



Cover

In 2020 the Atlantic Ocean has experienced a record season of hurricanes and many of these have hit the southern coast of the United States.

The latest to do so was Zeta, the twenty-seventh storm of the Atlantic hurricane season, which caused at least 6 deaths, mass power outages and floods in Louisiana.

Heavy rain caused by the passage of Hurricane Zeta over the Mississippi Valley caused the river and its tributaries to swell, carrying large amounts of mud and sediment into the Gulf of Mexico.

As shown in this image acquired by one of Copernicus Sentinel-3 satellites on 2 November 2020, the amount of sediment was such that it tainted the sea in a coastal area (about 500 km long) that runs from the mouth of the Mississippi River in Louisiana to the coast of Texas.

The Copernicus Sentinel-3 mission is composed of two twin satellites, Sentinel-3A and Sentinel-3B, which allow us to monitor water quality and sediment transport with coastal areas.

Vorgelegt von:

Leon Wahlefeld

(3104861)

Kommunikationsdesign mit Praxissemester

9. Fachsemester

Generative Gestaltung

Wie eine interdisziplinäre Herangehensweise die Zukunft des Designs revolutionieren könnte.

Erstprüferin:

Prof. DipL-Des. Ilka Helimg

Visuelle Konzeption und zeichnerische Gestaltung

Zweitprüferin:

Prof. Dr. phil. Melanie Kurz

Visuelle Konzeption und zeichnerische Gestaltung

1 Einleitung

2 Vorwort 4

3 Themenstellung 6

4 Zielsetzung 8

5 Inhalt 10

6 Copernicus 12

7 Image of the day 14

8 Umsetzung 16

9 Bild Prozess 30

10 Text Prozess 42

11 Archive

12 Plakate 62

13 Verzeichniss 182

14 Quellen 186

15 Impressum 187

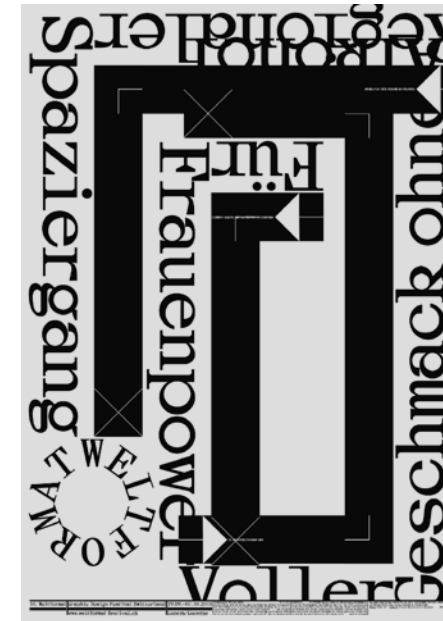


Einleitung

Vorwort

FIG.1

Die Generative Gestaltung ist ein Fachgebiet, welches ich schon seit Längerem mit Erstaunen beobachtet habe. Meine ersten Berührungspunkte waren Plakate zum *Weltformatfestival 2018* von *Ludovic Balland*. Bereits zu diesem Zeitpunkt war ich von der Arbeit und der Methodik Verfahren sehr beeindruckt, hatte allerdings noch zu viel Respekt davor, dieses Verfahren selbst anzuwenden. In meinem Praxissemester ein Jahr später, wurde ich erneut mit der Thematik konfrontiert. Diesmal im Rahmen eines Projekts, welches mit Hilfe eines generativen Layouting umgesetzt werden sollte. Ziel war es, ein umfangreiches Bildarchiv eines Architekturstudios in einem Buch zusammenzufassen. Hierzu wurde ein Algorithmus entwickelt, der alle Bilder inklusive Texte hintereinander platzierte, ohne jedoch das Format des Buches zu berücksichtigen. Dieser Algorithmus führte jedoch dazu, dass Bilder auf einer Seite teilweise abgeschnitten wurden. Der abgeschnittene Rest des Motivs wurde in diesen Fällen auf der Folgeseite abgebildet. Diese spezielle Art der Gestaltung, wäre ohne die Verwendung generativer Algorithmen nicht entstanden. Das liegt in erster Linie daran, dass der Computer keinerlei Ästhetik oder grafische Regeln kennt. Kaum ein Designer hätte eine solche Darstellungsweise des Bildmaterials gewählt. Aber genau das ist es, was in meinen Augen diese gestalterische Herangehensweise sehr spannend macht. Es ergeben sich neuartige Visualisierungen, die DesignerInnen – die unterbewusst



immer an Trends und Regeln gebunden sind – sonst nicht anstreben würden. Das damit einhergehende Konzept ist auch für Kunden eine spannende Methodik, wodurch die Generative Gestaltung mit hoher Wahrscheinlichkeit eine große Zukunft vor sich hat.

