

Git ?? ?? (Advanced)

1. ? Git ??? SHA-1 ?? ??

1.1 ??

Git 的 对象 (blob, tree, commit 等) 的 **SHA-1** 哈希值 是 唯一的 标识。

```
“对象 大小 40 字节 16 字节 哈希 值”
```

1.2 ??

```
e83c5163... ← 主分支, 当前分支 指向 该 哈希值
```

1.3 ?? ?? ??

- Git 的 对象 是 由 类型 + 大小 + 内容 组成的 哈希值。
例如: `blob 12\0Hello World\n`
- Git 的 SHA-1 哈希值 是 40 字节 的 字符串。

```
echo "Hello World" | git hash-object --stdin  
→ 557db03de997c86a4a028e1ebd3a1ceb225be238
```

2. Git? packfile ?? ??

2.1 ??

Git 的 对象 是 存储在 `.git/objects` 目录下的。每个对象 都有一个 唯一的 哈希值。当需要 查找 某个 对象 时，Git 会 根据 该 哈希值 来 定位 对象。如果 对象 不存在，Git 会 提示 错误。

2.2 packfile???

“Git 的 packfile 是啥？”

2.3 ?? ??

```
.git/
├─ objects/
│   └─ pack/
│       ├── pack-xxxxx.pack  ← 二进制文件
│       └─ pack-xxxxx.idx    ← 索引文件
```

2.4 ?? ???

```
git gc # 清理 packfile 垃圾 (garbage collection)
```

2.5 ??

| 操作 | 描述 |
|--------------------|-------------------------|
| clone | 克隆仓库，包含所有对象和索引 |
| clone, fetch, push | clone, fetch, push 操作 |
| packfile | packfile 是二进制文件，索引是文本文件 |

3. Git vs SVN ??

| 操作 | Git | SVN |
|----------------|----------------|----------------|
| 分支 | 分支 (DVCS) | 分支 (CVCS) |
| 提交 | 提交 (本地) | 提交 (服务器) |
| 合并 | 合并 (本地) | 合并 (服务器) |
| 撤销 | X (本地) | O (服务器) |
| 回滚 | 回滚 (本地) | 回滚 (服务器) |
| merge & rebase | merge & rebase | merge & rebase |

3.1 ?? ??

| Git📄 📄📄📄📄 📄 📄 📄📄 📄 , 📄📄📄 lightweight
→ 📄📄📄 , 📄 📄 📄📄📄 📄📄

| SVN📄 📄 📄 📄📄📄 📄📄📄📄 📄 📄
→ 📄 📄📄 📄📄 📄📄 📄

??

| 📄 | 📄 |
|------------|-------------------------|
| SHA-1 📄 | Git 📄📄 📄 +📄 📄📄📄 📄 📄 |
| packfile | 📄 📄📄📄 📄 📄 📄 📄 |
| Git vs SVN | Git📄 📄 & 📄 , SVN📄 📄 & 📄 |