

DATORER

PROGRAM

NÄTVERK

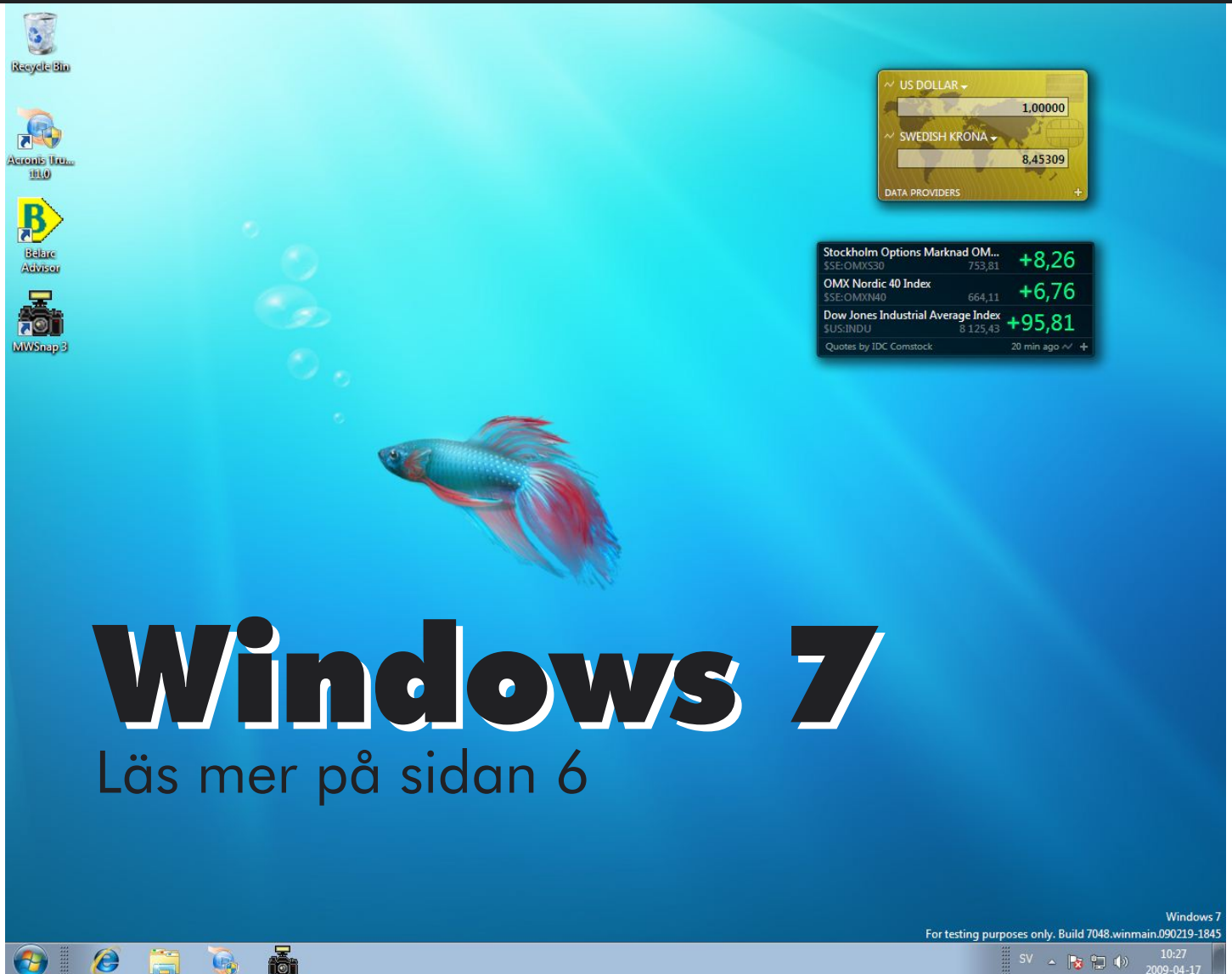
ABC BLADET

ABC-klubbens medlemstidning

Grundad 1980

Nr 1/2 2009

Pris 65 kr



Tommy Florstedt

Internet Explorer 8

E-post & IT-säkerhet

LinQ - fortsättning

Google Android

ABC-bladet utges av ABC-Klubben och
utkommer 4 ggr/år

Ansvarig utgivare

Bo Saltorp <7585>

Redaktionen

Elisabeth Sporrang <10892>

Anders Höglin <10216>

Peter Nermander <8130>

Erik Nyman <9520>

Sven-Erik Svensson <4394>

Sten Wickberg <1384>

Lars Magnusson <8827>

Johan Kullberg <5644>

Chefredaktör

Elisabeth Sporrang <10892>

Webb och e-post

Webbplats: www.abc.se

SSH och ftp: sun.abc.se

Kansliet: kansli@abc.se

ABC-bladet: red@abc.se

Postadress

Box 14 143, 167 14 Bromma

Tel. Klubblokalen/kansliet:

08-80 17 25

(Säkrast tisdag kvällar - klubbkväll)

Styrelsen

Dagmar Elleby - Ordförande

Eric Rowley - Vice ordf.

Bo Saltorp - Sekreterare

Tommy Florstedt - Kassör [avliden]

Jan Roos

Johan Kullberg

Valberedning

Allen Ristic <8646> (sammankallande)

Claes Tullbrink <6083>

Anders Höglin <10216>

Avgifter

Medlemsavgift: 300SEK/kalenderår

Bankgiro: 5986-2169

Org.nr: 802010-8174

Innehållet lagras och publiceras elektroniskt.

För obeställt material ansvaras ej.

Tryck och Distribution:

Botkyrka Offset

Utgivningsbevis: 12027

ISSN: 0349-3652

Innehåll

Några ord från ordföranden.....	3
Dagmar Elleby <10342>	
Tommy Florstedt.....	4
Medlemsenkät.....	5
Windows 7.....	6
Bo Saltorp <7585>	
Internet Explorer 8.....	9
Johan Kullberg <5644>	
E-post och IT-säkerhet.....	10
Lars Magnusson <8827>	
ABC-bladet Open Source.....	11
Mera LinQ.....	12
Anders Franén <5258>	
Google Android.....	18
Tomas Gustavsson <7305>	
Slutbilden.....	20
Kalendariet.....	20

Utgivningsplan

Nummer 3/4 2009 : Manusstopp 15 Nov.

Nummer 1 2010: Manusstopp 30 December

Nummer 2 2010: Manusstopp 30 Mars

Upprop från ABC-Bladet!

Bästa medlem. ABC-Bladet är i behov av att få fler eminenta artiklar att publicera; om operativsystem, netbooks, multimedia, hårdvara, mjukvara, programmering eller annan, för våra medlemmar, nyttig IT-information. Se på vad DU kan tillföra av kunskap, det finns alltid något.

Att skriva kan kännas pirrigt och ovant, men behöver du tips och hjälp, kontakta redaktionen. Vi är heller inte proffs, men vi har prövat på det själva. Och det handlar oftast om 2-4 A4 sidor, kanske med någon bild.

Som författare i Bladet har du hedern, äran och en erfarenhet du kan ta upp i din CV eller personbeskrivning, om du fortfarande underhåller en sådan ;-). Att ha publicerat en artikel i en offentligt registrerad tidning, med en upplaga på 1000 ex. Det är faktiskt något.

Några ord från ordföranden

Dagmar Elleby <10342>

Hej alla medlemmar.

På årsmötet i februari blev jag vald till ordförande. Jag kan bara säga att det är jättekul att arbeta och förverkliga mina mål inom ABC-klubben. Nummer ett är att få igång kontakt och samarbete mellan medlemmarna. Jag skall precisera mig lite bättre i en enkät som finns på annat ställe i bladet och på medlems-webben.

Jag vill framför allt att kontakten mellan medlemmarna förstärks, så att alla skall få tillfällen att träffas, delta i kurser och andra aktiviteter.

Rent praktiskt vill jag att vi skall ha en "Fadder-pool". Att medlemmar i Stockholms-området skall kunna ställa upp för medlemmar som bor längre bort, med att hjälpa till med övernattningar. Nya medlemmar bör ha en fadder som finns till hands när det uppkommer frågor.

Min framtidsvision är att klubblokalen sjuder av aktiviteter, kurser, föredrag och bytes-träffar, eller sånt som ni föreslår.

Medlemmar skall komma till lokalen i Alvik och använda våra datorer för att göra hemsidor, göra multimedia, designa affischer eller bara få hjälp att komma igång.

Se på frågeformuläret - fundera över frågorna och fyll sedan i på webben!

Formuläret är en viktig del i klubbens utveckling, vi skall målmedvetet arbeta med att skapa en klubb så som medlemmarna vill ha. Det är bara med medlemmarnas deltagande, med medlemmarnas engagemang och vilja vi kan göra klubben innehållsrik.



