DataRuler.NET

E-Beyanname Node (Yeni e-Beyan / REST)

Kurulum ve Kullanım Kılavuzu

Kaynak veriyi GİB KDV1 JSON formatına eşleyen DataRuler çıktı düğümü

Sürüm 1.0 · Kasım 2026

skalabilisim.com

İçindekiler

1. Giriş · · s. 3

2. Mimari ve Çalışma Mantığı · · s. 3

3. Kurulum (Derleme ve Yükleme) · · s. 4

4. Çözümün Kurulması · · s. 5

5. Node Ayarları (API) · · s. 5

6. Tablo Eşleştirmeleri · · s. 6

7. Değer ve Biçim Kuralları · · s. 7

8. Boş Tablo Davranışı · · s. 7

9. Çalıştırma ve Log Okuma · · s. 7

10. KDV1 Tablo Referansı · · s. 8

11. Uçtan Uca Örnek · · s. 9

12. Gerçek Gönderim (Sonraki Aşama) · · s. 10

13. Sık Karşılaşılan Durumlar · · s. 10

14. İletişim · · s. 11

1. Giriş

Bu kılavuz, DataRuler.NET için geliştirilen E-Beyanname Node (Yeni e-Beyan / REST) düğümünün kurulumunu ve kullanımını adım adım anlatır. Düğüm, Gelir İdaresi Başkanlığı'nın (GİB) yeni e-Beyan sisteminin REST tabanlı veri modeline uygun şekilde, kaynak verinizi doğrudan KDV1 JSON gövdelerine dönüştürür.

Eski yöntemden (Java/BDP gömme, Internet Explorer otomasyonu, zip paketleme) tamamen ayrılır. Yerine; kaynak veriyi GİB tablolarına eşleyen, ara ürün olarak JSON üreten ve gönderimi ayrı bir işlem adımı olarak ele alan sade bir mimari kurar.

Kime hitap eder

DataRuler ile çözüm geliştiren, KDV beyannamesi verilerini hazırlayan teknik kullanıcılara yöneliktir. Temel DataRuler kavramlarına (repository, node, Command Organizer, node bağlama) aşina olunduğu varsayılır.

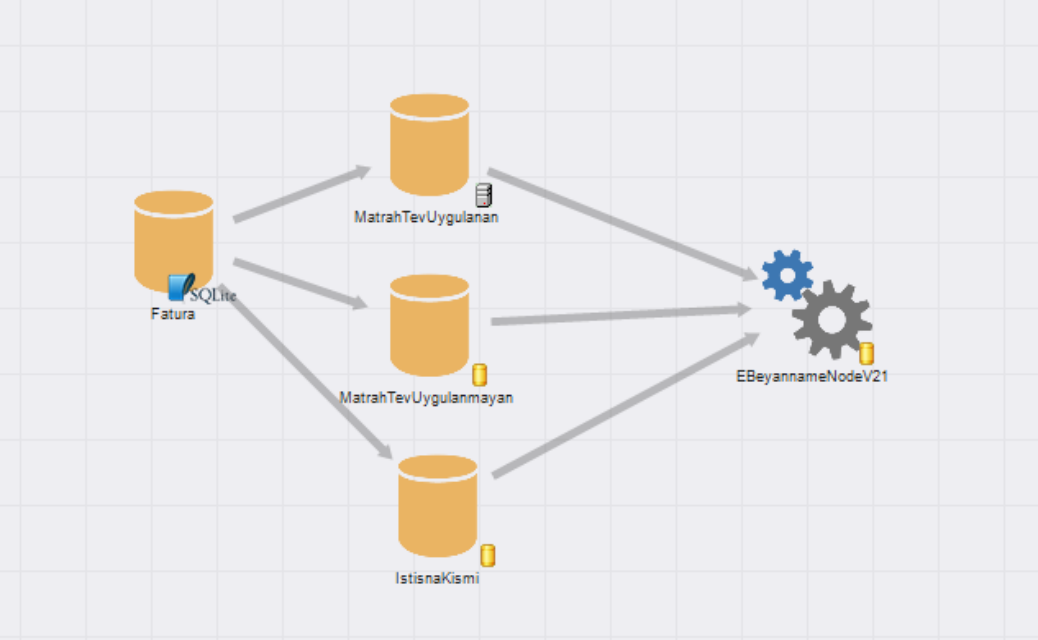
2. Mimari ve Çalışma Mantığı

Mimarinin temeli, sorumlulukların ayrılmasıdır: veriyi hazırlamak üst düğümlerin, onu GİB formatına çevirmek ise E-Beyanname düğümünün işidir. Böylece GİB'in şeması değişse bile yalnızca eşleştirme güncellenir; veri hazırlama akışına dokunulmaz.

