



Konferenca

SharePoint dnevi 2015

17 - 18 November, Terme Zreče

**25% popust na zgodnje prijave le do 14.9.!**

## Microsoft Partner

Gold Collaboration and Content

Gold Devices and Deployment

Silver Intelligent Systems

Silver Application Development

Silver Datacenter

Silver Learning

Silver Midmarket Solution Provider



Leading Learning Partners Association

ISO 9001

BUREAU VERITAS  
Certification



Kompas Xnet d.o.o.

Stegne 7

1000 Ljubljana

01 5136 990

info@kompas-xnet.si

www.kompas-xnet.si



Konferenca

SharePoint dnevi 2015

17 - 18 November, Terme Zreče

## 6. konferenca SharePoint dnevi

*in Exchange*

- 2 dni
- 6 tematskih sklopov
- 25+ predavateljev
- 50+ predavanj
- druženje in zabava



**Vljudno vabljeni,  
da se nam pridružite!**

**Pohitite, prijave so možne najkasneje do 5. novembra!**

[www.sharepoint-dnevi.si](http://www.sharepoint-dnevi.si)



Dragi prijatelji,

Letošnje poletje se je začelo, kot si le lahko želimo in to v letu, ko praznujemo dvojni jubilej: nadvse uspešno se je zaključil projekt v Bakuju; zaključna slovesnost iger je bila res nepozaben dogodek in poslovilna večerja pri naših naročnikih višek gostoljubja.

V Bakuju se odpirajo še nove priložnosti in verjetno bomo zelo kmalu spet na poti v Azerbajdžan.

Zaključili smo šolanje po pogodbi z Zavodom za zaposlovanje za večje število brezposelnih; odlične ocene si je Rok prislužil z Azure delavnico na Malti, ...

Počitnicam navkljub pa nismo počivali, bilo je razgibano poletje in veliko projektov v teku:

- Infrastrukturalna projekta:
  - nadgradnja okolja, virtualizacija, postavitve Exchange 2013 in selitev poštnih predalov
  - implementacija Skype For Business
- Razvoj aplikacij
  - Pomoč in svetovanje interni ekipi podjetja
  - Vzdrževanje, dodatne funkcionalnosti za več strank
  - Prenova CMS-ja
  - Spletna trgovina
- SharePoint projekti:
  - Namestitev in konfiguracija SP 2013, branding, implementacija KPI, projektno vodenje
  - Namestitev in konfiguracija SP 2013 (test & prod), branding, migracija vsebin iz SP 2007, novice, odsotnosti, telefonski imenik, ...
  - Pilot: Namestitev in konfiguracija SP 2013, branding, obvestila, dokumenti, baza znanja, oglasna deska, predlogi izboljšav, vprašanja zaposlenih, ankete, galerija, ...

- SharePoint dnevi 2015
- Šolanja
  - Julija smo imeli kar dobro zasedenost učilnic, avgusta nekoliko manj, za september pa se že obeta pravi bum. Učilnice so praktično zasedene ves mesec in le tu in tam, na kakšnem tečaju, so še prosta mesta. Če se po jutru dan pozna :) odlična napoved! So morda vendarle razmere postale bolj naklonjene znanju?

### Xnet – 20 let in PiKA 18 let

Kot ste že lahko prebrali v prejšnji številki Pike, je letošnje leto za nas jubilejno, saj praznujemo 20 let od takrat, ko smo stopili na samostojno pot. Hkrati pa naša PiKA obeležuje svojo polnoletnost. Res je, ta drobna knjižica izhaja že 18 let in vi jo radi prebirate, kar nas zelo veseli in nam, od številke do številke, daje spodbudo, da pripravimo zanimive članke in tako vedno znova opravičimo vaše zaupanje.

V preteklih 20 letih smo preživeli že več turbulentnih obdobij, ko so bile investije v IT na zelo nizkem nivoju ali pa jih praktično ni bilo. Kljub temu smo v največji možni meri vlagali v razvoj svojih kadrov, jih vsako leto napotili na usposabljanja v tujino in zgradili odlično ekipo izvrstnih specialistov.

Tako danes lahko ponosno ugotavljamo, da v nas prepoznate zaupanja vrednega partnerja, kjer vedno najdete pravega sogovornika, učinkovito podporo in popoln servis.

Vaše zaupanje nas dodatno zavezuje in nam nalaga posebno odgovornost, da storitve še izboljšujemo in svoje znanje ves čas izpopolnjujemo.

Kljub številnim težavam, ki se ves čas postavljajo na pot, ostajamo zvesti svojim načelom in vrednotam, saj trdno verjamemo, da je prav korekten in iskren partnerski odnos zagotovilo za uspešno sodelovanje na dolgi rok.

V Xnet-u imamo zagotovo **»jagodni izbor« najboljših specialistov**: Rok Bermež, Robi Vončina, Miha Pihler, Toni Uranjek, Dejan Sarka, David

Čamdžić, Gašper Kamenšek, če izpostavim samo nekatere. To pa je najboljše zagotovilo za vaše uspešne in pravočasno izvedene projekte.



### SharePoint dnevi 2015 in Exchange

Vabim vas, da zkoristite kar 25 % nižjo ceno kotizacije in si že v času zgodnjih prijav (do 14. septembra), zagotovite svoje mesto na konferenci. Letos pričakujemo večji obisk, saj se je konferenca širše uveljavila kot odličien dogodek na visoki strokovni ravni, z množico eminentnih predavateljev in kar **6 sklopov** zanimivih predavanj.

Večino izmed vas zagotovo zanima, kaj prinaša SharePoint 2016, je smiselno počakati novo različico ali implemetirati še SP 2013. Vse ključne novosti, prednosti in slabosti, boste spoznali prav na konferenci in to brez dlake na jeziku. Ponovno bomo predstavili nekaj zanimivih SharePoint projektov in izkušenj, ki izhajajo iz njih.

Novost, ki smo jo uvedli lani, ostaja v programu, saj ga še dodatno obogati, to je poseben sklop predavanj na temo Exchange in Office 365.

Prijavite se še danes, letos se bomo družili **17. in 18. novembra v Termah Zreče**. Bodite z nami, ne bo vam žal!

### Več znaš, več veljaš

V informatiki lahko ta rek še malo preoblikujemo, saj mora vsak dan več znati, da lahko sploh delaš. IT tehnologija se pač izjemno hitro razvija in kar težko je slediti vsem novostim.

Vzemite si čas in si pridobite nova znanja, da boste lažje in bolj učinkovito opravljali svoje delo. Z nasveti, tiste male skrivnosti velikih mojstrov, ki vam jih bodo na tečaju zaupali naši izkušeni predavatelji, se boste izognili prenekateri prepreki, ki bi se vam utegnila postaviti na pot in otežiti vaš projekt.

Tako se vam bo čas, ki ga boste namenili šolanju, hitro povrnil.

Pa tudi finančno breme ni tako nedosegljivo, če si le izberete kvaliteten šolski center in odličnega predavatelja, ki bo z vami delil še kaj več, kot le vsebino, ki je napisana v knjigi.



Bodite zahtevni, ko gre za vaš čas in vaše znanje – učite se od najboljših, pridite na Xnet, saj izobražujemo že dobrih 25 let. Zaradi angažiranosti in kvalitete nas je Microsoft tudi predlagal, da je prav Xnet predstavnik izobraževalnih centrov iz Slovenije v mednarodnem združenju **LLPA – Leading Learning Partner Association**.

In ne nazadnje, kotizacijo za večino tečajev lahko poravnate z SA voucherji. Poskrbite, da vam ne propadejo.

Hvala vsem, ki nam zaupate in sodelujete z nami. Priporočamo se vam za sodelovanje tudi v prihodnje!

**Branka Slinkar, direktorica**  
[branka.slinkar@kompas-xnet.si](mailto:branka.slinkar@kompas-xnet.si)

## Naše kompetence

### Microsoft Partner

Gold Collaboration and Content

Gold Devices and Deployment

Silver Intelligent Systems

Silver Application Development

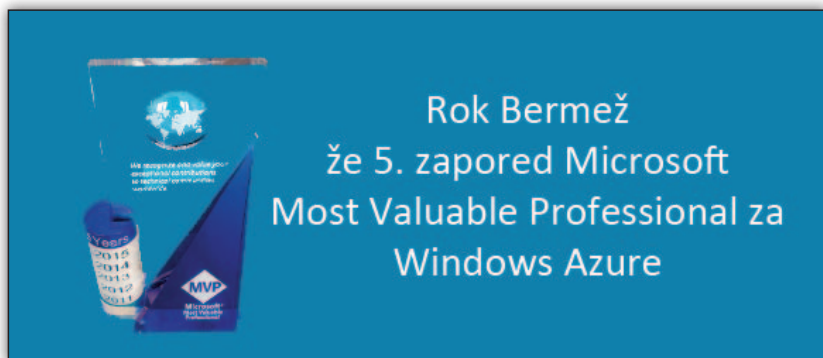
Silver Datacenter

Silver Learning

Silver Midmarket Solution Provider

# Kazalo

Ustvarite funkcijo za uvoz podatkov v Power Query Dodatku .....	6
Delo s tabelami .....	9
Kako učinkovito upravljati vire v večprojektnem okolju s pomočjo Teorije omejitev in Microsoft Project - 2. del .....	10
Filtriranje podatkov s filter gradnikom .....	13
SharePoint 2013 Search troubleshooting .....	15
Uporabniški vmesniki hibridnih mobilnih aplikacij .....	18
Singleton vzorec v C# programskem jeziku .....	19
Izboljšave orodij za ASP.Net .....	21
Powershell Direct .....	22
T-SQL for Statistics - Higher population moments (2) .....	23
PASS SQL Saturday Slovenija 2015 .....	26
Kolofon .....	27





# Ustvarite funkcijo za uvoz podatkov v Power Query Dodatku

Kot že naslov namiguje, bomo spoznali eno izmed precej naprednih tehnik za

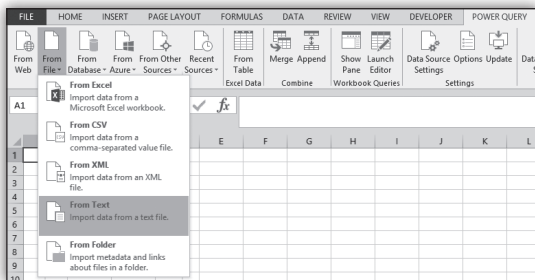
uvažanje podatkov v Excel. Kot zagotovo že veste (če pa še ne, pa »znova« preberite prvo številko Pike v letu 2015), je Power Query kot dodatek na voljo v verzijah Excela 2010 ter 2013, v verziji 2016 pa bo že kar del Excela in s tem bo uvoz podatkov v Excelu dvignil na popolnoma nov nivo.