Das oben genannte Projekt weckte also endgültig mein Interesse an der Generativen Gestaltung und sorgte zusätzlich dafür, meine Furcht vor Programmiersprache abzulegen.

Nach den ersten Nachforschungen zu diesem Thema, fiel mir ein Projekt ein, welches ich 2018 selber realisierte und sich ebenfalls im Bereich der Generativen Gestaltung bewegte. Dabei handelt es sich um ein Plakatarchiv, mit dem Namen *RIG - Revolution im Großformat*. Bei diesem Prozess standen mein Kommilitone und ich vor einer sehr ähnlich Problematik. Uns lag eine enorme Datenmenge in Form von Plakaten vor, die alle ihren Weg in ein gedrucktes Archiv finden mussten. Zu

1 diesem Zeitpunkt haben wir bereits unbewusst ein ana-
2 loges generatives Verfahren verwendet. Wir erlegten
3 uns selbst gestalterische Regeln auf, die in der Um-
4 setzung nicht gebrochen werden durften. Auch dieses
5 Projekt erhielt insbesondere durch dieses Verfahren
6 eine sehr eigene und spannende Gestalt.

7 Als Neueinsteiger in der Generativen Gestaltung, bedarf
8 es einer experimentierfreudigen Einstellung, um in die-
9 sem neuartigen Teilbereich zu einem Resultat zu ge-
10 langen. Aus diesem Grund sah ich die Gelegenheit, dies
11 in meiner Bachelor Arbeit zu thematisieren und mich
12 als angehender Grafikdesigner in dieser Gestaltungs-
13 disziplin weiterzubilden.

14 Themenstellung

15 Die Generative Gestaltung ist ein Teilbereich der
16 Gestaltung, die bislang noch recht unberührt und
17 unerforscht ist. Hierbei handelt es sich um eine
18 spezielle Herangehensweise, einen visuellen Output
19 zu erhalten. Dieses Prinzip kann in fast allen krea-
20 tiven und visuellen Bereichen seine Anwendung
21 finden. Dazu gehören sowohl Illustration, Bild,
22 Film als auch Layout.

23 Anders als bei einem herkömmlichen Entwurfs-
24 prozess, bei dem man Schritt für Schritt manuell
25 ein visuelles Ergebnis erhält – und darüber hinaus
26 als DesignerIn selbst in den Entstehungsprozess
27 eingreift – werden bei der Generativen Gestaltung
28 einige Komponenten durch ein analoges Regel-
29 werk oder einen Quellcode ersetzt. Generatives
30

Design gehört bereits in einigen Bereichen zu
einer weit verbreiteten Methodik, um neuartige
Ergebnisse zu erzielen. Dazu gehören beispiels-
weise das Produktdesign sowie das Ingenieur-
wesen. Im Grafikdesign ist Generative Gestal-
tung vor allem in der Verarbeitung von großen
Informationsmengen sowie der Entwicklung gro-
ßer Diversität, sehr vorteilhaft.

In der grafischen Gestaltung gibt es zwei Wege,
um einen generativen Output zu erhalten. Der
simplere und nachvollziehbare Weg, ist ein ana-
loges Regelwerk, welches dem Entwurfsprozess
bestimmt und dirigiert. Das bedeutet zu Beginn
einer Entwurfsphase, dass ein Regelwerk auf-
gesetzt werden muss, an dem man sich in der
Umsetzung des Entwurfs bindet. Die Gestaltung
ist also nicht frei, wie es normalerweise der Fall
ist, sondern äußert sich vielmehr in der Anwen-
dung selbst verfasster Richtlinien. Ist der daraus
resultierende Output nicht zufriedenstellend,
muss das Regelwerk bearbeitet werden – nicht
jedoch das entstandene Resultat. Sobald man
beginnt, Regeln zu brechen und Ausnahmen ein-
zufügen, wird der Sinn der Generativen Gestaltung
gebrochen. Die Visualisierung wird hier also nicht
nicht von Geschmack oder ähnlichen Normen
bestimmt, sondern allein durch ein Regelwerk
definiert.