İki katman

- Veri hazırlama (üst düğümler): Her GİB tablosu için ayrı bir kaynak/işlem düğümü, o tablonun satırlarını üretir (örn. matrah, indirim, istisna). Gruplama, filtre, hesaplama burada yapılır.
- Eşleştirme ve üretim (E-Beyanname düğümü): Bağlı kaynak tabloları, GİB tablolarına ve alanlarına eşler; doğru formatta JSON üretir.

Önemli: Tek bir E-Beyanname düğümü, beyannamenin tamamını üretir. Her bölüm için ayrı düğüm gerekmez; bölümler düğüm içindeki 'Tablo Eşleştirmeleri' koleksiyonunda birer satır olarak tanımlanır.



Örnek yapı: kaynak düğümler (bölüm başına bir tane) tek E-Beyanname düğümüne bağlanır.

İşlem adımları (komutlar)

Düğüm, Command Organizer'da dört ayrı komut sunar:

Komut	Görevi
1) Beyanname Hazırla	Kaynak veriyi eşleştirip GİB KDV1 JSON gövdelerini üretir (ara ürün).
2) Beyanname Gönder	Üretilen JSON'u GİB REST API'ye gönderir (entegratör erişimi ile).
3) Beyanname Onayla	Gönderilen taslağı onaylar.
4) Durum / Tahakkuk	Durum sorgular, tahakkuk/PDF alır.

Bu kılavuz ağırlıklı olarak 1) Beyanname Hazırla adımını (veri → doğru JSON) kapsar; gerçek gönderim için gerekenler Bölüm 12'de özetlenir.

3. Kurulum (Derleme ve Yükleme)

Düğüm, bağımsız bir proje olarak derlenir ve tek bir DLL (EBeyannameV2Node.dll) olarak DataRuler'ın Nodes klasörüne kopyalanır. DataRuler'ın hiçbir dosyası değiştirilmez.

3.1 Derleme

```
cd C:\DRSuite\Development\Tools\Build
powershell -ExecutionPolicy Bypass -File .\Derle_EBeyannameNodeV2.ps1
```

Başarılı olunca yeşil 'DERLEME BASARILI: ...EBeyannameV2Node.dll (NN KB)' satırı görünür. Hata olursa kırmızı 'DERLEME BASARISIZ' yazıp ilk 20 hatayı listeler; tam log Build klasöründeki derleme.log'dadır.

3.2 Yükleme (test kopyasına)

Yüklemeden önce o kurulumdaki DataRuler kapalı olmalıdır (DLL kilitlenmesin).

```
cd C:\DRSuite\Development\Packages\EBeyannameNodeV2
powershell -ExecutionPolicy Bypass -File .\Kur.ps1
```

'Kuruldu: C:\DRSuite\Test\DataRuler\Nodes\EBeyannameV2Node.dll' satırı görününce kurulum tamamdır. Geri almak için GeriAl.ps1 kullanılır. Asıl kuruluma geçiş yalnızca test bittikten sonra: Kur.ps1 -DataRulerKlasoru "C:\DRSuite\DataRuler".

Not: PowerShell komutunu kopyalarken başındaki 'PS C:\...>' istemini DAHİL ETMEYİN; yalnızca 'powershell ...' ile başlayan kısmı yapıştırın.

4. Çözümün Kurulması

DataRuler'ı test kopyasından açın. Düğüm, Tools penceresinde Output grubunda 'E-Beyanname (Yeni e-Beyan)' adıyla görünür.

