

Memahami konsep trunk, branch dan tags di subversion

Author : Putera Kahfi Email :puterakahfi@gmail.com blog: <http://codejunior.net>

Banyak yang bingung dengan konsep **trunk**, **branch** dan **tags** dalam proses *software development*, kebanyakan bingung dengan tujuan di buatnya konsep tersebut, dan walaupun melakukan *development* menggunakan *subversion* itu hanya berkisar di **trunk** saja , tidak melibatkan **branch** apalagi **tags** hal ini dapat di maklumi jika proses *software development* masih dalam skala yang kecil, bagaimana dengan software yang cukup besar, hal ini akan menyusahkan jika hanya berkisar di **trunk** saja

Ambil sebuah contoh kasus, kita *development* software Sistem Informasi Akuntansi versi 1.0, *software* ini kita release dan sudah banyak client yang menggunakan, kemudian kita develop versi 2.0 dengan fitur-fitur yang lebih banyak. Ditengah jalan ternyata *client* yang menggunakan versi 1.0 menemukan bugs di salah satu modulnya. Tentu kita tidak ingin develop versi 2.0 ke client yang masih development dan unstable (lagian client kan beli versi 1.0 masak di beri versi 2.0).

Bagaimana caranya agar bugs bisa di perbaiki dan di deliver ke client, tanpa harus mengganggu proses development yang sedang berlangsung ? Solusinya adalah menggunakan branch dan tag

Untuk memperjelas pemahaman anda, maka kita akan bahas satu persatu apa yang di maksud dengan trunk, branc dan tags dan implementasinya.

1. Trunk (batang)

Trunk berisi program terbaru, artinya trunk merupakan induk dari semua program yang paling update, disinilah biasanya proses development dilakukan, penambahan feature, bug fixing dilakukan di sini sampai software di anggap sudah siap masuk ke fase testing sebelum release.

2. Branch (cabang)

Kenapa di sebut branch(cabang), karena memang isinya adalah cabang dari trunk, kalo sebelumnya trunk berisi program terbaru, branch berisi program-program yang sebelumnya, di sinilah versi-versi software tersimpan (beta, cr, 1.0, dst) .

Setelah melalui fase development di trunk, dan di rasa sudah cukup untuk di release, maka software akan di buat branch (misalnya release-1.0), di fase branch software masih dalam tahap testing dan development untuk memperbaiki bug-bug yang di temukan.

3. Tag (label/penanda)

Setelah proses testing dan perbaikan bug-bug dilakukan di branch, sehingga software sudah dianggap cukup stabil, maka dia akan di tandai (tag) untuk menandai software tersebut (ex stable-1.0 atau release 1.0), perlu di ingat setelah di tag, tidak ada lagi perubahan-perubahan software, jika di temukan bugs baru setelah fase ini, maka akan di develop versi selanjutnya (versi minor, misalnya v.1.0.1) dan akan di buat tag baru dengan nama v.1.0.1, karena tujuan dari tags adalah untuk menandai proses development.

Dari keterangan diatas, diharapkan anda sudah mengetahui fungsi dari trunk, branch dan tag.

untuk Semakin memudahkan anda, maka akan dijelaskan dengan study case berikut ini.

Developer melakukan *commit* semua pekerjaan ke dalam trunk, mereka menambahkan fitur, bug fixing, development di sini

Setelah team merasa bahwa software siap untuk di release, trunk akan di copy kedalam branch (ex:release1.0), maka trunk akan di copy kedalam /branch/releasi-1.0

Team meneruskan proses development, disisi lain team juga melakukan development untuk versi terbaru (misalnya versi 2.0), maka mereka melakukan development kembali di dalam trunk, jika bug di temukan di kedua lokasi, maka dilakukan porting perbaikan secara bolak-balik
seperlunya.

Setelah proses terting dan dirasa software versi 1.0 sudah cukup stabil, maka akan di tag(tandai), dan inilah yang akan di releasi ke *customer*, dan branch release-1.0 akan di 'bekukan' tepat sebelum software di tag.

Project-project di dalam branch terus di *maintenance*, sembari proses development versi 2.0 terus di lakukan di trunk, bug-bug yang di temukan di dalam branch/release-1.0 di akumulasikan, dan manajemen mungkin akan memutuskan untuk merelease versi minor 1.0.1 sebagai perbaikan dari versi 1.0 sebelumnya. Maka program-program terbaru hasil update di branch akan di buat tag baru bernama 1.0.1 dan package siap untuk di release