Der digitale Ansatz weist keine großen Abwei-
chungen zu der analogen Version auf. Hier ist
allerdings die Kenntnis von Programmiersprache
Voraussetzung. Auch hier steht an erster Stelle

1 die Idee, gefolgt von der Erstellung der Regeln.
2 Diese Regeln werden nun aber in eine digitale
3 Form umgewandelt. Das bedeutet, es wird ein
4 Quellcode erstellt, der das Regelwerk im Zu-
5 sammenhang mit dem Inhalt in einen visuellen
6 Output umwandelt. An genau dieser Stelle wird
7 die Umsetzung der Regeln, aus dem analogen
8 Ansatz, durch den Computer übernommen. Das
9 bedeutet aber auch, dass die Kontrolle für De-
10 signerInnen in diesem Ansatz noch geringer ist,
11 da die Gestaltung komplett durch den Code be-
12 stimmt wird und es wesentlich schwerer ist, Aus-
13 nahmen einzubauen. Sollte der Output also nicht
14 den Wünschen entsprechen, muss der Quellcode
15 entsprechend angepasst werden. Dieser Vorgang
16 wird so lange wiederholt, bis das gewünschte
17 Ergebnis erzielt wurde.

18 Zielsetzung

19 Die Generative Gestaltung bietet eine Vielzahl
20 von Möglichkeiten, Projekte in die Realität umzu-
21 setzen. Die Herangehensweise an ein solches
22 Projekt ist frei zu wählen. Zu Beginn ist festzu-
23 legen, mit welcher Programmiersprache und auf
24 welcher Plattform der Code basieren soll. Diese
25 Entscheidung ist natürlich von Vorkenntnissen
26 verschiedener Programmiersprachen abhängig.
27 Ebenfalls abhängig von der Wahl der Programmier-
28 sprache ist, was genau umgesetzt werden soll.
29 Komplexere Verfahren, wie etwa automatisierte
30

Algorithmen, benötigen eine weitaus vielseitigere
Programmierensprache, als etwa eine reine Visua-
lisierung bestimmter Elemente.

In meinem Fall fiel die Entscheidung der Program-
miersprache auf das sogenannte *Processing*.
Processing hat den Charakter einer vereinfachten
Version der *Java* Programmiersprache und wurde
entwickelt, um Grafik, Simulation und Animatio-
nen zu erstellen. Es hat zum Ziel, einen grafisch
denkenden Menschen und den Code näher zu-
einander zu bringen. Die endgültige Entscheidung
für diese spezifische Programmiersprache ist
damit zu begründen, dass hier der Fokus auf der
Vereinfachung der Sprache sowie der direkten
Umsetzung in einen visuellen Output liegt. Also
genau die Instrumente, die notwendig sind, um
dieses Projekt umzusetzen.

Das Ziel war es, eine Plakatreihe zu entwerfen,
die auf der Grundlage eines Codes – und somit
einer Reglementierung – basiert. Mit Hilfe dieses
Verfahrens, sollten auf Knopfdruck eine beliebige
Menge an Plakaten generiert werden. Um dieses
Ziel zu gewährleisten, habe ich bestimmte Para-
meter festgelegt, die bei jedem Durchlauf des
Codes neu generiert werden. Jedes neu aus-
gespielte Plakat, ist dementsprechend anders
als sein Vorgänger.

Daraus sollte ein Verfahren entstehen, welches
man mit Inhalten füttert, die dann verarbeitet,
gesetzt und in Form einer fertigen PDF ausge-
worfen werden. Diese Technik soll vorrangig da-
zu dienen, eine große Menge an Daten in kürzes-

