

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

[Browse files](#)

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-
6 6 tags>]]")
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
16 16
17 17
18 18
19 19
20 20
21 21
22 22
23 23
24 24
25 25
26 26
27 27
28 28
29 29
30 30
31 31
32 32
33 33
34 34
35 35
36 36
37 37
38 38
39 39
40 40
41 41
42 42
43 43
44 44
45 45
46 46
47 47
48 48
49 49
50 50
51 51
52 52
53 53
54 54
55 55
56 56
57 57
58 58
59 59
60 60
61 61
62 62
63 63
64 64
65 65
66 66
67 67
68 68
69 69
70 70
71 71
72 72
73 73
74 74
75 75
76 76
77 77
78 78
79 79
80 80
81 81
82 82
83 83
84 84
85 85
86 86
87 87
88 88
89 89
90 90
91 91
92 92
93 93
94 94
95 95
96 96
97 97
98 98
99 99
100 100
101 101
102 102
103 103

```

```

90 104   endif()
91 105
106 +if(ENABLE_CCACHE)
107 +   message(STATUS "Enabling ccache")
108 +   set(LSTORE_COMPILER_WRAPPER "ccache ")
109 +   set(LSTORE_COMPILER_WRAPPER_CMAKE "-DCMAKE_C_COMPILER=ccache"
110 +                                     "-DCMAKE_CXX_COMPILER=ccache")
111 +   if(CMAKE_C_COMPILER)
112 +     list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
113 +              "-DCMAKE_C_COMPILER_ARG1=${CMAKE_C_COMPILER}")
114 +   else()
115 +     list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
116 +              "-DCMAKE_C_COMPILER_ARG1=cc")
117 +   endif()
118 +   if(CMAKE_CXX_COMPILER)
119 +     list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
120 +              "-DCMAKE_CXX_COMPILER_ARG1=${CMAKE_CXX_COMPILER}")
121 +   else()
122 +     list(APPEND LSTORE_COMPILER_WRAPPER_CMAKE
123 +              "-DCMAKE_CXX_COMPILER_ARG1=c++")
124 +   endif()
125 +   set_property(GLOBAL PROPERTY RULE_LAUNCH_COMPILE ccache)
126 +   set_property(GLOBAL PROPERTY RULE_LAUNCH_LINK ccache)
127 +else()
128 +   set(LSTORE_COMPILER_WRAPPER "")
129 +   set(LSTORE_COMPILER_WRAPPER_CMAKE "-DCMAKE_C_COMPILER=${CMAKE_C_COMPILER}"
130 +                                     "-DCMAKE_CXX_COMPILER=${CMAKE_CXX_COMPILER}")
131 +endif()
132 +
133 +# Detect compiler flags
134 +include(CheckCCompilerFlag)
135 +check_c_compiler_flag(-Werror=incompatible-pointer-types HAS_INCOMPATIBLE_POINTER_WERROR)
136 +if(HAS_INCOMPATIBLE_POINTER_WERROR)
137 +   set(CMAKE_C_FLAGS "${CMAKE_C_FLAGS} -Werror=incompatible-pointer-types")
138 +endif()
139 +
92 140   # Use gnu-style paths
93 141   include(GNUInstallDirs)
94 142
⚡ @@ -155,6 +203,23 @@ if(BUILD_TESTS)
155 203     endif()
156 204   endif()
157 205
206 +# Build documentation
207 +find_program(DOXYGEN_PATH doxygen
208 +            DOC "Path to doxygen executable")
209 +add_custom_target(docs COMMENT "Build documentation")
210 +add_custom_target(doxygen COMMAND "${DOXYGEN_PATH}"
211 +                  WORKING_DIRECTORY "${CMAKE_SOURCE_DIR}/doc/"
212 +                  COMMENT "Document code with doxygen")
213 +add_custom_target(sphinx-html COMMAND $(MAKE) "html"
214 +                  WORKING_DIRECTORY "${CMAKE_SOURCE_DIR}/doc/"
215 +                  COMMENT "Produce sphinx documentation (HTML)")
216 +add_custom_target(sphinx-man COMMAND $(MAKE) "man"
217 +                  WORKING_DIRECTORY "${CMAKE_SOURCE_DIR}/doc/"
218 +                  COMMENT "Produce sphinx documentation (Man Pages)")
219 +add_dependencies(sphinx-html doxygen)
220 +add_dependencies(sphinx-man doxygen)
221 +add_dependencies(docs sphinx-html sphinx-man)
222 +
158 223   # Import packaging
159 224   include(LStorePackaging)
160 225
⚡

```

14 ■ ■ ■ Jenkinsfile View

```

⚡ @@ -90,7 +90,7 @@ compile_map['scan-build'] = {
90 90     def scan_checks = "-enable-checker alpha.core.BoolAssignment -enable-checker alpha.core.CallAndMessageUnInitRefArg -enable-
91 91     dir('build') {
92 92         sh "mkdir clang-static-analyzer"
93 93     -     sh "CCC_CC=clang scan-build cmake -DCMAKE_EXPORT_COMPILE_COMMANDS=ON -DBUILD_TESTS=on -DCMAKE_INSTALL_PREFIX=local/ .."
94 94
95 95     }

```

```

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

```

9 ■ ■ ■ ■ cmake/LStoreExternals.cmake View

