**Software Components:**

NVM-> Nonvolatile memory (güç kapatıldıktan sonra depolanan veriler kaybolmaz) HDD, SSD, CD, DVD, ROM

Autosar içindeki Application Software, Atomik Software Türleri adı verilen bağımsız birimlerde düzenlenir.

Bu tür Atomik Software bileşenleri birlikte yazılımın eksiksiz işlevsel uygulamasını oluşturur.

**ASW:**

Application Software Component: Ağırlıklı olarak kullanılan software component türüdür.Software in çeşitli functionalitylerini tutar. Örnek: Calculations, Functional/Decision making alghorithms…

NVBlock Software Component: Kritik bir bilgiyi memoryde tutmaya gerek olduğunda kullanılır NV -> Non Volatile(Uçucu olmayan) Bu basic softwaredaki NVM manager a erişir. Uçucu olmayanın anlamı sen gücü kapattığında ve tekrar açtığında kaydettiğin veriler duruyorsa NVM e kaydetmişsindir.

Örnek: Bir aracın gittiği toplam yolu bir yere kaydetmen gerekiyor diye düşün bunu uçucuya kaydedersen araç kapanıp açılınca gider fakat NVM e kaydedersen bu güç kapanıp açılsa dahi orda durur.

Burda NVM managera SWC miz eriştikten sonra NVM manager veriyi non volatile olan bir yere kaydeder.

Complex Device Driver Software Component: Bu hardware e direkt erişim için de kullanılabilir. Time critic operasyonlarda bunu kullanmak mantıklı olabilir çünkü direkt hardware e erişimi vardır. Autosar ın desteklemediği işlemler için de bu kullanılabilir.

Service Software Component: Belirli bir kontrol ünitesi için hizmetleri yapılandırma için kullanılır

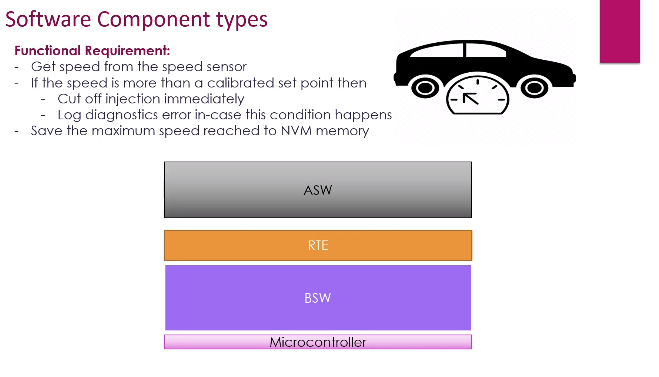
Service Proxy Software Component: Belirli bir hizmet bileşenine farklı kontrol birimlerinden erişileceği zaman kullanılır.

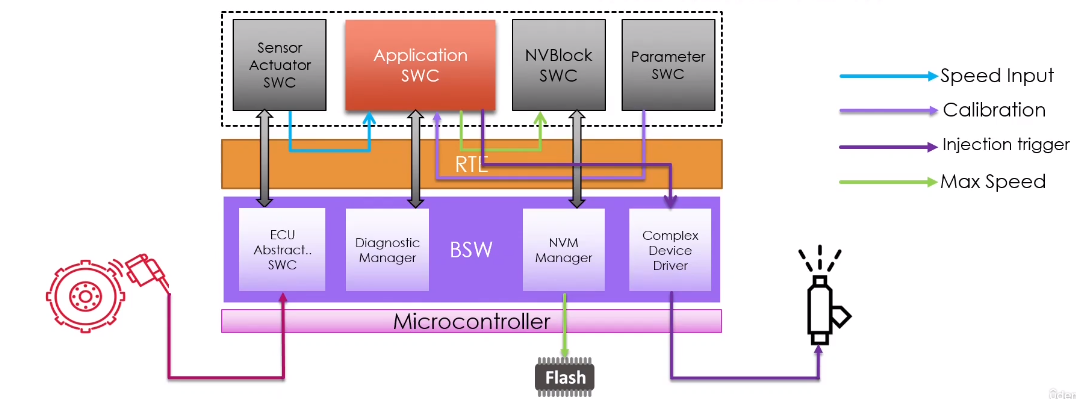
ECU Abstraction Software Component: BSW’ nin bir parçasıdır. MCAL ile Sensor Actuator bileşeni arasında interface gibi davranır.