1 ter Zeit in einen visuellen Output zu verwandeln.
2 Genau das macht diese Arbeitsweise für mich,
3 als einen visuell arbeitenden Gestalter, zu etwas
4 ganz Speziellem und sie geht mit einer Menge
5 Herausforderungen einher. So erstelle ich Plaka-
6 te normalerweise in *InDesign*. Dort kann ich in
7 der Entwurfsphase jeden Schritt visuell nach-
8 vollziehen und korrigieren. In der Generativen
9 Gestaltung sind diese Grundlagen jedoch nicht
10 vorhanden. Eine gestalterische Idee muss zu-
11 nächst genauestens überlegt werden. Ist dieser
12 Schritt getan, folgt die nächste Herausforderung.
13 Wie erkläre ich dem System, wie es die zuvor
14 definierte gestalterische Idee umsetzen soll? Der
15 Umsetzungs-Wunsch muss also zunächst in einen
16 Proccesing-Code übersetzt werden. Sind all diese
17 Schritte vollzogen, wirft das System einen visu-
18 ellen Entwurf aus, den ich als Gestalter dann zum
19 ersten mal nachvollziehen kann.

21 Inhalt

22 Während des Findungsprozesses des entsprechenden
23 Inhalts, gab es bestimmte Faktoren, die für mich wich-
24 tig waren. Diese orientierten sich an dem Wesen der
25 Generativen Gestaltung.

26 Wie bereits erwähnt, kann diese Methodik besonders
27 hilfreich sein, wenn es darum geht, eine große Daten-
28 menge zu verarbeiten. Generative Plakate haben den
29 Vorteil stets die neuesten Informationen zu beinhalten
30 zu können, da durch ein generativen Verfahren in kür-

zester Zeit ein automatisierter Output erfolgen kann.
Dementsprechend schien die Arbeit mit aktuellen Daten
eine passende Vorgehensweise zu sein, um die Plaka-
te mit Inhalten zu befüllen. Nun gibt es zur aktuellen
Zeit, insbesondere wenn man die Digitalisierung in Be-
tracht zieht, eine Menge aktueller Daten im Internet zu
finden. Sei es in Nachrichtenform, in den sozialen Me-
dien oder auch wissenschaftliche Daten, die sich jede
Sekunde aktualisieren. Die Auswahl ist dementsprechend
groß. Nun galt es, die Auswahl der möglichen Thema-
tiken zu reduzieren. Dies tat ich, indem ich drei Faktoren
definierte, die mir im Bezug auf das zu erstellende ge-
nerative Plakat besonders wichtig waren. Ich definierte
automatisiert, *aktuell* sowie *Entwicklung*. Diese Faktoren
in Kombination mit persönlichen Interessensgebieten
brachten mich zur Raumfahrt als möglichen inhaltlichen
Schwerpunkt. Diese Thematik bietet eine fast unend-
liche Vielfalt von verschiedenen Daten, die sekundlich
geschaffen und aktualisiert werden. Um jedoch aus
dieser Datenmenge diejenigen herauszufiltern, die letzt-
lich in die Gestaltung des Plakats einfließen, musste die
Auswahl an Daten eingegrenzt werden. Dazu wurden
bestimmte Parameter definiert. Zunächst war es wich-
tig, Bilder zu erhalten, die eine entsprechende Auflösung
bieten, um weiterverarbeitet zu werden. Außerdem
mussten zu jedem dieser Motive, entsprechende Infor-
mationen vorhanden sein, die in das Processing System
weitergeleitet werden können. Diese Kombination aus
Raumfahrt und Bildwelten, brachte mich schnell in den
Bereich von Satellitenbildern. Diese bieten eine Menge
spannender und beeindruckender Motive, die durch
ihre Plakativität sehr gut für ein Medium, wie das eines

1 Plakats geeignet sind. Zahlreiche Plattformen, darunter
2 die wohl bekanntesten wie *Google Earth*, *Apple Maps*
3 und *Bing Maps*, bieten dem Nutzer sogar den ganzen
4 Planeten in Form von Bildmaterial an. Diese Möglichkeit
5 bot mir zwar eine enorme Menge an Daten, stellte mich
6 aber im Bezug auf die Themenfindung nicht zufrieden.
7 Das lag im Wesentlichen an zwei Faktoren. Zum einen
8 kann ich aufgrund fehlender Qualifikation in der Astro-
9 nomie kaum bewerten, welcher Bildausschnitt eines
10 gesamten Planeten interessant oder gar besonders
11 wichtig ist. Zum anderen bieten rein zufällig ausgewählte
12 Ausschnitte aus *Google Earth* und Co.–ganz abgesehen
13 von der zu niedrigen Auflösung–keine Aktualität. Die
14 Daten mussten demnach einer Quelle entstammen, die
15 stets aktualisiertes Bildmaterial bietet, welches von
16 qualifiziertem Fachpersonal ausgesucht und veröffent-
17 licht wird.