```

✳ @@ -13,7 +13,7 @@ if(BUILD_APR OR (NOT APR-ACCRES_FOUND) OR (APR_LIBRARY MATCHES "^${EXTERNAL_INSTA
13 13         "--includedir=${EXTERNAL_INSTALL_DIR}/include/apr-ACCRES-1"
14 14         "--with-installbuilddir=${EXTERNAL_INSTALL_DIR}/lib/apr-ACCRES-1/buil
15 15         "--enable-static=yes" "--enable-shared=no" "--with-pic"
16 16         - "CC=${CMAKE_C_COMPILER}"
17 17         + "CC=${LSTORE_COMPILER_WRAPPER}${CMAKE_C_COMPILER}"
18 18         BUILD_COMMAND $(MAKE)
19 19         TEST_COMMAND $(MAKE) "test"
20 20         INSTALL_COMMAND $(MAKE) "install"
✳ @@ -43,7 +43,7 @@ if(BUILD_APU OR (NOT APRUTIL-ACCRES_FOUND) OR (APU_LIBRARY MATCHES "^${EXTERNAL_I
43 43         "--includedir=${EXTERNAL_INSTALL_DIR}/include/apr-util-ACCRES-1"
44 44         "--with-installbuilddir=${EXTERNAL_INSTALL_DIR}/lib/apr-util-ACCRES-1
45 45         "--enable-static=yes" "--enable-shared=no" "--with-pic"
46 46         - "CC=${CMAKE_C_COMPILER}"
47 47         + "CC=${LSTORE_COMPILER_WRAPPER}${CMAKE_C_COMPILER}"
48 48         BUILD_COMMAND $(MAKE)
49 49         TEST_COMMAND $(MAKE) "test"
50 50         INSTALL_COMMAND $(MAKE) "install"
✳ @@ -61,7 +61,7 @@ if(BUILD_JERASURE OR (NOT JERASURE_FOUND) OR (JERASURE_LIBRARY MATCHES "^${EXTER
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=${CMAKE_C_COMPILER}"
65 65         + "${LSTORE_COMPILER_WRAPPER_CMAKE}"
66 66         TEST_COMMAND ""
67 67         INSTALL_COMMAND $(MAKE) "install"
        )
✳ @@ -76,7 +76,8 @@ if((NOT CZMQ_FOUND) OR (CZMQ_LIBRARY MATCHES "^${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=${CMAKE_C_COMPILER}"
80 80         + "DCMAKE_C_FLAGS=-fPIC"
81 81         + "${LSTORE_COMPILER_WRAPPER_CMAKE}"
82 82         TEST_COMMAND $(MAKE) "test"
83 83         INSTALL_COMMAND $(MAKE) "install"
84 84         TEST_EXCLUDE_FROM_MAIN 1
✳

```

View

5		cmake/LStorePackaging.cmake
		@@ -48,7 +48,10 @@ set(CPACK_DEBIAN_PACKAGE_SHLIBDEPS ON)
48	48	# Targetski
49	49	add_custom_target(dist COMMAND \${CMAKE_MAKE_PROGRAM} package_source)
50	50	if(UNIX)
51		- ADD_CUSTOM_TARGET(rpm rpmbuild -ta \${CPACK_SOURCE_PACKAGE_FILE_NAME}.tar.gz)
	51	+ ADD_CUSTOM_TARGET(rpm rpmbuild --verbose
	52	+ -D "_rpmmdir" \${CMAKE_BINARY_DIR}/rpm_output"
	53	+ -D "_srcrpmmdir" \${CMAKE_BINARY_DIR}/srpm_output"
	54	+ -ta \${CPACK_SOURCE_PACKAGE_FILE_NAME}.tar.gz)
52	55	ADD_DEPENDENCIES(rpm dist)
53	56	ADD_CUSTOM_TARGET(deb dpkg-buildpackage -uc -us)
54	57	endif()

4		debian/rules	View
		@@ -20,6 +20,6 @@ export DH_BUILD_DDEBS=1	
20	20		
21	21	# dh_make generated override targets	
22	22	# This is example for Cmake (See https://bugs.debian.org/641051)	
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	# spaces. See also FILE_PATTERNS and EXTENSION_MAPPING	
772	772	# Note: If this tag is empty the current directory is searched.	
773	773		
774		-INPUT = ../src/toolbox/chksum.h	
	774	+INPUT = ../src/toolbox/tbx	
775	775		
776	776	# This tag can be used to specify the character encoding of the source files	
777	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	LStore - Logistical Storage.	
18	18		
19	19	%prep	
20		-%setup -n LStore-%{_version}	
	20	+%setup -q -n LStore-%{_version}	
21	21		
22	22	%build	
23	23	CFLAGS="-I%{_prefix}/include \$RPM_OPT_FLAGS"	
24	24	CMAKE_FLAGS="-DCMAKE_INSTALL_PREFIX=%{_prefix}"	
25	25	CMAKE_FLAGS="\$CMAKE_FLAGS -DINSTALL_YUM_RELEASE:BOOL=ON"	
26	26	CMAKE_FLAGS="\$CMAKE_FLAGS -DINSTALL_META:BOOL=ON"	
	27	+CMAKE_FLAGS="\$CMAKE_FLAGS -DCMAKE_BUILD_TYPE:BOOL=RelWithDebInfo"	
27	28	cmake \$CMAKE_FLAGS .	
28		-make	
	29	+make %{?_smp_mflags}	
29	30		
30	31	%install	
	32	+rm -rf \$RPM_BUILD_ROOT	
31	33	make DESTDIR=\$RPM_BUILD_ROOT install	
32	34		
33	35	%clean	
		@@ -39,8 +41,8 @@ rm -rf \$RPM_BUILD_ROOT	
39	41		
40	42	%files	
41	43	%defattr(-,root,root,-)	

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

2		■■■■ scripts/generate-docker-base.sh	View
✳		@@ -65,6 +65,7 @@ for DISTRO in "\${DISTROS[@]}"; do	
65	65	rpm)	
66	66	ADDITIONAL_PACKAGES=(
67	67	autoconf	
68	68	+ ccache	
68	69	curl	
69	70	createrepo	
70	71	czmq-devel	
✳		@@ -85,6 +86,7 @@ for DISTRO in "\${DISTROS[@]}"; do	
85	86	ADDITIONAL_PACKAGES=(
86	87	autoconf	
87	88	ca-certificates	
88	89	+ ccache	
88	90	cmake	
89	91	curl	
90	92	debhelper	
✳			

120		■■■■■ scripts/package-internal-separate.sh	View
...	...	@@ -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 +     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 $SOURCE_BASE &&
87 +         ( git update-index -q --refresh &>/dev/null || true ) && \
88 +         git describe --abbrev=32 --dirty="-dev" --candidates=100 \
89 +         --match ROOT --always)"
90 + fi
91 +
92 + TAG_NAME=${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 +
98 +     build_lstore_package $PACKAGE $SOURCE_BASE $TAG_NAME \
99 +         $PACKAGE_DISTRO
100 +     mkdir -p $PACKAGE_REPO
101 +     (
102 +         umask 000
103 +         cp *.{PACKAGE_SUFFIX} $PACKAGE_REPO
104 +         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 +
10 11 set -eu
11 12 ABSOLUTE_PATH=$(cd `dirname "${BASH_SOURCE[0]}"` && pwd)
12 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 + CMAKE_ARGS="-DINSTALL_DEB_RELEASE=ON"
25 27 ;;
26 28 centos-*)
27 29 PACKAGE_INSTALL="rpm -i"
28 30 PACKAGE_SUFFIX=rpm
31 + CMAKE_ARGS="-DINSTALL_YUM_RELEASE=ON"
29 32 ;;
30 33 *)
31 34 fatal "Unexpected distro name $DISTRO_NAME"
32 35 ;;
33 36 esac
35 38 # todo could probe this from docker variables
36 -PACKAGE_BASE="/tmp/lstore-release"
37 39 REPO_BASE=$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
40 +PACKAGE_BASE=/tmp/lstore-package
41 +
50 42 note "Beginning packaging at $(date) for $PACKAGE_SUBDIR"
51 43
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

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	83	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		
	35	## Set up ccache to speed multiple compilations	
	36	+if [[-z "\${CCACHE_DIR:-}"]]; then	
	37	+ CCACHE_DEFAULT=1	
	38	+ CCACHE_DIR=/tmp/source/build/ccache	
	39	+ mkdir -p \$CCACHE_DIR	
	40	+fi	
	41	+	
35	42	if [[! -z "\${HOST_VOLUME_PATH:-}" && ! -z "\${CONTAINER_VOLUME_PATH:-}"]]; then	
36	43	LSTORE_RELEASE_RELATIVE="\${HOST_VOLUME_PATH}/\${realpath \$(pwd) --relative-to "\$CONTAINER_VOLUME_PATH")"	
	44	+ CCACHE_DIR_RELATIVE="\${HOST_VOLUME_PATH}/\${realpath "\$CCACHE_DIR" --relative-to "\$CONTAINER_VOLUME_PATH")"	
37	45	else	
38	46	LSTORE_RELEASE_RELATIVE="\$LSTORE_RELEASE_BASE"	
	47	+ CCACHE_DIR_RELATIVE="\$CCACHE_DIR"	
	48	+fi	
	49	+	
	50	+note "ccache default: \${CCACHE_DEFAULT:-} ccache_dir \$CCACHE_DIR"	
	51	+if [[-z "\${CCACHE_DEFAULT:-}" && -d "\$CCACHE_DIR"]]; then	
	52	+ EXTRA_ARGS="\$EXTRA_ARGS -e CCACHE_DIR=/tmp/ccache"	
	53	+ EXTRA_ARGS="\$EXTRA_ARGS -v \$CCACHE_DIR_RELATIVE:/tmp/ccache"	
	54	+else	
	55	+ EXTRA_ARGS="\$EXTRA_ARGS -e CCACHE_DIR=\$CCACHE_DIR"	
39	56	fi	
40	57		
41	58	for DISTRO in "\${DISTROS[@]"; do	
42	59	note "Starting docker container to package \$DISTRO"	
	60	+ case \$DISTRO in	
	61	+ centos*)	
	62	+ INTERNAL_CMD="/tmp/source/scripts/package-internal.sh \$DISTRO"	
	63	+ ;;	
	64	+ debian* ubuntu*)	
	65	+ INTERNAL_CMD="/tmp/source/scripts/package-internal.sh \$DISTRO"	
	66	+ ;;	
	67	+ esac	
43	68	set -x	
44	69	docker run --rm=true -v \$LSTORE_RELEASE_RELATIVE:/tmp/source \	
45	70	\$EXTRA_ARGS \	
46	71	lstore/builder:\${DISTRO} \	
47	-	/tmp/source/scripts/package-internal.sh \$DISTRO	
	72	+ \$INTERNAL_CMD	
48	73	set +x	
49	74	done	