Preden preidemo na temo, ki je omenjena v naslovu, pa le še opišimo situacijo, ki pripelje do potrebe po svojih funkcijah za uvoz v Excelu. Na tem mestu, bi še enkrat predlagal, da znova preberete članek iz pike 2015/1. Namreč ko enkrat vidite Power Query in razumete njegovo delovanje, ugotovite, da medtem, ko je na začetku morda z samim uvozom nekaj dela, se vsoto delo mnogokratno povrne v obliki ponovne uporabe naših »uvoznih postopkov«. Vendar pa se tu lahko tudi zaplete. Predstavljajte si situacijo, kjer bi radi poleg vsebine datoteke uvozili še imena datoteke, ali pa kar primer na katerem bomo mi delali, predstavljajte si, da te datoteke niso na isti lokaciji. Bodisi zaradi reda ali pa avtomatizacije procesov, so sicer po obliki identične datoteke razpršene na več lokacijah. Zelo

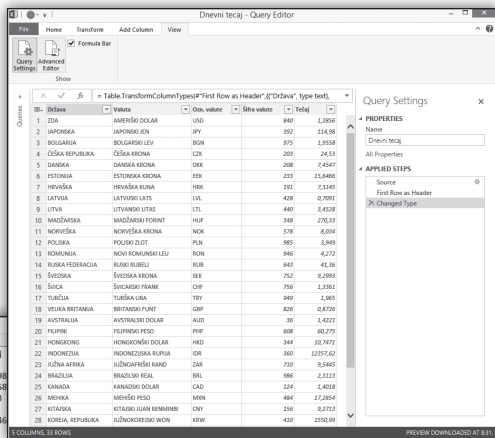
redundantno se zdi ustvarjati za vsako datoteko posebej celoten proces uvoza. In takrat ustvarimo svojo funkcijo. Funkcijo pa ustvarimo na primeru že narejena uvoza.

Torej dve datoteki, na različnih lokacijah.

Torej najprej na podlagi ene izmed teh naredimo uvoz preko **POWER QUERY/From File/From Text** in potem poiščemo eno izmed datotek...



Pridemo v naše najljubše okno v Excelu :), nastavimo celoten uvoz, tipe podatkov, uredimo podatke in jih preoblikujemo v željeno obliko, nato pa izberemo **View/Advanced Editor...**



In pojavi se slavni in zagotovo vaš najljubši jezik (M).

In že če pogledamo drugo vrstico kode

Source = Csv.Document(File.Contents("C:...

Vidimo, da je v kodi pot do te datoteke, ki bi jo lahko teoretično spreminjali vsakič na roko in večkrat uporabili to kodo. Mnogo bolj elegantna pot pa je ustvarjanje funkcije, ki vzame lokacijo in ime datoteke kot parameter, in potem ponovi vse postopke, ki jih je pri prvem uvozu.

V kodi bomo naredili dve spremembi. V prvo vrstico bomo dodali

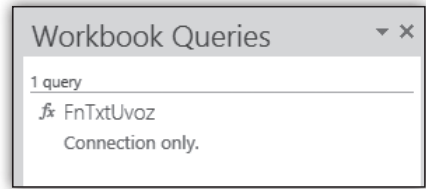
(filepath)=>

V oklepajih poimenujete parameter, ki ga kasneje uporabite v svoji funkciji, lahko bi bili tudi prazni a tako kot v Excelu, funkcija mora imeti oklepaje. Oklepajem in parametrom pa mora slediti enačaj ter znak večje.

Zatem pa bomo pot, ki je kazala na specifično datoteko spremenili v naš parameter, zato, da

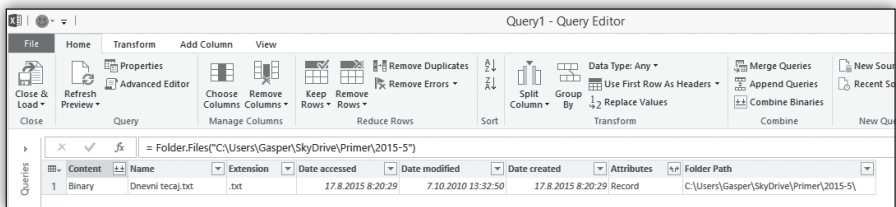
pa je, da damo funkciji smiselno ime v Query Settings podoknu.

In imamo našo funkcijo.

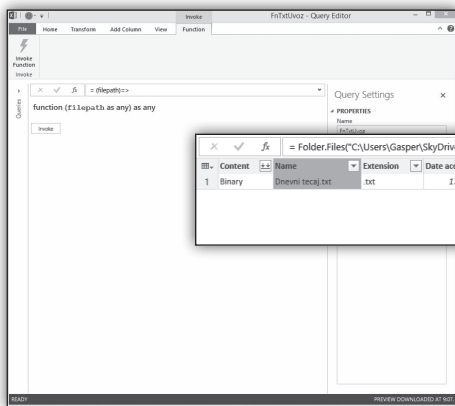


Sedaj pa jo uporabimo na drugi datoteki. Najprej izberemo **Power Query/From File/From Folder** in izberemo lokacijo datoteke.

Sedaj v Query Editorju dobimo seznam vseh datotek v mapi. Ta seznam po potrebi s pomočjo filtrov omejimo le na željene datoteke (lahko jih je tudi več!).

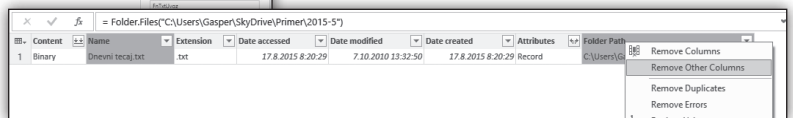


bomo lahko funkcijo ponovno uporabili.



Po pritisku na gumb Done, bi se moral izgled okna malo spremeniti in odraziti spremembe, da smo zares naredili funkcijo. Zelo pomembno

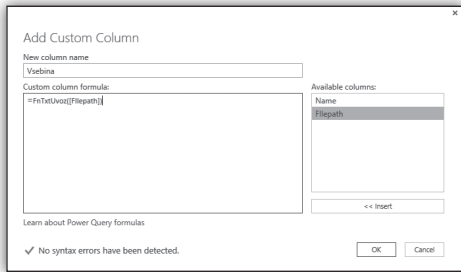
Sedaj odstranimo vse stolpce razen **Name in Folder Path**. To naredimo tako, da jih označimo in iz seznama ukazov, ki ga dobimo z desnim klikom izberemo **Remove Other Columns**.



Sedaj izberemo **Add Column/Add Custom Column** ter s pomočjo operatorja & združimo pot ter ime datoteke. Ime stolpca ni pomembno ampak zaradi jasnosti sem dal ime filepath.

Sedaj lahko stolpec **Folder Path** odstranimo.

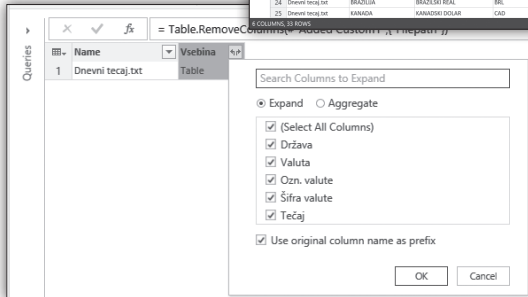
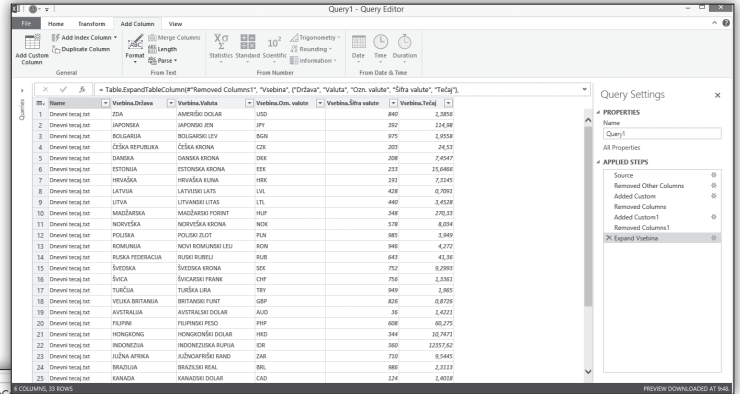
Sedaj pa pride čarovnija, zopet dodamo nov stolpec, definiramo ga pa kot



stolpca našega uvoza. Pozor na kljuciko spodaj! Omogoča nam, da bi kot predpono imenom stolpca dodali ime našega stolpca po meri. Torej namesto Valuta bi pisalo Vsebina.Valuta. To zelo odsvetujem, je pa res, da če bi to želeli potem ime stolpca je pomembno!

Končni rezultat in večna sreča, saj smo ohranili tudi izvor podatkov (ime datoteke). :)

Tukaj je pomembno, da vemo kakšno ime smo dali naši funkciji. Ime stolpca je popolnoma nepomembno. Ko pritisnemo OK, dobimo rezultat v obliki Tabele. Glavni čudež pa se skriva za ikono ob imenu stolpca, ki namiguje na tabelo in pojavijo se vsi



**Gašper Kamenšek**  
 MCP, MOS, MOS Master, MOS Expert,  
 MCTS, Adobe Certified Expert - Photoshop,  
 Predavatelj  
[gasper.kamensek@kompas-xnet.si](mailto:gasper.kamensek@kompas-xnet.si)