18 Copernicus

19 *Copernicus* ist das Erdbeobachtungsprogramm
20 der Europäischen Union, das sich mit unserem
21 Planeten und seiner Umwelt zum größtmöglichen
22 Nutzen aller europäischen Bürger befasst. Es
23 bietet Informationsdienste auf der Grundlage von
24 satellitengestützter Erdbeobachtung und In-situ-
25 Daten (Nicht-Weltraum Daten) an. Das Programm
26 wird von der Europäischen Kommission koor-
27 diniert und verwaltet. Es wird in Partnerschaft
28 mit den Mitgliedstaaten, der Europäischen
29 Weltraumorganisation (ESA), der Europäischen
30

Organisation für die Nutzung meteorologi-
scher Satelliten (EUMETSAT), dem Europäischen
Zentrum für mittelfristige Wettervorhersagen
(ECMWF), den EU-Agenturen und Mercator
Océan umgesetzt.

Copernicus wurde speziell für die Anforderungen
der Nutzer entwickelt. Basierend auf Satelliten-
und In-situ-Beobachtungen liefern die *Coperni-
cus*-Dienste auf globaler Ebene nahezu in Echt-
zeit Daten, die für lokale und regionale Bedürf-
nisse genutzt werden können, um unseren
Planeten besser zu verstehen und die Umwelt,
in der wir leben, nachhaltig zu schützen.

Copernicus wird von einer Reihe von speziellen
Satelliten (die Sentinel-Familien) und beitragende
Missionen (bestehende kommerzielle und staat-
liche Satelliten) bedient. Die *Sentinel*-Satelliten
sind speziell auf die Bedürfnisse der Copernicus-
Dienste und ihrer Nutzer zugeschnitten. Seit dem
Start von *Sentinel-1A* in 2014 hat die Europäische
Union einen Prozess in Gang gesetzt, um bis 2030
eine Konstellation von fast 20 weiteren Satelliten
in die Umlaufbahn zu bringen.

Copernicus sammelt auch Informationen von
Nicht-Weltraum Daten, wie etwa Bodenstationen.
Sie liefern Daten, die von einer Vielzahl von Sen-
soren am Boden, auf See oder in der Luft erfasst
werden.

Die Dienste von *Copernicus* wandeln diese Fülle
von Daten in Mehrwert Informationen um. Über
Jahre und Jahrzehnte zurückreichende Daten-
sätze, werden vergleichbar und durchsuchbar

1 gemacht, so dass die Überwachung von Ver-
2 änderungen gewährleistet ist. Muster werden
3 untersucht und genutzt, um bessere Vorhersagen
4 zum Beispiel über den Ozean und die Atmosphä-
5 re zu treffen. Karten werden aus Bildern erstellt,
6 Merkmale und Anomalien identifiziert und statis-
7 tische Informationen extrahiert.

8 9 Image of the day

10 Die Webseite copernicus.eu bietet eine Reihe
11 von offenen Archiven. Eines dieser Archive, ist
12 das *Image of the day*. Diese Sammlung liefert
13 täglich neue Datensätze in Form von Bildern
14 und den zugehörigen Informationen. Bilder von
15 Umweltkatastrophen, Erderwärmung, Wetter-
16 ereignissen und noch unzählige weitere interes-
17 sante sowie erschreckende Fakten, gepaart mit
18 einer entsprechenden Bildwelt.

19 Diese Datenquelle bot sich als ein idealer Inhalt
20 für meine Plakate an. Das Archiv gab mir die
21 Möglichkeit, die Idee eines auf aktuellen Daten
22 basierenden Plakates umzusetzen. Nun konnte
23 ich mit täglich aktualisierten Bildwelten sowie
24 Texten Plakate generieren.