```

2  scripts/test-repo-internal.sh View
@@ -23,7 +23,7 @@ gpgcheck=0
23 23 protect=1
24 24 EOF
25 25     yum install -y epel-release
26 26 -     yum install -y accre-lio which
27 27 +     yum install -y lstore which
28 28     yum clean all
29 29     ;;
    ubuntu-*|debian-*)

```

```

12  src/ibp/ibp_config.c View
@@ -147,7 +147,7 @@ int ibp_rw_submit_coalesce(tbx_stack_t *stack, tbx_stack_ele_t *ele)
147 147     op_generic_t *gop = (op_generic_t *)tbx_get_stack_ele_data(ele);
148 148     ibp_op_t *iop = ibp_get_iop(gop);
149 149     ibp_context_t *ic = iop->ic;
150 150 -     ibp_op_rw_t *cmd = &(iop->rw_op);
151 151 +     ibp_op_rw_t *cmd = &(iop->ops.rw_op);
152 152     rw_coalesce_t *rwc;
153 153     tbx_pch_t pch;
@@ -195,7 +195,7 @@ int ibp_rw_coalesce(op_generic_t *gop1)
195 195     ibp_op_t *iop1 = ibp_get_iop(gop1);
196 196     ibp_op_t *iop2;
197 197     ibp_context_t *ic = iop1->ic;
198 198 -     ibp_op_rw_t *cmd1 = &(iop1->rw_op);
199 199 +     ibp_op_rw_t *cmd1 = &(iop1->ops.rw_op);
200 200     ibp_op_rw_t *cmd2;
201 201     op_generic_t *gop2;
202 202     ibp_rw_buf_t **rwbuff;
@@ -266,7 +266,7 @@ int ibp_rw_coalesce(op_generic_t *gop1)
266 266     do {
267 267         gop2 = (op_generic_t *)tbx_get_stack_ele_data(ele);
268 268         iop2 = ibp_get_iop(gop2);
269 269 -         cmd2 = &(iop2->rw_op);
270 270 +         cmd2 = &(iop2->ops.rw_op);
271 271
272 272         /** Unlink the element if needed
273 273         if (gop2 != gop1) {
@@ -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);
311 311 -             cmd2 = &(iop2->rw_op);
312 312 +             cmd2 = &(iop2->ops.rw_op);
313 313             rwbuff[n] = &(cmd2->buf_single);
314 314             iov_sum += cmd2->n_tbx_iovec_total;
315 315             workload += cmd2->size;
@@ -374,8 +374,8 @@ void ibp_op_free(op_generic_t *gop, int mode)
374 374     if (gop->op->cmd.coalesced_ops != NULL) { /** Coalesced R/W op so free the stack
375 375         log_printf(15, "gid=%d Freeing rwbuff\n", gop_id(gop));
376 376         iop = ibp_get_iop(gop);
377 377 -         tbx_pch_release(iop->ic->coalesced_gop_stacks, &(iop->rw_op.rwcg_pch));
378 378 -         free(iop->rw_op.rwbuff);
379 379 +         tbx_pch_release(iop->ic->coalesced_gop_stacks, &(iop->ops.rw_op.rwcg_pch));
380 380 +         free(iop->ops.rw_op.rwbuff);
381 381         gop->op->cmd.coalesced_ops = NULL;
382 382     }
383 383     gop_generic_free(gop, OP_FINALIZE); /** I free the actual op

```