Berätta om Dig själv, vad Du tycker, vad Du vill ha hjälp med, eller vad Du är bra på och vill/kan dela med Dig. Jag har också planerat att ha en del aktiviteter, som en byteshelg med picknick, föreläsningar som vi även streamar ut över nätet för er som inte kan komma till Stockholm. Och så bör vi ha fester, där vi bara träffas och umgås, och äter en middag tillsammans.

Och till sist önskar jag Er alla en trevlig höst.

Dagmar

Tommy Florstedt har gått bort



Det är med stor sorg, som vi i ABC-klubben fått kännedom om Tommy Florstedts hastiga bortgång den 8 juli.

Tommy har under en lång följd av år, över 10 år, gjort klubben stora tjänster, bl.a. som kassör, styrelseledamot och sysop.

Medlemmar som haft möjlighet att besöka klubblokalen i Alvik på klubbkvällarna, tisdagskvällar, har också kunnat lära känna Tommy personligen. Vi har haft möjligheten att diskutera vitt skilda datorproblem med Tommy. Baserat på den stora kunskap han hade vad gäller datorer, lämnade han gärna både hjälp och råd till oss, som hade problem.

Vi vill nu å ABC-klubbens vägnar till Tommys familj framföra vårt djupa deltagande i den sorg och förlust som har drabbat oss alla.

ABC-klubbens styrelse

Medlemsenkät

För att föreningen bättre skall kunna planera inför framtiden, har Styrelsen tagit fram en web-enkät (ni finner den på www.abc.se/member/enkaet.html). Styrelsen skulle uppskatta om du tog dig tid att gå in på sidan för att besvara frågorna.

Frågorna presenteras även här (men måste alltså besvaras på webben):

- Vet du vad ABC-KOM är? (Ja/Nej):.....
- Använder du konferenssystemet idag? (Ja/Nej):.....
- Har du använt konferenssystemet men slutat? (Ja/Nej):.....
- Tycker du att konferenssystemet är krångligt? (Ja/Nej):.....
- Skulle du hellre använda ett annat konferenssystem, som t.ex. news-grupper eller e-postlistor? (Ja/Nej):.....
- Om Ja på förra frågan, vilket system:.....
- Har du någon annan ISP (Internetuppkoppling) än ABC-Klubben? (Ja/Nej):.....
- Läser du medlemstidningen ABC-Bladet? (Ja/Nej):.....
- Kan du tänka dig ABC-Bladet enbart som webbtidning (Ja/Nej):.....
- Använder du e-post på ABC-Klubben? (Ja/Nej):.....
- Använder du Usenet/Netnews på ABC-Klubben? (Ja/Nej):.....

Vilka av klubbens tjänster känns viktiga för dig? Svara i valfri ordning med allt som känns viktigt för dig:

- a. Konferenssystemet ABC-KOM.....
- b. E-post-servicen.....
- c. Medlemstidningen ABC-Bladet.....
- d. Hemsida-utrymmet (egen WWW-sida).....
- e. News-servicen (usenet news).....
- f. Möteslokalen i Alvik.....
- g. Kurser i klubbens regi (t.ex. bildbehandling, Macromedia Flash, Linux m.fl.).....
- Har du tittat på klubbens hemsida (www.abc.se) någon gång det senaste året? (Ja/Nej)....
- Har du varit i klubbens möteslokal i Alvik? (Ja/Nej).....
- Använder du dig av klubbens Unix-miljö? (Ja/Nej).....
- Har du ringt till klubben någon tisdagskväll för hjälp eller frågor? (Ja/Nej).....
- Vad vill du få ut av ABC-Klubben i framtiden?

- Är det något som du vill framföra?

Ett par nya frågor:

- Mitt medlemsnummer är:.....
- Namn.....
- Gata.....
- Postnummer.....
- Ort.....
- Jag kan vara fadder till en ny medlem (ja/nej):.....
- Jag kan vara fadder och hjälpa med sovplats i Stockholm(ja/nej):.....
- Jag har plats i bil till Stockholm (ja/nej).....
- Jag kan hålla kurs(ja/nej):.....
- Jag kan komma en tisdag kväll och föreläsa om.....

Jag vill ha hjälp med :

- hemsida(ja/nej):.....
- photoshop(ja/nej):.....
- uppkoppling till internet(ja/nej):.....
- byta hårddisk (ja/nej):.....
- litteratur till.....

Tack för att Du tog Dig tid!

Windows 7

Bo Saltorp <7585>

Nästa operativsystem är nu på väg från Microsoft (MS). Det är lätt att glömma hur ofta MS kommer med nya OS.

Windows 98 SE kom 1999, Windows 2000 och ME år 2000. Windows XP kom 2001 men sedan var det en lång paus innan Windows Vista kom år 2007.

Jag har inte lyckats lista ut varifrån 7:an i Windows 7 kommer. I början av 90-talet fanns ju Windows 3.0 och 3.1, men sedan upphörde numreringen i och med att Windows 95 kom.

Nog om detta.

Vad är nu Windows 7? Enkelt uttryckt är det en förbättring av Vista, där MS har lyssnat på den kritik, som kommit från användare av Vista.

Hårdvarukraven är de samma som för Vista vilket innebär 1GHz Processor, 1GB RAM och 16 GB Hårddisk.

Att det är en uppgradering av Vista märker man redan vid installationen, där man får välja mellan att göra en Upgrade (från Vista) eller att göra en Custom (Advanced) installation. Jag valde det senare och installerade således Windows 7 på en tom partition. Installationen löper i stort automatiskt, man behöver endast göra ett par ingrepp.

Om man väljer Upgrade måste installationen göras på den partition som innehåller Vista. Då följer automatiskt de inställningar man hade i Vista med. Motsvarande automatik för Windows XP finns inte, så här får man föra över allt själv.

Nyheter i stort

De flesta förändringar ligger i användargränssnittet. Här har Taskbar (Aktivitetsfältet) fått en utökad roll,

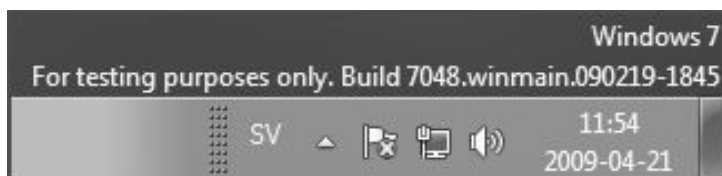


Bild 2: Meddelandefältet har färre ikoner.



Bild 1: Taskbar har fått större betydelse.

bl.a. beroende på att Quick Launch Bar har (Snabbstartfältet) försvunnit. Taskbar har också blivit högre, för att få plats med större och utökade ikoner.

Metoden att skapa ett hemmanätverk har förenklats genom att man vid installationen kan välja en s.k. Homegroup och ange ett lösenord för denna. Då kommer automatiskt övriga datorer att kunna kommunicera med varandra. Detta förutsätter dock att de övriga datorerna på hemmanätverket också kör Windows 7.

Vidare har Windows Mail (=Outlook Express i XP), Windows Movie Maker och Windows Photo Gallery försvunnit.

Det nya användargränssnittet mera i detalj

Som tidigare nämnts har Taskbar (Bild 1) nu fått en större betydelse. Här kan man nu placera ikoner för icke startade program, vilka dock inte längre är försedda med någon text. Man drar program, som finns under Start > All Programs och släpper dem på Taskbar.

Denna möjlighet ersätter således Quick Launch verktygsfältet, som har försvunnit.

På Taskbar ser man skillnad på startade/icke startade program genom att de startade programmens ikon är försedda med en ram.

Meddelandefältet (= listen till höger om Taskbar) innehåller nu mycket färre ikoner än tidigare (Bild 2). Kvar finns endast ljudvolym- och nätverksikonerna samt en ny ikon, som heter Action Center. Under denna har funktionerna för Network firewall, Virus, Malware, UAC och Internet Security m.fl. samlats. Här kan speciellt nämnas att UAC (User Account Control), som infördes i Vista men som irriterade många nu fått 4 nivåer, mot tidigare 2 (På eller Av). Under Action Center (Bild 3) finns också Windows Update och Windows Backup.

