

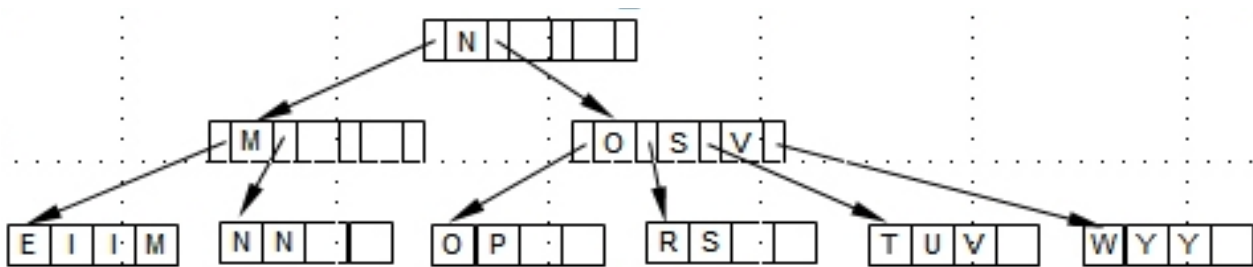
Indeksit

a) Mitä tarkoitetaan B-Tree:llä? Miksi se on tärkeä tietokannoille?

Vastaus:

B-puu on tietorakenne, joka soveltuu erittäin hyvin tiedon tallentamiseen tehokkaasti kiintolevyille. Se on tasapainotettu puumainen tietorakenne, jossa kaikkien haarojen korkeus on sama. Alla esimerkki miltä B-puu näyttää. Asiasta on kerrottu myös osoitteessa:

[http://technet.microsoft.com/en-us/library/ms189051\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms189051(v=sql.105).aspx).



b) Mitä tarkoitetaan clustered index? Mitä merkitystä sillä on käytännössä?

Vastaus:

Clusteroidulla indeksillä tarkoitetaan indeksia, joka määrää taulun tietorakenteen. Data rivit on siis tallennettu taulun tietorakenteeseen levyille clusteroidun indeksin määräämässä järjestyksessä. Sen vuoksi clustetorituja indeksejä voi olla vain yksi. Asiasta on kerrottu lisää osoitteessa:

[http://technet.microsoft.com/en-us/library/ms177443\(v=SQL.105\).aspx](http://technet.microsoft.com/en-us/library/ms177443(v=SQL.105).aspx).

c) Mitä tarkoitetaan non-clustered indeksillä?

Vastaus:

Non-clustered indeksi poikkeaa edellisestä clustered indeksistä kahdella tavalla. Non clustered indeksejä voi olla useita, koska kyseessä on looginen indeksi tietorakenne, jossa taulun data rivit eivät tarvitse olla tämän indeksin määräämässä järjestyksessä. Asiasta on kerrottu lisää osoitteessa:

[http://technet.microsoft.com/en-us/library/ms177484\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms177484(v=sql.105).aspx).

d) Mitä tietokannassa kannattaa indeksoida? Mitä asioita on otettava huomioon ja minkälaista työjärjestystä kannattaa käyttää?

Vastaus:

Yleensä tietokannoissa indeksoidaan oletuksena kaikki pääavaimet. Jotkut indeksoivat oletuksena myös ulkoiset viite-eheysavaimet (foreign keys). Ne kannattaa siis indeksoida, jollei niitä siis ole indeksoitu tietokannan toimesta. Erityisesti näin kannattaa tehdä jos tauluja liitetään yhteen select ... join lauseessa kyseisiä pääavaimia ja ulkoisia viite-eheysavaimia käyttäen. Jos lisäksi kyseisiä select lauseita käytetään paljon ja kyseisissä tauluissa on riittävän paljon datarivejä. Jos datarivejä on vähän, niin indekseistä ei ole hyötyä, sillä query optimizer voi päättää olla käyttämättä niitä. Indekskejä voi lisätä myöhemmin create index lauseella sitä mukaa kuin huomataan, että niistä olisi hyötyä tietokannan tuotannollisen ajon alkaessa. Asiasta on kerrottu lisää osoitteessa:

[http://technet.microsoft.com/en-us/library/ms179613\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/ms179613(v=sql.105).aspx).

- e) Minkälaisia indeksejä löytyy SQL Server:stä ja millaisessa tilanteessa niitä kannattaa käyttää? Vertaa löytyneitä indeksi tyyppejä MariaDB/MySQL:stä löytyviin.

Vastaus:

SQL Server:stä löytyy clustered, non-clustered, unique, index with included columns, full-text, spatial, filtered ja XML indeksit. Ne on esitelty tarkemmin SQL Server 2008 R2:n osalta osoitteessa: [http://msdn.microsoft.com/en-us/library/ms175049\(v=sql.105\)](http://msdn.microsoft.com/en-us/library/ms175049(v=sql.105)) ja SQL Server 2012:n osalta osoitteessa: <http://msdn.microsoft.com/en-us/library/ms175049>.

MySQL:n reference manual:in mukaan MySQL:ssä on B-Puu indeksejä ja Hash indeksejä. Myös Full-text indeksejä voi käyttää. Spatial indeksit ovat R-puu:n mukaisia tietorakenteita ja niitä voi käyttää vain MyISAM tietokannan yhteydessä. InnoDB:stä sitä ei siis löydy ja lisäksi kaikki indeksit ovat B-puu tietorakenteita (ks. <http://dev.mysql.com/doc/refman/5.6/en/column-indexes.html>). MyISAM tietokannoissa ei puolestaan voi käyttää foreign key viittauksia, jotka ovat mahdollisia InnoDB tietokannoissa. MySQL:n indekseistä on kerrottu osoitteessa: <http://dev.mysql.com/doc/refman/5.6/en/optimization-indexes.html>. Myös osoite <http://dev.mysql.com/doc/refman/5.6/en/optimization-indexes.html> on mielenkiintoinen, sillä siinä kerrotaan create index lauseen esittelyn yhteydessä miten kyseisiä indeksejä voi luoda.