## Aktualni tečaji



**PPIVOT: PowerPivot: 7.9.2015 (izvedba zagotovljena!)**

**Excel 2010 nadaljevalni: 14. – 16.9.2015 (izvedba zagotovljena!)**

**Poslovna inteligenca za uporabnike Excela 2013, SELF SERVICE BI: 14. – 17.12.2015**



## Delo s tabelami



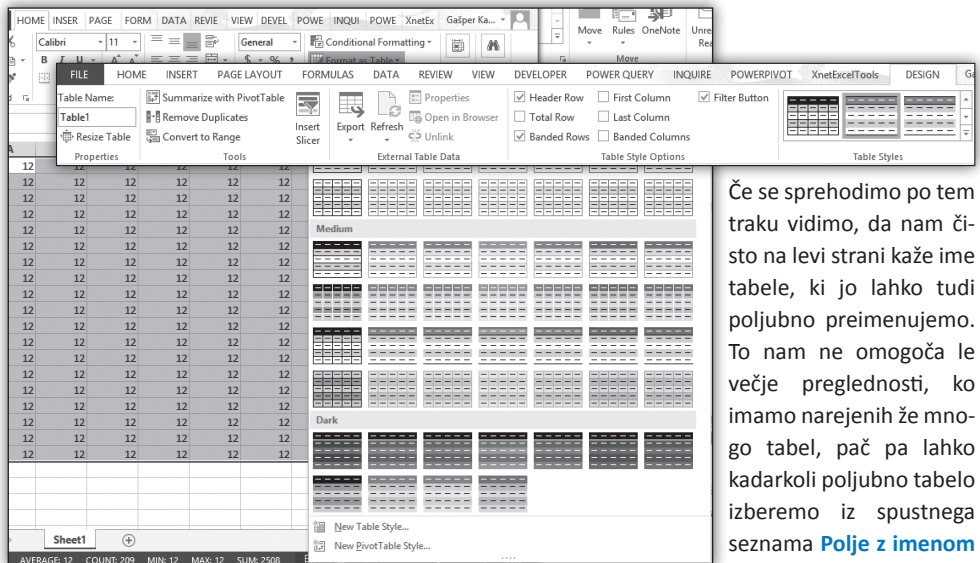
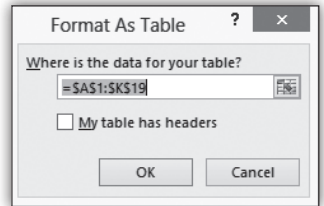
V Excelu 2007 in vseh verzijah od takrat naprej, je bila vključena povsem nova funkcionalnost, namenjena olajšanju dela z velikimi količinami podatkov. Govorimo seveda o tabelah. Do verzije 2007 smo, če odštejemo poskus z seznamami (katerih naslednice so tabele), lahko podatke le vnašali v posamezne celice in jih nato oblikovali, z njimi računali, ... Tabele pa nam omogočajo povsem nov, dinamičen način dela. Da lahko začnemo z delom s tabelami, moramo imeti nek obseg podatkov. Priporočljivo je, da glavo vsakega stolpca v obsegu, torej vsako celico neposredno nad našim obsegom, poimenujemo. Najbolje bo, če že ime stolpca nakazuje na vrsto podatkov, ki jih stolpec vsebuje, na primer: Ime, Priimek, Plača, ... Ključno pa je, da stolpec v tabeli vsebuje uniformirane podatke, torej podatke istega tipa.

Ko si obseg podatkov pripravimo, je naslednji korak, da tabelo ustvarimo. Da to naredimo, se postavimo na katerokoli celico v obsegu, gremo na trak **Vstavljanje (Insert)** in kliknemo na gumb **Oblikuj kot Tabela (Format as Table)**.

Postopek lahko tudi skrajšamo z bližnjico **Ctrl + T**. Ko pritisnemo na gumb, ali uporabimo bližnjico, nam bo Excel avtomatično izbral obseg podatkov, ki jih bo vključil v tabelo, vpraša pa nas, če naša tabela vsebuje glave (naslovno vrstico).

Če smo glave naredili, to možnost obkljukamo, sicer pa nam bo Excel glave naredil sam, a jim bo dodelil generična imena (Stolpec 1, Stolpec 2, ...). Ta imena lahko kadarkoli tudi spremenimo. Ko smo končali, le še pritisnemo **OK**.

Na prvi pogled dobimo le bolj pregleden obseg, ki se od prejšnjega razlikuje le po barvitosti in po glavah stolpcev. A že takoj opazimo, da imajo glave stolpcev spustni seznam, ki nam omogoča filtriranje in razvrščanje podatkov iz posameznega stolpca. Ena od opaznejših sprememb pa je tudi, da smo dobili tudi povsem nov trak **Orodja za tabele (Table Tools) oz. DESIGN** v verziji 2013.



Če se sprehodimo po tem traku vidimo, da nam čisto na levi strani kaže ime tabele, ki jo lahko tudi poljubno preimenujemo. To nam ne omogoča le večje preglednosti, ko imamo narejenih že mnogo tabel, pač pa lahko kadarkoli poljubno tabelo izberemo iz spustnega seznama **Polje z imenom**

**(Name Box)**. Na ta način si prihranimo ogromno časa, ki bi ga sicer porabili za iskanje tabele po delovnem zvezku.

Funktionalnosti na tem traku je še mnogo, na primer odstranjevanje duplikatov po poljubnih stolpcih, a ena od najbolj uporabnih je verjetno **Vrstica z rezultati (Total Row)**. Ko obkljukamo to možnost, se nam na dnu tabele pojavi nova vrstica. Vsaka celica te vrstice ima na voljo spustni seznam, ki nam ponuja nekatere najpogostejše uporabljane funkcije, kot so **Vsota (Sum)**, **Min/Max**, **Štetje (Count)**, izberemo pa lahko tudi možnost **Več funkcij (More Functions)** in s seznama poiščemo poljubno funkcijo. Vsaka funkcija naredi izračun le za svoj stolpec.

Možnosti na traku je še kar nekaj, a raje bi namenil še nekaj besed načinu delovanja tabele. Če se postavimo v katerokoli celico ob desnem ali spodnjem robu tabele in začnemo vpisovati nove podatke, se bo tabela razširila in vključila te nove podatke. Če podatke vpisujemo pod že obstoječi stolpec, bo tabela začela vključevati nove vrstice, če pa vpisujemo v sosednje stolpce, bo tabela naredila povsem nov stolpec, ki pa ga bomo verjetno želeli preimenovali.

Tabele nam pridejo zelo prav tudi pri vpisovanju formul. Če želimo na primer v novem stolpcu prikazati vsoto prejšnjih dveh stolpcev, moramo

formulo vpisati samo v prvo celico novega stolpca, vse ostale celice bo Excel zapolnil sam.

Funkcij je še mnogo, kot prikaz moči Excelovih tabel pa naj omenimo še funkcijo **Bliskovita zapolnitev (Flash Fill)**, ki je sicer novost v Excelu **2013**. Denimo, da imamo seznam ljudi, ki so zaposleni v naši firmi. V seznamu imamo posebej stolpec za njihova imena in posebej za priimke, sedaj pa bi želeli, da bi bila ime in priimek združena v eni celici. S tabelami nič lažjega. Na poljubno mesto vstavimo nov stolpec, torej začnemo pisati v prvi prazen stolpec ob tabeli, ali pa stolpec vrinemo na poljubno mesto. Nato v prvo prazno celico tega stolpca vpišemo tako ime kot priimek osebe iz tiste vrstice. Pritisnemo Enter in v naslednji vrstici ponovimo postopek. Nekje na pol poti bi nam moral Excel že ponuditi možnost bliskovite zapolnitve, torej da nam sam zapolni prazne celice v našem novem stolpcu z imenom in priimkom osebe iz tiste vrstice. Včasih se lahko zgodi, da ima Excel svojo predstavo o tem, kaj smo želeli. Takrat preprosto ročno vpisujemo podatke, dokler se ne nauči, kaj od njega želimo in nam ponudi ustrezno možnost. Nato jo le še sprejmemo in se čudimo nad njegovo močjo :)

**Klemen Vončina**

*klemen.voncina@kompas-xnet.si*



## Kako učinkovito upravljati vire v večprojektnem okolju s pomočjo Teorije omejitev in Microsoft Project - 2. del

**Zakaj tradicionalno vodeni projekti mnogokrat ne dosežejo zelenih rezultatov?**

Zelo težko je primerjati izvedbo istega projekta po klasični metodologiji in metodologiji kritične verige ter spreminjati parametre, saj ne moremo zagotoviti istih pogojev (tudi če imajo isto vsebino), preveč je izgovorov in razlogov za

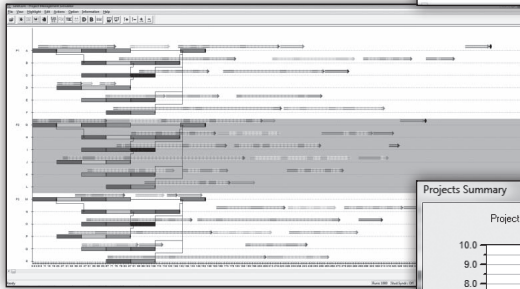
odstopanje, preveč je spremenljivosti v nalogah projekta in se vsaka rešitev (problem) spreminja sproti. Dosti bolje je, če lahko s pomočjo simulatorja analiziramo projekt ali skupino projektov večkrat in pogledamo trende ter spreminjamo različne parametre. V ta namen si bomo pomagali s PmSim simulatorjem [7].

Na spodnjih slikah so prikazani rezultati simulacije izvedbe projektov v večprojektnem okolju

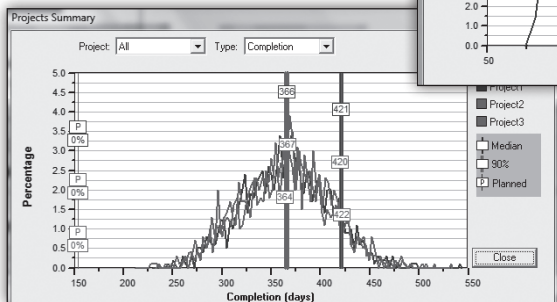
na dva načina – na klasičen način in s pomočjo metodologije kritične verige.

Vsak projekt ima predviden čas izvedbe 152 dni (konec označen s črnim krogom za sivim kvadratom), viri (barva določa vir) so med projekti deljeni, vključen je študentski sindrom in 75 % virov napačno poroča predčasno dokončanje naloge (Parkinsonov zakon). Simulacija je izvedena s 1000 ponovitvami.

Ne glede na to, da gre za prikaz simulacije, vidimo, da nobeden izmed projektov po klasični metodologiji ne bo dokončan v predvidenem roku (slika 6 in slika 7). Predviden čas dokončanja projekta je daljši za več kot trikrat glede na planiran čas izvedbe (iz predvidenih 152 dni na 422 dni z 90 % zanesljivostjo).

