

# UniRak ESG -net- A

Mischfonds - ISIN: LU0718558728



## WERTENTWICKLUNG IN DER VERGANGENHEIT\*

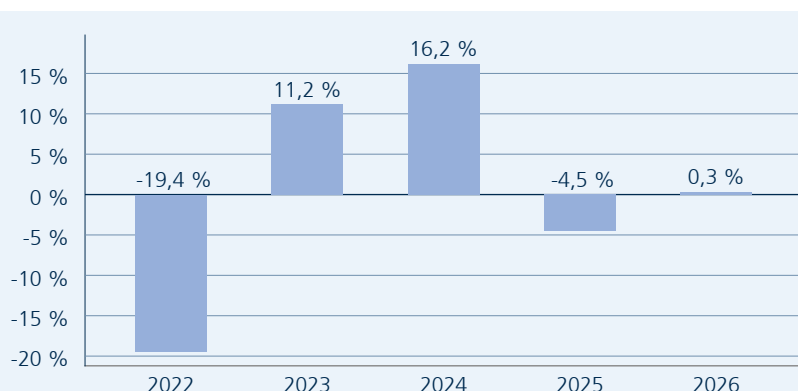
### Indexierte Wertentwicklung



## WERTENTWICKLUNG ÜBER VERSCHIEDENE ZEITRÄUME

1 Monat	3 Monate	6 Monate	lfd. Jahr	1 Jahr
0,18 %	1,01 %	1,29 %	0,34 %	-5,72 %
3 Jahre p.a.	5 Jahre p.a.	10 Jahre p.a.	Seit Auflage p.a.	
6,08 %	2,42 %	4,97 %	5,41 %	
3 Jahre	5 Jahre	10 Jahre	Seit Auflage	
19,39 %	12,72 %	62,48 %	106,21 %	

## WERTENTWICKLUNG PRO KALENDERJAHR



## ROLLIERENDE 12-MONATS ENTWICKLUNG

23.02.22-23.02.23	23.02.23-23.02.24	23.02.24-23.02.25	23.02.25-23.02.26
-6,92 %	12,27 %	12,80 %	-5,72 %

\*Historische Wertentwicklungen sind keine Garantie für eine ähnliche Entwicklung in der Zukunft. Diese ist nicht prognostizierbar. Die Berechnung der Wertentwicklung erfolgt gemäß BVI-Methode.