```

134  src/ibp/ibp_op.c View
@@ -431,10 +431,10 @@ void set_ibp_vec_read_op(ibp_op_t *op, ibp_cap_t *cap, int n_vec, ibp_tbx_iovec_
431 431     op_generic_t *gop = ibp_get_gop(op);
432 432

```

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;
	434	+ op->ops_rw_op.n_ops = 1;
	435	+ op->ops_rw_op.n_tbx_iovec_total = n_vec;
	436	+ op->ops_rw_op.buf_single.n_iovec = n_vec;
	437	+ op->ops_rw_op.buf_single.iovec = vec;
438	438	
439	439	gop->op->cmd.send_command = vec_read_command;
440	440	}
✳		@@ -466,7 +466,7 @@ op_status_t vec_read_command(op_generic_t *gop, tbx_ns_t *ns)
466	466	ibp_rw_buf_t *rwbuf;
467	467	op_status_t err;
468	468	
469		- cmd = &(op->rw_op);
	469	+ cmd = &(op->ops_rw_op);
470	470	
471	471	used = 0;
472	472	
✳		@@ -522,7 +522,7 @@ op_status_t read_command(op_generic_t *gop, tbx_ns_t *ns)
522	522	op_status_t err;
523	523	ibp_op_rw_t *cmd;
524	524	
525		- cmd = &(op->rw_op);
	525	+ cmd = &(op->ops_rw_op);
526	526	
527	527	if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {
528	528	// tbx_ns_chksum_reset(&(op->ncs));
✳		@@ -599,7 +599,7 @@ op_status_t read_recv(op_generic_t *gop, tbx_ns_t *ns)
599	599	double swait;
600	600	ibp_rw_buf_t *rwbuf;
601	601	
602		- cmd = &(op->rw_op);
	602	+ cmd = &(op->ops_rw_op);
603	603	
604	604	/** Need to read the depot status info
605	605	log_printf(15, "read_recv: ns=%d starting command size=" OT "\n", tbx_ns_getid(ns), cmd->size);
✳		@@ -687,7 +687,7 @@ op_generic_t *new_ibp_write_op(ibp_context_t *ic, ibp_cap_t *cap, ibp_off_t offs)
687	687	op_generic_t *gop = new_ibp_rw_op(ic, IBP_WRITE, cap, offset, buffer, bpos, len, timeout);
688	688	// ibp_op_t *iop = ibp_get_iop(gop);
689	689	
690		-//log_printf(15, "new_ibp_write_op: gid=%d next_block=%p\n", gop_id(gop), iop->rw_op.next_block); tbx_flush_log();
	690	+//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();
691	691	return(gop);
692	692	}
693	693	
✳		@@ -700,10 +700,10 @@ void set_ibp_vec_write_op(ibp_op_t *op, ibp_cap_t *cap, int n_iovec, ibp_tbx_iov)
700	700	op_generic_t *gop = ibp_get_gop(op);
701	701	
702	702	set_ibp_rw_op(op, IBP_WRITE, cap, 0, buffer, bpos, len, timeout);
703		- op->rw_op.n_ops = 1;
704		- op->rw_op.n_tbx_iovec_total = n_iovec;
705		- op->rw_op.buf_single.n_iovec = n_iovec;
706		- op->rw_op.buf_single.iovec = iovec;
	703	+ op->ops_rw_op.n_ops = 1;
	704	+ op->ops_rw_op.n_tbx_iovec_total = n_iovec;
	705	+ op->ops_rw_op.buf_single.n_iovec = n_iovec;
	706	+ op->ops_rw_op.buf_single.iovec = iovec;
707	707	
708	708	gop->op->cmd.send_command = vec_write_command;
709	709	
✳		@@ -736,7 +736,7 @@ op_status_t vec_write_command(op_generic_t *gop, tbx_ns_t *ns)
736	736	ibp_op_rw_t *cmd;
737	737	ibp_rw_buf_t *rwbuf;
738	738	
739		- cmd = &(op->rw_op);
	739	+ cmd = &(op->ops_rw_op);
740	740	
741	741	used = 0;
742	742	

✳	@@ -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
802	802 // log_printf(10, "write_command: gid=%d next_block=%p\n", gop_id(gop), cmd->next_block);
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 *rwbuf;
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 log_printf(15, "write_recv: Start!!! ns=%d\n", tbx_ns_getid(ns));
929	929
930	930
931	- cmd = &(op->rw_op);
931	+ cmd = &(op->ops.rw_op);
932	932
933	933 tbx_ns_chksum_read_disable(ns);
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	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 *rwbuf;
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	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_VALIDATE_CHKSUM]));</code>
1194	1194	<code>op->dop.cmd.hostport = strdup(hoststr);</code>
1195	1195	
1196		- <code>cmd = &(op->validate_op);</code>
	1196	+ <code>cmd = &(op->ops.validate_op);</code>
1197	1197	<code>cmd->correct_errors = correct_errors;</code>
1198	1198	<code>cmd->n_bad_blocks = n_bad_blocks;</code>
1199	1199	
✳		@@ -1227,7 +1227,7 @@ op_status_t get_chksum_command(op_generic_t *gop, tbx_ns_t *ns)
1227	1227	<code>ibp_op_t *op = ibp_get_iop(gop);</code>
1228	1228	<code>char buffer[1024];</code>
1229	1229	<code>op_status_t err;</code>
1230		- <code>ibp_op_get_chksum_t *cmd = &(op->get_chksum_op);</code>
	1230	+ <code>ibp_op_get_chksum_t *cmd = &(op->ops.get_chksum_op);</code>
1231	1231	
1232	1232	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d\n",</code>
1233	1233	<code>IBPv040, IBP_GET_CHKSUM, cmd->key, cmd->typekey, cmd->chksum_info_only,</code>
✳		@@ -1251,7 +1251,7 @@ op_status_t get_chksum_recv(op_generic_t *gop, tbx_ns_t *ns)
1251	1251	<code>char buffer[1024];</code>
1252	1252	<code>op_status_t err;</code>
1253	1253	<code>int status, fin;</code>
1254		- <code>ibp_op_get_chksum_t *cmd = &(op->get_chksum_op);</code>
	1254	+ <code>ibp_op_get_chksum_t *cmd = &(op->ops.get_chksum_op);</code>
1255	1255	<code>char *bstate;</code>
1256	1256	<code>tbx_tbuf_t buf;</code>
1257	1257	
✳		@@ -1324,14 +1324,14 @@ void set_ibp_get_chksum_op(ibp_op_t *op, ibp_cap_t *mcap, int chksum_info_only,
1324	1324	<code>init_ibp_base_op(op, "get_chksum", timeout, op->ic->other_new_command, NULL, 1, IBP_VALIDATE_CHKSUM, IBP_NOP);</code>
1325	1325	<code>op_generic_t *gop = ibp_get_gop(op);</code>
1326	1326	
1327		- <code>cmd = &(op->get_chksum_op);</code>
	1327	+ <code>cmd = &(op->ops.get_chksum_op);</code>
1328	1328	
1329	1329	<code>parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);</code>
1330	1330	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_VALIDATE_CHKSUM]));</code>
1331	1331	<code>op->dop.cmd.hostport = strdup(hoststr);</code>
1332	1332	
1333	1333	<code>cmd->cap = mcap;</code>
1334		- <code>cmd = &(op->get_chksum_op);</code>
	1334	+ <code>cmd = &(op->ops.get_chksum_op);</code>
1335	1335	<code>cmd->chksum_info_only = chksum_info_only;</code>
1336	1336	<code>cmd->cs_type = cs_type;</code>
1337	1337	<code>cmd->cs_size = cs_size;</code>
✳		@@ -1371,7 +1371,7 @@ op_status_t allocate_command(op_generic_t *gop, tbx_ns_t *ns)
1371	1371	<code>char buffer[1024];</code>
1372	1372	<code>// apr_time_t atime, now;</code>
1373	1373	<code>op_status_t err;</code>
1374		- <code>ibp_op_alloc_t *cmd = &(op->alloc_op);</code>
	1374	+ <code>ibp_op_alloc_t *cmd = &(op->ops.alloc_op);</code>
1375	1375	
1376	1376	<code>log_printf(10, "allocate_command: cs_type=%d\n", cmd->disk_chksum_type);</code>
1377	1377	
✳		@@ -1409,7 +1409,7 @@ op_status_t split_allocate_command(op_generic_t *gop, tbx_ns_t *ns)