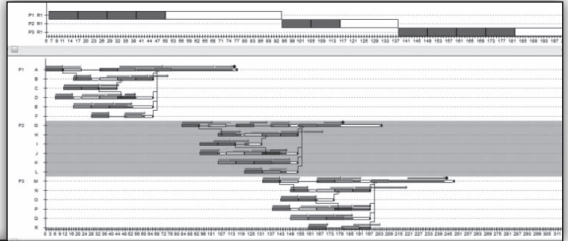


Slika 6: Rezultati simulacije izvedbe projektov po klasični metodologiji

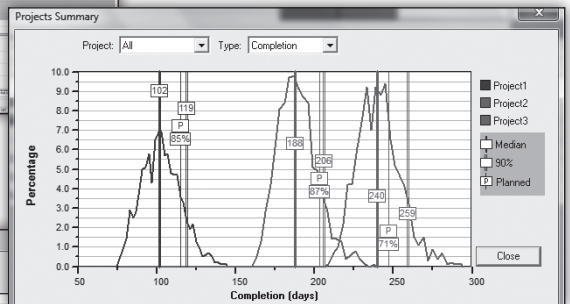


Slika 7: Čas dokončanja projektov po klasični metodologiji

Na sliki 8 smo izvedli plan projekta po metodologiji kritične verige in določili ključni vir (rdeča barva) na nivoju portfelija projektov. Le-ta je povzročil, da so se plani projektov po prioriteti podredili njegovi razpoložljivosti. Prav tako smo dodali še dodaten blažilec (100%) med posameznimi projekti za zaščito ključnega vira (tudi ključni vir ima spremenljivost izvedbe nalog projekta) – blažilec ključnega vira (ang. Drum Buffer – DB).



Slika 8: Rezultati simulacije izvedbe projektov v večprojektnejem okolju po metodologiji kritične verige



Slika 9: Čas dokončanja izvedbe projektov po metodologiji kritične verige

Kot je razvidno iz slike 9, so časi dokončanja vseh treh projektov bistveno krajši. Prvi projekt je z 90 % zanesljivostjo končan v 116 dneh, drugi v 176 dneh, tretji pa v 212 dneh. Vsi trije projekti so izvedeni skoraj dvakrat hitreje kot v primeru, ko viri izvajajo večopravnost med projekti – z 90% zanesljivi-

vostjo je prvi projekt izveden v 116 dneh oz. ~28% časa (116/421), drugi v 206 dneh oz. ~49% časa (206/420) in tretji v 259 dneh oz. ~61% časa (259/422). Prednosti na nivoju podjetja so očitne, saj se projekti lahko začnejo bistveno hitreje tržiti in izboljšamo denarni tok podjetja.

## Zaključek

S sedanjim zmanjševanjem sredstev za investicije in zaostreno konkurenco poizkuša vsak potencialni ponudnik storitev ponuditi najnižjo, a komaj vzdržno ceno. Tega se podjetja močno zavedajo, zato iščejo načine, kako se izboljševati čim bolj racionalno, na način, ki bo prinesel želeno rezultate. V mnogih primerih odobravanje vedno novih projektov, zgolj dodajanje novih virov in investicij na strani izvajalcev ne prinese želenih rezultatov. Zastavlja se vprašanje, kako lahko izboljšamo upravljanje projektnega portfelja, kako lahko izboljšamo sodelovanje med izvajalci in naročnikom, kako lahko skrajšamo čas za izvedbo projektov, povečamo kvaliteto dela in povečamo dobiček izvajalcev in povečamo zadovoljstvo naročnikov (kot uporabnikov rezultatov projektov)?

Odgovore na navedena vprašanja smo iskali sistematično s pomočjo aplikacij in orodij poznanih pod skupnim imenom Teorija omejitvev (ang. Theory Of Constraints). Teorija omejitvev predpostavlja, da je znotraj vsake organizacije mnogo procesov (virov), ki so med seboj povezani in soodvisni. Na ta način se postavlja analogija delovanja organizacije z močjo »verige«, kjer je moč celotne verige omejena z močjo najšibkejšega člana.

V našem prispevku smo prikazali, kako lahko s Teorijo omejitvev definiramo usmeritve, ki imajo na nivoju projektnega vodenja največji doprinos. Prikazali smo, kako lahko s projektno metodologijo kritične verige skrajšamo čas potreben za izvedbo nalog projekta in posledično izboljšamo rezultate podjetja in njegovo konkurenčno prednost. Še posebno lahko pridobimo če imamo na razpolago ustrezna programska

orodja (npr. MS Project s CCPM add-in) in na ta način pridobimo zadostno konkurenčno prednost pred drugimi (tujimi) podjetji, ki uporabljajo tradicionalne pristope. S tem dobimo Win-Win-Win situacijo za vse vpletene.

## Literatura

- Eliyahu M. Goldratt, Jeff Cox. The Goal: A Process of Ongoing Improvement, North River Press, 2004
- The Standish Group. CHAOS MANIFESTO 2013, Objavljeno na <http://versionone.com/assets/img/files/ChaosManifesto2013.pdf> (zadnji ogled 4. 4. 2015)
- Tomaž Aljaž, Lidija Grmek Zupanc, Branka Jarc Kovačič, Gabrijela Krajnc, Mateja Demšar. Kako s pomočjo Teorije omejitve delati manj, a narediti več, 2014
- Programska oprema PMSim version 2.03
- Eliyahu M. Goldratt. Critical Chain, North River Press, 1997
- Programska oprema cc-(M)Pulse version 4.0.0.0

**dr. Tomaž Aljaž, EMA d.o.o., Celje**

*Dr. Tomaž Aljaž je certificiran strokovnjak na področju Teorije omejitvev (angl. Theory Of Constraints – TOC) z znanjem in referencami na tem področju in 18 let izkušenj v gospodarstvu in bančništvu ter več kot 7 let kot predavatelj doma in v tujini. V zadnjem obdobju daje poudarek na izboljšavi rezultatov dela z vpeljavo »vitkih« metodologij dela v multinacionalnih organizacijah, vodenje oddelkov in ljudi, strokovnih ekip, vzpostavitev meril dela, priprava celovitih rešitev in vodenje projektov. Svoje znanje s področja vpeljuje »vitkih« metodologij dela je v letih 2013 in 2014 izpopolnjeval na Washington State University, ZDA, ki je specializirana za vpeljevanje »vitkih« metod dela v organizacije, projektna vodenja in druge metode s ciljem izboljšanja rezultatov v podjetjih. Na omenjeni univerzi je bil jeseni 2014 tudi zunanji predavatelj. V aprilu 2014 je pridobil tudi certifikat na mednarodni organizaciji Theory Of Constraints International Certification Organization (TOCICO).*



# Filtriranje podatkov s filter gradnikom

## 1. Uvod

V primeru prikazovanja vsebine Excel datotek na SharePoint-u lahko naletimo na nekaj omejitev, predvsem kar se tiče filtriranja podatkov. Če imamo filtriranje podatkov v Excelu urejeno z uporabo »Data validation-a«, bomo imeli težavo s filtriranjem podatkov v gradniku.

V SharePoint-u so temu namenjeni posebni gradniki za filtre, ki nam pri tem pomagajo.

Poglejmo si enostaven primer.

## 2. Postopek v Excelu

V Excelu imamo enostavno tabelo s prikazom regij in števila prebivalcev. Podatki so izmišljeni.

Slika 1: Podatki v Excel datoteki

Izberemo si poljubno polje, ki bo služilo kot filter

Regija	Število prebivalcev
Gorenjska	312313
Dolenjska	296133
Osrednjeslovenska	557500
Primorska	347850
Štajerska	223698
Pomurska	199585

in ga ustrezno poimenujemo. V našem primeru bo to polje B3 poimenovano »FilterRegije«.

Slika 2: Dodajanja polja za filter

FilterRegije	
A	B
1	
2	
3	

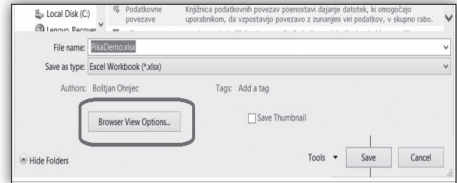
V naslednjo poljubno celico vnesemo formulo s katero želimo filtrirati podatke glede na vneseni filter. V našem primeru je bila v polju B4 uporabljena formula, ki nam iz naše tabele vrne število prebivalcev glede na ime regije in v celici izpiše »Ni izbire« v primeru, da ni vnesen noben filter.

Slika 3: formula za filter

C	D	E	F
=IFERROR(VLOOKUP(FilterRegije;Table1;2;0);"Ni izbire")			
Ni izbire			

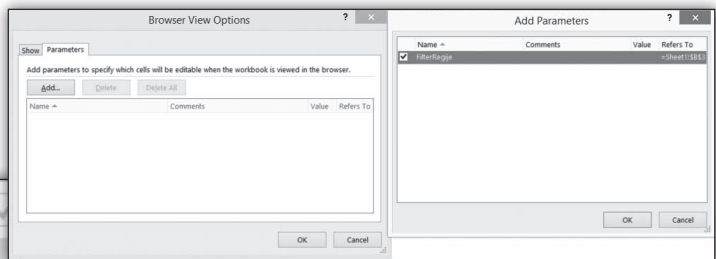
Ko imamo oboje nastavljeno XLSX datoteko shranimo v poljubno SharePoint knjižnico. Pred shranitvijo je potrebno v »Save as« dialogu izbrati »Browser view options«

Slika 4: Shranjevanje XLSX datoteke



in v zavihku »Parameters« obklicati naš filter.

Slika 5: označba filtra v XLSX datoteki

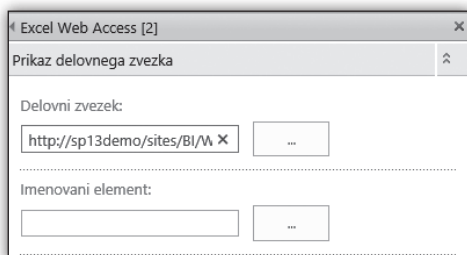


## 3. Postopek v SharePoint-u

Ko imamo datoteko shranjeno v knjižnici jo lahko prikažemo v obliki »Excel Web Access« gra-

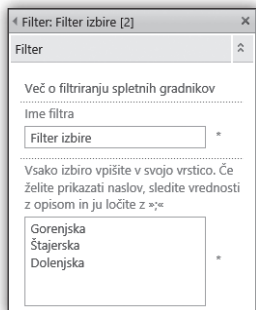
dnika. Gradnik dodamo na stran in pokažemo pot do naše XLSX datoteke.

Slika 6: nastavev Excel web access gradnika



Nad ta gradnik dodamo še gradnik za filter imenovan »Filter izbire«. V nastavitvah gradnika naštejemo možnosti za filtracijo. Možnosti morajo biti enake kot v Excelu, v našem primeru torej nazivi regij. Vrednosti v nastavitvah gradnika dodajamo eno pod drugo.

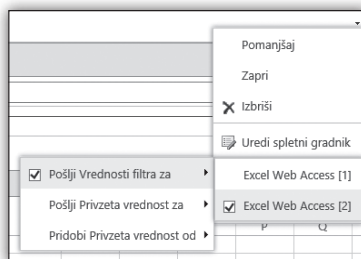
Slika 7: Vnos filtrov



Ko imamo vrednosti vnesene je potrebno na koncu oba gradnika še povezati. To naredimo v možnostih gradnika za filtriranje, kjer preko menija »Povezave > Pošlji vrednosti filtra za« izberemo gradnik,

ki prikazuje našo Excel datoteko, v našem primeru torej »Excel Web Access 2«.

