

# Kemerdekaan Teknologi

*Romi Satria Wahono*

*Peneliti LIPI dan Koordinator Umum IlmuKomputer.Com*

17 Agustus bagi anak bangsa memiliki arti yang sangat penting, suatu tanggal bersejarah dimana republik ini menyatakan kemerdekaan. Pernyataan merdeka mengandung makna bahwa telah bebas, bebas bukan hanya dari satu penjajah, tapi juga seluruh penjajah yang telah, sedang dan akan menjajah republik kita. Bebas memilih partner dan teman, bebas dalam bekerjasama dengan bangsa apapun di dunia ini. Meskipun secara formal republik sudah merdeka, secara informal republik ini belum bebas. Belum bebas memilih partner dan bekerjasama dengan bangsa lain. Masih banyak tekanan dan paksaan dari bangsa yang lebih kuat dari segi ekonomi, politik maupun militer yang membuat kita sulit bergerak. Ini yang sering disebut oleh para pengamat politik sebagai kita belum merdeka.

Kemerdekaan dalam sudut pandang teknologi memiliki konsep yang sama dengan kemerdekaan berbangsa, meskipun sedikit berbeda dalam penerapan. Kemerdekaan bagi seorang teknolog, engineer atau profesional adalah kebebasan dalam menggunakan teknologi, metodologi dan approach apapun dalam menyelesaikan masalah. Teknologi, metodologi dan approach bukanlah agama yang perlu difanatikkan, dia bukanlah sesuatu yang kekal hidup di dunia ini. Mereka itu adalah ciptaan manusia yang bisa dihapuskan, bisa diganti, dan bisa diperbaiki ketika mungkin sudah tidak efektif dan efisien dalam penyelesaian masalah.

Ketika 15 tahun lalu pertama kali menggunakan PC dengan sistem operasi MS DOS dengan dosshellnya yang canggih, saya berpikir bahwa dengan menguasainya saya bisa memecahkan banyak masalah (menulis, berhitung, bermain game, dsb). Tapi tiga tahun kemudian Windows 3.1 datang dan ini memungkinkan pemecahan masalah dengan lebih baik lagi. Demikian juga tahun 1995 muncul satu sistem operasi buatan Microsoft yang lebih baik lagi yaitu Windows 95. Dan saya berpikir bahwa cukup dengan itu saya bisa melakukan banyak hal, mengerjakan laporan, berhitung, manipulasi image, dsb. Tapi lagi-lagi meleset. Masuk kampus di Saitama University, semua berbasis Unix (Sun Microsystem). Saya harus mengerjakan semua tugas dengan text editor bernama Emacs, menulis laporan dengan LaTeX, mengolah data dengan Gnuplot, menulis code dengan C, bermain-main web programming dengan perl dan CGInya. Di rumah karena tidak mungkin saya membeli Sun Sparc, saya menggunakan Linux untuk melanjutkan tugas-tugas kuliah. Saya mencoba berbagai distribusi Linux dari Slackware (era itu termasuk yang populer), Redhat, Turbolinux, Debian, Vinelinux, dsb.

Dengan menguasai MS Windows, Unix, dan Linux, saya pikir sudah cukup matang dan luas skill saya. Sejak tingkat 2 program undergraduate di Saitama University (1996), saya mencoba parttime (arubaito) di berbagai perusahaan IT Jepang. Saya cukup terkejut karena saya harus mengelola mesin lain di luar itu. Berbagai perusahaan menginginkan supaya saya juga bisa mengelola mesin berplatform Macintosh (MAC) dan BSD Unix. Macintosh yang pertama membingungkan saya, akhirnya menjadi sahabat saya karena saya putuskan memakai di rumah disamping Linux dengan distro Slackwarenya. BSD Unix saya gunakan kemudian untuk berbagai server di kampus. Meskipun ini baru sampai ke sistem operasi, maintenance dan pengelolaan server, saya sudah mulai bisa menarik kesimpulan bahwa teknologi berkembang dengan varian yang beraneka ragam. Mereka masing-masing sangat unik dan saling melengkapi dalam proses pemecahan masalah. Sekali lagi, secara teknologi

tidak ada yang bisa mengatakan bahwa suatu platform adalah terbaik dalam segala hal. Yang ada adalah terbaik dalam suatu sisi. Macintosh dengan kestabilan desktopnya, Linux dalam keterbukaan source dan kestabilan server, MS Windows dalam kemudahan pemakaian, BSD dalam sekuriti, dsb. Kelebihan dalam suatu sisi itu yang melengkapi penyelesaian masalah secara global masyarakat di dunia.