1409	1409	<code>ibp_op_t *op = ibp_get_iop(gop);</code>
1410	1410	<code>char buffer[1024];</code>
1411	1411	<code>op_status_t err;</code>
1412		- <code>ibp_op_alloc_t *cmd = &(op->alloc_op);</code>
	1412	+ <code>ibp_op_alloc_t *cmd = &(op->ops.alloc_op);</code>
1413	1413	
1414	1414	<code>if (cmd->disk_chksum_type == CHKSUM_DEFAULT) { /** Normal split allocation</code>
1415	1415	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d %d " I64T " %d\n",</code>
✳		@@ -1445,7 +1445,7 @@ op_status_t allocate_recv(op_generic_t *gop, tbx_ns_t *ns)
1445	1445	<code>char rcap[1025], wcap[1025], mcap[1025];</code>
1446	1446	<code>char *bstate;</code>
1447	1447	<code>op_status_t err;</code>
1448		- <code>ibp_op_alloc_t *cmd = &(op->alloc_op);</code>
	1448	+ <code>ibp_op_alloc_t *cmd = &(op->ops.alloc_op);</code>
1449	1449	
1450	1450	<code>/** Need to read the depot status info</code>
1451	1451	<code>log_printf(15, "allocate_recv: ns=%d Start\n", tbx_ns_getid(ns));</code>
✳		@@ -1516,7 +1516,7 @@ void set_ibp_alloc_op(ibp_op_t *op, ibp_capset_t *caps, ibp_off_t size, ibp_depo

1516	1516	<code>init_ibp_base_op(op, "alloc", timeout, op->ic->other_new_command, strdup(hoststr), 1, IBP_ALLOCATE, IBP_NOP);</code>
1517	1517	<code>//log_printf(15, "set_ibp_alloc_op: after init_ibp_base_op\n");</code>
1518	1518	
1519		- <code>cmd = &(op->alloc_op);</code>
1519	1519	+ <code>cmd = &(op->ops.alloc_op);</code>
1520	1520	<code>cmd->caps = caps;</code>
1521	1521	<code>cmd->depot = depot;</code>
1522	1522	<code>cmd->attr = attr;</code>
☼		@@ -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	<code>init_ibp_base_op(op, "split_allocate", timeout, op->ic->other_new_command, NULL, 1, IBP_SPLIT_ALLOCATE, IBP_NOP);</code>
1563	1563	
1564	1564	
1565		- <code>cmd = &(op->alloc_op);</code>
1565	1565	+ <code>cmd = &(op->ops.alloc_op);</code>
1566	1566	
1567	1567	<code>parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);</code>
1568	1568	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_SPLIT_ALLOCATE]));</code>
1569	1569	<code>op->dop.cmd.hostport = strdup(hoststr);</code>
1570	1570	
1571		- <code>cmd = &(op->alloc_op);</code>
1571	1571	+ <code>cmd = &(op->ops.alloc_op);</code>
1572	1572	<code>cmd->caps = caps;</code>
1573	1573	<code>cmd->attr = attr;</code>
1574	1574	
☼		@@ -1610,7 +1610,7 @@ op_status_t rename_command(op_generic_t *gop, tbx_ns_t *ns)
1610	1610	<code>ibp_op_t *op = ibp_get_iop(gop);</code>
1611	1611	<code>char buffer[1024];</code>
1612	1612	<code>op_status_t err;</code>
1613		- <code>ibp_op_alloc_t *cmd = &(op->alloc_op);</code>
1613	1613	+ <code>ibp_op_alloc_t *cmd = &(op->ops.alloc_op);</code>
1614	1614	
1615	1615	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %d\n",</code>
1616	1616	<code>IBPv040, IBP_RENAME, cmd->key, cmd->typekey, (int)apr_time_sec(gop->op->cmd.timeout));</code>
☼		@@ -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	<code>init_ibp_base_op(op, "rename", timeout, op->ic->other_new_command, NULL, 1, IBP_RENAME, IBP_NOP);</code>
1641	1641	
1642	1642	
1643		- <code>cmd = &(op->alloc_op);</code>
1643	1643	+ <code>cmd = &(op->ops.alloc_op);</code>
1644	1644	
1645	1645	<code>parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);</code>
1646	1646	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_RENAME]));</code>
☼		@@ -1678,7 +1678,7 @@ op_status_t merge_command(op_generic_t *gop, tbx_ns_t *ns)
1678	1678	<code>ibp_op_t *op = ibp_get_iop(gop);</code>
1679	1679	<code>char buffer[1024];</code>
1680	1680	<code>op_status_t err;</code>
1681		- <code>ibp_op_merge_alloc_t *cmd = &(op->merge_op);</code>
1681	1681	+ <code>ibp_op_merge_alloc_t *cmd = &(op->ops.merge_op);</code>
1682	1682	
1683	1683	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %s %s %d\n",</code>
1684	1684	<code>IBPv040, IBP_MERGE_ALLOCATE, cmd->mkey, cmd->mtypekey, cmd->ckey, cmd->ctypekey, (int)apr_time_sec(gop->op->cmd.timeou</code>
☼		@@ -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	<code>init_ibp_base_op(op, "rename", timeout, op->ic->other_new_command, NULL, 1, IBP_RENAME, IBP_NOP);</code>
1710	1710	
1711	1711	
1712		- <code>cmd = &(op->merge_op);</code>
1712	1712	+ <code>cmd = &(op->ops.merge_op);</code>
1713	1713	
1714	1714	<code>parse_cap(op->ic, mcap, host, &port, cmd->mkey, cmd->mtypekey);</code>
1715	1715	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_MERGE_ALLOCATE]));</code>
☼		@@ -1747,7 +1747,7 @@ op_status_t alias_allocate_command(op_generic_t *gop, tbx_ns_t *ns)
1747	1747	<code>ibp_op_t *op = ibp_get_iop(gop);</code>
1748	1748	<code>char buffer[1024];</code>
1749	1749	<code>op_status_t err;</code>
1750		- <code>ibp_op_alloc_t *cmd = &(op->alloc_op);</code>
1750	1750	+ <code>ibp_op_alloc_t *cmd = &(op->ops.alloc_op);</code>
1751	1751	
1752	1752	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s " I64T " " I64T " %d %d\n",</code>
1753	1753	<code>IBPv040, IBP_ALIAS_ALLOCATE, cmd->key, cmd->typekey, cmd->offset, cmd->size, cmd->duration, (int)apr_time_sec(gop->op-</code>
☼		@@ -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	<code>init_ibp_base_op(op, "rename", timeout, op->ic->other_new_command, NULL, 1, IBP_ALIAS_ALLOCATE, IBP_NOP);</code>
1780	1780	
1781		- <code>cmd = &(op->alloc_op);</code>
	1781	+ <code>cmd = &(op->ops.alloc_op);</code>
1782	1782	
1783	1783	<code>parse_cap(op->ic, mcap, host, &port, cmd->key, cmd->typekey);</code>
1784	1784	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_ALIAS_ALLOCATE]));</code>
☼		@@ -1829,7 +1829,7 @@ op_status_t modify_count_command(op_generic_t *gop, tbx_ns_t *ns)
1829	1829	<code>op_status_t err;</code>
1830	1830	<code>ibp_op_probe_t *cmd;</code>
1831	1831	
1832		- <code>cmd = &(op->probe_op);</code>
	1832	+ <code>cmd = &(op->ops.probe_op);</code>
1833	1833	
1834	1834	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d %d\n",</code>
1835	1835	<code>IBPV040, cmd->cmd, cmd->key, cmd->typekey, cmd->mode, cmd->captype, (int)apr_time_sec(gop->op->cmd.timeout));</code>
☼		@@ -1851,7 +1851,7 @@ op_status_t alias_modify_count_command(op_generic_t *gop, tbx_ns_t *ns)
1851	1851	<code>op_status_t err;</code>
1852	1852	<code>ibp_op_probe_t *cmd;</code>
1853	1853	
1854		- <code>cmd = &(op->probe_op);</code>
	1854	+ <code>cmd = &(op->ops.probe_op);</code>
1855	1855	
1856	1856	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d %s %s %d\n",</code>