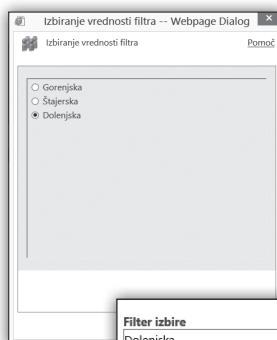
Slika 8: Povezava obeh gradnikov



## 4. Rezultat

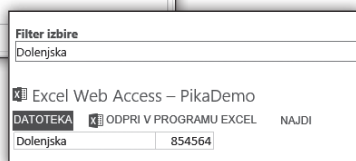
Končni rezultat je vnos filtra v gradniku filter, ki ga izberemo v pojavnem oknu.

Slika 9: Filtriranje podatkov



Povezan gradnik nam prikaže vrednost tega filtra glede na podatke v Excel datoteki.

Slika 10: Rezultat filtriranja



**Boštjan Ohnjec**

[bostjan.ohnjec@kompas-xnet.si](mailto:bostjan.ohnjec@kompas-xnet.si)

## Aktualni tečaji

20332: Advanced Solutions of SharePoint Server 2013: 7. – 11.9.2015

20346: Managing Office 365 Identities and Services: 5. – 9.10.2015 **NOVO!**

55028: SharePoint 2013 Power User: 23. - 24.11.2015





# SharePoint 2013 Search troubleshooting

## 1. Uvod

Pri delu s SharePoint strežnikom 2013 vedno bolj do izraza prihaja uporaba Search infrastrukture, saj nam ta omogoča prikaz najrazličnejših podatkov iz različnih virov v različnih oblikah. Vendar pa tudi ta funkcionalnost SharePoint-a lahko, od časa do časa odpove in v tem članku je zbranih nekaj najbolj pogostih napak s katerimi se lahko srečamo in upam, da bodo opisane rešitve delovale tudi v vašem okolju.

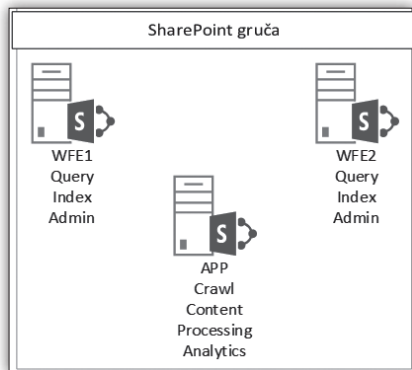
Opisane napake in reševanje težav s temi napakami, pridejo v poštev za tista podjetja, ki imajo več strežniško okolje in kjer so search komponente porazdeljene med več strežniki.

Primer postavitve, kjer imamo strežnika WFE1 in WFE2 s komponentami:

- Query processing
- Index component
- Administration component.

Strežnik APP ima v tem primeru postavitve vloge:

- Crawl component
- Content processing component
- Analytics component



## 2. Search komponenta v »failed state«

### Opis

Na enem izmed strežnikov odpremo Centralno administracijo in navigiramo do:

Service applications -> Search\_SA in na administrativni strani opazimo, da ima search komponenta rdeč krogec z belim »X«. To pomeni, da je komponenta v failed state in ne deluje.

Primer:

### Predlagana rešitev:

Predlagana rešitev je restart strežnika za katerega kaže, da je komponenta v failed state.

## 3. Prikaz novih elementov v rezultatih iskanja

Za težave s prikazom novih elementov v rezultatih iskanja je lahko več vzrokov. Za odkrivanje napake, moramo odpreti crawl log v centralni administraciji na enem od strežnikov

Odpremo centralno administracijo -> Service applications -> Search\_SA -> Crawl log (na levi strani) -> Local SharePoint sites -> crawl history in kliknemo na prvi crawl log.

Pojavi se okno:

Search\_SA: Crawl Log - URL

Content Source | Host Name | Crawl History | Error Breakdown | Databases | URL View

Search for documents that have been crawled.

Type a URL or host name. Use the \* character as a wildcard.  Exact match

Filters

Content source: Local SharePoint sites

Status: All

Message: All

Complete message: All

Start Time: 12 AM 00 End Time: 12 AM 00

---

Search\_SA: Crawl Log - URL

Content Source | Host Name | Crawl History | Error Breakdown | Databases | URL View

Search for documents that have been crawled.

Type a URL or host name. Use the \* character as a wildcard.  Exact match

Filters

Content source: Local SharePoint sites

Status: All Errors

Message: All

Complete message: All

Start Time: 7/10/2015 8 AM 00 End Time: 7/10/2015 3 PM 00

V tem oknu lahko izberemo status: »**All errors**« in omejimo iskanje s »**Start Time**« in »**End time**« ter kliknemo »**Search**«

**Search komponente so OK, vendar se novi elementi ne prikazujejo v rezultatih iskanja 1**

### Opis

V crawl logih se lahko pojavi napaka:

»The crawler could not communicate with the server. Check that the server is available and that the firewall access is configured correctly. If the repository was temporarily unavailable, an incremental crawl will fix this error«

V tem primeru je odkrivanje napake odvisno od postavitve samega SharePoint okolja. Problem, ki se tukaj pojavi je, da crawl komponenta ne more indeksirati Web strežnikov. Kar lahko naredimo v tem primeru je, da se poskusimo prijaviti na vse web strežnike v naši gruči in poskusimo

odpreti portal v brskalniku in preverimo ali se stran pravilno naloži.

Pogosto se na APP strežnikih nastavi tudi, da strežnik indeksira samega sebe, da ne obremenjuje po nepotrebem ostalih web strežnikov in s tem vpliva na uporabniško izkušnjo. V tem primeru, bi bilo potem potrebno odpreti portal na strežniku APP in preveriti delovanje IIS-ja.

### Predlagana rešitev:

Običajno pomaga že restart IIS na strežniku APP. V primeru, da to ne bi pomagalo se naredi restart strežnika APP.

**Search komponente so OK, vendar se novi elementi ne prikazujejo v rezultatih iskanja 2**

### Opis

V crawl logs se pojavi napaka:

»The item could not be indexed successfully because the item failed in the indexing subsystem. ( The item could not be indexed successfully because the item failed in the indexing subsystem.«

Za to napako je običajno kriva primarna index komponenta, ki je na enem od strežnikov:

- WFE1,
- WFE2.

da bi ugotovili kateri od strežnikov gosti primarno Index ali administrativno komponento, je potrebno prenesti in zagnati skripto za preverjanje stanja Search infrastrukture. Skripto si lahko prenesete z naslednje povezave:

<https://goo.gl/3ju4ll>

Zagon te skripte izpiše rezultate v obliki:



```

Search Topology health check
=====

10. julij 2015 14:25:16
SSA: Search_SA

job-application-server-admin-service
-----
WFE1: job-application-server-admin-service recently ran 18 seconds ago
WFE2: job-application-server-admin-service recently ran 21 seconds ago
APP: job-application-server-admin-service recently ran 27 seconds ago

job-application-server
-----
WFE1: job-application-server recently ran 25 seconds ago
WFE2: job-application-server recently ran 19 seconds ago
APP: job-application-server recently ran 9 seconds ago

Component                Server                Partition State
-----                -
AdminComponent1 (Primary) WFE1                Active
AdminComponent1          WFE2                Active
AnalyticsProcessingComponent1 APP                Active
ContentProcessingComponent1 APP                Active
CrawlComponent0         APP                Active
IndexComponent1 (Primary) WFE1                0                Active
IndexComponent1          WFE2                0                Active
QueryProcessingComponent1 WFE1                Active
QueryProcessingComponent1 WFE2                Active

Analytics Processing Job Status:
  SearchAnalytics : Idle
    Last successful start time: 10.7.2015 0:00:01
    Last completed time: 10.7.2015 0:08:44
    Next scheduled run time: 11.7.2015 0:00:00
  UsageAnalytics : Idle
    Last successful start time: 10.7.2015 1:00:01|
    Last completed time: 10.7.2015 1:10:03
  SearchReports : Idle
    Last successful start time: 10.7.2015 3:10:00
    Last completed time: 10.7.2015 3:10:11
    Next scheduled run time: 11.7.2015 3:01:00

Searchable items: 2078
Crawler is idle

Search service overall state: OK

```

Pomemben podatek za reševanje težav je, da vidimo na katerem strežniku se nahaja primarna Index komponenta. V primeru izpisa je to strežnik WFE1 ->

Component	Server	Partition	State
CrawlComponent0	APP		Active
IndexComponent1 (Primary)	WFE1	0	Active
IndexComponent1	WFE2	0	Active

#### Predlagana rešitev:

Za odpravo te napake, je potrebno narediti restart na strežniku kjer se nahaja primarna Index komponenta.

## 4. Ogled crawl logov ni na voljo

### Opis

V Centralni administraciji navigiramo do administracije Search servisne aplikacije in želimo npr. odpreti loge indeksiranja vsebin. Pojavi se nam napaka:

»Retrieving the COM class factory for component with CLSID {0FF1CE15-0005-0000-0000-000000000000} failed due to the following error: 800703fa Illegal ope-

ration attempted on a registry key that has been marked for deletion. (Exception from HRESULT: 0x800703FA).«

Razlog napake je, da je na primarni administrativni komponenti ključ v registru označen za brisanje in v primeru, da želimo odpreti Crawl logs na strežniku, na katerem ni primarna administrativna komponenta, nam SharePoint javi to napako.

### Predlagana rešitev:

Pri odpravljanju te napake, se moramo prav tako poslužiti skripte, ki je omenjena v točki 3.2. V tem primeru nas zanima kateri od strežnikov gosti primarno Admin komponento. V našem primeru je to strežnik WFE1.

Za odpravo napake, bi morali narediti ponoven zagon strežnika WFE1.

Component	Server	Partition	State
AdminComponent1 (Primary)	WFE1		Active
AdminComponent1	WFE2		Active

## 5. Zaključek

Upam, da napotkov za reševanje težav s SharePoint strežniki ne boste nikoli potrebovali, če pa že, pa upam, da vam zgoraj opisani postopki prihranijo marsikatero uro raziskovanja in razbijanja glave.

V primeru, da imate dodatna vprašanja in želite pomoč pri reševanju težav s SharePoint strežniki me vedno lahko kontaktirate po elektronski pošti.