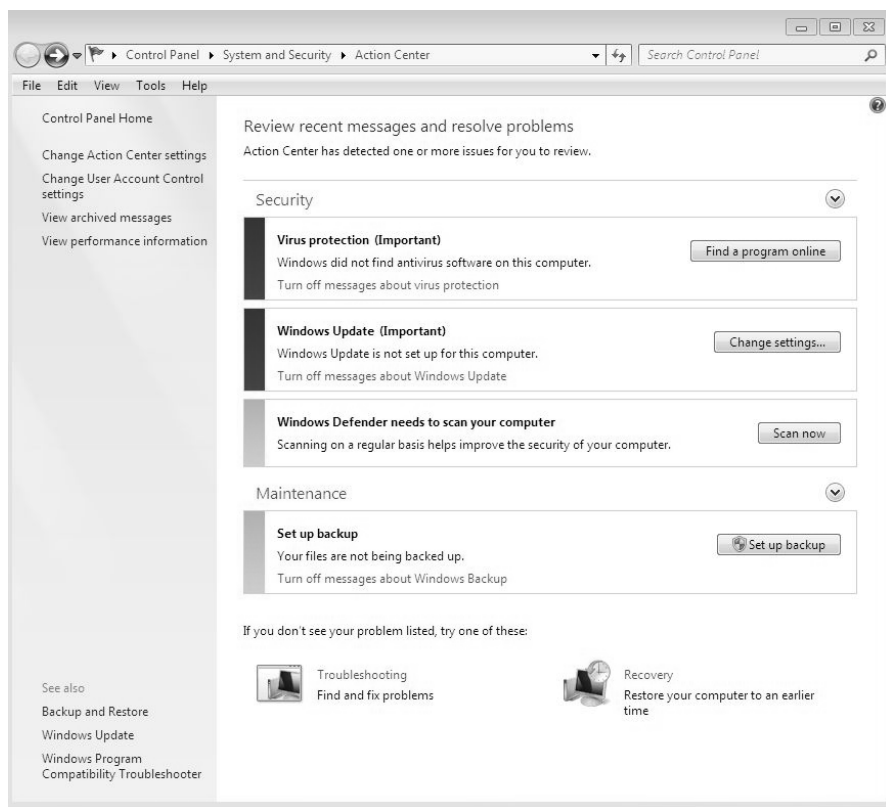


Bild 3: Det ny Action Center innehåller bland annat virusskydd, brandvägg och Windows Update.

I start-menyn (Bild 4) har bl.a. tillkommit en länk till Devices and Printers, vilken ger direkt tillgång till alla anslutna enheter. Vidare har den i Vista svårbegripliga Shut down nu fått betydelsen Stäng av.

I Vista innebar som bekant Shut down att datorn gick i ide.

Vidare har MS infört begreppet jump lists, som är ett slags snabbänkar. Se Bild 1, som visar de snabbänkar för Windows Explorer (Filhanteraren) och som i detta exempel utgörs av de mappar och filer (=dokument) som senast besökts.

Sidpanelen (Gadgets panel i Vista) har försvunnit. I stället kan nu gadgets placeras var som helst på skärmen. För den som inte känner till Vista kan nämnas att Gadgets är förstorade ikoner, som i realtid kan visa t.ex. börskurser eller valutakurser.

Eftersom normalt alla gadgets täcks av applikationsfönster kan man snabbt se alla gadgets genom att föra muspekarn ner i det högra hörnet av skärmen. Då blir alla öppna fönster genomskinliga. Om man vänsterklickar, då muspekarn står i högra, nedre hörnet, minimeras alla fönster ned till Taskbar, vilket ju är detsamma som Show the Desktop i XP och Vista.

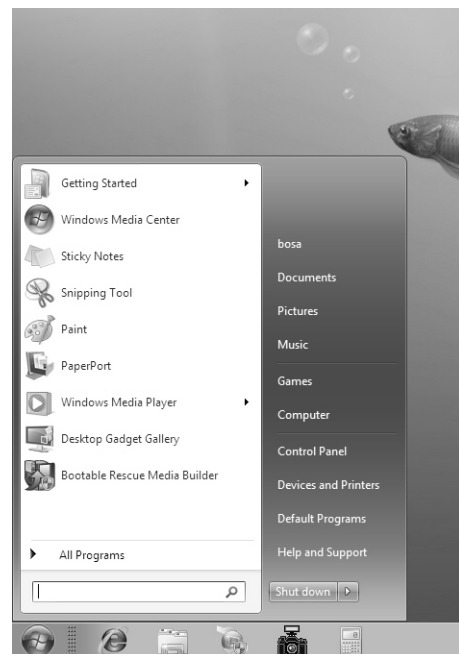


Bild 4: Startmenyn har fått en del nya ikoner.



Bild 5: Gadgets kan nu placeras var som helst på skärmen.

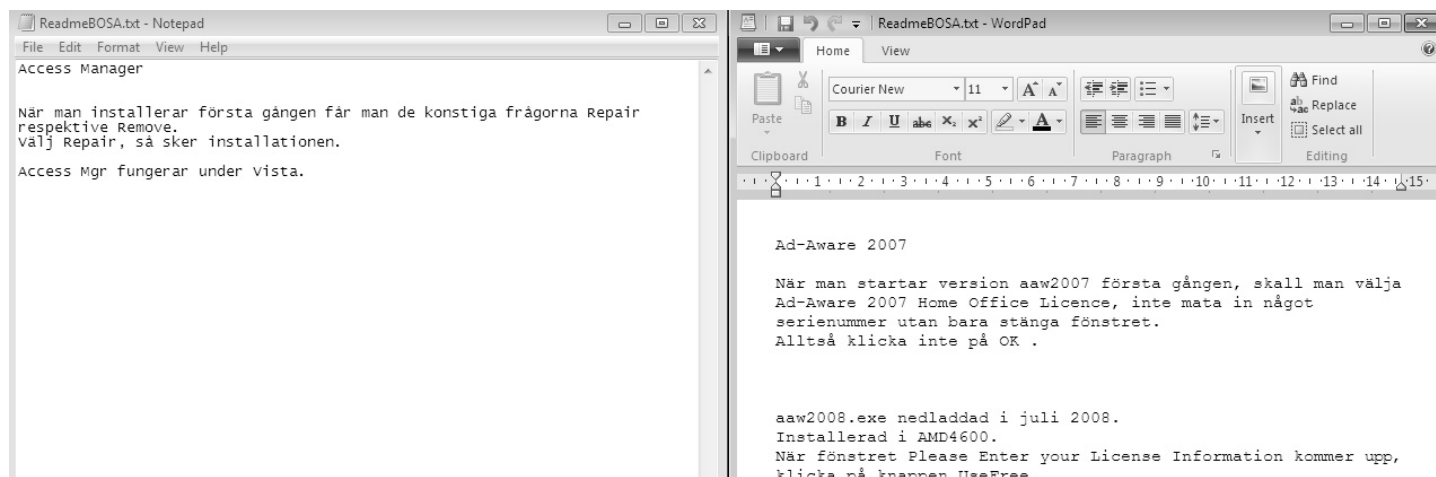


Bild 6: det är enkelt att placera två fönster sida vid sida.

Övriga ändringar

Om man exempelvis har två fönster på skärmen, i mitt exempel ett Notepad-dokument och ett Wordpad-dokument och vill granska dessa samtidigt, kan man enkelt få dem sida vid sida och lika stora.

Man drar då det ena fönstret tills muspekarn träffar den vänstra skärmkanten. Då ändras automatiskt storleken på detta fönster så att det precis täcker den halva, vänstra skärmytan.

Gör man sedan motsvarande manöver med det andra fönstret, men för muspekaren mot den högra skärmkanten, så placeras detta fönster så att det täcker den högra halvan (Bild 6).

Andra nyheter

Drivrutiner

Eftersom Windows 7 bygger på Vista med SP1 (SP2 hade inte kommit, då denna artikel skrevs), så kommer det sannolikt inte blir några problem med dessa.

Energisparfunktioner

Processorkärnor som används väldigt lite, eller inte alls, kan stängas av. Denna funktion stöds bland annat av Intels Nahalem-processorer. Windows 7 nöjer sig dock inte med att stänga av oanvända kärnor, operativsystemet kan även samla ihop flera program på en processorkärna för att inte belasta andra kärnor i onödan.

Det är vanligt att bärbara datorer används för att visa DVD-film. Windows Media Player 12, som kommer i Windows 7, använder ett antal tricks för att förbättra batteritiden. Bland annat buffras större delar av filmen i minnet vilket låter DVD-spelaren varva ned mellan läsningarna. Dessa småförbättringar uppges ge totalt mellan tio och femton procent längre batteritid vid DVD-uppspelning.

SSD-diskar

Något som inte märks på ytan är att Windows 7 nu förberetts för SSD-diskar. Dessa har andra egenskaper än dagen magnetiska diskar och kräver en annan teknik vid skrivning och läsning.

Utan att gå in i detaljer på detta, så måste hänsyn tas till att de inte tål att skrivas till hur många gånger som helst. Vidare har defragmentering ingen betydelse för söktiderna. Därför stängs schemalagd defragmentering av för att förlänga diskens livslängd.

Virtuella kataloger

Vidare införs begreppet virtuella kataloger, vilka benämns libraries (bibliotek). Dessa innebär att man kan ha filer och mappar på olika anslutna hårddiskar samlade under en logisk katalog.

