Support Toolbox Manual

Chapter 5

Object Distribution Tools

Chapter 5 - Object Distribution Tools

A. Introduction

This chapter describes four tools which are used to package and unpackage toolboxes and software objects.

A.1. kcksum — *check checksums created by kcrsum*

Object Information

Category: *Command Line Tools*
Subcategory: *Object Distribution*
Operator: *Check Checksums*
Object Type: *script*
In Cantata: *No*

Description

The script will check the checksum's for the file in in a toolboxes' object. If nothing specified on the command line than all toolboxes found in the files specified by KHOROS_TOOLBOX and all the objects within are checked. You can specify multiple toolboxes by doing:

```
% kcksum -tb toolbox1,toolbox2
```

You can specify multiple objects by doing, but the object will have to be in every toolbox:

```
% kcksum -oname object1,object2
```

If you combine the two options the objects specified must be in every toolbox specified.

```
% kcksum -tb toolbox1,toolbox2 -oname object1,object2
```

By default the current checksum for the files in the object are compared to the previous checksum for the object's files located in

{toolbox}/objects/{object type}/{object}/db/{object}.cksum.

Where {toolbox} is the path to the toolboxes' top directory, {object type} is library, script, kroutine, etc. and {object} is the object name.

If -version is specified than the checksum for the files in the object are compared against that version. The checksum for the object under that version by default is located in {toolbox}/objects/{object type}/{object}/db/{version}/{object}.cksum

If -dir1 is specified than the default checksum directory is changed from

{toolbox}/{toolbox}/objects/{object type}/{object}/db

to the directory specified.

If -dir2 is specified -dir1 must also be specified. The checksum's for the files in the objects are not calculated, but the object file checksum's found in -dir1 are compared to those in -dir2. If -oname specified just those specified are compared. You can not specify -version or -tb when -dir1 and -dir2 are specified.

See Also

klint, kcrsum, kpack, kunpack

A.2. kcrsum — *creates a checksum file for object(s)*

Object Information

Category:	*Command Line Tools*
Subcategory:	*Object Distribution*
Operator:	*Create Checksum File*
Object Type:	*script*
In Cantata:	*No*

Description

The script will create a checksum file for every objects specified. It will have the format:

{path to file relitive to toolbox}/{filename}:{SYSV checksum}:{# of 512 blocks}:{size in bytes}

If nothing specified on the command line than all toolboxes found in the files specified by KHOROS_TOOLBOX and all the objects within are checked. You can specify multiple toolboxes by doing:

% kcrsum -tb toolbox1,toolbox2

You can specify multiple objects by doing, but the object will have to be in every toolbox:

% kcrsum -oname object1,object2

If you combine the two options the objects specified must be in every toolbox specified.

% kcrsum -tb toolbox1,toolbox2 -oname object1,object2

If -version is specified than the checksum for the files in the object are stored in that version. The checksum for the object under that version by default is located in

{toolbox}/objects/{object type}/{object}/db/{object}.cksum.

Where {toolbox} is the path to the toolboxes' top directory, {object type} is library, script, kroutine, etc. and {object} is the object name.

If -dir is specified than the default checksum directory is changed from

{toolbox}/objects/{object type}/{object}/db

to the directory specified.

See Also

klint, kcksum, kpack, kunpack

A.3. kpack — *pack up toolboxes or objects*

Object Information

Category:	*Command Line Tools*
Subcategory:	*Object Distribution*
Operator:	*Pack Object*
Object Type:	*script*
In Cantata:	*No*

Description

The script will pack up toolboxes or objects. It will use checksum created by kcrsum to do this if -cksum specified. Otherwise, it packs up the entire toolbox or object. If nothing specified on the command line than all toolboxes found in the files specified by KHOROS_TOOLBOX and all the objects within are packed. By default no pack file is create until -create used, but all the files to be pack will be place in the current working directory or the directory specified by -packdir. Each toolbox will be contained within a single tar file call {toolbox}.tar.Z. A file called INDEX will also be created that will tell kunpack the contents of the pack file and the operations to perform.

You can specify multiple toolboxes by doing:

% kpack -tb toolbox1,toolbox2

You can specify multiple objects by doing, but the object will have to be in every toolbox:

% kpack -oname object1,object2

If you combine the two options the objects specified must be in every toolbox specified.

% kpack -tb toolbox1,toolbox2 -oname object1,object2

If -version is specified than the checksum for the files in the object are compared againts that version. The checksum for the object under that version by default is located in

{toolbox}/objects/{object type}/{object}/db/{version}/{object}.cksum

Where {toolbox} is the path to the toolboxes' top directory, {object type} is library, script, kroutine, etc. and {object} is the object name.

If -dir is specified than the default checksum directory is changed from

{toolbox}/objects/{object type}/{object}/db

to the directory specified.

If -cksum is specified than a checksum is preformed on only those files with the object that are different are pack'ed.

If -packdir is specified than pack will be built in this directory. All the temporary files will also be created there.

If -create is specified than the output of kpack will be the file specified. If -tb and -oname not specified and a INDEX file exists in the pack directory a pack is built without adding anything new. If no INDEX file exists than a new pack file is created with all the toolboxes in KHOROS_TOOLBOX. If either -tb or -oname specified the pack directory is updated and than the pack file is built.

If -add is specified than the toolbox(es) specified will be added to the pack directory.

If -delete is specified than the toolbox(es) specified will be deleted from the pack directory.

If -info is specified than the pack information is printed out. It can be used inconjunction will -create, -add and -delete. If used by itself it will just print out the information and no changes to the pack will occur.

If -dtb is specified than those toolbox(es) specified will be stored in the INDEX file and when the pack is unpacked those toolbox(es) will be deleted.

If -dobject is specified than that toolbox object will be stored in the INDEX file and when the pack is unpacked that toolbox object will be deleted.

If -clean is specified than the intermediate toolbox tar files and the INDEX file will be deleted. This should only be used with -create is specified.

This routine uses klint to verify to integrity of the object. It is recommended that if klint fails that the toolbox or object not be packed.

See Also

klint, kcksum, kcrsum, kunpack

A.4. kunpack — *unpack a pack file created by kpack*

Object Information

Category:	*Command Line Tools*
Subcategory:	*Object Distribution*
Operator:	*Unpack Object*
Object Type:	*script*
In Cantata:	*No*

Description

The script will unpack a pack file created by kpack. It will determine the contents of the pack file and prompt the user for which toolboxes or objects to unpack and which files to delete if applicable.

This page left intentionally blank

Table of Contents

This page left intentionally blank