

عموما دو الگوی اصلی caching در برنامه‌ها وجود دارند: cache aside و cache trough .
در الگوی cache trough ، سیستم caching داخل DAL (که در اینجا همان NHibernate است)، تعبیه می‌شود؛ مانند سطح اول caching که پیشتر در مورد آن صحبت شد. در این حالت cache از دید سایر قسمت‌های برنامه مخفی است و DAL به صورت خودکار آن را مدیریت می‌کند.
در الگوی cache aside ، کار مدیریت سیستم caching دستی است و خارج از NHibernate قرار می‌گیرد و DAL هیچگونه اطلاعی از وجود آن ندارد. در این حالت لایه caching موظف است تا هنگام به روز شدن بانک اطلاعاتی، اطلاعات خود را نیز به روز نماید. این لایه عموماً توسط سایر شرکت‌ها یا گروه‌ها برنامه نویسی تهیه می‌شود.
NHibernate جهت سهولت کار با این نوع cache providers خارجی، نقاط تزریق ویژه‌ای را تدارک دیده است که به second level cache معروف است. هدف از second level cache فراهم آوردن دیدی کش شده از بانک اطلاعاتی است تا فراخوانی‌های کوئری‌ها به سرعت و بدون تماس با بانک اطلاعاتی صورت گیرد.
در حال حاضر (زمان نگارش این مطلب)، entity framework این لایه‌ی دوم caching یا به عبارتی دیگر، امکان تزریق ساده‌تر cache providers خارجی را به صورت توکار ارائه نمی‌دهد.
در NHibernate طول عمر second level cache در سطح session factory (یا به عبارتی طول عمر تمام برنامه) تعریف می‌شود و برخلاف سطح اولیه caching محدود به یک سشن نیست. در این حالت هر زمانیکه یک موجودیت به همراه ID منحصر بفرد آن تحت نظر NHibernate قرارگیرد و همچنین سطح دوم caching نیز فعال باشد، این موجودیت در تمام سشن‌های برنامه بدون نیاز به مراجعه به بانک اطلاعاتی در دسترس خواهد بود (بنابراین باید دقت داشت که هدف از این سیستم، کار سریعتر با اطلاعاتی است که سطح دسترسی عمومی دارند).

در ادامه لیستی از cache providers خارجی مهیا جهت استفاده در سطح دوم caching را ملاحظه می‌نمائید:

AppFabric Caching Services : بر اساس Microsoft's AppFabric Caching Services که یک پلتفرم caching محسوب می‌شود (+). (این پروژه پیشتر به نام Velocity معروف شده بود و قرار بود تنها برای ASP.NET ارائه شود که سیاست آن به گونه‌ای جامع‌تر تغییر کرده است)

MemCache : بر اساس سیستم معروف MemCached تهیه شده است (+).

NCache : (+)

ScaleOut : (+)

Prevalence : (+)

SysCache : بر اساس همان روش آشنای متداول در برنامه‌های ASP.NET به کمک System.Web.Caching.Cache کار می‌کند؛ یا به قولی همان IIS caching

SysCache2 : همانند SysCache است با این تفاوت که SQL dependencies ویژه SQL Server را نیز پشتیبانی می‌کند.

SharedCache : یک سیستم distributed caching نوشته شده برای دات نت است (+).

این موارد و پروایدها جزو پروژه‌ی nhcontrib در سایت سورس فورج هستند (+).

مطالب تکمیلی:

[مستندات NHibernate](#)

[توضیحات مفصلی در مورد سطح اول و دوم caching در NHibernate](#)

[مقایسه‌ای در مورد مبحث caching در EF و NHibernate](#)

[چگونه fluent NHibernate را جهت استفاده از سطح دوم caching تنظیم کنیم؟](#)
[توضیحات جامعی در مورد استفاده از SysCache](#)

نظرات خوانندگان

نویسنده: Ahmad

تاریخ: ۱۳۹۰/۰۱/۱۲ ۱۸:۲۷:۰۱

سلام

سطح دوم فقط برای وب استفاده میشه یا برای ویندوز اپلیکیشن هم کاربرد داره؟
درک کردن سطح دوم یه کمی سخته!!!

نویسنده: وحید نصیری

تاریخ: ۱۳۹۰/۰۱/۱۲ ۳۰:۰۶:۰۱

برای برنامه‌های تک کاربره شاید لزومی نداشته باشه چون هدف اصلی آن کم کردن بار سرور است
یک سری از موارد سایت عمومی است مثلاً آمار سایت حالا سوال اینجا است که اگر 10 هزار نفر هم اکنون به سایت مراجعه
کردند باید 10 هزار بار به بانک اطلاعاتی جهت دریافت یک سری اطلاعات با سطح دسترسی عمومی مراجعه کرد؟ اینجا است که
سطح دوم کش ذکر شده معنا پیدا می‌کند