Sensor Actuator Software Component: Ecu Abs Layer ile etkileşim kurmak için Application Layer da kullanılır.

Parameter Software Component:

Compositions Software Component:





**Ports:**

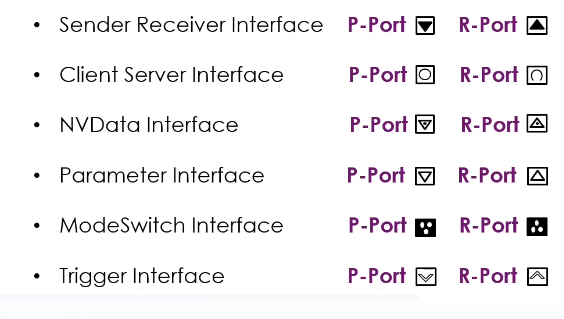
Autosar mimarisi, autosar modülleri arasındaki iletişim modu olarak portları önerir.

Provider Port (P-Port): Bir SW-C (Software Component) den gönderilmesi gereken bir bilgi olduğunda kullanılır.

Receiver Port (R-Port): Diğer SW-C lerden bazı bilgiler almamız gerektiğinde kullanılır.

ProviderReceiver Port (PR-Port): Simultane olarak bir SW-C den veri alma ve yollama için kullanılır.

**metin içeren bir resim

Açıklama otomatik olarak oluşturulduPort Interfaces:**

**SR Interface:** SW-C lar arası iletişim kurmak

için kullanılır. Projelerde majör olarak kullanılan

metin içeren bir resim

Açıklama otomatik olarak oluşturulduinterfacelerdendir.

**CS Interface:** Başka SW-C lerden servisler veya

fonksiyonlar çağırmak için kullanılır.Projelerde

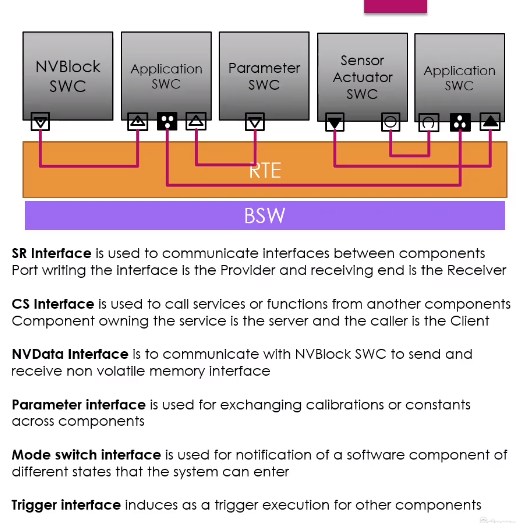
majör olarak kullanılan interfacelerdendir.

**NVData Interface:** NVBlock SWC ile iletişim kurmak için kullanılır.

**Parameter Interface:** Bileşenler arasında kalibrasyonlar veya sabit alışverişi için kullanılır.

**ModeSwitch Interface:** Sistemin girebileceği farklı durumların bir yazılım bileşenine bildirimi için kullanılır.

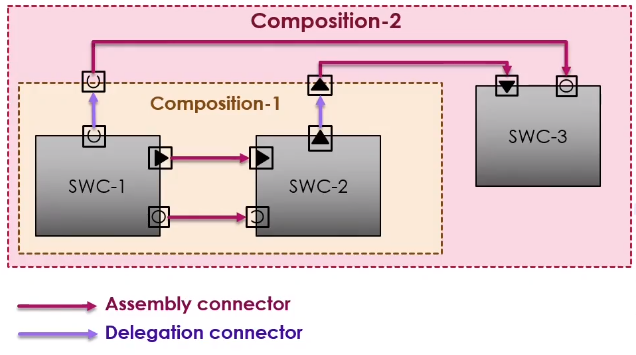
**Trigger Interface:** Diğer bileşenler için bir tetikleyici yürütme olarak indükler. Belli triggerlarda gerçekleşmesi gereken bir event olduğunda kullanılır.



