NAME

git-pack-refs - Pack heads and tags for efficient repository access

SYNOPSIS

git pack-refs [--all] [--no-prune]

DESCRIPTION

Traditionally, tips of branches and tags (collectively known as *refs*) were stored one file per ref in a (sub)directory under $GIT_DIR/refs directory. While many branch tips tend to be updated often, most tags and some branch tips are never updated. When a repository has hundreds or thousands of tags, this one-file-per-ref format both wastes storage and hurts performance.

This command is used to solve the storage and performance problem by storing the refs in a single file, $GIT_DIR/packed-refs. When a ref is missing from the traditional $GIT_DIR/refs directory hierarchy, it is looked up in this file and used if found.

Subsequent updates to branches always create new files under $GIT_DIR/refs directory hierarchy.

A recommended practice to deal with a repository with too many refs is to pack its refs with --all once, and occasionally run git pack-refs. Tags are by definition stationary and are not expected to change. Branch heads will be packed with the initial pack-refs --all, but only the currently active branch heads will become unpacked, and the next pack-refs (without --all) will leave them unpacked.

OPTIONS

--all

The command by default packs all tags and refs that are already packed, and leaves other refs alone. This is because branches are expected to be actively developed and packing their tips does not help performance. This option causes branch tips to be packed as well. Useful for a repository with many branches of historical interests.

--no-prune

The command usually removes loose refs under $GIT_DIR/refs hierarchy after packing them. This option tells it not to.

BUGS

Older documentation written before the packed-refs mechanism was introduced may still say things like .git/refs/heads/<branch> file exists when it means branch <branch> exists.

GIT

Part of the **git(1)** suite