Apakah cuman masalah sistem operasi? Ternyata tidak! Dalam bahasa pemrograman juga seperti itu. Ketika saya menganggap bahwa bahasa C dan Perl cukup, karena hampir semua laporan dan project di kampus menggunakan bahasa C serta sebagian Perl. Saya dikejutkan karena kebutuhan perusahaan tempat saya bekerja part time adalah bahasa Java untuk beberapa project. Untung saya sudah membiasakan diri di semester ke 4 (awal 1997) di mata kuliah Computer Graphics. Dan ini otodidak, karena pemrograman berorientasi object baru diajarkan di tingkat 3. Saya juga mulai mengganti kebiasaan Perl dengan PHP, khusus untuk web programming karena kebutuhan lebih banyak di situ.

Demikian juga dengan hobi saya menggunakan notasi OMT (Object Modeling Technique) milik James Rumbaugh dalam desain object-oriented, harus berubah karena muncul UML (Unified Modeling Language) yang akhirnya diusulkan menjadi standard oleh James Rumbaugh, Grady Booch dan Ivar Jacobson di OMG (Object Management Group).

Dunia penelitian juga tak lepas dari itu, kita harus lebih banyak membaca jurnal-jurnal ilmiah terbaru untuk mengupdate informasi dan pengetahuan kita tentang berbagai approach, teknologi, metodologi, formula baru yang telah ditemukan oleh berbagai peneliti lain di dunia. Kadang dengan pemikiran baru itu kita bergerak dan harus meninggalkan berbagai tema penelitian yang sudah tidak terlalu dibutuhkan oleh masyarakat.

Mari kita renungkan bersama, suatu hal yang lucu kalau kita terbelenggu oleh teknologi. Walaupun kita sangat suka komputer tabung, tentu kita tidak bisa lagi menggunakannya di era saat ini. Walaupun kita fanatik terhadap pascal dan quickbasic, kita sudah mulai kesulitan mencari kompilernya saat ini. Walaupun kita sangat lihai bermain MS DOS, ngoprek dosshell, tentu harus ditinggalkan dengan sistem windowing (X window, Mac window, window ala MS Windows, dsb)

Sekali lagi teknologi, approach, metodologi dalam bentuk sistem operasi, bahasa pemrograman, software, dsb hanyalah tool yang harus dikuasai dan digunakan bagi para teknolog dan profesional untuk memecahkan masalah. Dia bersifat tidak kekal, dia bukanlah agama yang harus dianut dan difanatikkan seumur hidup. Ketergantungan terhadap sebuah tool adalah kebodohan. Debat kusir tentang tool dan saling mengumpat atau membela mati-matian sebuah tool adalah tindakan sia-sia.

Kemerdekaan bagi seorang teknolog adalah kebebasan dan kemampuan dalam memilih, memilah dan menggunakan berbagai teknologi, tool dan approach dalam memecahkan masalah. Ketika kita terbelenggu dan terpaksa dalam menggunakan suatu teknologi, maka itu menandakan bahwa kita belum merdeka, dan perlu perdjoeangan untuk memerdekakan diri :)

Tentu kemerdekaan teknologi yang lebih tinggi tingkatnya adalah apabila kita bukan lagi sebagai pengguna, tapi juga sebagai pencipta dari teknologi. Sehingga kita tidak perlu lagi menggunakan teknologi enkripsi khususnya untuk kunci publik yang diciptakan Martin Hellman dan Adi Shamir, maupun algoritma kompresi data (misal dalam format ZIP) yang

merupakan karya Abraham Lempel dan Jacob Ziv. Perlu diketahui bahwa empat orang yang saya sebut baru saja adalah orang Israel :(

17 Agustus 2006, menyambut kemerdekaan RI yang ke-61, jadikanlah tanggal ini suatu timing yang tepat untuk mulai memikirkan kemerdekaan teknologi, meskipun kita mulai dari hal-hal kecil.

Merdeka !!