1. Her GİB tablosu için bir kaynak/işlem düğümü hazırlayın (matrah, indirim, istisna ...). Her biri kendi çıktı tablosunu üretsin.
2. E-Beyanname düğümünü çözüme ekleyin.
3. Kaynak düğümleri E-Beyanname düğümüne 'Add Link' ile bağlayın.

Çalışma anında, her bağlı düğümün çıktısı E-Beyanname düğümüne ayrı bir tablo olarak gelir; tablo adı, kaynak düğümün çıktı tablosu adıdır.

5. Node Ayarları (API)

Düğüme tıklayıp Properties panelindeki 'API Ayarları' bölümünü doldurun:

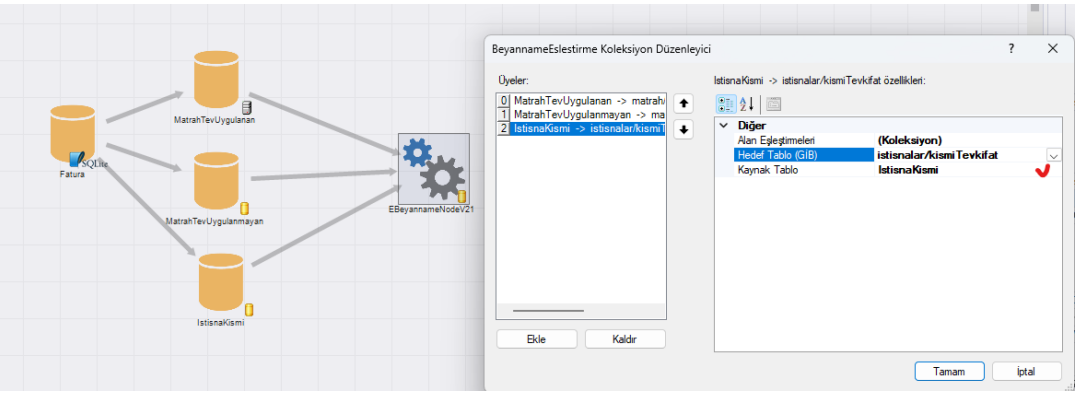
Ayar	Açıklama
Ortam	PreProd (test) veya Prod (canlı). İstemci adresini belirler.
API Token (Bearer)	GİB e-Beyan 'Entegrasyon Yönetimi'nden alınan kimlik anahtarı.
Mükellef VKN	Beyannamenin ait olduğu mükellefin VKN'si.
Entegratör (VKN-Ünvan)	Entegratör firma bilgisi; alfanumerik, Türkçe karaktersiz, en çok 20 hane.
Beyanname Türü	KDV1 (ileride KDV2, DAMGA vb.).

Dönem (YYYY/AA)	Beyanname dönemi.

Token hassas bir bilgidir; paylaşmayın ve gerektiğinde portalden yenileyin.

6. Tablo Eşleştirmeleri

Beyannamenin tüm bölümleri, Properties'teki 'Eşleştirme (Çoklu Tablo)' kategorisinde yer alan 'Tablo Eşleştirmeleri' koleksiyonunda tanımlanır. Yanındaki '...' düğmesi standart koleksiyon editörünü açar.



Koleksiyon editörü: her üye bir kaynak tabloyu bir GİB tablosuna eşler.

'Ekle' ile istediğiniz kadar üye eklersiniz. Her üyede üç alan vardır:

Alan	Ne yazılır
Hedef Tablo (GİB)	Açılır listeden GİB KDV1 tablosu. Seçince alanlar otomatik dolar.
Kaynak Tablo	Bağlı kaynağın tablo adı. (Boş bırakılırsa ilk tablo kullanılır.)
Alan Eşleştirmeleri	Her satır alanı için kural. '...' ile düzenlenir.

6.1 Alan kuralı: sabit mi, kolon mu?

Her alanın 'Değer'i iki biçimde yazılır:

- Başında '=' VARSA → sabit değer. Örn: =KDV1_1100, =0.2
- Başında '=' YOKSA → kaynak kolon adı; değer o kolondan satır satır gelir. Örn: NET_TUTAR

6.2 Kolaylıklar

- Hedef Tablo seçilince, o tablonun alanları otomatik listelenir; siz yalnızca 'Değer'leri doldurursunuz. Hedef tabloyu değiştirirseniz (henüz değer girmediyseniz) alanlar yenilenir.

