

عنوان: به روز رسانی ارجاعات یک اسمبلی دارای امضای دیجیتال بدون کامپایل مجدد

نویسنده: وحید نصیری

تاریخ: ۱۳۹۰/۰۱/۱۰ ۰۹:۰۹:۰۰

آدرس: www.dotnettips.info

برچسب‌ها: NHibernate

سؤال: امروز NHibernate به روز شده اما Fluent NHibernate خیر! چکار باید کرد؟!

Fluent NHibernate کتابخانه‌ای است جهت رهایی برنامه نویس‌ها از نوشتن فایل‌های XML نگاشت کلاس‌ها به جداول به همراه قابلیت‌های دیگری مانند نگاشت خودکار و غیره. بنابراین این کتابخانه بدون NHibernate اصلی بدون کاربرد است. تیم توسعه آن هم با تیم اصلی NHibernate یکی نیست. عموماً NHibernate به روز می‌شود اما Fluent NHibernate ممکن است تا دو ماه بعد از آن هم به روز نشود. در این مواقع چه باید کرد؟

دو کار را می‌توان انجام داد:

الف) سورس Fluent NHibernate را دریافت کنیم و سپس ارجاعات قبلی به NHibernate قدیمی را حذف و ارجاعاتی را به اسمبلی‌های جدید آن اضافه و پروژه را کامپایل کنیم. Fluent NHibernate در طی این مدت به اندازه کافی رشد کرده و به قولی پخته است. کاری را هم که ادعا می‌کند به خوبی انجام می‌دهد. اما چون اسمبلی‌های اصلی NHibernate و همچنین Fluent NHibernate دارای امضای دیجیتال هستند، نمی‌توان از اسمبلی‌های جدید NHibernate به همراه Fluent NHibernate قدیمی استفاده کرد. خطای حاصل شبیه به عبارات ذیل خواهد بود:

```
System.IO.FileLoadException: Could not load file or assembly 'nameOfAssembly', Version=specificVersion, Culture=neutral, PublicKeyToken=publicKey' or one of it's dependencies. The located assembly's manifest definition does not match the assembly reference. (Exception from HRESULT: 0x80131040)
```

حذف ارجاعات به NHibernate قدیمی و افزودن مجدد ارجاعات به فایل‌های جدید و کامپایل نهایی پروژه یک راه حل است.

ب) راه حل دیگر استفاده از ویژگی [bindingRedirect](#) است بدون دریافت سورس، حذف و افزودن ارجاعات و کامپایل مجدد:

```
<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="NHibernate"
        publicKeyToken="aa95f207798dfdb4"
        culture="neutral" />
      <bindingRedirect oldVersion="3.0.0.4000"
        newVersion="3.1.0.4000"/>
    </dependentAssembly>
  </assemblyBinding>
</runtime>
```

در این مثال، پس از افزودن تعاریف فوق به فایل config برنامه، به سادگی می‌توان از اسمبلی اصلی NHibernate دارای نگارش 3.1.0.4000 به جای اسمبلی قدیمی‌تر 3.0.0.4000 آن استفاده کرد (همان نگارشی که Fluent NHibernate ما بر اساس آن کامپایل شده) و دیگر نیازی هم به کامپایل مجدد پروژه‌ای که از یک اسمبلی قدیمی Fluent NHibernate استفاده می‌کند، نخواهد بود.