**Robi Vončina, uni.dipl.ekon**  
MVP, MCT, MCITP, MCSA, MCTS  
[robi.voncina@kompas-xnet.si](mailto:robi.voncina@kompas-xnet.si)



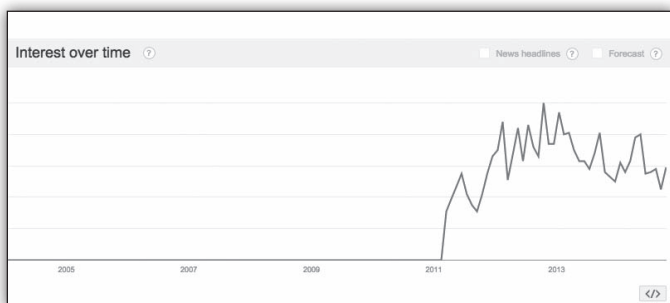
## Uporabniški vmesniki hibridnih mobilnih aplikacij

V zadnjih nekaj letih se intenzivno ukvarjam z razvojem mobilnih aplikacij. Predvsem z izdelavo hibridnih aplikacij na podlagi ogrodja AngularJS. V iskanju dobre knjižice za izdelavo mobilnih uporabniških vmesnikov, ki so združljive z AngularJS, sem spoznal nekaj opcij: IonicFramework, [Famo.us](http://Famo.us), Framework7, OnsenUI ter Material Design.

### Kaj je hibridna mobilna aplikacija?

Hibridna mobilna aplikacija je 'domorodna' aplikacija, ki je razvita na temelju spletnih standardov: HTML5, CSS in JavaScripta. Domo-rodna aplikacija vsebuje le skriti spletni brskal-

nik (WebView) v katerem prikazujemo naše HTML datoteke. S pomočjo uporabe knjižnic Apache Cordova oz. PhoneGap pa lahko dostopamo do vseh funkcionalnosti telefona, četudi le-te niso izpostavljene spletnim stranem.



V bistvu to pomeni, da lahko HTML5 aplikacijo prevedete v domorodno mobilno aplikacijo in jo objavite v trgovinah z aplikacijami (Google Play, Apple Store, Windows Store, ...). Takšen pristop omogoča hitro izdelavo mobilnih aplikacij, poleg tega pa omogoča, da aplikacijo enkrat napišemo in potem prevedemo za vse pomembnejše mobilne platforme (Android, iOS in Microsoft Phone).

### Uporabniški vmesniki

V preteklosti bi za razvoj hibridnih mobilnih aplikacij najverjetneje uporabili jQuery knjižnice. S tem ni nič narobe in mnogo hibridnih aplikacij je še danes narejenih na osnovi jQuery komponent. Je pa na ta način težko zagotoviti, da bo mobilna aplikacija odzivna in da uporabnik ne bo imel občutka, da gre za spletno stran. Zaradi tega se v zadnjem času pojavljajo ogrodja, ki nam omogočajo izdelavo mobilnih uporabniških vmesnikov.

### IonicFramework

Je ogrodje, ki še ni dopolnilo eno leto. Mobilni razvijalci so mu izredno naklonjeni, saj se osvežuje skoraj na dnevni bazi. Prav tako se lahko pohvali z izredno razširjeno bazo razvijalcev, ki so to ogrodje posvojili in v njem razvijajo aplikacije. IonicFramework je ogrodje, ki je zgrajeno nad ogrodjem AngularJS. Poleg samih knjižnic ponujajo še dodatne strežniške in razvijalske storitve.

### Famo.us + AngularJS

[Famo.us](#) je mobilno ogrodje, ki je še novejšo od IonicFrameworka. Namenjen je vse, katerih glavni skrbi sta odzivnost in hitrost. Omogoča animacije s hitrostjo 60 slik na sekundo. V osnovi [Famo.us](#) uporablja lasten JavaScript pogon, ki zna delati z CSS3 3D transformacijskimi funkcijami, ki jih pospešuje GPU.

### Framework7

Je okolje z veliko razvitimi komponentami za razvoj aplikacij za iOS. Lahko ga uporabljamo tudi za razvoj splošnih hibridnih aplikacij, saj ponuja bogat nabor UI elementov.

### OnsenUI

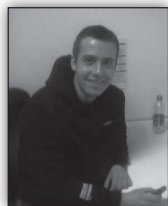
Je konkurenčen product IonicFrameworku. Prihaja s podporo za AngularJS in ponuja vse kar ponuja Ionic. OnsenUI je bil narejen kot odgovor za Cordovo in PhoneGap razvijace, ki so se trudili z razvojem uporabniških vmesnikov.

### Material Design

Je Googlov concept za razvoj uporabniških vmesnikov, ki temelji na izgledu komponent operacijskega Sistema Android L. Namen projekta je ponuditi AngularJS elemente v 'materialni' obliki. Trenutno izgleda obetajoče, vendar je za karkoli konkretnega še prezgodaj. Neka j primerov najdemo na naslovu <http://material.angularjs.org>.

Dejan Mauer

[dejan.mauer@cent.si](mailto:dejan.mauer@cent.si)



## Singleton vzorec v C# programskem jeziku

Singleton je najbolj pogosto uporabljen programski vzorec v svetu razvoja programske opreme. Osnoven princip

tega vzorca je, da imamo v danem momentu na voljo le eno instanco Singletona, ki je implementiran kot class ter da imamo tudi zelo enostaven dostop do le-te. Prav tako je cilj Sin-

gleton vzorca tudi lazy loading oz. da se instance ustvari šele takrat, ko jo zares potrebujemo in ne že ob samem zagonu naše aplikacije. Obstaja več načinov, kako implementiramo Singleton vzorec v C#-u.

Vse te različne implementacije si delijo naslednje lastnosti oz. pravila:

- le en private constructor brez parametrov

- sealed class, kar lahko pripomore k bolj optimalnem delovanju kode
- statična variabla, ki drži referenco na instanco objekta
- public static metodo, ki vrača instance in omogoča globalen dostop do le-te

Singleton vzorec v multithreaded okolju mora biti še posebej natančno implementiran. Prva verzija je neprimerna v multi-threaded okolju, ker bi lahko dva ali več različnih threadov instancirala instance Singleton classa.

### Verzija 1:

```
public sealed class Singleton
{
    private static Singleton instance = null;
    private Singleton() { }
    public static Singleton Instance
    {
        get
        {
            if (instance == null)
            {
                instance = new Singleton();
            }
            return instance;
        }
    }
}
```

Zgornja verzija, ki je tudi najbolj osnoven način, sploh pri programerjih začetnikih, ni varna za uporabo v multi-threaded okolju. Dva različna thread-a (niti) bi lahko izvedla test `if (instance == null)` in kreirala instanci ter kršila osnovno pravilo Singleton vzorca.

Naslednja verzija je pa najbolj optimalna implementacija Singleton vzorca v C#-u (zahteva je .NET 4 framework):

### Verzija 2:

```
public sealed class Singleton
{
    private Singleton() { }
    private static readonly Lazy<Singleton>
    lazy = new Lazy<Singleton>(() => new
    Singleton());
}
```

```
public static Singleton Instance { get {
return lazy.Value; } }
}
```

Zgornja verzija ustreza vsem potrebam in zahtevam implementacije Singleton vzorca v C#-u.

Omembe vredna je implementacija vzorca s statično instanco generičnega tipa `Lazy<>`, ki je na voljo od verzije .NET frameworka 4 dalje.

Če `Lazy<>` tipa nimamo na voljo, je naslednja verzija implementacije prav tako ustrezna:

### Verzija 3:

```
public sealed class Singleton
{
    private Singleton() { }

    public static Singleton Instance { get {
return Nested.instance; } }

    private class Nested
    {
        static Nested() { }

        internal static readonly Singleton
        instance = new Singleton();
    }
}
```

Lazy instanciranje Singleton vzorca je v zgornjem primeru doseženo z gnezdenim classom ter statičnim constructorjem class-a `Nested`. Na ta način zagotovimo pravilno delovanje implementacije v multi-threaded okolju ter omogočimo lazy loading samega Singleton vzorca.

David Čamdžić

[david.camdzic@kompas-xnet.si](mailto:david.camdzic@kompas-xnet.si)

## Aktualni tečaji

**20480: Programming in HTML5 with JavaScript and CSS3:**  
21. – 25.9.2015

**20486: Developing ASP.NET MVC 4 Web Applications:**  
12. – 16.10.2015



# Izboljšave orodij za ASP.Net

Prejšnji mesec je prinesel novi Visual Studio 2015, ki prinaša kar nekaj sprememb in izboljšav na področju spletnega razvoja.

## JSON urejevalnik

JSON je postal prvorazredna izkušnja v Visual Studiojevemu urejevalniku, s podporo za potrjevanje JSON Schema, INTELLISENSEom, in podporo za [SchemaStore.org](http://SchemaStore.org) pisanje. Tako urejanje JSON vsebine še nikoli ni bilo tako enostavno. Prav tako je dodan tudi intellisense p o d p o r o bower.json in package.json datotek za Bower in NPM upravljalni-ka paketov.

```
Schema: http://json.schemastore.org/bower
1  {
2    "name": "My ASP.NET App",
3    "devDependencies": {
4      "bootstrap": "3.3.5",
5      "angular": "1.4.3",
6      "react":
7    },
8    "scripts": {
9      "react-simple",
10     "react-quick",
11     "react-quick-jquery",
12     "react-typeahead",
13     "react",
14     "react-o11y",
15     "react-addons",
16     "react-animate-plus",
17     "react-anything-sortable"
18   }
19 }
```

## HTML urejevalnik

HTML urednik je v tej dopolnitvi prejel veliko pozornosti. Podpira in sledi najnovejšim HTML 5 standardom in ima bogato podporo priljubljenih novih 'frameworkom' in knjižnicam. Prav tako sedaj zagotavlja bogat z ikono označen intellisense za Bootstrap css razrede (ki so bili že prej vključeni kot del spletne predloge).

```
9  <div class="jumbotron page">
10 <h1>ASP.NET</h1>
11 <p class="lead">AS
12 <p><a href="http://asp.net">
13 </div>
```

Dobili smo tudi podporo za nastajajoč standard za uvoz spletnih komponent,

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title></title>
5 <script src="bower_components/platform/platform.js"></script>
6 <link rel="import"
7   href="bower_components/bit-avatar/dist/bit-avatar.html" />
8 </head>
9 <body>
10
11 <bit-avatar user-id="scottgu@microsoft.com"></bit-avatar>
12
13 </body>
14 </html>
```

Ter intellisense za AngularJS javascript direktive in attribute z ustrezno ikono, ki vam pove kje prožite AngularJS funkcionalnosti.