- Kaynak Tablo açılır listesi, '1) Beyanname Hazırla' bir kez çalıştıktan sonra gerçek tablo adlarıyla dolar. (Tasarım anında düğüm bağlı tabloları göremediği için ilk seferde elle yazılabilir.)

7. Değer ve Biçim Kuralları

GİB KDV1 modeli, alanların biçimini kesin belirler. Düğüm bunları otomatik uygular:

Alan tipi	Biçim	Örnek
Para alanları	İki ondalıklı metin (string)	"276075.00"
Oran (kdvOrani, tevkifatOrani)	Sayı; KESİR, yüzde değil	0.2 (yani %20)
Tür/kod alanları	Referans kod (metin)	"KDV1_1100", "KDV1_108"

7.1 Ondalık ayraç

Sayısal kolonlar (Double vb.) metne çevrilmeden doğrudan işlenir; böylece Türkçe sistemdeki virgül ondalığı ile İngilizce nokta ondalığı karışmaz. Örneğin 41808,46 değeri "41808.46" olarak doğru üretilir.

7.2 Oran: kesir kuralı

kdvOrani 0.01 / 0.10 / 0.20 gibi KESİR beklenir. Kaynağınız oranı yüzde olarak (ör. 20) tutuyorsa, sorguda 100'e bölün (KDV_ORAN/100.0) ya da ilgili alanı sabit (=0.2) tanımlayın.

8. Boş Tablo Davranışı

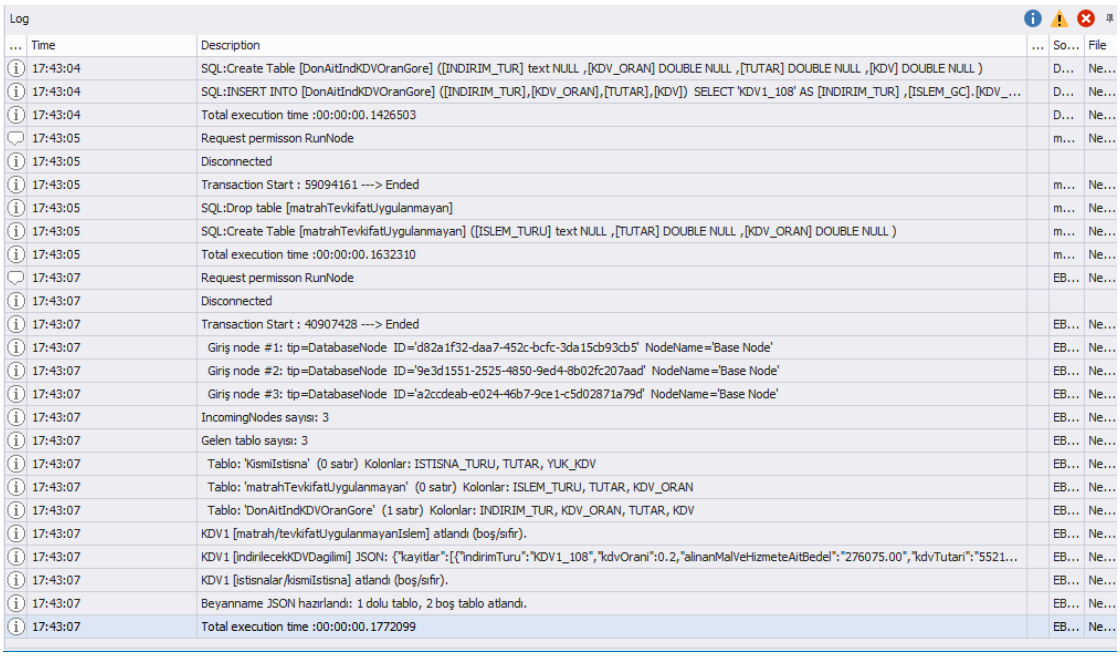
Beyannameye boş/sıfır bölüm gönderilmez. Düğüm, bir tabloyu yalnızca anlamlı veri içeriyorsa üretir:

- Kaynak tabloda hiç satır yoksa → üretilmez.
- Tüm para alanları 0/boş ise → üretilmez.
- En az bir satırda bir para alanı sıfırdan farklıysa → üretilir.