**Compositions:**

Composition diğer Software Componentları kümeleyen veya toplayan Software Component türüdür.A container that holds a group of software components.

**Connectors:**

Receiver ve Providerların hangisinin hangisiyle bağlanacağı connector configuration da belirlenir.

**Assembly Connector:** Aynı composition içindeki portları bağlamak için

kullanılır.“Same parent”

**Delegation Connector:** İç bağlantı noktasını bileşenlerden dış bileşime

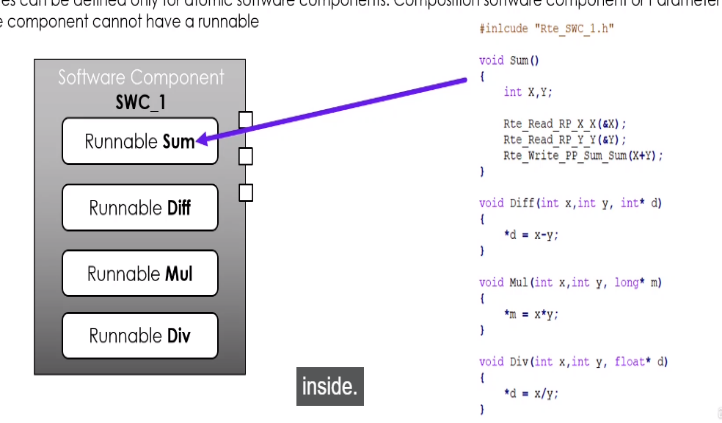
uzatır

**Pass Through Connector**

**Runnables:**

Runnable Entities, bileşen tarafından sağlanan en küçük kod parçalarıdır.C dosyalarındaki her fonksiyon autosar yapılandırılmasında runnable olarak tanımlanmalıdır.

Composition Software Component veya Parameter Software Component “Runnable” içeremez.



**tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Events:**

Eventler, runnable bir dosyanın nasıl çağrılacağını veya zamanlanacağını işletim sistemine belirten, runnable için ek yapılandırmadır.

**General Events**

**Init Event:** Runnable’ı yalnızca bir kez çağırmak içindir ve runnable yalnızca başlatma sırasında yazılımın başlatılması içindir.

**Timing Event:** Bir zamanlama periyodunda Runnableları çağırmak için kullanılır. 100 ms de bir kullanılan bir runnable için mesela.

**Internal Trigger Occured Event:** Internal bir triggerlanma olayı olduğunda runnable çalışır.

**External Trigger Occured Event:** External bir triggerlanma olayı olduğunda runnable çalışır.

**Background Event:** Arkaplanda çalışması gereken runnablelar için kullanılır.

**Client Server Events:**

**Operation Invoked Events:** Server side.

**Asynchronous Server Call Result Event:** Asynchronous server call bittiğinde çağırılır.

**Data Related Events:**

**Data Write Complete Event**

**Data Send Complete Event**

**Data Receive Event**

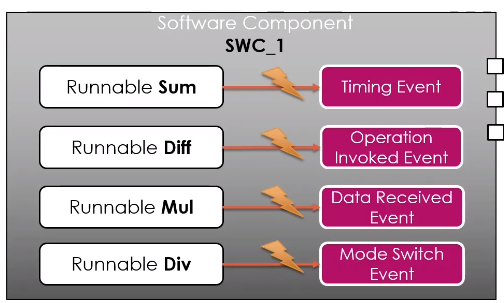
**Data Receive Error Event**

**Mode Events:**

**Mode Switch Event**

**Mode Manager Error Event**

**Mode Switch Ack Event**

****

**RTE**

Software Component’lar arası iletişimi sağlar.

Mesaj tutarlılığını sağlar.

Scheduling of runnables.

**Autosar Methodology:**