1857	1857	<code>IBPV040, cmd->cmd, cmd->key, cmd->typekey, cmd->mode, cmd->captype, cmd->mkey, cmd->mtypekey, (int)apr_time_sec(gop->op->cmd.timeout));</code>
☼		@@ -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	<code>init_ibp_base_op(op, "modify_count", timeout, op->ic->other_new_command, NULL, 1, command, mode);</code>
1919	1919	
1920		- <code>cmd = &(op->probe_op);</code>
	1920	+ <code>cmd = &(op->ops.probe_op);</code>
1921	1921	
1922	1922	<code>parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);</code>
1923	1923	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[command]));</code>
☼		@@ -1993,7 +1993,7 @@ op_status_t modify_alloc_command(op_generic_t *gop, tbx_ns_t *ns)
1993	1993	<code>int atime;</code>
1994	1994	<code>ibp_op_modify_alloc_t *cmd;</code>
1995	1995	
1996		- <code>cmd = &(op->mod_alloc_op);</code>
	1996	+ <code>cmd = &(op->ops.mod_alloc_op);</code>
1997	1997	
1998	1998	<code>atime = cmd->duration - time(NULL); /** This is in sec NOT APR time</code>
1999	1999	<code>if (atime < 0) atime = cmd->duration;</code>
☼		@@ -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	<code>init_ibp_base_op(op, "modify_alloc", timeout, op->ic->other_new_command, NULL, 1, IBP_MANAGE, IBP_CHNG);</code>
2031	2031	
2032		- <code>cmd = &(op->mod_alloc_op);</code>
	2032	+ <code>cmd = &(op->ops.mod_alloc_op);</code>
2033	2033	
2034	2034	<code>parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);</code>
2035	2035	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_MANAGE]));</code>
☼		@@ -2072,7 +2072,7 @@ op_status_t alias_modify_alloc_command(op_generic_t *gop, tbx_ns_t *ns)
2072	2072	<code>int atime;</code>
2073	2073	<code>ibp_op_modify_alloc_t *cmd;</code>
2074	2074	
2075		- <code>cmd = &(op->mod_alloc_op);</code>
	2075	+ <code>cmd = &(op->ops.mod_alloc_op);</code>
2076	2076	
2077	2077	<code>atime = cmd->duration - time(NULL); /** This is in sec NOT APR time</code>
2078	2078	<code>if (atime < 0) atime = cmd->duration;</code>
☼		@@ -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	<code>init_ibp_base_op(op, "alias_modify_alloc", timeout, op->ic->other_new_command, NULL, 1, IBP_ALIAS_MANAGE, IBP_CHNG);</code>
2110	2110	
2111		- <code>cmd = &(op->mod_alloc_op);</code>
	2111	+ <code>cmd = &(op->ops.mod_alloc_op);</code>
2112	2112	
2113	2113	<code>parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);</code>
2114	2114	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_ALIAS_MANAGE]));</code>

		@@ -2152,7 +2152,7 @@ op_status_t truncate_command(op_generic_t *gop, tbx_ns_t *ns)
2152	2152	op_status_t err;
2153	2153	ibp_op_modify_alloc_t *cmd;
2154	2154	
2155		- cmd = &(op->mod_alloc_op);
	2155	+ cmd = &(op->ops.mod_alloc_op);
2156	2156	
2157	2157	snprintf(buffer, sizeof(buffer), "%d %d %s %s %d " I64T " %d\n",
2158	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	
2184	2184	init_ibp_base_op(op, "truncate_alloc", timeout, op->ic->other_new_command, NULL, 1, IBP_MANAGE, IBP_CHNG);
2185	2185	
2186		- cmd = &(op->mod_alloc_op);
	2186	+ cmd = &(op->ops.mod_alloc_op);
2187	2187	
2188	2188	parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
2189	2189	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;
2265	2265	
2266		- cmd = &(op->probe_op);
	2266	+ cmd = &(op->ops.probe_op);
2267	2267	
2268	2268	snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d 0 0 %d \n",
2269	2269	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	
2302	2302	status = atoi(tbx_stk_string_token(buffer, " ", &bstate, &fin));
2303	2303	if ((status == IBP_OK) && (fin == 0)) {
2304		- p = op->probe_op.probe;
	2304	+ p = op->ops.probe_op.probe;
2305	2305	log_printf(15, "probe_recv: p=%p QWERT\n", p);
2306	2306	
2307	2307	p->readRefCount = atoi(tbx_stk_string_token(NULL, " ", &bstate, &fin));
		@@ -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);
	2340	+ cmd = &(op->ops.probe_op);
2341	2341	
2342	2342	parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);
2343	2343	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;
2382	2382	
2383		- cmd = &(op->probe_op);
	2383	+ cmd = &(op->ops.probe_op);
2384	2384	
2385	2385	snprintf(buffer, sizeof(buffer), "%d %d %s %s %d %d %d \n",
2386	2386	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	
2419	2419	status = atoi(tbx_stk_string_token(buffer, " ", &bstate, &fin));
2420	2420	if ((status == IBP_OK) && (fin == 0)) {
2421		- p = op->probe_op.alias_probe;
	2421	+ p = op->ops.probe_op.alias_probe;
2422	2422	p->read_refcount = atoi(tbx_stk_string_token(NULL, " ", &bstate, &fin));
2423	2423	p->write_refcount = 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	
2446	2446	init_ibp_base_op(op, "alias_probe", timeout, op->ic->other_new_command, NULL, 1, IBP_ALIAS_MANAGE, IBP_PROBE);
2447	2447	
2448		- cmd = &(op->probe_op);
	2448	+ cmd = &(op->ops.probe_op);
2449	2449	
2450	2450	parse_cap(op->ic, cap, host, &port, cmd->key, cmd->typekey);

2451	2451	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_ALIAS_MANAGE]));</code>
✳		<code>@@ -2491,7 +2491,7 @@ op_status_t copyappend_command(op_generic_t *gop, tbx_ns_t *ns)</code>
2491	2491	<code>op_status_t err;</code>
2492	2492	<code>ibp_op_copy_t *cmd;</code>
2493	2493	
2494		<code>- cmd = &(op->copy_op);</code>
	2494	<code>+ cmd = &(op->ops.copy_op);</code>
2495	2495	
2496	2496	<code>if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {</code>
2497	2497	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %s %s %s " I64T " " I64T " %d %d %d\n",</code>
✳		<code>@@ -2527,7 +2527,7 @@ op_status_t copy_recv(op_generic_t *gop, tbx_ns_t *ns)</code>
2527	2527	<code>ibp_op_copy_t *cmd;</code>
2528	2528	<code>double swait;</code>
2529	2529	
2530		<code>- cmd = &(op->copy_op);</code>
	2530	<code>+ cmd = &(op->ops.copy_op);</code>
2531	2531	
2532	2532	<code>/** Need to read the depot status info</code>
2533	2533	<code>log_printf(15, "copy_recv: ns=%d Start\n", tbx_ns_getid(ns));</code>
✳		<code>@@ -2566,7 +2566,7 @@ void set_ibp_copyappend_op(ibp_op_t *op, int ns_type, char *path, ibp_cap_t *src</code>
2566	2566	
2567	2567	<code>init_ibp_base_op(op, "copyappend", src_timeout, op->ic->rw_new_command + size, NULL, size, IBP_SEND, IBP_NOP);</code>
2568	2568	
2569		<code>- cmd = &(op->copy_op);</code>
	2569	<code>+ cmd = &(op->ops.copy_op);</code>
2570	2570	
2571	2571	<code>parse_cap(op->ic, srccap, host, &port, cmd->src_key, cmd->src_typekey);</code>