Log'da her tablo için ya JSON ya da 'atlandı (boş/sıfır)' satırı görünür; sonda 'N dolu tablo, M boş tablo atlandı' özeti yazılır.

9. Çalıştırma ve Log Okuma

E-Beyanname düğümüne sağ tıklayıp '1) Beyanname Hazırla' komutunu çalıştırın. Log panelinde tanılama ve üretilen JSON görünür.



Örnek Log: bir dolu tablo üretildi, iki boş tablo atlandı.

Log'da göreceğiniz başlıca satırlar: gelen tablo sayısı ve adları/kolonları; her hedef tablo için üretilen JSON ya da 'atlandı'; en sonda dolu/boş özeti. Üretilen JSON, düğümün 'SonUretilenJson' alanında da saklanır (Gönder adımı bunu kullanır).

10. KDV1 Tablo Referansı

Eşleştirmede kullanılan başlıca GİB KDV1 tabloları ve ekleme sırasında gönderilen alanları:

Hedef Tablo (yol)	Alanlar
matrah/tevkifatUygulanmayanIslem	islemTuru, matrah, kdvOrani
matrah/kismiTevkifatUygulananIslem	islemTuru, matrah, kdvOrani, tevkifatOrani
matrah/digerIslemler	islemTuru, matrah, kdvOrani
digerIndirimler	indirimTuru, indirilecekKdvTutari
indirilecekKDVDagilimi	indirimTuru, kdvOrani, alinanMalVeHizmeteAitBedel, kdvTutari
istisnalar/kismiIstisna	istisnaTuru, teslimVeHizmetTutari, yuklenilenKdv
istisnalar/tamIstisna	istisnaTuru, teslimVeHizmetTutari, malBedeli, yuklenilenKdv
istisnalar/kismiTevkifat	islemTuru, teslimVeHizmetTutari, iadeyeKonuOlanKdv

Not: 'vergi' gibi sunucunun hesapladığı alanlar eklemede GÖNDERİLMEZ. Tür/kod değerleri referans kod olmalıdır (örn. işlem türü KDV1_1100, indirim türü KDV1_108). Tam liste için portaldaki ilgili açılır listelerin kodları esas alınır.

11. Uçtan Uca Örnek

'İndirilecek KDV Tutarının Oranlara Göre Dağılımı' bölümünü örnekleyelim. Kaynak düğüm (örn. DonAitIndKDVOranGore) şu kolonları üretsin: INDIRIM_TUR, KDV_ORAN, TUTAR, KDV.

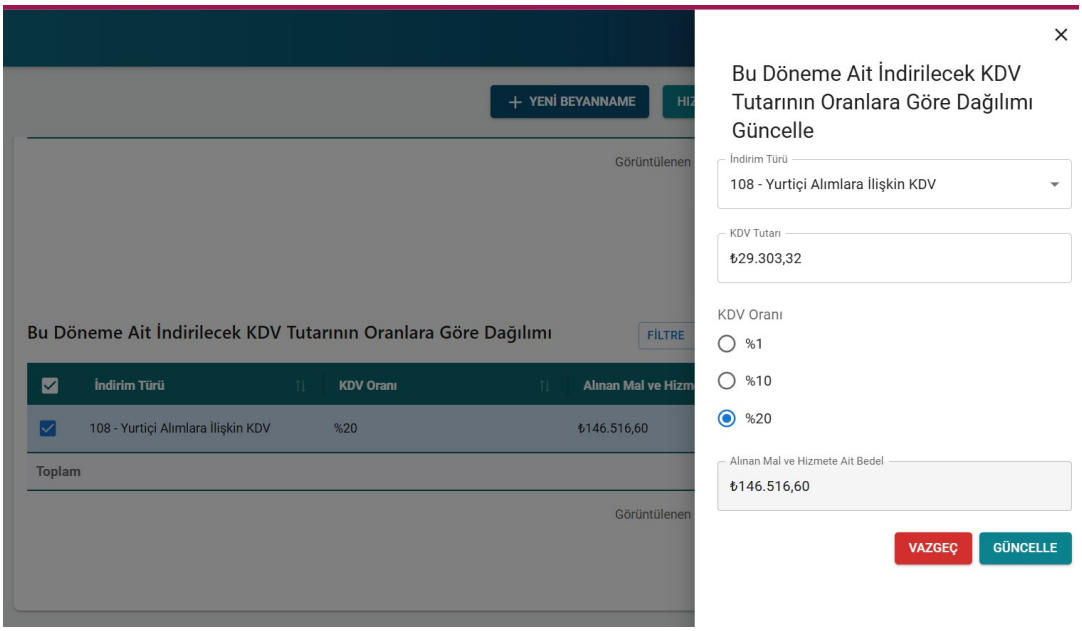
Eşleştirme

```
Hedef Tablo : indirilecekKDVDagilimi
Kaynak Tablo: DonAitIndKDVOranGore
Alanlar:
   indirimTuru               = =KDV1_108   (sabit)  ya da kod kolonu
   kdvOrani                  = KDV_ORAN
   alinanMalVeHizmeteAitBedel = TUTAR
   kdvTutari                 = KDV
```

