc);</code>
604 +	<code>cmd = &(iop->ops.alloc);</code>
605 605	
606 606	<code>/* Fill in the depot structure</code>
607 607	<code>res2ibp(res, &(cmd->depot));</code>

2 src/lio/ds_ibp_priv.h View

	<code>@@ -53,7 +53,7 @@ typedef struct {</code>
53 53	<code>union {</code>
54 54	<code> ds_ibp_alloc_op_t alloc;</code>
55 55	<code> ds_ibp_truncate_op_t truncate;</code>
56 -	<code>};</code>

	56	+ } ops;
	57	} ds_ibp_op_t ;
	58	
	59	typedef struct {
diff		

4	src/lio/ex3_types.h	View
	51	@@ -51,9 +51,7 @@ typedef struct {
	52	typedef struct {
	53	int type;
	54	- union {
	55	- exnode_text_t text;
	56	- } ;
	57	+ exnode_text_t text;
	58	} exnode_exchange_t;
	59	#ifdef __cplusplus
diff		

2	src/lio/segment_lun.c	View
	2470	@@ -2470,7 +2470,7 @@ op_status_t seglun_clone_func(void *arg, int id)
	2471	dtus = gop->op->cmd.end_time - gop->op->cmd.start_time;
	2472	dts = (1.0*dtus) / (1.0*APR_USEC_PER_SEC);
	2473	ibp_op_t *iop = ibp_get_iop (gop);
	2473	- ibp_op_copy_t *cmd = &(iop->copy_op);
	2473	+ ibp_op_copy_t *cmd = &(iop-> ops .copy_op);
	2474	status = gop_get_status (gop);
	2475	log_printf (5, "clone_dt src=%s dest=%s gid=%d status=(%d %d) dt=%lf\n", cmd->srrcap, cmd->destcap, gop_id (gop), status.op_
	2476	
diff		

32	src/toolbox/packer.c	View
	63	@@ -63,7 +63,7 @@ int tbx_pack_write_flush(tbx_pack_t *pack) {
	64	int pack_read_zlib(tbx_pack_t *pack, unsigned char *data, int len)
	65	{
	66	- tbx_pack_zlib_t *p = &(pack->zlib);
	66	+ tbx_pack_zlib_t *p = &(pack-> data .zlib);
	67	int nbytes;
	68	
	69	p->z.avail_out = len;
diff		
	89	@@ -89,7 +89,7 @@ int pack_read_zlib(tbx_pack_t *pack, unsigned char *data, int len)
	90	int pack_read_new_data_zlib(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
	91	{
	92	- tbx_pack_zlib_t *p = &(pack->zlib);
	92	+ tbx_pack_zlib_t *p = &(pack-> data .zlib);
	93	int err = 0;
	94	
	95	if (p->z.avail_in > 0) err = PACK_ERROR;
diff		
	110	@@ -110,7 +110,7 @@ int pack_read_new_data_zlib(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
	111	int pack_write_zlib(tbx_pack_t *pack, unsigned char *data, int len)
	112	{
	113	- tbx_pack_zlib_t *p = &(pack->zlib);
	113	+ tbx_pack_zlib_t *p = &(pack-> data .zlib);
	114	int nbytes;
	115	
	116	if (len == 0) return (0);
diff		
	134	@@ -134,7 +134,7 @@ int pack_write_zlib(tbx_pack_t *pack, unsigned char *data, int len)
	135	void pack_write_resized_zlib(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
	136	{
	137	- tbx_pack_zlib_t *p = &(pack->zlib);
	137	+ tbx_pack_zlib_t *p = &(pack-> data .zlib);

```

138 138     int offset;
139 139     assert(bufsize >= (p->bufsize - p->z.avail_out));
140 140 
@@ -153,7 +153,7 @@ void pack_write_resized_zlib(tbx_pack_t *pack, unsigned char *buffer, unsigned i
153 153     void pack_consumed_zlib(tbx_pack_t *pack)
154 154     {
155 155 - tbx_pack_zlib_t *p = &(pack->zlib);
156 156 + tbx_pack_zlib_t *p = &(pack->data.zlib);
157 157     p->z.avail_out = p->bufsize;
158 158     p->z.next_out = p->buffer;
159 159 @@ -165,7 +165,7 @@ void pack_consumed_zlib(tbx_pack_t *pack)
165 165     void pack_end_zlib(tbx_pack_t *pack)
166 166     {
167 167 - tbx_pack_zlib_t *p = &(pack->zlib);
168 168 + tbx_pack_zlib_t *p = &(pack->data.zlib);
169 169     if (pack->mode == PACK_READ) {
170 170         inflateEnd(&(p->z));
171 171 @@ -182,7 +182,7 @@ void pack_end_zlib(tbx_pack_t *pack)
182 182     int pack_used_zlib(tbx_pack_t *pack)
183 183     {
184 184 - tbx_pack_zlib_t *p = &(pack->zlib);
185 185 + tbx_pack_zlib_t *p = &(pack->data.zlib);
186 186     return(p->bufsize - p->z.avail_out);
187 187 }
188 188 @@ -194,7 +194,7 @@ int pack_used_zlib(tbx_pack_t *pack)
194 194     int pack_write_flush_zlib(tbx_pack_t *pack)
195 195     {
196 196 - tbx_pack_zlib_t *p = &(pack->zlib);
197 197 + tbx_pack_zlib_t *p = &(pack->data.zlib);
198 198     int err;
199 199     log_printf(5, "start avail_out=%d\n", p->z.avail_out);
200 200 @@ -221,7 +221,7 @@ int pack_write_flush_zlib(tbx_pack_t *pack)
221 221     void pack_init_zlib(tbx_pack_t *pack, int type, int mode, unsigned char *buffer, unsigned int bufsize)
222 222     {
223 223 - tbx_pack_zlib_t *p = &(pack->zlib);
224 224 + tbx_pack_zlib_t *p = &(pack->data.zlib);
225 225     memset(pack, 0, sizeof(tbx_pack_t));
226 226 @@ -274,7 +274,7 @@ void pack_init_zlib(tbx_pack_t *pack, int type, int mode, unsigned char *buffer,
274 274     int pack_read_raw(tbx_pack_t *pack, unsigned char *data, int len)
275 275     {
276 276 - tbx_pack_raw_t *p = &(pack->raw);
277 277 + tbx_pack_raw_t *p = &(pack->data.raw);
278 278     int nbytes;
279 279     nbytes = (len > p->nleft) ? p->nleft : len;
280 280 @@ -297,7 +297,7 @@ int pack_read_raw(tbx_pack_t *pack, unsigned char *data, int len)
297 297     int pack_read_new_data_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
298 298     {
299 299 - tbx_pack_raw_t *p = &(pack->raw);
300 300 + tbx_pack_raw_t *p = &(pack->data.raw);
301 301     int err = 0;
302 302     if (p->nleft > 0) err = PACK_ERROR;
303 303 @@ -319,7 +319,7 @@ int pack_read_new_data_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int
319 319     int pack_write_raw(tbx_pack_t *pack, unsigned char *data, int len)
320 320     {