2572	2572	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_SEND]));</code>
✳		<code>@@ -2631,7 +2631,7 @@ op_status_t pushpull_command(op_generic_t *gop, tbx_ns_t *ns)</code>
2631	2631	<code>op_status_t err;</code>
2632	2632	<code>ibp_op_copy_t *cmd;</code>
2633	2633	
2634		<code>- cmd = &(op->copy_op);</code>
	2634	<code>+ cmd = &(op->ops.copy_op);</code>
2635	2635	
2636	2636	<code>if (tbx_ns_chksum_is_valid(&(op->ncs)) == 0) {</code>
2637	2637	<code>snprintf(buffer, sizeof(buffer), "%d %d %d %s %s %s %s " I64T " " I64T " " I64T " %d %d %d\n",</code>
✳		<code>@@ -2673,7 +2673,7 @@ void set_ibp_copy_op(ibp_op_t *op, int mode, int ns_type, char *path, ibp_cap_t</code>
2673	2673	
2674	2674	<code>init_ibp_base_op(op, "copy", src_timeout, op->ic->rw_new_command + size, NULL, size, IBP_SEND, IBP_NOP);</code>
2675	2675	
2676		<code>- cmd = &(op->copy_op);</code>
	2676	<code>+ cmd = &(op->ops.copy_op);</code>
2677	2677	
2678	2678	<code>parse_cap(op->ic, srccap, host, &port, cmd->src_key, cmd->src_typekey);</code>
2679	2679	<code>set_hostport(hoststr, sizeof(hoststr), host, port, &(op->ic->cc[IBP_SEND]));</code>
✳		<code>@@ -2738,7 +2738,7 @@ op_status_t depot_modify_command(op_generic_t *gop, tbx_ns_t *ns)</code>
2738	2738	<code>op_status_t err;</code>
2739	2739	<code>ibp_op_depot_modify_t *cmd;</code>
2740	2740	
2741		<code>- cmd = &(op->depot_modify_op);</code>
	2741	<code>+ cmd = &(op->ops.depot_modify_op);</code>
2742	2742	
2743	2743	<code>snprintf(buffer, sizeof(buffer), "%d %d %s %d %s %d\n " I64T " " I64T " " TT "\n",</code>
2744	2744	<code>IBPV040, IBP_STATUS, cmd->depot->rid.name, IBP_ST_CHANGE, cmd->password, (int)apr_time_sec(gop->op->cmd.timeout),</code>
✳		<code>@@ -2761,7 +2761,7 @@ op_status_t depot_modify_command(op_generic_t *gop, tbx_ns_t *ns)</code>
2761	2761	<code>void set_ibp_depot_modify_op(ibp_op_t *op, ibp_depot_t *depot, char *password, ibp_off_t hard, ibp_off_t soft,</code>
2762	2762	<code>int duration, int timeout)</code>
2763	2763	<code>{</code>
2764		<code>- ibp_op_depot_modify_t *cmd = &(op->depot_modify_op);</code>
	2764	<code>+ ibp_op_depot_modify_t *cmd = &(op->ops.depot_modify_op);</code>
2765	2765	
2766	2766	<code>init_ibp_base_op(op, "depot_modify", timeout, op->ic->other_new_command, NULL,</code>
2767	2767	<code>op->ic->other_new_command, IBP_STATUS, IBP_ST_CHANGE);</code>
✳		<code>@@ -2802,7 +2802,7 @@ op_status_t depot_inq_command(op_generic_t *gop, tbx_ns_t *ns)</code>
2802	2802	<code>op_status_t err;</code>
2803	2803	<code>ibp_op_depot_inq_t *cmd;</code>
2804	2804	
2805		<code>- cmd = &(op->depot_inq_op);</code>
	2805	<code>+ cmd = &(op->ops.depot_inq_op);</code>

2806	2806	
2807	2807	<code>tbx_ns_chksum_write_clear(ns);</code>
2808	2808	
✳ @@ -2881,7 +2881,7 @@ op_status_t depot_inq_rcv(op_generic_t *gop, tbx_ns_t *ns)		
2881	2881	<code>char *bstate;</code>
2882	2882	<code>ibp_op_depot_inq_t *cmd;</code>
2883	2883	
2884		- <code>cmd = &(op->depot_inq_op);</code>
	2884	+ <code>cmd = &(op->ops.depot_inq_op);</code>
2885	2885	
2886	2886	<code>tbx_ns_chksum_read_clear(ns);</code>
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	<code>char hoststr[MAX_HOST_SIZE];</code>
2930	2930	<code>char pchost[MAX_HOST_SIZE];</code>
2931		- <code>ibp_op_depot_inq_t *cmd = &(op->depot_inq_op);</code>
	2931	+ <code>ibp_op_depot_inq_t *cmd = &(op->ops.depot_inq_op);</code>
2932	2932	
2933	2933	<code>ibppc_form_host(op->ic, pchost, sizeof(pchost), depot->host, depot->rid);</code>
2934	2934	<code>set_hostport(hoststr, sizeof(hoststr), pchost, depot->port, &(op->ic->cc[IBP_STATUS]));</code>
✳ @@ -2989,7 +2989,7 @@ op_status_t depot_version_rcv(op_generic_t *gop, tbx_ns_t *ns)		
2989	2989	<code>int pos, nmax, status, fin;</code>
2990	2990	<code>ibp_op_version_t *cmd;</code>
2991	2991	
2992		- <code>cmd = &(op->ver_op);</code>
	2992	+ <code>cmd = &(op->ops.ver_op);</code>
2993	2993	
2994	2994	<code>err = ibp_success_status;</code>
2995	2995	<code>pos = 0;</code>
✳ @@ -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	<code>char hoststr[MAX_HOST_SIZE];</code>
3037	3037	<code>char pchoststr[MAX_HOST_SIZE];</code>
3038		- <code>ibp_op_version_t *cmd = &(op->ver_op);</code>
	3038	+ <code>ibp_op_version_t *cmd = &(op->ops.ver_op);</code>
3039	3039	
3040	3040	<code>ibppc_form_host(op->ic, pchoststr, sizeof(pchoststr), depot->host, depot->rid);</code>
3041	3041	<code>set_hostport(hoststr, sizeof(hoststr), pchoststr, depot->port, &(op->ic->cc[IBP_STATUS]));</code>
✳ @@ -3097,7 +3097,7 @@ op_status_t query_res_rcv(op_generic_t *gop, tbx_ns_t *ns)		
3097	3097	<code>op_status_t err;</code>
3098	3098	<code>int fin, n, i, status;</code>
3099	3099	<code>char *p, *bstate;</code>
3100		- <code>ibp_op_rid_inq_t *cmd = &(op->rid_op);</code>
	3100	+ <code>ibp_op_rid_inq_t *cmd = &(op->ops.rid_op);</code>
3101	3101	
3102	3102	<code>err = ibp_success_status;</code>
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	<code>char hoststr[MAX_HOST_SIZE];</code>
3144	3144	<code>char pchoststr[MAX_HOST_SIZE];</code>
3145		- <code>ibp_op_rid_inq_t *cmd = &(op->rid_op);</code>
	3145	+ <code>ibp_op_rid_inq_t *cmd = &(op->ops.rid_op);</code>
3146	3146	
3147	3147	<code>ibppc_form_host(op->ic, pchoststr, sizeof(pchoststr), depot->host, depot->rid);</code>
3148	3148	<code>set_hostport(hoststr, sizeof(hoststr), pchoststr, depot->port, &(op->ic->cc[IBP_STATUS]));</code>
✳		

42 ■■■■ src/ibp/ibp_op.h

View

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

239	-int sub_cmd; /** sub command, if applicable
240	-tbx_ns_chksum_t ncs; /** chksum associated with the command
241	-union { /** Holds the individual commands options
242	-ibp_op_validate_chksum_t validate_op;
243	-ibp_op_get_chksum_t get_chksum_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	-};
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_chksum_t ncs; /** chksum associated with the command
241	+ union { /** Holds the individual commands options
242	+ ibp_op_validate_chksum_t validate_op;
243	+ ibp_op_get_chksum_t get_chksum_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
256	256 } ibp_op_t;
257	257


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

2 src/lcio/ds_ibp_priv.h		View
⚙	@@ -53,7 +53,7 @@ typedef struct {	
53	53 union {	
54	54 ds_ibp_alloc_op_t alloc;	
55	55 ds_ibp_truncate_op_t truncate;	
56	- };	

