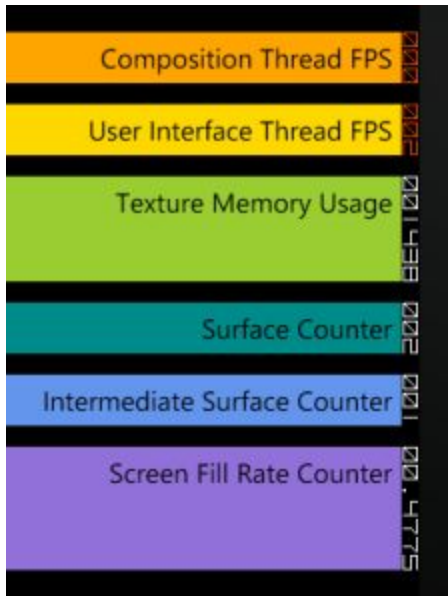


Frame Rate Counter kullanımı

Windows Phone ortamında geliştirilen uygulamaların performanslı çalışması çok önemlidir. Yazacağınız uygulamalar, en düşük sistem özelliği olan cihazlarda dahi hiçbir performans kaybı yaşanmadan çalışabilmelidir. Bunu, yazacağınız kodları optimize ederek sağlayabilirsiniz. Windows Phone SDK'sı bu noktada bazı kolaylıklar sunar.

Windows Phone Emulator ile birlikte gelen Frame Rate Counter, uygulamanızın o anki performans değerlendirmesini ekranda görüntüler. Uygulamanın her anında çeşitli performans testleri gerçekleştirerek, uygulamanızın performansı konusunda sürekli olarak bilgi sahibi olmanıza olanak sağlar. Bu testler aşağıda listelenmiştir:

- Composition Thread FPS
- User Interface Thread FPS
- Texture Memory Usage
- Surface Counter
- Intermediate Surface Counter
- Screen Fill Rate Counter



Uygulamanızdaki App.xaml.cs dosyasına girdiğinizde şu satırı görürsünüz:

```
Application.Current.Host.Settings.EnableFrameRateCounter = true;
```

Bu satır, Frame Rate Counter'ı yani yukarıdaki resimde bulunan sağa yaslanmış sayıları görüntülemenize olanak sağlar. Bu ifadeye false (yanlış) değerini atarsanız, uygulamanızın herhangi bir

noktasında bu performans testi çalışmaz ve görüntülenmez. Bu işlemi uygulamayı Release modunda derleyip Dev Center'a göndermeden önce yapmanız gerekir. Bunun dışında kalan durumlarda, uygulamanızın test işlemlerini gerçekleştirmek için true (doğru) değerinde tutulması işinize yarayacaktır. Testleri açıklamaya geçmeden önce, Composition Thread konusunu açıklamalıyız. Composition Thread, Silverlight 5 ile birlikte gelen, multi-threading işlemler yapmanıza olanak sağlayan ve bunu yaparken ayrı bir thread'de (composition thread) çalıştığından hiçbir şekilde UI Thread üzerinde işlem yapmayan çok kanallı programlama tekniğidir. Composition Thread olmasaydı, Windows Phone uygulamalarımız tek kanallı (Single-Threaded) çalışırdı. Bu da, arka planda uzun süren işlemlerin varsa UI Thread'in kilitlenmesine ve dolayısıyla, uygulamanızın istenmeyen bir şekilde kapanmasına veya telefonun performansının ciddi ölçüde düşmesine neden olurdu. Windows Phone üzerinde uygulama geliştirirken hiçbir şekilde UI Thread üzerinde işlem yaptırmamamız gerekir. Bu yüzden, Composition Thread bu tarz yoğun işlem gücü gerektiren işlerde kullanılır.

Aşağıda, bu performans testleri kısaca açıklanmıştır:

- **Composition Thread FPS:** Composition Thread'de çalışan ve bir saniyede gösterilen frame sayısını verir. Normal bir uygulamada önerilen değer 45 FPS'dir. Bu değer 30'un altına düşerse, performans optimizasyonlarına gereksinim duyulur. Uygulamayı ilk açtığınızda "0" değerini görürseniz endişelenmeyin. Bu değer, aslında Windows Phone uygulamasının o anda hiçbir işlem yapmadığını belirtir.
- **User Interface Thread FPS:** UI Thread üzerinde çalışan, bir saniyede gösterilen kare sayısını verir. Bu thread, Windows Phone arayüzüyle ilgili, kontrollerin yüklenmesi, stillerin verilmesi gibi işlemleri yapar. Ayrıca, olay yakalayıcılarının çalışması da bu thread üzerinde gerçekleştirilir. Bu thread, composition thread kadar büyük değildir. 15 FPS'in altına inmemesine özen gösterilmelidir. 30 FPS ideal bir değerdir.
- **Texture Memory Usage:** Bu, daha çok oyunlarda kullanılan dokuların cihaz üzerinde kapladığı alanı ifade eder. Herhangi bir önerilen değer veya kısıtlama sınırı mevcut değildir.
- **Surface Counter:** GPU üzerinde işlem yapılmak üzere cihaz tarafından ayrılan alana işaret eder. Daha çok oyun programlaması yapanlar için geçerlidir. Diğer bir deyişle, UI Thread'den Composition Thread'e aktarılan değeri ifade eder.
- **Intermediate Surface Counter:** Intermediate Surface Counter, Composition Thread tarafından Surface Counter'a ne kadar alan ayrılacağını belirler. Daha çok oyun programlaması yaparken kullanılır.
- **Screen Fill Rate Counter:** Ekrandaki nesneleri çizmek için saniye başına çizilen piksel sayısını ifade eder. 480x800 çözünürlüğü için bu değer 2.5'i geçmemelidir. 3 olduğu zaman kırmızı renkte gösterilir ve bu tehlikeli bir durumu, performansta gözle görülür bir kayıp yaşandığını belirtir. Örnek olarak, animasyonlar kullanılan bir uygulamada, bu değer sürekli değişecektir, çünkü her bir piksel her defasında yeniden çizilecektir. Bu nedenle, uygulamanın performanslı çalışmasını sağlayacak kodlar yazmaya özen gösterilmelidir. UI Thread'de çalışacak kontrollerin

sayısının azaltılması, boyutlarının küçültülmesi, gereksiz imaj kullanımından kaçınılması performans artışı sağlar.

Elbette, bu testleri Windows Phone Emulator üzerinde yaptığınızda, [Hyper-V](#) gibi bir sanallaştırma ürünü kullanıldığından, test işlemleri normalden hızlı gerçekleşecektir. Asıl test işlemleri, Windows Phone 8 cihazları üzerinde yapılmalıdır.