```

5/19/2016

Merge 120ff4dbc33db1b410bbbcc681f4fd53c125c1e1 into e7213e748c17a9904... · accre/lstore@808a152

```

322 | 322 | - tbx_pack_raw_t *p = &(pack->raw);
323 | 323 | + tbx_pack_raw_t *p = &(pack->data.raw);
324 | 324 |
325 | 325 |     nbytes = (len > p->nleft) ? p->nleft : len;
326 | @@@ -342,7 +342,7 @@@ int pack_write_raw(tbx_pack_t *pack, unsigned char *data, int len)
342 | 342 |
343 | 343 |     void pack_write_resized_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
344 | 344 |     {
345 | 345 |         tbx_pack_raw_t *p = &(pack->raw);
346 | 346 |         tbx_pack_raw_t *p = &(pack->data.raw);
347 | 347 |         assert(bufsize >= p->bpos);
348 | 348 |
349 | @@@ -357,7 +357,7 @@@ void pack_write_resized_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
357 | 357 |
358 | 358 |     void pack_consumed_raw(tbx_pack_t *pack)
359 | 359 |     {
360 | 360 |         tbx_pack_raw_t *p = &(pack->raw);
361 | 361 |         tbx_pack_raw_t *p = &(pack->data.raw);
362 | 362 |         p->bpos = 0;
363 | 363 |         p->nleft = p->bufsize;
364 | @@@ -378,7 +378,7 @@@ void pack_end_raw(tbx_pack_t *pack)
378 | 378 |
379 | 379 |     int pack_used_raw(tbx_pack_t *pack)
380 | 380 |     {
381 | 381 |         return(pack->raw.bpos);
382 | 382 |         return(pack->data.raw.bpos);
383 | 383 |
384 | 384 | //*****
```

4  src/toolbox/packer.h View

```

25 | @@@ -25,6 +25,8 @@@
25 | 25 | #include "zlib.h"
26 |
27 | #ifdef __cplusplus
28 | +id (*write_resized)(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize);
29 |
30 | 30 | extern "C" {
31 | 31 | #endif
32 | @@@ -48,7 +50,7 @@@ struct tbx_pack_t {
33 |
34 | 48 |     union {
35 | 49 |         tbx_pack_raw_t raw;
36 | 50 |         tbx_pack_zlib_t zlib;
37 |
38 | 51 |         };
39 | 53 |     } data;
40 |
41 | 52 |     void (*end)(tbx_pack_t *pack);
42 | 53 |     void (*write_resized)(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize);
43 | 54 |     int (*write)(tbx_pack_t *pack, unsigned char *data, int nbytes);
44 | @@@
```

0 comments on commit 808a152

🔒 Lock conversation

Write

Preview

AA

B

i

“

<

›

≡

≡

≡

←

→

@

★

5/19/2016

Merge 120ff4dbc33db1b410bbbcc681f4fd53c125c1e1 into e7213e748c17a9904... · accre/lstore@808a152

Leave a comment

Attach files by dragging & dropping, [selecting them](#), or pasting from the clipboard.

 Styling with Markdown is supported

[Comment on this commit](#)



[◀×](#) [Unsubscribe](#)

You're receiving notifications because you're subscribed to this repository.

© 2016 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Contact](#) [Help](#)



[Status](#) [API](#) [Training](#) [Shop](#) [Blog](#) [About](#)