En annan detalj är den mycket förbättrade Calculator (miniräknaren), som nu har funktioner för vetenskapliga beräkningar (sinus, cosinus), sortomvandlingar (t.ex. gallon till liter) och statistiska beräkningar (standardavvikelser, medelvärden).

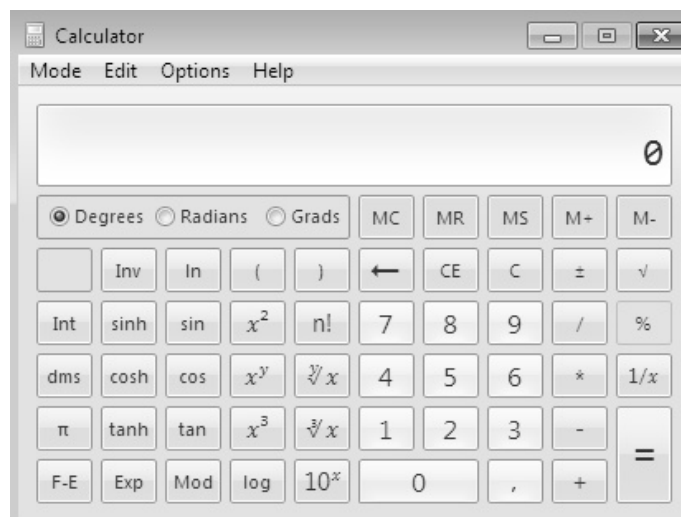


Bild 7: Kalkylatorn har fått en hel del nya funktioner.

Dags för IE8?

Johan Kullberg <5644>

Microsoft Internet Explorer (MSIE) version 8 finns nu tillgänglig. Den innehåller bl a följande nyheter (läs mer på t ex

<http://www.microsoft.com/windows/internet-explorer/default.aspx>).

Sökningar går vidare till andra sajter

Från ett ord i en webbsida kan man gå vidare till t ex en söktjänst, en karttjänst eller ett lexikon såsom Wikipedia. Webbläsarens sökfunktion är även utökad till att kunna söka vidare via s k search providers - det kan vara t ex Wikipedia, Yahoo etc.

Det går också att genom funktionen web slices att få veta att en webbsida uppdaterats, t ex en artikel som uppdaterats eller om nya email har kommit. Tjänsterna bygger på att webbleverantörerna har implementerat stöd för funktioner.

Låtsas vara version 7

Version 8 kan på olika sätt maskera sig som version 7. Man kan själv ställa in detta för den aktuella sidan man surfar till eller ha en lista över webbsidor som skall hanteras som om det var 7:an. Det är även möjligt att koda webbsidor med en meta-tag, eller ställa in IIS eller Apache-servrarna så att de skickar ut en header för att styra detta.

Webbsidekrasher isoleras

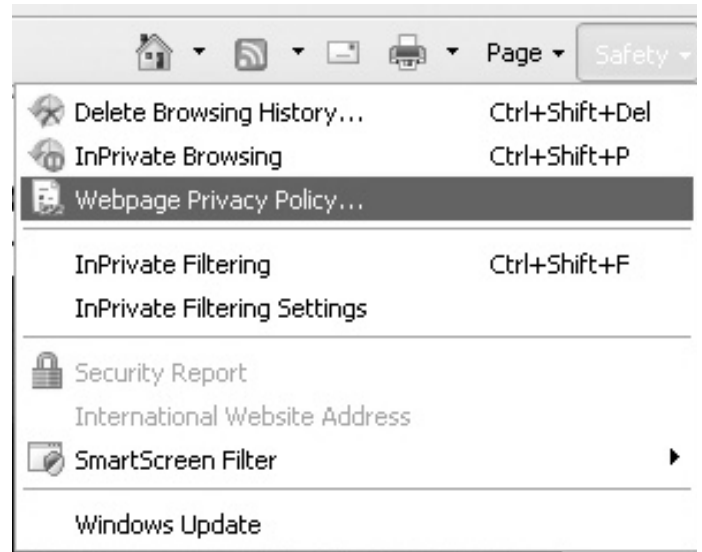
Om en webbsida i egen flik "kraschar", så påverkas bara denna flik och man skall kunna ta sig tillbaka till sidan man var på före kraschen.

Säkerhetsfunktioner

InPrivate Browsing - man kan för valbar flik välja att surfa i ett InPrivate-läge. Därmed sparas ingen historik kring vad man besöker.

InPrivate Filtering - detta används för att blockera "tredjepartssajter", som sådana som försöker att spåra dina surfvanor, t ex marknadsföringsunderlag.

Filtreringen måste dock slås på av användaren inför varje session (jag hoppas att i framtida versioner blir det möjligt att ställa in generellt). Denna funktion blockerar inte något som verkligen hör till den sajt man besöker.



Internet Explorer 8 har flera nya säkerhets och integritetsfunktioner.

SmartScreen Filter - Med detta påslaget så blir användaren varnad om man går till en sajt som är betraktad som osäker, vad gäller t ex phishing och annat otyg. Är dessutom filter för att motverka Cross Site Scripting (XSS).

Det finns även utökad kontroll av att inga insticksprogram eller annat man laddar ned under surfandet försöker sig på att skriva i minne där det inte skall skriva.

I URL-fältet är domänen "highlighted" så att man lättare ser om webbläsarens omdirigeras till önskad sida/server.

Stöd för nya standarder

Stöd för HTML 5 är under framtagande. Se t ex: <http://www.w3.org/TR/2008/WD-html5-diff-20080610/> för mer detaljer om detta. Det finns även stöd för CSS 2.1 och de mer "populära" delarna av CSS 3.

Gamla standardversioner kommer även att fungera i version 8, antagligen lång tid framöver.

Och på grund av den ökade hotbilden på nätet, är det kanske dags att inte vänta, utan nu byta till IE8.

E-post och IT-säkerhet

Lars Magnusson <8827>

I våras, månadsskiftet mars-april, fick rätt många av klubbens medlemmar ett e-brev från "systemadministratör". Brevet talade på halvdålig svenska om att det hade uppstått ett fel och att systemadministratörerna därför behövde medlemmens namn, adress, användar-ID och lösenord, för att lösa problemet.

Som tur var insåg de som fick brevet att det var något fiffel och båg med det. Vi fick flera förfrågningar till Sysop@abc.se, som klubbens administrationsgrupp nu heter, och inte "systemadministratör", så problemet kunde hanteras utan att klubben säkerhet riskerades.

Men under ett samtal inom sysopgruppen vid årsmötet, kom vi fram till att det behövs nog en liten refresh avseende e-post och IT-säkerhet.

Bassäkerhet

För att börja i den ände som nu gällde; det så kallade "phishing"-försöket ovan, så kom ihåg följande grundregel:

Klubbens funktionärer frågar aldrig, aldrig er medlemmar om era lösenord.

Vi kan, om så skulle behövas för att lösa något problem, komma in på era konton på andra och mycket säkrare sätt. Skulle det värsta uppstå, byter vi erat lösenord tillfälligt. Men vi frågar aldrig om lösenord och ni skall aldrig någonsin sända det till oss med e-post. Det gäller alltid.

Nu har vi för de flesta ändamål börjat se på annan distribution av lösenord. Det gäller ännu inte alla system med separata lösenord, men för de viktigare som inte använder det vi kallar för shell-loginet. Ansvarig sysop lägger då helt enkelt en textfil med det nya lösenordet i användarens hemmakatalog på sun.abc.se och meddelar via e-post att det finns där.

I några fall sänds lösenord fortfarande ut med e-post, men vi ser på att ändra processen där. T.ex. för de som beställt tillgång till Usenet/Netnews, så kommer lösenordet fortfarande med e-brev.

Men det kommer då inte ihop med användar-ID, utan i ett eget e-brev. Allt för att undvika att lösenord skall kunna användas av fel personer.



Detta betyder, att all begäran om att få ditt lösenord till något av klubbens system skall ses som falska. Det är vad man kallar "phishing" eller "fiskning", i avsikt att kunna bryta sig in på ditt ABC-konto.

Faktum är att de flesta Internetleverantörer idag har gått ifrån att fråga efter lösenord vid problem, man byter rutinmässigt ut lösenordet i stället. Några, som Spray Mail, gör det fortfarande inte i en del fall, men de är ett krympande fåtal.

Hur skyddar jag mig?

"Men kan ingen lyssna på när jag kopplar upp för att hämta posten och då få lösenordet?"

Jo den risken finns. Tyvärr har e-post aldrig fått en bra krypteringsmetod som blivit universell. För själva breven så finns PGP och S/MIME-kryptering till flera klienter, medan i stort sett alla e-postklienter idag klarar att köra själva uppkopplingen över SSL eller TLS, dvs. samma som för säker web.

Man måste emellertid konfigurera om klienten för att köra med SSL mot klubben, vilket dock starkt rekommenderas av föreningens sysop:ar.