```

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


```

4  src/lio/ex3_types.h View

```

@@ -51,9 +51,7 @@ typedef struct {
51 51
52 52     typedef struct {
53 53         int type;
54 -         union {
55 -             exnode_text_t text;
56 -         };
54 +         exnode_text_t text;
57 55     } exnode_exchange_t;
58 56
59 57 #ifdef __cplusplus

```

2  src/lio/segment_lun.c View

```

@@ -2470,7 +2470,7 @@ op_status_t seglun_clone_func(void *arg, int id)
2470 2470     dtus = gop->op->cmd.end_time - gop->op->cmd.start_time;
2471 2471     dts = (1.0*dtus) / (1.0*APR_USEC_PER_SEC);
2472 2472     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 2475     log_printf(5, "clone_dt src=%s dest=%s gid=%d status=(%d %d) dt=%lf\n", cmd->srccap, cmd->destcap, gop_id(gop), status.op
2476 2476

```

32  src/toolbox/packer.c View

```

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

```

138	138	<code>int offset;</code>
139	139	
140	140	<code>assert(bufsize >= (p->bufsize - p->z.avail_out));</code>
✳		@@ -153,7 +153,7 @@ void pack_write_resized_zlib(tbx_pack_t *pack, unsigned char *buffer, unsigned i
153	153	
154	154	<code>void pack_consumed_zlib(tbx_pack_t *pack)</code>
155	155	<code>{</code>
156		- <code>tbx_pack_zlib_t *p = &(pack->zlib);</code>
	156	+ <code>tbx_pack_zlib_t *p = &(pack->data.zlib);</code>
157	157	
158	158	<code>p->z.avail_out = p->bufsize;</code>
159	159	<code>p->z.next_out = p->buffer;</code>
✳		@@ -165,7 +165,7 @@ void pack_consumed_zlib(tbx_pack_t *pack)
165	165	
166	166	<code>void pack_end_zlib(tbx_pack_t *pack)</code>
167	167	<code>{</code>
168		- <code>tbx_pack_zlib_t *p = &(pack->zlib);</code>
	168	+ <code>tbx_pack_zlib_t *p = &(pack->data.zlib);</code>
169	169	
170	170	<code>if (pack->mode == PACK_READ) {</code>
171	171	<code>inflateEnd(&(p->z));</code>
✳		@@ -182,7 +182,7 @@ void pack_end_zlib(tbx_pack_t *pack)
182	182	
183	183	<code>int pack_used_zlib(tbx_pack_t *pack)</code>
184	184	<code>{</code>
185		- <code>tbx_pack_zlib_t *p = &(pack->zlib);</code>
	185	+ <code>tbx_pack_zlib_t *p = &(pack->data.zlib);</code>
186	186	
187	187	<code>return(p->bufsize - p->z.avail_out);</code>
188	188	<code>}</code>
✳		@@ -194,7 +194,7 @@ int pack_used_zlib(tbx_pack_t *pack)
194	194	
195	195	<code>int pack_write_flush_zlib(tbx_pack_t *pack)</code>
196	196	<code>{</code>
197		- <code>tbx_pack_zlib_t *p = &(pack->zlib);</code>
	197	+ <code>tbx_pack_zlib_t *p = &(pack->data.zlib);</code>
198	198	<code>int err;</code>
199	199	
200	200	<code>log_printf(5, "start avail_out=%d\n", p->z.avail_out);</code>
✳		@@ -221,7 +221,7 @@ int pack_write_flush_zlib(tbx_pack_t *pack)
221	221	
222	222	<code>void pack_init_zlib(tbx_pack_t *pack, int type, int mode, unsigned char *buffer, unsigned int bufsize)</code>
223	223	<code>{</code>
224		- <code>tbx_pack_zlib_t *p = &(pack->zlib);</code>
	224	+ <code>tbx_pack_zlib_t *p = &(pack->data.zlib);</code>
225	225	
226	226	<code>memset(pack, 0, sizeof(tbx_pack_t));</code>
227	227	
✳		@@ -274,7 +274,7 @@ void pack_init_zlib(tbx_pack_t *pack, int type, int mode, unsigned char *buffer,
274	274	
275	275	<code>int pack_read_raw(tbx_pack_t *pack, unsigned char *data, int len)</code>
276	276	<code>{</code>
277		- <code>tbx_pack_raw_t *p = &(pack->raw);</code>
	277	+ <code>tbx_pack_raw_t *p = &(pack->data.raw);</code>
278	278	<code>int nbytes;</code>
279	279	
280	280	<code>nbytes = (len > p->nleft) ? p->nleft : len;</code>
✳		@@ -297,7 +297,7 @@ int pack_read_raw(tbx_pack_t *pack, unsigned char *data, int len)
297	297	
298	298	<code>int pack_read_new_data_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)</code>
299	299	<code>{</code>
300		- <code>tbx_pack_raw_t *p = &(pack->raw);</code>
	300	+ <code>tbx_pack_raw_t *p = &(pack->data.raw);</code>
301	301	<code>int err = 0;</code>
302	302	
303	303	<code>if (p->nleft > 0) err = PACK_ERROR;</code>
✳		@@ -319,7 +319,7 @@ int pack_read_new_data_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int
319	319	
320	320	<code>int pack_write_raw(tbx_pack_t *pack, unsigned char *data, int len)</code>
321	321	<code>{</code>

```

322     -   tbx_pack_raw_t *p = &(pack->raw);
322     +   tbx_pack_raw_t *p = &(pack->data.raw);
323     int nbytes;
324
325     nbytes = (len > p->nleft) ? p->nleft : len;
✱ @@ -342,7 +342,7 @@ int pack_write_raw(tbx_pack_t *pack, unsigned char *data, int len)
342
343     void pack_write_resized_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize)
344     {
345     -   tbx_pack_raw_t *p = &(pack->raw);
345     +   tbx_pack_raw_t *p = &(pack->data.raw);
346
347     assert(bufsize >= p->bpos);
348
✱ @@ -357,7 +357,7 @@ void pack_write_resized_raw(tbx_pack_t *pack, unsigned char *buffer, unsigned in
357
358     void pack_consumed_raw(tbx_pack_t *pack)
359     {
360     -   tbx_pack_raw_t *p = &(pack->raw);
360     +   tbx_pack_raw_t *p = &(pack->data.raw);
361
362     p->bpos = 0;
363     p->nleft = p->bufsize;
✱ @@ -378,7 +378,7 @@ void pack_end_raw(tbx_pack_t *pack)
378
379     int pack_used_raw(tbx_pack_t *pack)
380     {
381     -   return(pack->raw.bpos);
381     +   return(pack->data.raw.bpos);
382     }
383
384     //*****
✱ @@ -396,7 +396,7 @@ int pack_write_flush_raw(tbx_pack_t *pack)
396
397     void pack_init_raw(tbx_pack_t *pack, int type, int mode, unsigned char *buffer, unsigned int bufsize)
398     {
399     -   tbx_pack_raw_t *p = &(pack->raw);
399     +   tbx_pack_raw_t *p = &(pack->data.raw);
400
401     memset(pack, 0, sizeof(tbx_pack_t));
402
✱

```

4 ■ ■ ■ ■ src/toolbox/packer.h View

```

✱ @@ -25,6 +25,8 @@
25     #include "zlib.h"
26
27     #ifdef __cplusplus
28     +id (*write_resized)(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize);
29     +
28     30     extern "C" {
29     31     #endif
30     32
✱ @@ -48,7 +50,7 @@ struct tbx_pack_t {
48     union {
49     51     tbx_pack_raw_t raw;
50     52     tbx_pack_zlib_t zlib;
51     -   };
52     +   } data;
53     void (*end)(tbx_pack_t *pack);
54     void (*write_resized)(tbx_pack_t *pack, unsigned char *buffer, unsigned int bufsize);
55     int (*write)(tbx_pack_t *pack, unsigned char *data, int nbytes);
56
✱

```

0 comments on commit 808a152

Lock conversation



Write

Preview

AA B i


“ <> ↻

⋮ ≡ ☰


↶ @ *

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.