Üretilen JSON

```
{"kayitlar":[{
   "indirimTuru":"KDV1_108",
   "kdvOrani":0.2,
   "alinanMalVeHizmeteAitBedel":"276075.00",
   "kdvTutari":"55215.00"
}],"beyannameId":"<GIB-taslak-id>"}
```

GİB formundaki alanlarla birebir örtüşür: İndirim Türü → indirimTuru (kod), KDV Oranı → kdvOrani, Alınan Mal ve Hizmete Ait Bedel → alinanMalVeHizmeteAitBedel, KDV Tutarı → kdvTutari.



GİB e-Beyan formu: alan adları düğümdeki eşleştirmeyle örtüşür.

12. Gerçek Gönderim (Sonraki Aşama)

Veri hazırlama tarafı tamamlanmıştır: düğüm, gerçek verinizden doğru formatta KDV1 JSON üretir. Beyannameyi GİB'e fiilen göndermek için gerekenler:

1. Entegrasyon Başvurusu: Dijital Vergi Dairesi üzerinden 'e-Beyan Entegrasyon Başvurusu' (idari adım).
2. Ortam Testleri: PreProd/pilot API üzerinde entegrasyon ve yük testleri.
3. API Key (Token): e-Beyan 'Entegrasyon Yönetimi' menüsünden alınır; düğümdeki 'API Token' alanına girilir.
4. Canlı Gönderim: ebeyanapi.gib.gov.tr üzerinde HTTPS/Bearer ile gönderim.

Düğümün transport katmanı (HTTPS, Bearer token, CHANNEL/MUKELLEF-VKN/ENTEGRATOR başlıkları, PreProd ↔ Prod geçişi) hazırdır. Tamamlanması gereken tek şey, entegratör REST referansındaki (Swagger) GERÇEK endpoint yollarının Gönder/Onayla/Durum komutlarına girilmesi ve PreProd'da ilk gönderim testidir.

13. Sık Karşılaşılan Durumlar

Belirti	Olası neden / çözüm
kayitlar:[] (boş)	Kaynak tabloda satır yok ya da para alanları 0. Kaynak sorguyu kontrol edin.
Alan değerleri boş	Alan 'Değer'leri doldurulmamış. '...' ile kolon adı/sabit yazın.
kdvOrani 20 geldi	Oran yüzde olarak gelmiş; KESİR olmalı (0.2). Kaynakta

	/100 veya =0.2.
Tür alanı metin geldi	Kod yerine açıklama yazılmış. Referans kodu kullanın (KDV1_108).
Kaynak Tablo listesi boş	Henüz çalıştırılmadı. '1) Beyanname Hazırla'yı bir kez çalıştırın.
Yeni ayar görünmüyor	DLL güncellenmemiş. Derle + Kur yapıp DataRuler'ı yeniden açın.

14. İletişim

Skala Bilişim — İşletme Çözümleri & Danışmanlık

E-posta: info@skalabilisim.com

Web: skalabilisim.com

Bu kılavuz DataRuler.NET E-Beyanname Node (Yeni e-Beyan / REST) içindir. © 2026 Skala Bilişim.