Om ni loggar in på klubbens medlemssidor, finns ett par bra beskrivningar för hur man byter till SSL för ett par av de vanligaste klienterna, Outlook, Thunderbird och Eudora. Jag har själv använt Eudora- och Outlook-beskrivningarna för att ställa in SSL-trafik i min egen Pegasus Mail, de är relativt lika, så finns inte er klient, se om någon annan kan hjälpa.

Notera att man inte som vanligt hämtar posten på port 110 och sänder på port 25, utan man måste byta till att hämta på port 995 och sända på port 26 eller port 587.

Sitter man på ett annat nät, som Telia, Tele2, ComHem eller Bredbandsbolaget, så har de liknande möjligheter, men där måste man läsa deras respektive instruktioner.

Ett par leverantörer, främst Telia, stängde under april dessutom av möjligheten att sända ut e-brev med klubben som avsändaradress, via deras system, vilket ställde till problem för några medlemmar.

Om vi medlemmar vill, kan vi då använda klubben för att sända, även om vi sitter på något av dessa nät. Då måste man använda port 587 och användarautentisering mot klubbens brevservrar, enligt instruktionerna på medlemswebben. Självt har jag gjort så i 6-7 års tid, trots att jag kör via Telia.

Men om man vill kryptera breven?

Jo, det vore en bra idé. Självt kör jag Lotus Notes på jobbet och den är alltid krypterad, så jag kan utan vidare sända känsligt material inom jobbet. Då jag dagligen arbetar med IT-säkerhet, så hanterar jag ibland känsliga personuppgifter så det behövs. Men, det är på jobbet.

Min privata Pegasus-klient kan jag ställa in för att använda PGP, det finns färdiga pluginmoduler för det, men kryptonyckelutbytet som skall till, är fortfarande för krångligt. Samma gäller nog Outlook, Thunderbird och andra klienter.

Orsaken till att Notes är så enkelt, är att där är det en del av adressbokfunktionen. Ute på Internet är det manuellt nyckelutbyte som gäller för PGP.

S/MIME är något enklare, men många klienter, som trots allt har visst stöd för PGP, stödjer inte S/MIME och tvärt om. Outlook och Thunderbird tycks främst ha stöd för S/MIME, medan Pegasus och Eudora har bara enkla PGP plugin:er.

Och ingen av dem verkar enkla att använda, inte som med Notes. Jag måste medge att jag är förvånad över att ingen bra standardisering ännu har dykt upp kring nyckelutbyte för e-post, så det blev lika enkelt i dessa system, som i Notes.

Problemet är att verifiera att den som använder en nyckel är den han/hon säger att man är. I de flesta fall ligger detta på ISP:erna, men de verkar ointresserad att utveckla funktionerna, trots att samhället går mot att vi behöver alternativet.

Slutledning

Lämna aldrig ut lösenord.

Se i varje fall till att ändra er uppkoppling till klubbens e-postsystem, till att köra SSL. I dagens fientliga Internet är det absolut rekommendabelt att kryptera både POP/IMAP (hämtning) som SMTP (sändning).

Vad gäller kryptering av breven, vill ni pröva med någon, så kom först överens om metod, PGP eller S/MIME, och leta efter instruktioner för era respektive klienter och pröva. Men räkna inte med att allmänt kunna använda brevkryptering.

Jag kommer senare, i en andra artikel, att berätta lite vad man i övrigt kan göra med vår e-post.

ABC-bladet goes Open Source

Det här numret av ABC-bladet har tagit ett stort steg mot Open Source. Layouten är gjort i layout-programmet Scribus, bilder är redigerade med GIMP, och dessutom är brödtext-typsnittet Linux Libertine, ett fritt typsnitt.

Hade vi behövt rita vektorbilder kunde vi dessutom använt Inkscape.

Det intressanta med dessa program är att de inte bara är Open Source, de finns dessutom till både Linux, Windows och OS X. För Windows finns det dessutom PortableApps-versioner av alla tre.

Läs mer:

www.scribus.net

www.gimp.org

<http://linuxlibertine.sourceforge.net/>

www.inkscape.org

<http://portableapps.com>

LINQ - fortsättning

Anders Franzén <5258>

I min förra artikel (ABC-Bladet Nr. 3/4 2008) tittade vi lite grann på det nya frågespråket LINQ (Language Integrated Query) som introducerades i och med Microsoft släppte .NET Framework 3.5 i slutet av 2007.

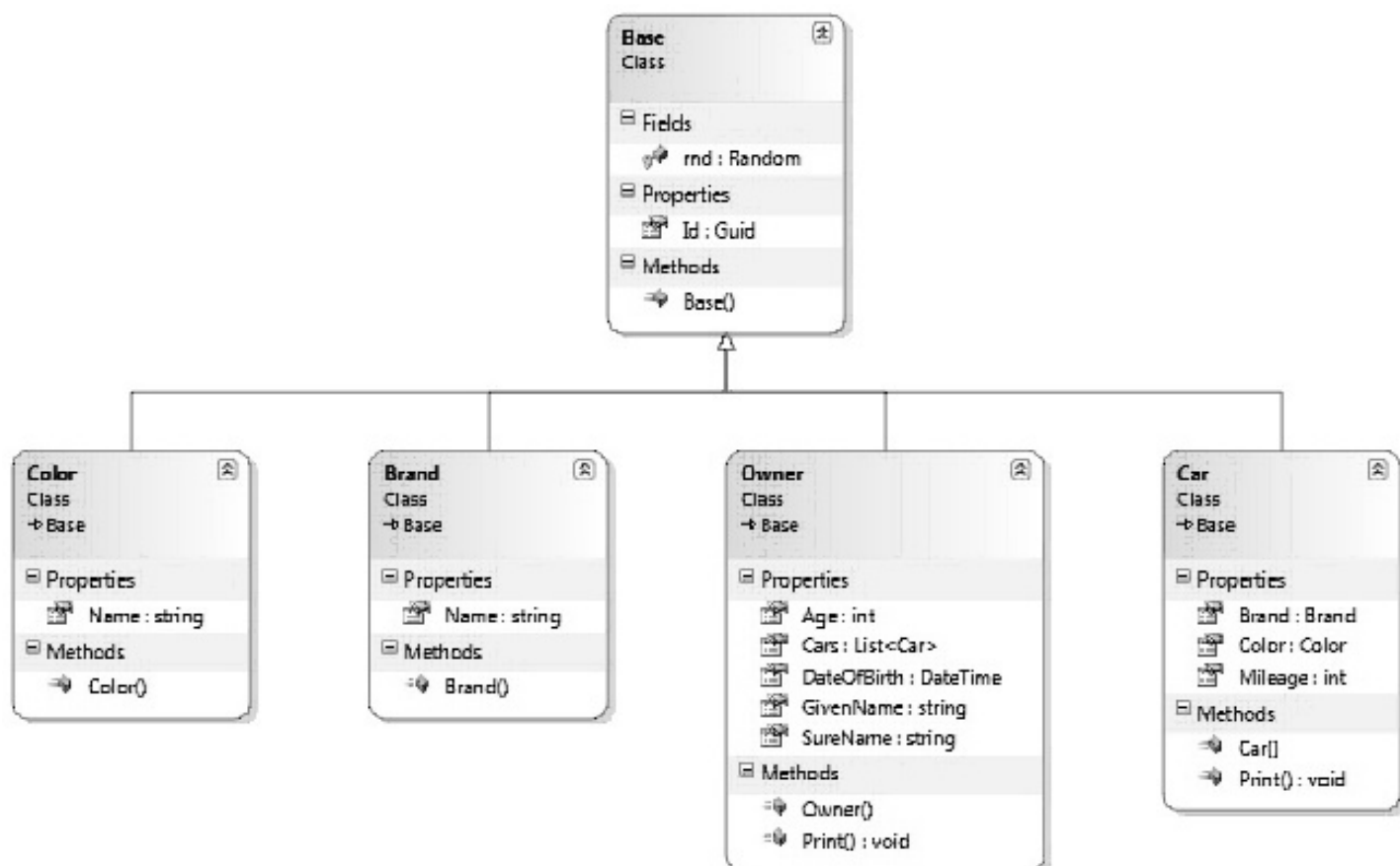
Denna gång ska vi fördjupa oss lite mer i frågespråket och se hur man kan ställa lite mer komplicerade frågor. Läs gärna den förra artikeln om LINQ om du inte redan gjort det innan du börjar läsa denna.

