Kako z uporabo modula Parallel pospešimo program na večjedrnem procesorju?

V .NET ogrodju za paralelizacijo uporabljamo TPL in PLINQ.

TPL(Task Parallel Library) je množica javnih objektov in API-jev znotraj imenskih prostorov System.Threading in System.Threading.Tasks. Razvijalcem omogoča večjo produktivnost, ker poenostavi uporabo vzporednega programiranja. TPL zna dinamično nastaviti nivo vzporednosti na dosegljivih procesorjih, vzpostavljanje niti z razredom ThreadPool in druge nizko-nivojske podrobnosti. V verziji .NET 4.0 je TPL priporočena knjižnica za vse asinhrone procese. Vseeno pa je treba dobro premisliti kdaj sploh uporabiti paralelizacijo, včasih je zaradi overhead-a asinhron program lahko tudi počasnejši.

PLINQ je verzija LINQ(Language Integrated Query) imenskega prostora za vzporedno programiranje. PLINQ poizvedbe pišemo skoraj identično kot LINQ, obratujejo na IEnumerable vmesniku, ki deluje na in-memory objektih. Glavna razlika je v tem, da PLINQ poskuša razdeliti delo na vse procesorje/jedra v sistemu. To doseže tako, da razdeli podatke podatke po delih, nato po vsakem delu izvaja poizvedbo znotraj ločene niti na posameznem procesorju.

PLINQ dobro deluje na podatkih, ki so v dovolj preprosti obliki. Če v poizvedbi dodamo "order by" stavek, bo program najprej zbral vse podatke in jih nato vrnil v pravilnem zaporedju, kar ne more izboljšati z ločenimi procesi. Sinhrono izvajanje kverijev v LINQ dosežemo s AsParallel ukazom, ki ga dodamo enumerable objektu na katerem izvajamo poizvedbe.