```
<input type="text" name="name" ng-model
  ng-model
  ng-model-options
```

## JavaScript urejevalnik

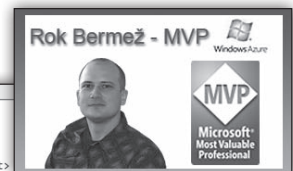
Tudi Javascript urejevalnik je dobil podporo za urejanje AngularJS struktur. Vključno z regulatorji, storitvami, tovarnami, direktivami in animacijami. Podprt je tudi celoten nov ECMAScript 6 standard z vsemi svojimi super novostmi. Dodana je tudi nova navigacijska vrstica, ki vam pomaga krmari med glavnimi elementi in JSDoc podporo za intellisense.

Toliko na kratko o najbolj očitnih spremembah v urejevalnikih. Naslednjič si bomo pogledali pa na kakšen način uporabljamo nove upravljalnike JavaScript paketov, kot sta na primer Grunt in Gulp.

```
1 /**
2  * Add two numbers that are submitted
3  * @param {Number} addend1 First number to add
4  * @param {Number} addend2 Second number to add
5  * @returns {Number} Sum of the two values
6  */
7 var sum = function (addend1, addend2) {
8   return addend1 + addend2;
9 }
10
11 sum()
  Number (Number addend1, Number addend2):
  Add two numbers that are submitted
  addend1: First number to add
```

**Rok Bermež**

MVP, MCT, MCTS, MCPD, MCITP  
[rok.bermez@kompas-xnet.si](mailto:rok.bermez@kompas-xnet.si)





## Powershell Direct

Ena od novosti, ki jih bomo lahko v prihodnje, začenši z windows 10, uporabljali pri upravljanju naših virtualnih

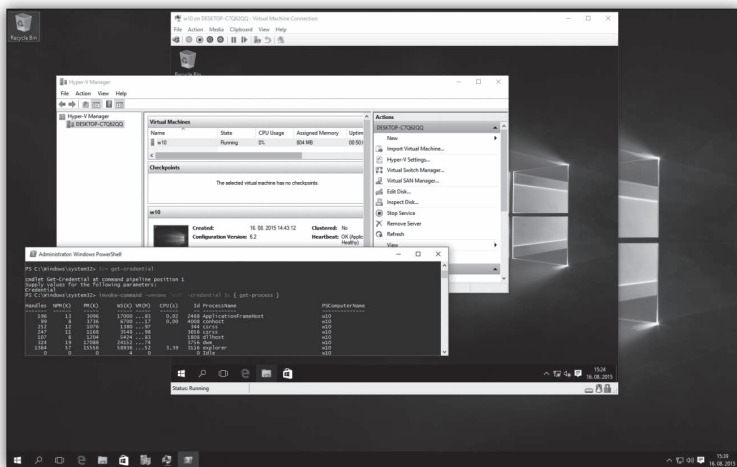
okolij, je Powershell Direct. Do sedaj smo te stvari počeli z RDP ali powershell remoteingom.

Prvega težko avtomatiziramo, za drugega moramo poskrbeti za ustrezne nastavitve omrežja in požarnega zidu ( enable-PSremoteing). Z Windows 10 in Serverjem 2016 pa bo Powershell za komunikacijo med gostom in hostom

začel uporabljati tehnologijo VMbus , ki je sicer prisotna že nekaj časa (no od windows server 2008 oz hipervizorja tipa 1) in sicer skrbi za interno komunikacijo med obema. V Windows server 2012R2 smo lahko preko nje pretakali datoteke med sistemoma (copy-VMfile), sedaj pa nam bo služila za popolno powershell izkušnjo, v kateremkoli stadiju našega virtualnega okolja, saj komunikacija poteka po spominskem toku in zanjo ne rabimo vzpostavljati in konfigurirati omrežja. Ukaze lahko prožimo tako z Invoke-Command kot z Enter-PSsession

Npr:

```
$cred=Get-Credentiaql
Invoke-Command -VMname »virtualka«
-ScriptBlock { ukazi }
```



Trenutno lahko to tehnologijo uporabljamo z določenimi omejitvami. Tako host kot gost morata biti Windows 10 ali Windows Server 2014 TP, proces na hostu mora imeti pravice Hyper-V administrator, potrebujemo uporabniški pristo do virtualke, za administratorske posege pa seveda tudi privilegirane.

**Aleš Lipušček**

MCP, MCTS, MCITP

[ales.lipusecek@kompas-xnet.si](mailto:ales.lipusecek@kompas-xnet.si)

## Aktualni tečaji

**20412: Configuring Advanced Windows Server 2012 Services: 21. – 25.9.2015**

**20411: Administering Windows Server 2012: 26. – 30.10.2015**

**Xnet**  
izobraževanja



## T-SQL for Statistics - Higher population moments (2)

Mean, standard deviation, skewness, and kurtosis, which I will focus on in this article, are also called the four population moments. Mean uses the values on the first degree in the calculation; therefore, it is the first population moment. Standard deviation uses the squared values and is therefore the second population moment. Skewness is the third, and kurtosis is the fourth population moment. All together, they give you a very good estimation of the population distribution.

### Kurtosis

Kurtosis characterizes the relative peakedness or flatness of a distribution compared with the normal distribution. A positive kurtosis means a higher peak around the mean and some extreme values on any side tail. A negative kurtosis indicates a relatively flat distribution. For a peaked distribution, consider that values far from the mean in any direction might be correct. The formula for the kurtosis is

$$\text{Kurt} = \frac{n * (n + 1)}{(n - 1) * (n - 2) * (n - 3)} * \sum_{i=1}^n \left( \frac{v_i - \mu}{\sigma} \right)^4 - \frac{3 * (n - 1)^2}{(n - 2) * (n - 3)}$$

As with the skewness, you can see that the formula for the kurtosis also includes the mean and the standard deviation. To get an efficient query, I start with expanding the subtraction again:

$$(v_i - \mu)^4 = v_i^4 - 4v_i^3\mu + 6v_i^2\mu^2 - 4v_i\mu^3 + \mu^4$$

After that, I can use the fact that the sum is distributive over the product again, and calculate the kurtosis with a single scan of the data, as the following query shows:

```
WITH KurtCTE AS
(
SELECT SUM(salesamount) AS rx,
SUM(POWER(salesamount,2)) AS rx2,
SUM(POWER(salesamount,3)) AS rx3,
SUM(POWER(salesamount,4)) AS rx4,
COUNT(salesamount) AS rn,
STDEV(salesamount) AS stdv,
AVG(salesamount) AS av
FROM dbo.SalesAnalysis
)
SELECT
(rx4 - 4*rx3*av + 6*rx2*av*av -
4*rx*av*av*av + rn*av*av*av*av)
/ (stdv*stdv*stdv*stdv) * rn * (rn+1) /
(rn-1) / (rn-2) / (rn-3)
- 3.0 * (rn-1) * (rn-1) / (rn-2) / (rn-3)
AS kurtosis
FROM KurtCTE;
```

The result for the kurtosis of the salesamount variable is

```
kurtosis
-----
68.7756494703992
```

The value for the kurtosis tells you that there are many values of the salesamount variable around the mean, because there is a high peak in the distribution around the mean; however, the variable has quite a few values far away from the mean in either or both of the tails as well.

### Skewness and kurtosis with CLR UDAs

The queries for calculating skewness and kurtosis are more complex than the queries earlier in the module. Now imagine you need to calculate these two values in groups. Calculating mean and standard deviation in groups is simple, because you have appropriate T-SQL aggregate functions, which can be used together with the GROUP BY clause. However, calculating skewness and kurtosis in groups with T-SQL expressions only leads to more complex queries.

Calculating skewness and kurtosis in groups would be simple if there were appropriate T-SQL aggregate functions. You can actually expand the list of the T-SQL aggregate functions with user-defined aggregate functions. However, you can't define a user-defined aggregate (UDA) in T-SQL. You need a Common Language Runtime (CLR) language for this—for example, Visual C#. You can use either a SQL Server Database Project template or a simple Class Library template in Microsoft Visual Studio with the appropriate programming languages installed. Refer to books online for details on how to create such a project. Then you can simply copy the following C# code for the skewness and kurtosis UDAs, which uses the same mathematics as the T-SQL solutions for an efficient calculation with a single scan of the data:

```
using System;
using System.Data;
using System.Data.SqlClient;
using System.Data.SqlTypes;
using Microsoft.SqlServer.Server;

[Serializable]
[SqlUserDefinedAggregate(
    Format.Native,
    IsInvariantToDuplicates = false,
    IsInvariantToNulls = true,
    IsInvariantToOrder = true,
    IsNullIfEmpty = false)]
public struct Skew
{
    private double rx; // running sum of
    current values (x)
    private double rx2; // running sum of
    squared current values (x^2)
    private double r2x; // running sum of
    doubled current values (2x)
    private double rx3; // running sum of
    current values raised to power 3 (x^3)
    private double r3x2; // running sum of
    tripled squared current values (3x^2)
    private double r3x; // running sum of
    tripled current values (3x)
    private Int64 rn; // running count of
    rows

    public void Init()
    {
        rx = 0;
        rx2 = 0;
        r2x = 0;

```

```

        rx3 = 0;
        r3x2 = 0;
        r3x = 0;
        rn = 0;
    }

    public void Accumulate(SqlDouble inpVal)
    {
        if (inpVal.IsNull)
        {
            return;
        }
        rx = rx + inpVal.Value;
        rx2 = rx2 + Math.Pow(inpVal.Value,
2);
        r2x = r2x + 2 * inpVal.Value;
        rx3 = rx3 + Math.Pow(inpVal.Value,
3);
        r3x2 = r3x2 + 3 * Math.Pow(inpVal.
Value, 2);
        r3x = r3x + 3 * inpVal.Value;
        rn = rn + 1;
    }

    public void Merge(Skew Group)
    {
        this.rx = this.rx + Group.rx;
        this.rx2 = this.rx2 + Group.rx2;
        this.r2x = this.r2x + Group.r2x;
        this.rx3 = this.rx3 + Group.rx3;
        this.r3x2 = this.r3x2 + Group.r3x2;
        this.r3x = this.r3x + Group.r3x;
        this.rn = this.rn + Group.rn;
    }

    public SqlDouble Terminate()
    {
        double myAvg = (rx / rn);
        double myStDev = Math.Pow((rx2 - r2x
* myAvg + rn * Math.Pow(myAvg, 2))
/ (rn - 1), 1d
/ 2d);
        double mySkew = (rx3 - r3x2 * myAvg
+ r3x * Math.Pow(myAvg, 2) -
rn * Math.
Pow(myAvg, 3)) /
Math.Pow(myStDev, 3)
* rn / (rn - 1) / (rn - 2);
        return (SqlDouble)mySkew;
    }
}