Även denna gång är testprogrammet ett vanligt konsolprogram, dvs det skriver ut resultatet i ett vanligt kommandopromptsfönster. Denna gång har vi dock lite fler objekt som ordnas till en struktur som vi sedan ställer en fråga mot. I vårt exempel kommer vi att ha en lista med bilägare som har en eller flera bilar, varje bil är av ett visst märke och har en viss färg. Vi har alltså totalt fyra objekt som vi kallar Owner, Car, Brand och Color. Jag har valt engelska namn eftersom man ofta programmerar på engelska. Det kan vara svårt ibland att översätta svenska begrepp och då kan det vara bättre att behålla svenska namn på saker och ting men här är det ju fråga om enkla saker. Eftersom programspråket C# är på engelska passar det helt enkelt bättre med engelska namn på variabler osv.

Klassdiagrammet

Med Visual Studio kan man skapa och editera ett klassdiagram som visar sambandet mellan klasserna. Man kan antingen börja med att skapa diagrammet varvid programkod automatskapas samtidigt eller så kan man skapa programkod varvid diagrammet automatiskt skapas. Diagram och programkod kommer alltså alltid att vara i fas med varandra när man senare ändrar i antingen kod eller diagram, det garanteras av utvecklingsverktyget. Tyvärr klarar inte Visual Studio att visa associationer, dvs relationer mellan två objekt. En Owner kan ha flera Car så vi borde ha något sorts streck mellan Owner och Car men det supportar alltså inte verktyget vilket egentligen gör klassdiagramsfunktionen tämligen värdelös.

Diagrammet för testprogrammet i denna artikel bifogas artikeln, titta gärna på det samtidigt som du läser vidare! Som synes i diagrammet har vi en basklass som heter Base och som samtliga objekt ärver från. Detta arv fyller egentligen ingen funktion för vårt test av LINQ men om man vill laborera vidare kan det vara bra att ha en basklass. Varje instans av bilar, personer, bilmodeller och färger har en unik identitet som inte heller används i detta test men vi lagrar den i basklassen om någon vill experimentera vidare. Där



Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace LinqTest2
{
    class Program
    {
        static void Main(string[] args)
        {
            Test test = new Test();
            test.Start();
        }
    }
}
```

Test.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using LinqTest2.Classes;
namespace LinqTest2
{
    public class Test
    {
        private List<Owner> Owners = new List<Owner>();
        private DateTime StartTime;

        public void Start()
        {
            CreateTestData();
            Test1();
            Console.WriteLine(new string('-', 40));
            Console.WriteLine("Done");
            Console.ReadKey();
        }

        private void Test1()
        {
            PrintHeader("30-åringar födda på en söndag med en  
vit Opel som kört minst 2000 mil");
            var query =
                from owner in Owners
                from car in owner.Cars
                where
                    owner.DateOfBirth.DayOfWeek ==
DayOfWeek.Sunday &&
                    owner.Age == 30 &&
                    car.Brand.Name == "Opel" &&
                    car.Color.Name == "vit" &&
                    car.Mileage >= 2000
                orderby
                    owner.GivenName,
                    owner.SureName
                select owner;
            PrintResult(query.ToList());
        }
    }
}
```

```
private void CreateTestData()
{
    for (int i = 0; i < 100000; i++)
        Owners.Add(new Owner());
}

private void PrintHeader(string header)
{
    Console.WriteLine(new string('-', 40));
    Console.WriteLine(header);
    StartTime = DateTime.Now;
}

private void PrintResult(List<Owner> result)
{
    TimeSpan timeSpan = DateTime.Now - StartTime;
    foreach(Owner owner in result.ToList().Take(10))
    {
        owner.Print();
        foreach(Car car in owner.Cars)
            car.Print();
    }
    Console.WriteLine("{0} rows selected, time {1}ms",
        result.Count.ToString(),
        Math.Truncate(timeSpan.TotalMilliseconds));
}
}
```

Owner.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace LinqTest2.Classes
{
    public class Owner : Base
    {
        public string GivenName { get; set; }
        public string SureName { get; set; }
        public DateTime DateOfBirth { get; set; }
        public List<Car> Cars { get; set; }
        public Owner()
        {
            string[] given = {"Adam", "Bertil", "Ceasar",
"David", "Erik", "Fredrik",
"Gustav", "Helge", "Ivar", "Johan",
"Kalle", "Ludvig" };
            string[] surePart1 = { "Sten", "Lund", "Berg",
"Gran", "Björk", "Sjö" };
            string[] surePart2 = { "bäck", "vall", "man",
"dal", "gren", "blad", "löv" };
            GivenName = given[rnd.Next(given.Length - 1)];
            SureName = surePart1[rnd.Next(surePart1.Length -
1)] +
                surePart2[rnd.Next(surePart2.Length - 1)];
            DateOfBirth = new
DateTime(DateTime.Today.AddYears(-18).Year, 1, 1) -
                new TimeSpan(rnd.Next(30000), 0, 0,
0);
        }
    }
}
```



```

        Cars = new List<Car>();
        int numberOfCars = rnd.Next(1, 6);
        for (int i = 0; i < numberOfCars; i++)
        {
            Cars.Add(new Car());
        }
    }
    public int Age
    {
        get
        {
            TimeSpan ts = DateTime.Today - DateOfBirth;
            return (int)(ts.TotalDays / 365.25);
        }
    }
    public void Print()
    {
        Console.WriteLine("{0, -18} {1} {2, 3}",
            GivenName + " " + SureName,
            DateOfBirth.ToShortDateString(),
            Age);
    }
}
}

```

Car.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace LinqTest2.Classes
{
    public class Car : Base
    {
        public Brand Brand { get; private set; }
        public Color Color { get; private set; }
        public int Mileage { get; set; }
        public Car()
        {
            Brand = new Brand();
            Color = new Color();
            Mileage = rnd.Next(300, 3000);
        }
        public void Print()
        {
            Console.WriteLine(" {0, -10} {1, -8} {2, 5}",
                Brand.Name, Color.Name, Mileage);
        }
    }
}

```

Brand.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace LinqTest2.Classes
{
    public class Brand : Base
    {
        public string Name { get; private set; }
        public Brand()
        {
            string[] names = {"Volvo", "Saab", "Ford", "Opel",
                "Mercedes", "Toyota", "Honda"};
            Name = names[rnd.Next(names.Length - 1)];
        }
    }
}

```

Color.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace LinqTest2.Classes
{
    public class Color : Base
    {
        public string Name { get; private set; }
        public Color()
        {
            string[] names = {"Gul", "Blå", "Svart", "Vit",
                "Gul", "Silver", "Grön"};
            Name = names[rnd.Next(names.Length - 1)];
        }
    }
}