## 23. FEBRUAR 2026

### Anlageziel

Das Fondsvermögen wird überwiegend in internationale Aktien angelegt. Daneben wird in Anleihen weltweiter Aussteller in Euro beziehungsweise überwiegend währungsgesichert investiert. Mindestens 80% des Netto-Fondsvermögens werden in Vermögensgegenstände investiert, die unter nachhaltigen Gesichtspunkten ausgewählt wurden. Der Fonds bewirbt ökologische und/oder soziale Merkmale im Rahmen der Anlagestrategie im Sinne des Artikel 8 der Verordnung (EU) 2019/2088 (Offenlegungsverordnung). Die Erreichung der ökologischen und/oder sozialen Merkmale des Fonds wird anhand von Nachhaltigkeitsindikatoren gemessen. Zu den Nachhaltigkeitsindikatoren gehört die Einhaltung der Ausschlusskriterien. Im Anschluss, werden mögliche Anlagen einer eingehenden Nachhaltigkeitsanalyse unterzogen. Hier werden die vergangenen, gegenwärtigen und angekündigten Nachhaltigkeitsaktivitäten von Unternehmen auf Basis eines Best-In-Class-Ansatzes und/oder eines Transformationsansatzes analysiert und bewertet. Anhand des verbleibenden nachhaltigen Anlageuniversums prüft das Fondsmanagement, ob die Anlage nachhaltigen und wirtschaftlichen Anforderungen genügt und ein positiver Nutzen für den Fonds zu erwarten ist. Mit dem Fonds werden darüber hinaus auch nachhaltige Investitionen gemäß Artikel 2 Ziffer 17 der Offenlegungsverordnung getätigt. Weiterhin werden beim Erwerb von Wertpapieren, Geldmarktinstrumenten und Investmentanteilen nachteilige Auswirkungen auf Nachhaltigkeitsfaktoren berücksichtigt. Die Anlagestrategie orientiert sich an einem Vergleichsmaßstab, wobei versucht wird, dessen Wertentwicklung zu übertreffen. Das Fondsmanagement kann durch aktive Über- und Untergewichtung einzelner Vermögenswerte wesentlich – sowohl positiv als auch negativ – von diesem Vergleichsmaßstab abweichen. Darüber hinaus sind Investitionen in Titel, die nicht Bestandteil des Vergleichsmaßstabs sind, jederzeit möglich.

```

class MirrorX(object):
    """This adds an 'ob' attribute to the object"""

    def __init__(self, ob):
        self.ob = ob

    def __getattr__(self, attr):
        if attr == 'ob':
            return self.ob
        else:
            return getattr(self.ob, attr)

    def __setattr__(self, attr, value):
        if attr == 'ob':
            self.ob = value
        else:
            setattr(self.ob, attr, value)

    def __delattr__(self, attr):
        if attr == 'ob':
            del self.ob
        else:
            delattr(self.ob, attr)

    def __str__(self):
        return str(self.ob)

    def __repr__(self):
        return repr(self.ob)

    def __eq__(self, other):
        return self.ob == other

    def __neq__(self, other):
        return self.ob != other

    def __lt__(self, other):
        return self.ob < other

    def __gt__(self, other):
        return self.ob > other

    def __le__(self, other):
        return self.ob <= other

    def __ge__(self, other):
        return self.ob >= other

    def __add__(self, other):
        return self.ob + other

    def __sub__(self, other):
        return self.ob - other

    def __mul__(self, other):
        return self.ob * other

    def __div__(self, other):
        return self.ob / other

    def __mod__(self, other):
        return self.ob % other

    def __pow__(self, other):
        return self.ob ** other

    def __call__(self):
        return self.ob

    def __getitem__(self, item):
        return self.ob[item]

    def __setitem__(self, key, value):
        self.ob[key] = value

    def __delitem__(self, key):
        del self.ob[key]

    def __iter__(self):
        return iter(self.ob)

    def __len__(self):
        return len(self.ob)

    def __contains__(self, item):
        return item in self.ob

    def __getattr__(self, attr):
        if attr == 'ob':
            return self.ob
        else:
            return getattr(self.ob, attr)

    def __setattr__(self, attr, value):
        if attr == 'ob':
            self.ob = value
        else:
            setattr(self.ob, attr, value)

    def __delattr__(self, attr):
        if attr == 'ob':
            del self.ob
        else:
            delattr(self.ob, attr)

    def __str__(self):
        return str(self.ob)

    def __repr__(self):
        return repr(self.ob)

    def __eq__(self, other):
        return self.ob == other

    def __neq__(self, other):
        return self.ob != other

    def __lt__(self, other):
        return self.ob < other

    def __gt__(self, other):
        return self.ob > other

    def __le__(self, other):
        return self.ob <= other

    def __ge__(self, other):
        return self.ob >= other

    def __add__(self, other):
        return self.ob + other

    def __sub__(self, other):
        return self.ob - other

    def __mul__(self, other):
        return self.ob * other

    def __div__(self, other):
        return self.ob / other

    def __mod__(self, other):
        return self.ob % other

    def __pow__(self, other):
        return self.ob ** other

    def __call__(self):
        return self.ob

    def __getitem__(self, item):
        return self.ob[item]

    def __setitem__(self, key, value):
        self.ob[key] = value

    def __delitem__(self, key):
        del self.ob[key]

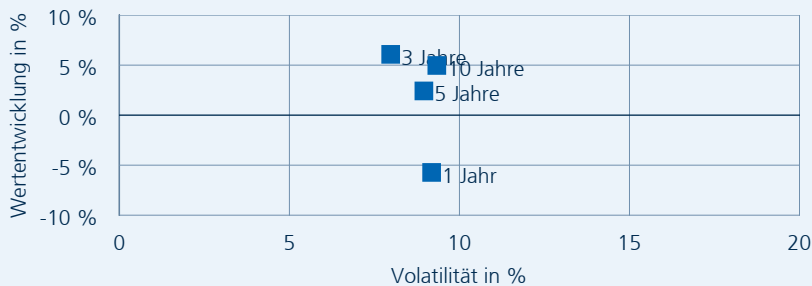
    def __iter__(self):
        return iter(self.ob)

    def __len__(self):
        return len(self.ob)

    def __contains__(self, item):
        return item in self.ob
    
```