Umsetzung

Bei der Umsetzung eines generativen Plakats, wird man als GestalterIn früh mit einigen Herausforderungen konfrontiert. Nach der Wahl der Programmiersprache, muss jeder Entwurfszyklus genauestens durchdacht werden. In diesem Fall stand bereits frühzeitig fest, ein Plakat aus Bild und Text zu erstellen. Sowohl Bild als auch Text sollten unabhängig voneinander generativ verarbeitet werden, um die Anzahl des unterschiedlichen Outputs zu erhöhen. Die Idee, aus einem Inhalt, diverse visuelle Outputs zu schaffen, stand hierbei im Vordergrund. Sobald die Grundparameter des Plakats feststehen, liegt der Fokus auf der Vielfältigkeit der Gestaltung – nicht etwa vorwiegend auf der Ästhetik. Dieser Fokus war insbesondere beim Layout des Textes eine neue Herausforderung. Ich entschied mich dafür, Bild und Text zu Beginn separiert voneinander zu bearbeiten. Diese Entscheidung ermöglichte mir eine bessere Übersicht. Sied hatte aber ebenso den Nachteil, dass ich mich zunächst im Westlichen auf den Bild-Code konzentrieren musste, da dieser die Grundlage des Plakats liefert. Die Gestaltung sollte sich von Grund auf an der Methodik des Generativen Designs ableiten. Aus diesem Grund hatte ich mich bereits zu Beginn dafür entschieden, eine stringente Gestaltung zu generieren. Dies sollte in Form eines strengen Textrasters visualisiert werden. Die Bildebene war ebenfalls so zu bearbeiten, sodass sie den technischen Charakter des Projekts aufzeigt.

Um Elemente in *Processing* umzusetzen, ist eine bestimmte Herangehensweise an das Projekt elementar. So ist immer zu beachten, was genau umgesetzt werden soll und wie es am besten in ein System, in Form eines *Processing*-Codes, ein-

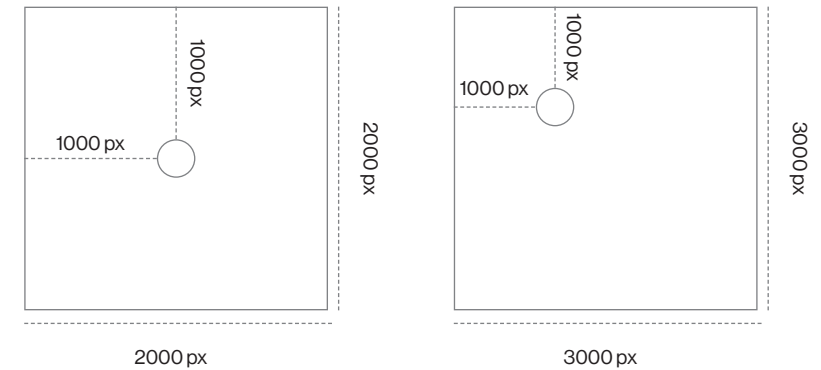


FIG.3

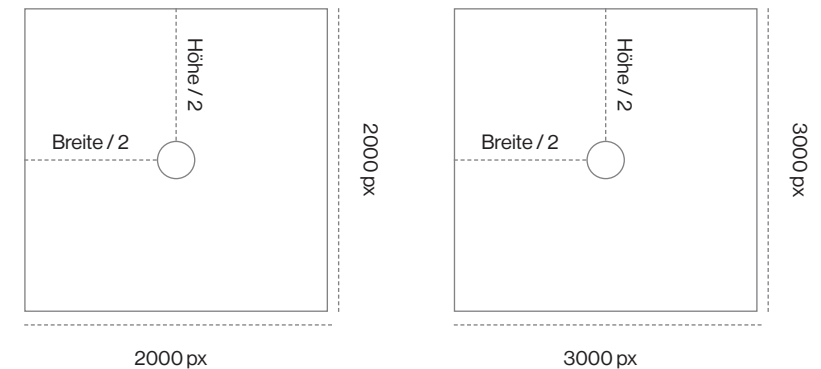


FIG.2

FIG.3

gepflegt wird. Das ist besonders herausfordernd, wenn es in den Bereich des Layouts geht. Möchte man beispielsweise einen Text mittig in ein 2000x2000 Pixel Dokument setzen, muss der Text anhand von X und Y Koordinaten platziert werden. Bei diesem Beispiel wäre die Mitte 1000x1000 Pixel. Verändert sich im Nachhinein allerdings die Größe des Dokuments, beispielsweise auf 3000x3000 Pixel, sitzt der Text nicht mehr mittig. In diesem Fall muss der Processing Code also so definiert werden, dass der Text immer mittig platziert wird - unabhängig von der Größe des Dokuments. Der Code wird in diesem Falle so angepasst, dass der Text immer auf der Hälfte der Dokumentehöhe sowie der Hälfte der Dokumentenbreite platziert wird. Nun kann das Dokument beliebig vergrößert oder verkleinert werden, ohne dass die Positionierung des Textes korrigiert werden muss. Eine kor-