[Serializable]
[SqlUserDefinedAggregate(
    Format.Native,
    IsInvariantToDuplicates = false,
    IsInvariantToNulls = true,
    IsInvariantToOrder = true,

```



```

        IsNullIfEmpty = false)]
public struct Kurt
{
    private double rx; // running sum of
current values (x)
    private double rx2; // running sum of
squared current values (x^2)
    private double r2x; // running sum of
doubled current values (2x)
    private double rx4; // running sum of
current values raised to power 4 (x^4)
    private double r4x3; // running sum of
quadrupled current values raised to power 3
(4x^3)
    private double r6x2; // running sum of
squared current values multiplied by 6 (6x^2)
    private double r4x; // running sum of
quadrupled current values (4x)
    private Int64 rn; // running count of
rows

    public void Init()
    {
        rx = 0;
        rx2 = 0;
        r2x = 0;
        rx4 = 0;
        r4x3 = 0;
        r4x = 0;
        rn = 0;
    }

    public void Accumulate(SqlDouble inpVal)
    {
        if (inpVal.IsNull)
        {
            return;
        }
        rx = rx + inpVal.Value;
        rx2 = rx2 + Math.Pow(inpVal.Value,
2);
        r2x = r2x + 2 * inpVal.Value;
        rx4 = rx4 + Math.Pow(inpVal.Value,
4);
        r4x3 = r4x3 + 4 * Math.Pow(inpVal.
Value, 3);
        r6x2 = r6x2 + 6 * Math.Pow(inpVal.
Value, 2);
        r4x = r4x + 4 * inpVal.Value;
        rn = rn + 1;
    }

    public void Merge(Kurt Group)
    {
        this.rx = this.rx + Group.rx;
        this.rx2 = this.rx2 + Group.rx2;
        this.r2x = this.r2x + Group.r2x;
        this.rx4 = this.rx4 + Group.rx4;
        this.r4x3 = this.r4x3 + Group.r4x3;
        this.r6x2 = this.r6x2 + Group.r6x2;

```

```

        this.r4x = this.r4x + Group.r4x;
        this.rn = this.rn + Group.rn;
    }

    public SqlDouble Terminate()
    {
        double myAvg = (rx / rn);
        double myStDev = Math.Pow((rx2 - r2x
* myAvg + rn * Math.Pow(myAvg, 2))
/ (rn - 1), 1d
/ 2d);
        double myKurt = (rx4 - r4x3 * myAvg
+ r6x2 * Math.Pow(myAvg, 2)
- r4x * Math.
Pow(myAvg, 3) + rn * Math.Pow(myAvg, 4)) /
Math.Pow(myStDev, 4)
* rn * (rn + 1)
/ (rn - 1) /
(rn - 2) / (rn - 3) -
3 * Math.Pow((rn -
1), 2) / (rn - 2) / (rn - 3);
        return (SqlDouble)myKurt;
    }
}

```

Then you can build the project and later deploy the assembly (the .dll file) to your SQL Server instance. Of course, you can also use the project with the source code or the pre-built assembly provided with the accompanying code for the book.

Once your assembly is built, you need to deploy it. The first step is to enable the CLR in your SQL Server instance:

```

EXEC sp_configure 'clr enabled', 1;
RECONFIGURE WITH OVERRIDE;

```

Then you deploy the assembly. Deploying an assembly means importing it into a database. SQL Server does not rely on .dll files on disk. You can deploy an assembly from Visual Studio directly, or you can use the T-SQL CREATE ASSEMBLY command. The following command assumes that the DescriptiveStatistics.dll file exists in the C:\temp folder.

```

CREATE ASSEMBLY DescriptiveStatistics
FROM 'C:\temp\DescriptiveStatistics.dll'
WITH PERMISSION_SET = SAFE;

```

After you import the assembly to your database, you can create the user-defined aggregates

with the CREATE AGGREGATE command. The following two commands create the skewness and the kurtosis UDAs:

```
-----
-- Skewness UDA
CREATE AGGREGATE dbo.Skew(@s float)
```

employeecountry	mean	standarddeviation	skewness	kurtosis
UK	665.538450	1151.18458127003	5.25006816518545	34.9789294559445
USA	615.269533	992.247411594278	7.56357332529894	88.6908404800727

```
RETURNS float
EXTERNAL NAME DescriptiveStatistics.Skew;
GO
-- Kurtosis UDA
CREATE AGGREGATE dbo.Kurt(@s float)
RETURNS float
EXTERNAL NAME DescriptiveStatistics.Kurt;
GO
-----
```

After you deploy the UDAs, you can use them in the same way as you do standard T-SQL aggregate functions. For example, the following query calculates the skewness and the kurtosis for the salesamount variable:

```
-----
SELECT dbo.Skew(salesamount) AS skewness,
       dbo.Kurt(salesamount) AS kurtosis
FROM dbo.SalesAnalysis;
-----
```

The result of the query is

skewness	kurtosis
6.78219499972001	68.7756494631014

Of course, now when you have the UDAs, you can use them to calculate the skewness and the kurtosis in groups as well. The following query calculates the four population moments for the salesamount continuous variable in groups of the employeecountry discrete variable:

```
-----
SELECT employeecountry,
       AVG(salesamount) AS mean,
       STDEV(salesamount) AS standarddeviation,
       dbo.Skew(salesamount) AS skewness,
       dbo.Kurt(salesamount) AS kurtosis
FROM dbo.SalesAnalysis
GROUP BY employeecountry;
-----
```

The result is

You can see from the higher values for the skewness and the kurtosis that the salesamount variable has a more important right tail for the USA employees. You can probably conclude that there are some big customers from the

USA that placed quite bigger orders than the average customers.

## Conclusion

In this article, you have learned again how important it is to know the mathematics behind the calculation. In addition, knowing how to write CLR aggregate functions and a CLR language like Visual C# is also very useful. With this knowledge, I was able to write aggregate functions that extend the T-SQL built-in functions set.

**Dejan Sarka**

MVP

[dsarka@solidq.com](mailto:dsarka@solidq.com)



## PASS SQL Saturday Slovenia 2015

Dogodek PASS SQL Saturday postaja tradicionalen. Letos bo seveda spet v soboto, tokrat 12. decembra, in sicer v prostorih FRI, Ljubljana (<http://www.sqlsaturday.com/460/eventhome.aspx>). Že predlani in predvsem lani se je izkazalo, da toliko uglednih mednarodnih predavateljev ne pride na nobeno drugo računalniško konferenco v Sloveniji. S seznama predlaganih predavanj (<http://www.sqlsaturday.com/460/Sessions/SubmittedSessions.aspx>) je že sedaj očitno, da se bo zgodba letos pono-

vila. Seveda poleg kakovosti ostaja enaka tudi cena za udeležbo, torej je udeležba brezplačna.

Spomnimo se, kako je to mogoče. Predavatelji nismo plačani, še več, stroške si krijemo sami, ob tem, da porabljamo svoj prosti čas. Sponzorji omogočijo predavalnice ter nekaj hrane in pijače. Nekaj denarja primakne PASS (<http://www.sqlpass.org/>), predvsem pa je vedno zraven velika podpora s strani Microsofta. PASS tudi vzdržuje spletno aplikacijo, ki je seveda pomemben del infrastrukture dogodka.

Ne nazadnje pa potrebujemo tudi vas, udeležence. Upamo, da bomo tudi tokrat ohranili lanskoletni nivo, ko smo dosegli, da je bilo razmerje med prisotnimi in prijavljenimi najboljše na svetu, ko je bil osip veliko pod svetovnim povprečjem.

Ne pozabite, da SQL Saturday ni namenjen le uveljavljenim predavateljem; dobrodošli so tudi začetniki, tudi tisti, ki bi radi stopili na pot predavanj. Če mislite, da lahko začnete delati kariero na tem področju, se registrirajte kot PASS uporabnik, prijavite na spletno stran našega dogodka (<http://www.sqlsaturday.com/460/callforspeakers.aspx>), ter pošljite predloge za predavanja. Svetujemo, da pošljete predloge in tudi predavate v angleškem jeziku. Le tako lahko računate tudi na morebitni prodor v tujino.

Poleg same konference tudi letos organiziramo dva predkonferenčna seminarja, ki bosta v petek, 11. decembra, v prostorih našega sponzorja Kompas Xnet-a (<http://www.kompas-xnet.si/>). Seminarja ne bosta brezplačna, bo pa cena zelo razumna. En seminar bo posvečen relacijskemu delu, en pa poslovnih inteligenci:

- Aaron Bertrand: 50 Things All SQL Server Developers Need to Know;
- Davide Mauri: Agile BI & Data Warehousing.

Celoten članek si lahko preberete na spletni strani [www.kompas-xnet.si/dogodki](http://www.kompas-xnet.si/dogodki)

## Kolofon

Kompas Xnet d.o.o., Stegne 7, Ljubljana  
 Telefon: 01 5136 990  
 Fax: 01 5136 999  
 Email: info@kompas-xnet.si  
 Web: <http://www.kompas-xnet.si>

### Direktorica

Branka Slinkar

### Urednica in oblikovalka

Urška Brus

### Člani uredništva

Aleš Lipušček, Anja Gruden, Boštjan Onhjec, David Čamdžić, Dejan Mauer, Dejan Sarka, Gašper Kamenšek, Klemen Vončina, Robert Vončina, Rok Bermež, dr. Tomaž Aljaž

Že petič MVP. Bravo!	Rok
Baby loading ..... juhuhu	Robi
Kjer je bil sam Napoleon :)	David
Mesec Amerike potem pa na projekte	Miha
Namesto na reggae kar na veselico	Boštjan
Lovil ptičke ali kilometre?	Gašper
Koledar, prijavnice, ponudbe, letaki, ...	Urška
»Hiking in Tenerife.... With schnaps«	Dejan S.
Magisterij – čisto blizu	Jošt
Seattle, prihajava ...	Rok & Robi
Že drugič fotr, čestitamo	Dejan M.
Na morju s Tevžem – neprecenljivo	Mojca
Kakšna kriza! Luksuz, nič računa :(	Anja
Mavričarji, tudi če je dež	Aleš



## Microsoft Partner

Gold Collaboration and Content  
Gold Devices and Deployment  
Silver Intelligent Systems  
Silver Application Development  
Silver Datacenter  
Silver Learning  
Silver Midmarket Solution Provider

**KOMPAS Xnet d.o.o.**

**1514 Ljubljana**

<http://www.kompas-xnet.si>

Poštnina  
plačana  
pri pošti  
1102 Ljubljana

**TISKOVINA**