```

Base.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace LinqTest2.Classes
{
    public class Base
    {
        public Guid Id { get; private set; }
        static protected Random rnd = new Random(314159);
        public Base()
        {
            Id = Guid.NewGuid();
        }
    }
}

```

finns dessutom slumpgeneratorn och den är definierad som static vilket betyder att alla instanser delar på en och samma slumpgenerator. I och med att den ligger i basklassen kan vi enkelt nå den från alla andra objekt.

Klassen Car beskriver en bil. Varje bil är av ett visst märke och har därför en variabel av typen Brand som vi kallar Brand (mer om denna namnstandard senare). Variabeln deklarerar med get- och set-operatorer, dvs den blir automatiskt ett property som kan användas utifrån. Notera att get är public medan set är private, dvs man kan inte ändra bilmärket på en bil utifrån. Klassen Brand i sin tur har ett namn som finns i variabeln Name och även denna är definierad med get och set där get är public och set är private. Om vi har en variabel av typen Car med namnet myCar kan vi med syntaxen myCar.Brand.Name läsa av bilmärkets namn men vi kan inte tilldela ett nytt värde; om vi försöker med myCar.Brand.Name="Lådbil" kommer vi att få ett syntaxfel när vi kompilerar programmet. En Car har dessutom en färg och den lagras som en variabel av typen Color och vi kallar variabeln Color.

Den uppmärksamma läsaren blir kanske lite förvillad av namnsättningen på klasser och variabler här, variabeln Brand i Car har samma namn som klassen Brand. Det kan vara intressant att veta att rent syntaxmässigt i C# är det tillåtet med variabler med samma namn som typen de är av. Naturligtvis kan man (kanske till och med borde?) hitta bättre namn så att man tydligt kan skilja på klass och variabel men varför har jag gjort så här? I språket C# finns en konvention som säger att man namnger klassvariabler, alltså sådana som deklarerar i klassen, med ett inledande underscore, i vårt fall skulle variablerna hetat _car respektive _color om vi inte använt get och set. Variabler som är lokala i en enskild metod kallar man car och color utan underscore. Properties brukar namnges med inledande versal, dvs Car och Color. I vårt fall använder vi en av nyheterna i senaste versionen av språket C#, att automatiskt deklarerar ett property samtidigt som variabeln deklarerar med hjälp av get och set. Hur ska man namnge en sådan variabel som samtidigt blir ett property? Jag har valt att följa namnstandarden för properties.

Slutligen har vi klassen Owner som beskriver en person. Varje Owner har ett förnamn, ett efternamn, ett födelsedatum och en lista med bilar som personen äger. Listan med bilar ligger i variabeln Cars som är av typen List<Car>. List-klassen är inbyggd i .NET och den är jättebra, man anger vad man vill lagra i listan inom hakarna och man får då en typsäker lista, dvs det går bara att lagra objekt av typen Car i den. Vi har även ett property som returnerar personens ålder men det är inget vi lagrar utan åldern beräknas dynamiskt utifrån dagens datum varje gång som vi frågar efter den.

Initiering

Låt oss nu studera programlogiken! Exekveringen startas i klassen Program som innehåller en Main-metod. Där instansierar vi klassen Test och anropar metoden Start som innehåller all programlogik som styr testet. Det första som sker i metoden Start är att vi initierar vårt testdata genom ett anrop till metoden CreateTestData. För vårt test slumpar vi fram hundratusen personer som var och en kan ha upp till sex bilar. Varje objekt är självt ansvarigt för att fylla sig med lämpligt slumpdata och detta sker i respektiva konstruktörer som exekveras när nya instanser av klasserna skapas.

Det enda som CreateTestData gör är alltså att i en loop som snurrar etthundratusen gånger skapa upp objekt av typen Owner som sparas i en lista som vi kallar Owners. Konstruktören för klassen Owner slumpar fram förnamn, efternamn och födelsedatum och skapar dessutom en Brand och en Color. Konstruktörerna i klasserna Brand och Color slumpar i sin tur fram ett namn för bilmärket respektive en färg. När detta är klart har vi i vår Test-klass en variabel Owners av typen List<Owner> som innehåller etthundratusen bilägare.

Notera gärna hur vi gör för att slumpa fram varje persons efternamn! Vi väljer från ett antal prefix och suffix och sätter ihop dem, t ex Stenberg, Stenbäck, Stenvall osv. Det gör testet måhända lite roligare. Födelsedatumet slumpas fram genom att utgå från dagens datum, dra bort 18 år (man kan ju ta körkort först när man är 18 år gammal) och därefter dra bort ett slumpmässigt antal dagar.

LINQ-frågan

När testdatat är initierat anropas metoden Test1 som utför själva testet. Vi har bara ett test i detta testprogram men den som vill kan själv addera Test2, Test3 osv. Vår testmetod börjar med att skriva ut en rubrik och startar stoppuret med ett anrop till PrintHeader . Därefter definieras en LINQ-fråga som exekreras när metoden PrintResult anropas. Notera att när frågan definieras så exekveras den inte, det sker när man använder resultatvariabeln och det sker i vårt fall med ett anrop till ToList som konverterar resultatet till en List-klass i samband med att PrintResult anropas. Själva frågan lagras alltså inte i resultatvariabeln utan sparas under skalet så att säga och exekveras först när man refererar variabeln varvid variabeln tilldelas resultatet av frågan. LINQ använder fördröjd exekvering.

Vad vi vill göra är att ta fram alla personer som är exakt 30 år, är födda på en söndag och som äger minst en vit Opel som kört minst 2000 mil. Vi skulle kunna loopa igenom alla personer, loopa igenom varje persons bilar och jämföra alla värden manuellt men det kräver ju att vi kodar en massa logik. Det är betydligt enklare att göra en LINQ-fråga!

Låt oss studera vår LINQ-fråga i detalj. Först anges varifrån vi vill hämta datat och det sker med en from-sats "from owner in Owners". Det som står efter "in" anger varifrån vi vill läsa, i vårt fall från variabeln Owners som är en variabel som innehåller listan med alla Owner. Precis före "in" anger vi ett alias eller en liten lokal variabel som vi använder i resten av frågan. Vi kallar den "owner" med liten bokstav för att skilja den från klassen Owner, C# skiljer ju på små och stora bokstäver.

Eftersom vi dessutom vill ställa villkor mot varje persons bilar som är en lista inuti varje Owner deklarerar vi ytterligare en from-sats "from car in owner.Cars" och det som står efter "in" är alltså varifrån vi vill läsa, denna gång använder vi det nyss definierade aliaset eller den lilla variabeln "owner" och eftersom den är av typen Owner kan vi nå dess lista med bilar som finns i variabeln Cars. Före "in" anger vi ett nytt alias eller en ny liten variabel som vi kallar "car".

Nu är det dags att ange sökvillkoren för vår fråga. Här kan vi använda de två små variablerna som vi nu



definierat, "owner" och "car". Först vill vi kontrollera att personen är ett söndagsbarn och det kan vi göra genom att använda variabeln DateOfBirth som finns i Owner. Denna variabel är av typen DateTime som är en typ som finns inbyggd i .NET och som har en massa bra metoder och properties, bl a DayOfWeek som tar fram veckodagen för ett visst datum. Vi jämför veckodagen med en konstant som också är inbyggd i .NET för att se att det är en söndag.

Därefter vill vi verifiera att personen är 30 år gammal, det kan vi göra genom att använda propertyt Age som finns på en Owner. Värdet för detta property lagras inte i en Owner utan beräknas varje gång det används. Sedan kollar vi bilmärket, bilens färg och hur långt bilen gått med hjälp av den lilla variabeln "car". Notera att "owner" och "car" som vi använder i LINQ-frågan inte är variabler i vanlig mening men man kan låtsas att de är det för att underlätta förståelsen av frågesyntaxen.

Vi sorterar frågeresultatet på förnamn och sedan efternamn och allra sist anger vi vad vi vill att frågan ska returnera, i vårt fall en "owner" för varje träff och det är ju en instans av en Owner. Således kommer frågan att returnera en lista med en massa Owner, en LINQ-fråga returnerar alltid en lista.

Slutligen skrivs resultatet ut, vi skriver ut varje hittad person och för varje person dess bilar. Utskriften sker i Print-metoder i klasserna Owner och Car. Notera hur vi formaterar utskriften, vi paddar på blanka på för- och efternamn så att vi ska få fina kolumner, -18 betyder att vi vill padda på med mellanslag till höger så att första parametern (förnamn+efternamn) fylls ut till 18 tecken. Den tredje parametern (ålder) fylls ut med tre inledande mellanslag till vänster. I formateringssträngen anger man parametrar inom

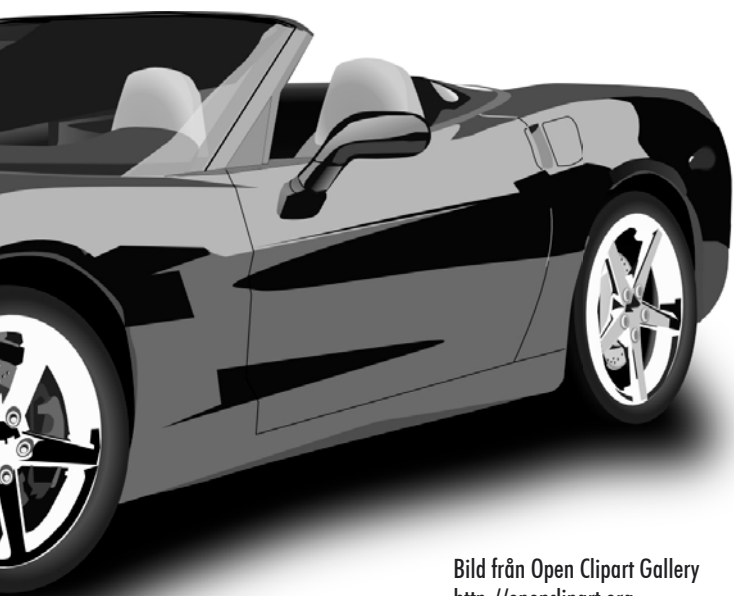


Bild från Open Clipart Gallery
<http://openclipart.org>

klamrar och de numreras från noll och uppåt. Eventuell redigering som utfyllnad med mellanslag anges efter kommatecknet inom klammrarna där tecknet anger vilken sida som ska paddas, plus för höger och minus för vänster.

Resultatet från min testkörning bifogas artikeln, se konsolfönstret. Som synes har personerna som listas flera bilar, vi skriver ju ut alla bilarna för de hittade personerna som uppfyller villkoret. Minst en av varje persons bilar är alltså en vit Opel. Den sista personen har till och med två vita Opel men endast en av dem har gått minst 2000 mil och det är tack vare denna som personen kommer med.

Slutsats

I detta exempel har vi enbart jobbat med data som ligger i internminnet. Detta förutsätter att vi själva skapat datat i programkod men man kan lika gärna läsa in datat från en databas eller liknande. Vi kan även använda LINQ to SQL för att ställa LINQ-frågor direkt mot relationsdatabaser utan att behöva läsa in datat till minnet. Då kommer vår LINQ-fråga att under skalet översättas till en eller flera SQL-frågor som skickas till databasen. Man kan även använda LINQ för att ställa frågor mot XML-dokument.

När man som i vårt exempel har en datastruktur med sammansatta objekt ser man verkligen nyttan med LINQ. Det går betydligt fortare (när man väl kan syntaxen!) att skriva en LINQ-fråga än att skriva programkod som ska loopa igenom datat för att ta fram samma sak. Det är enklare att underhålla en sådan fråga om man vill ändra på något jämfört med att gå in och rota i en stökig programlogik med en massa for-loopar. Vill man dessutom ställa flera olika frågor mot samma datamängd inser man att här kan man spara tid och tid är som bekant pengar. Jag är därför övertygad om att LINQ kommer att användas mer och mer ju fler som upptäcker nyttan.

