

Taslak

utfeight

November 26, 2023

Contents

1 Giriş	1
1.1 Merge Geliştirme Aşamaları	2
1.1.1 Kaynak Kodu	2
1.2 Merge Kullanım Alanları	2
1.2.1 Test Temelli Yazılım Geliştirme - Test Driven Development (TDD)	2
1.2.2 Birden fazla bilgisayarın bir programı çalıştırması (Soket Tabanlı IPC - Socket Based IPC)	3
1.3 Merge Standartları	3
1.3.1 Dosya Uzantısı	3

Proje Ana Alanı : Yazılım

Proje Tematik Alanı: Robotik Kodlama

Proje Adı (Başlığı): Merge

[width=.9]../..../logo/logo-inkscape

1 Giriş

Gelişen teknoloji ile birlikte yazılım alanında kullanılan metotların karmaşıklaşması yazılan programların soyutlaştırılması (abstraction¹) yazılım geliştiriciliğinin en temel ve önemli noktasıdır.

Bu soyutlama basamakları sayesinde çoğu bilgisayar yazılımcısı alt seviyede (low level²) ne olduğuna vakıf olmasa bile bilgisayarlar için program geliştirebilmekte.

¹[https://en.wikipedia.org/wiki/Abstraction_\(computer_science\)](https://en.wikipedia.org/wiki/Abstraction_(computer_science))

²<https://www.bbc.co.uk/bitesize/guides/z4cck2p/revision/2>

Bu soyutlama evresinin göze çarpan en önemli elemanlarından birisi de programlama dilleridir. ³ Bu diller sayesinde makina seviyesinde ne olduğunu bilmeden makinalar ile kolay bir biçimde konuşabiliriz.

Ben de birden fazla programlama dilini bir arada kullanabilmek için merge meta programlama dilini⁴ geliştirdim. Ana hedefim basit bir sözdizimi (syntax [[https://en.wikipedia.org/wiki/Syntax_\(programming_languages\)](https://en.wikipedia.org/wiki/Syntax_(programming_languages))]) ile kolaylıkla kullanılabilen, mümkün olduğu kadar hızlı ve basit bir programlama dili yazmaktı. [TODO: fix these grammatical mistakes]

Programlar arasında yapılacak olan veri paylaşımını mümkün olan en güvenli şekilde uygulayabilmek için günümüz modern işletim sistemlerinde var olan protokollerden yararlandım. Önceliklerimden birisi performans olduğundan genel adı Inter Process Communication Mekanizmalarından yararlandım (shared mem + file io for testing purposes)

1.1 Merge Geliştirme Aşamaları

1.1.1 Kaynak Kodu

Merge Programlama Dili Git proje organizasyon yazılımı ile Github bünyesinde geliştirilmiştir.

Merge Dilini geliştirirken özenle seçtiğimiz ana yazılım dili olan Rust'ın bize gelişim aşamasında sağladığı katkılardan bahsetmek isterim.

1. Derleme (Compilation)
2. Yüksek Seviye Sözdizimi, Yüksek Performans sistem programlama dilleri arasında oldukça yeni olan Rust, üst üste 8 yıl en çok istenilen programlama dili⁵ olarak öne çıkmıştır.

1.2 Merge Kullanım Alanları

1.2.1 Test Temelli Yazılım Geliştirme - Test Driven Development (TDD)

Yazılım geliştiricileri için önerilen alışkanlıklar arasında “birim testleri” (Unit Testing) ilk sıralarda bulunur. Kısaca programınızı birimlere bölerek her birimin sağlıklı çalıştığından ve istenilen sonuçları verdiğinden emin olmak için yazılımcı tarafından testler yapılmasıdır.

³https://en.wikipedia.org/wiki/Programming_language

⁴Merge programlama dilinin matematiksel kısmı diğer programlama dillerine dayanabilecek biçimde büyük çoğunluğu soyutlanmış biçimde tasarlanmıştır. Bu nedenle merge diğer programlama dillerinin üstünde (meta) bir programlama dilidir.

Bilgisayar bilimcileri arasında çok yaygın olan bu birim testlerinin geliştirilmesini kolaylaştırmak adına bir çok kütüphane yazılmıştır. Örnek olarak (C++ için) GTest ve Catch2 (Rust için) cargo test (Python için) PyUnit ve Pytest örnek verilebilir. [TODO: clean up]

1.2.2 Birden fazla bilgisayarın bir programı çalıştırması (Soket Tabanlı IPC - Socket Based IPC)

Bu senaryo, Merge programlama dilinin deneysel özelliklerinden birisi olup istisnai durumlar için faydalıdır.

1. Artı Yönleri Bazı donanımsal kısıtlamalardan dolayı yazılımların istenilen biçimde çalışmadığı durumda Soket Tabanlı IPC iletişim protokollerini kullanarak işlemci gücü, dahili hafıza birimlerinin kapasitesi, internet iletişim paralelleştirme⁶
2. Eksi Yönleri
Bilgisayarlar arası veri aktarımı, yeterli donanıma sahip olunmadığı zaman işleyiş süresi performansını (runtime performance) inanılmaz seviyede düşürebilir.

1.3 Merge Standartları

Yeni bir programlama dili geliştirmenin en zor yanlarından birisi olan sürdürülebilirliği (maintainability) sağlamak adına belirli standartlar belirlememiz gerekti. Başta programın formatı olmak üzere pek çok konu üzerinde belirlediğimiz standartlar sayesinde merge kullandığımız işletim sistemine bağlı olmadan çalışabilmektedir.

1.3.1 Dosya Uzantısı

Merge meta programlama dilinin özelliklerini kullanmanın iki farklı yolu var: Merge ile program yazmak, Yazdığımız program hakkında Merge Derleyicisine (compiler) bilgi vermek.

Eğer Merge programı yazmak istiyorsanız ‘.mg’ Eğer Merge Derleyicisini kullanıyorsanız ‘.mg.<dil>’ (‘<dil>’ yerine kullandığımız programlama dilinin uzantısını yazın. Örneğin Rust Programlama Dili için ‘.rs.mg’ uzantısını kullanmalısınız)

⁶birden fazla bilgisayar ile bir ya da birden çok erişim ağı ile bağlantı kurarak interneti maksimum etkiyle kullanabilmeyi sağlar