toolview

Albert Mietus'

CFLOW

De meeste programma's bestaan uit functies, waarbij zo'n functie gezien kan worden als het kleinste en belangrijkste onderdeel van een programma. Veel belangrijker dan bijvoorbeeld een file of een coderegel. Het is daarom verbazend hoe weinig aandacht er is voor de relaties tussen functies. Elke programmeur weet hoe belangrijk 'callGraph', of 'invocatieboom' is. Maar slechts weinigen weten exact hoe functies elkaar aanroepen, zelfs in hun eigen code. Het tooltje cflow kan dit automatisch bepalen

Files en Functies

Een programma is beschreven in één of meer files. Vaak in veel files, die in een directory-structuur opgeslagen worden. Een goede programmeur zorgt voor een logische structuur, waarbij alle functies in een file het kleinste onderdeel vormen van die structuur. Zoals 'static' functies, die slechts een betekenis hebben binnen die file. En waarbij de filenaam aangeeft waar een belangrijke functie als 'main()' te vinden is.
De werkelijkheid is vaak anders; de file-functie structuur is vaak meer historisch bepaald dan volgens logica. En functies roepen net zo vaak 'verre' functies aan als locale. Omdat dat de dagelijkse werkelijkheid is, valt het nauwelijks nog op.

Embedded software

Bij embedded software ontbreekt vaak zelfs de klassieke top van de invocatieboom; er is óf geen main-functie óf er zijn er meerdere. Zo is een driver, qua code, niet meer dan een verzameling losse functies, waarbij er juist dan strikte regels zijn welke functies aangeroepen mogen worden. Zo zal de main-routine nooit, ook niet indirect, een driver-functie aanroepen. Dit gebeurt slechts via de OS-laag.

Zoeken en Vinden

Met een overzicht van hoe functies elkaar aanroepen kan zo'n driver makkelijk gevonden worden. Het is een cluster van functies, die niet door een applicatie aangeroepen wordt. En, een applicatiefunctie die nooit aangeroepen wordt, zit aan de invocatieboomtop en is dus een main-routine.
Met 'cflow' kunnen we van één, een paar, of desnoods alle C-files een overzicht maken dat laat zien wat de flow door een C (of C++) programma is. Zowel een normale (voorwaartse) invocatieboom, als een achterwaarts overzicht (welke functies gebruiken een bepaalde functie) is mogelijk. Met zo'n overzicht kunnen we functies vinden op basis van plaats in het programma; dat is vaak krachtiger dan zoeken op zijn naam.

Invocatie bomen tekenen

Met 'cflow' zijn handige overzichtjes te maken. De output is tekstueel. Als we die opslaan in een file, is dit overzicht weliswaar 10 of 20 maal compacter dan de C-file; maar toch nog omvangrijk. Met een tool als 'dot', dat eerder besproken is, en een klein scriptje in awk of perl, is de output te converteren naar een plaatje. Dan is de structuur van zo'n historisch gegroeid programma van honderden functies te vangen in een plaatje.

Groeien of Snoeien

Dit plaatje laat de invocatieboom van je programma(deel) zien. Dat zou een boom moeten zijn. Of iets wat daar op lijkt. Maar het laat vaak ook zien waar scheefgroei ontstaan is. Niet elke boom is groot en sterk gegroeid; soms is 't alleen maar groot gegroeid. Met 'cflow' kun je daarom ook gericht gaan snoeien; mocht er tijd voor zijn. Maar minimaal kun je iets gerichter groeien.

Have fun!

---

toolview

Elke ingenieur gebruikt tools en iedereen heeft zijn eigen voorkeur; vaak gebaseerd op wat er toevallig als eerste beschikbaar was. Zo ook de software ingenieur. Ook hij gebruikt tools, software-tools wel te verstaan. Daarvan zijn er oneindig veel, waardoor kiezen erg moeilijk is. Veel *handige tools* worden niet gebruikt, eenvoudigweg omdat niet bekend was dat er een tool is, specifiek voor dat probleem.

Het leren gebruiken van een tool kost tijd. Maar als je de naam en het doel van het juiste tool kent, loont het gebruik vaak al de moeite. Het inleren en inzetten kost meestal minder tijd dan aanmodderen zonder goed tool.

Deze rubriek stelt dergelijke *handige tools* voor en bouwt soms verder op reeds besproken tools. Alle besproken ToolViews zijn daarom ook online beschikbaar op: http://albert.mietus.nl/read.IT

Mail me, voor meer info
ALbert dot Mietus at PTS dot nl

door & more

Dit programma is oud! Uit de prehistorie zou je kunnen zeggen, de eerste versies zijn verspreid via de Usenet newsgroup comp.sources.unix; een voorloper van het internet van nu. Het is door talloze mensen samengesteld - meer dan tien jaar geleden - en staat vermeld in een file uit 1995!

Omdat cflow een script is met twee hulp-programmaatjes en verder nog gebruik maakt van GCC, is het nog steeds prima bruikbaar.

Een echte web-home is er dus niet. Via Google kunnen we meerdere sites vinden.
Opgelet: Niet elke hit is bruikbaar; zo is er ook een cflow dat (IP) netwerk verkeer analyseert; dat is een ander tool. Wel goed is:
http://www.freshports.org/devel/cflow/