```
file:///C:/Kod/LinqTest/LinqTest2/bin/Release/LinqTest2.EXE
-----
30-åringar födda på en söndag med en vit Opel som kört minst 2000 mil
Adam Björkgren      1978-03-12  30
  Volvo      Silver      427
  Mercedes   Vit         939
  Opel       Vit         2306
  Mercedes   Gul         2679
Gustav Stenvall     1978-02-05  30
  Mercedes   Blå         1372
  Saab       Blå         1725
  Opel       Vit         2859
Kalle Grangren      1978-12-10  30
  Opel       Vit         1208
  Mercedes   Vit         399
  Toyota     Blå         1427
  Opel       Vit         2423
  Toyota     Gul         448
3 rows selected, time 125ms
-----
Done
```


Google Android - ett öppet operativsystem för mobiltelefoner

Tomas Gustavsson <7305>

I oktober presenterade Google tillsammans med HTC och mobiloperatören T-mobile i USA slutligen Androidplattformen. Android hade då varit omtalat en längre tid och efterlängtat i utvecklarkretsar. Mottagandet har hittills varit överväldigande med stor uppslutning både bland open source-utvecklare och kommersiella aktörer.

Vad är då Android?

Kort sagt är det en mjukvaruplattform för avancerade mobiltelefoner, så kallade smart phones. Android bygger på pekskärmstekniken och fungerar väldigt likt Apples iPhone, dvs man använder fingrarna för att manövrera i operativet istället för enbart tangenter eller en skrivpenna, vilket tidigare varit det vanligaste i Symbian eller Windows Mobile.

Android är helt öppet och källkoden kan laddas ner och studeras av vem som helst. Applikationer för Android skrivs i Java och det finns ett mycket bra utvecklingskit, att ladda ner. Något som skiljer Android från de flesta andra operativsystem är att alla applikationer är på samma nivå. Om du inte tycker om den telefonbok som kommer med din telefon kan du göra en egen och använda istället.



Foto: Frederik Hermann (netzkobold on flickr.com)
Used under Attribution-Noncommercial-Share Alike 2.0 Generic

Vilka ligger bakom Android?

Bakom Android ligger först och främst Google, som gjort det mesta av utvecklingen hittills. De har dock bildat ett konsortium, Open Handset Alliance, för att säkerställa plattformens framtid och att många telefoner kommer att finnas tillgängliga. De flesta större telefontillverkare är medlemmar i detta konsortium.

Vilka applikationer finns?

Trots att Androidtelefoner bara funnits i några månader finns redan hundratals applikationer att ladda ner. Allt från spel och Facebookhjälpmedel till kompasser och program som utnyttjar GPS för att hålla reda på dina joggingpass. Den bästa ssh-klienten för mobiltelefoner finns till Android vilket gör det lätt att logga in på ABC-klubbens servrar. Webläsaren i Android är mycket kapabel och stöder både Javascript och flera fönster. Detta gör att man kan surfa som vanligt på de flesta websajter. De flesta sajter utom de som är byggda i Flash fungerar som man förväntar sig. Stöd för Flash är dessutom på väg.

Att använda Android

Att använda Android på mobiltelefonen är mycket smidigt. Det finns en desktop som är bredare än skärmen och fyller tre skärmar. Man kan panorera mellan de tre delarna av desktopen med ett enkelt sidodrag med fingret. På desktopen kan man lägga genvägar till favoritprogrammen. För att ringa trycker man bara på ring-knappen så får man upp en applikation med olika flikar för manuell nummerslagning, återuppringning eller telefonboken. SMS-applikationen är verkligen trevlig att använda. Meddelanden presenteras i trådar för de olika personer man SMS:ar med, och det liknar mer ett chatprogram än traditionella SMS. För att svara på SMS skriver man sitt meddelande och trycker Enter, inget krångel. I övrigt startar man applikationer genom att trycka på applikationsikonen med fingret. Det fungerar på samma sätt oavsett om det är förinstallerade applikationer eller om det är program som du installerar själv. För att installera nya program kan man använda Android market, en marknadsplats för applikationer, eller klicka på en länk till ett program på en websida. Android är väldigt lättanvänt och fungerar snabbt och smidigt utan direkta väntetider, dessutom är det snyggt.

Att utveckla med android.

Att utveckla applikationer med Android är enkelt. Plattformen är öppen, väldokumenterad och kommer med massor av exempel och tutorials. Utvecklingsspråket är Java vilket är på hög nivå och välkänt av många utvecklare. Utvecklingsmiljön är Eclipse, vilket alla javautvecklare känner till och dessutom finns det en utmärkt emulator i utvecklingsmiljön så man kan utveckla och testa sin applikation utan tillgång till en telefon. Plattformen säkerställer sedan att din applikation skall kunna installeras och köras på alla Androidtelefoner.

I äldre system som Symbian är det krångligt att utveckla program, och programmen utvecklas i princip för en viss telefon så de är ej kompatibla mellan t ex Sony-Ericsson och Nokia. Utvecklingsmiljön är heller inte fri vilket gör tröskeln att börja utveckla hög. Tröskeln för att börja utveckla i Android är i princip så låg den kan bli idag.

Vilka telefoner finns?

I skrivande stund finns endast en telefon på marknaden, G1 från HTC, en välkänd taiwanesisk telefontillverkare. Telefonen har en 3.2 tums pekskärm och ett riktigt tangentbord som fälls fram genom att skjuta upp skärmen. Om ett riktigt tangentbord är bra eller dåligt, jämfört med ett virtuellt på skärmen, är en smaksak. Personligen tycker jag om det då det går snabbare att skriva på, och man samtidigt har tillgång till hela skärmen. Android på G1 fungerar ypperligt och det enda jag egenligen kan klaga på är den begränsade batteritiden. I övrigt är telefonen utrustad med allt jag kan önska mig av en modern telefon. 3G och GPRS, GPS, 3.2 megapixel kamera med autofokus, kompass samt qwerty-tangentbord.

Telefoner utan tangentbord, mer iPhone-lik kommer dyka upp under 2009, och Android borde då passa de flesta smaker. Flera olika telefontillverkare har annonserat att de kommer släppa Androidtelefoner under 2009.



This image is reproduced from work created and shared by Google and used according to terms described in the Creative Commons 3.0 Attribution License

Någon Androidtelefon går idag ej att köpa i Sverige utan den enda möjligheten, vilken jag har utnyttjat, är att beställa en utvecklartelefon från Google. Man måste då registrera sig som Androidutvecklare hos Google för 25 dollar och kan då beställa en upplåst telefon. Något krångligt men leveransen var snabb med UPS och det blev inte dyrare än att köpa senaste Windowstelefonen från Sony-Ericsson utan abonnemang.

Är detta framtiden?

Efter att ha använt olika telefoner under mina år är det fullkomligt klart att iPhone och Android visar framtiden för mobiltelefoner. Dessa gränssnitt känns väldigt moderna och i jämförelse känns de vanliga gränssnitten väldigt primitiva och det märks att de är gamla. De tillverkare som inte hakar på utvecklingen kommer att få det svårt. Både Motorola, Samsung, Sony-Ericsson och OpenMoko har dock redan annonserat att de utvecklar för Android så 2009 blir ser ut att kunna bli ett spännande år på mobiltelefonfronten.

Läs mer

Android <http://code.google.com/android/>

Open Handset Alliance <http://www.openhandsetalliance.com/>

Avsändare
ABC-klubben
Box 14143
167 14 Bromma

Glöm inte meddela kansliet din nya adress om du flyttar, kansli@abc.se.
Postens tjänst Begränsad eftersändning har upphört.

Foton från klubbens grillfest

Foto: Kjell Järbin <700>



Kalendariet

September	8,15,23,29	Medlemskväll Alvik
Oktober	6,13,20,27	Medlemskväll Alvik
	15	Manusstopp ABC-Bladet, nr 3-4, 2009
November	3,10,17,24	Medlemskväll Alvik
December	1,8,15,22	Medlemskväll Alvik
	27	Manusstopp ABC-Bladet, nr 1, 2010

(Information om planlagda medlemsaktiviteter kan meddelas redaktionen)