**RISIKOINDIKATOR\***

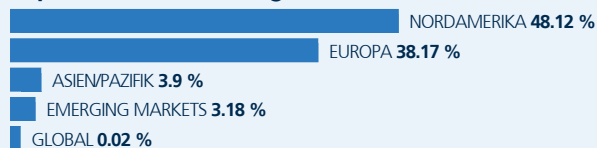
## Risiko-Rendite-Diagramm



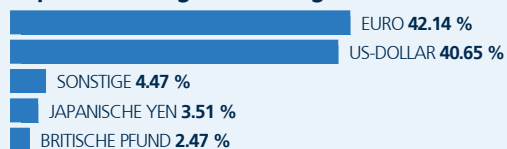
\*SRI gemäß Basisinformationsblatt

## PORTFOLIOSTRUKTUR

### Top 5 Länderverteilung



## Top 5 Währungsverteilung



## Top 5 Holdings



## Rechtliche Hinweise

Datenquelle für Fondsinformationen: cleversoft GmbH. Die von der cleversoft GmbH bereitgestellten Informationen stellen keine Empfehlungen dar und dienen nicht der Anlageberatung. Insbesondere stellen diese Informationen keine Finanzanalyse im Sinne von § 34 b WpHG dar. Die cleversoft GmbH trifft keinerlei Aussage zur bisherigen und zukünftigen Wertentwicklung der angezeigten Wertpapiere. Wegen der Dynamik der Finanzmärkte muss jegliche Haftung im Zusammenhang mit der Nutzung der Informationen oder dem (evtl. auch irrtümlichen) Vertrauen auf deren Richtigkeit, Vollständigkeit oder Aktualität ausgeschlossen werden. Informieren Sie sich daher vor dem Fondserwerb auf der Internetseite der jeweiligen Fondsgesellschaft.

## Fonds-Fakten

Fondskategorie	Mischfonds
WKN	A1JQ11
ISIN	LU0718558728
Auflagedatum	01. Juni 2012
Fondsvolumen	2,87 Mrd. EUR (23.02.2026)
Kapitalverwaltungs- gesellschaft	Union Investment Luxembourg S.A.
Fondsmanagement	Union Investment Gruppe
Depotbank	DZ PRIVATBANK S.A.
Sitzland	Luxemburg
Fondswährung	EUR
Rücknahmepreis	98,31 EUR
Ertragsverwendung	Ausschüttend
Geschäftsjahr	01. April - 31. März
Transparenzverord- nung (EU) 2019/2088	Art. 8 TVO ESG Merkmale

## Fonds-Konditionen

Ausgabeaufschlag	Ein Ausgabeaufschlag fällt im Rahmen der fondsgebundenen Rentenversicherungen von R+V nicht an.
Max. Verwaltungsvergütung p. a.	2,00 %
Max. Fondsmanagement Gebühr p.a.	2,00 %
Max. Depotbankvergütung p.a.	0,00 %
Laufende Kosten p.a.	1,90 % (06.10.2025)
Erfolgsabhängige Vergütung	0,00 %

## Kennzahlen

Volatilität		Maximum Drawdown	
1 Jahr	+9,18 %	1 Jahr	-12,95 %
3 Jahre	+7,98 %	3 Jahre	-13,85 %
5 Jahre	+8,95 %	5 Jahre	-20,49 %
Sharpe Ratio		Verlustdauer in Monaten	
1 Jahr	-0,82	1 Jahr	3
3 Jahre	0,40	3 Jahre	3
5 Jahre	0,07	5 Jahre	3

```

11 operation == "MIRROR X":
12     mirror_mod.use_x = True
13     mirror_mod.use_y = False
14     mirror_mod.use_z = False
15 elif operation == "MIRROR Y":
16     mirror_mod.use_x = False
17     mirror_mod.use_y = True
18     mirror_mod.use_z = False
19 elif operation == "MIRROR Z":
20     mirror_mod.use_x = False
21     mirror_mod.use_y = False
22     mirror_mod.use_z = True
23
24 #selection at the end -add back the deselected
25 mirror_ob.select= 1
26 modifier_ob.select=1
27 bpy.context.scene.objects.active = modifier_ob
28 print("Selected" + str(modifier_ob)) # modifier
29
30 #1000
31 modifier_ob.select = 0
32 bone = bpy.context.selected_objects[0]
33 bone.parent = modifier_ob

```

Informationen erhalten Sie in den Volksbanken und Raiffeisenbanken, R+V-Agenturen sowie bei der Direktion der Gesellschaften der R+V Versicherungsgruppe, Raiffeisenplatz 1, 65189 Wiesbaden.

Telefon: 0800 533-1171

Kostenfrei aus allen deutschen Fest- und Mobilfunknetzen.

[www.ruv.de](http://www.ruv.de)

R+V Lebensversicherung AG