

# MESSAGE STACK

“A new method for Interprocess Communication /  
Süreçlerarası İletişim için yeni bir metod”

Murat Balaban

([murat@enderunix.org](mailto:murat@enderunix.org))

IPC, ingilizcede Interprocess Communication (süreçlerarası iletişim) kelimelerinin kısaltmasıdır. Geleneksel olarak bu terim, aynı makinada çalışan farklı süreçler arasındaki mesaj alışverişini tarif eder.

Geçen 40 yılda UNIX ve türevi işletim sistemlerinin gelişim sürecinde mesaj alışverişi için bazı mekanizmalar icat edildi ve kullanıldı. Bunlar Pipe'lar, Named Pipe'lar (FIFO), Shared Memory ve SYSV/Posix message queue'lar olarak sıralanabilir.

Kısaca üzerinden geçecek olursak, Pipe'lar yaygın olarak kullanılan ilk iletişim biçimiydi. Pipe'ların yalnızca anne-çocuk prosesler arasında kullanılabilmesi, Named Pipe olarak da isimlendirilen FIFO'ların geliştirilmesine neden oldu.

1980'lerin başında ise System V çekirdeklere SYSV message queue'lar olarak adlandırılan IPC mekanizması eklendi. Bunlar Pipe'lardan farklı olarak anne-çocuk ilişkisi olmayan prosesler arasında da kullanılabilmekte, FIFO'lardan farklı olarak da stack mantığı içinde, gelen mesajları belirli bir üst sınıra kadar saklayabilmekteydi.

Günümüzde SYSV message queue'lar çoğu ticari uygulamada hem süreçler arasında haberleşmeyi sağlamak için, hem de mesaj tampon (buffer) bölgesi olarak kullanılmaktadır. UNIX ve türevi sistemlerde birden fazla girdi çıktı aygıtını bazı olaylara (event) karşı kontrol etmek için (poll), **select** sistem çağrısı kullanılmaktadır. Böylece teker teker mevcut bütün aygıtları kontrol etmek yerine, ki aygıtlarda veri olmaması durumunda okuma yazma bütün prosesi bloke edebilir, bir çağrıyla bütün aygıtlarla işimizi görmüş oluruz. SYSV mesaj queue'ların en önemli problemi, mimari bazı engellerle bu şekilde bir yapıyı desteklememesi, bir prosesin birden fazla SYSV message queue'yu multiplex edememesidir.

Çalıştığım şirket bünyesinde geliştirdiğimiz programlarda SYSV message queue'ları yoğunlukla kullanmaktayız. Soket bağlantılarını ve message queue'ları kontrol (poll) eden proseslerimiz bu bakımdan çok büyük performans sınırlamalarıyla karşılaşmaktalar.

Bu projemde önerdiğim metod, message stack, Pipe'ların ve FIFO'ların sağladığı I/O multiplexing (select) özelliğini ve message queue'ların sağladığı stack veri depolamaolanağını sunmakla beraber, Pipe ve FIFO'ların sınırlı mesaj tamponlayabilme (buffer) sınırlamasını ve message queue'ların bahsettiğimiz sorununu barındırmayacaktır. Proje hem çekirdek seviyesine hem de kullanıcı seviyesinde (userspace) çalışmalar içermektedir. Ticari hiçbir UNIX sağlayıcısı kernel kodunu modifiye etmek için vermemektedir. Bu nedenle FreeBSD, Linux gibi her satır koduna istediğimiz ölçüde müdahalede bulunabileceğimiz işletim sistemlerinin olması bizim için büyük bir şanstır.