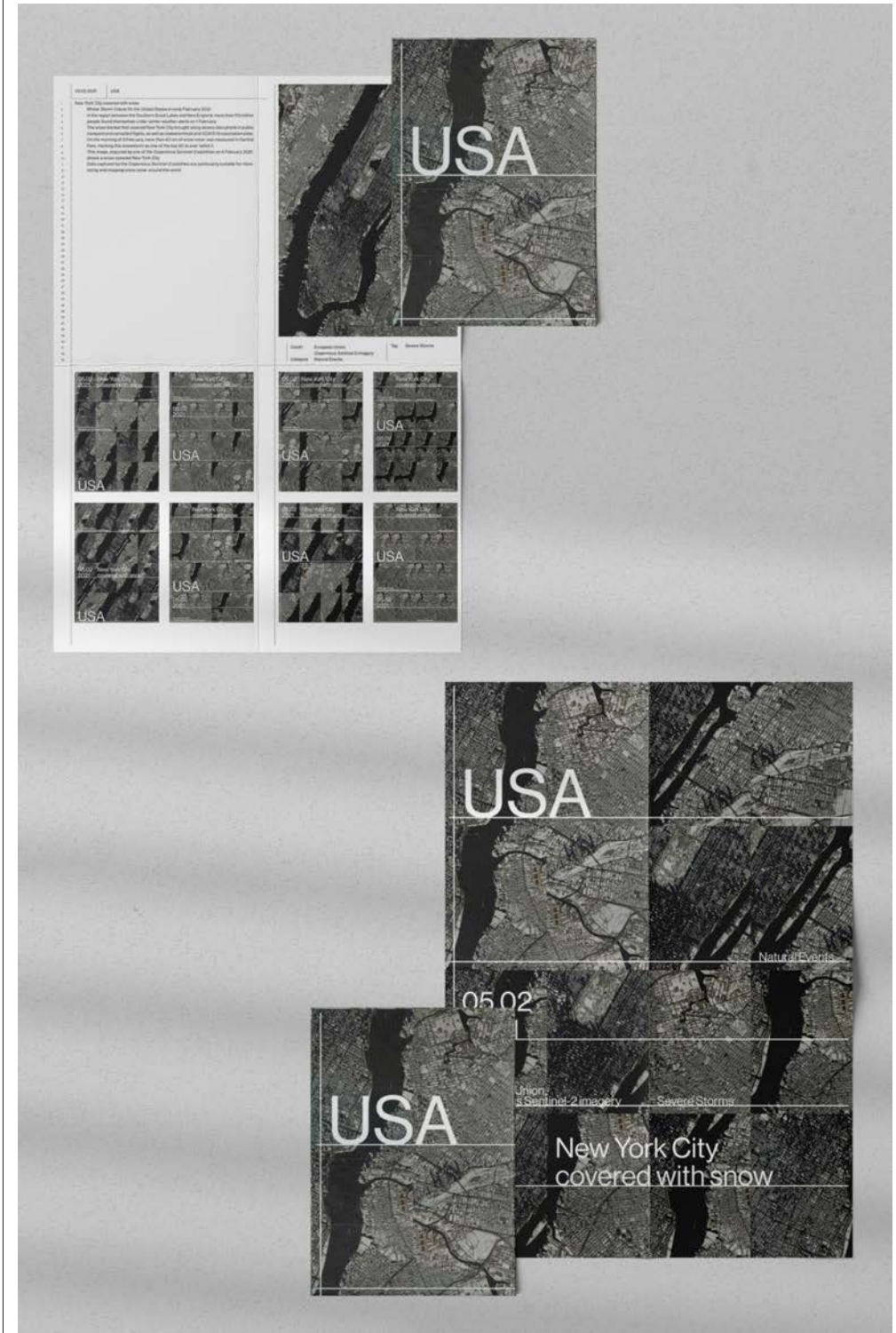
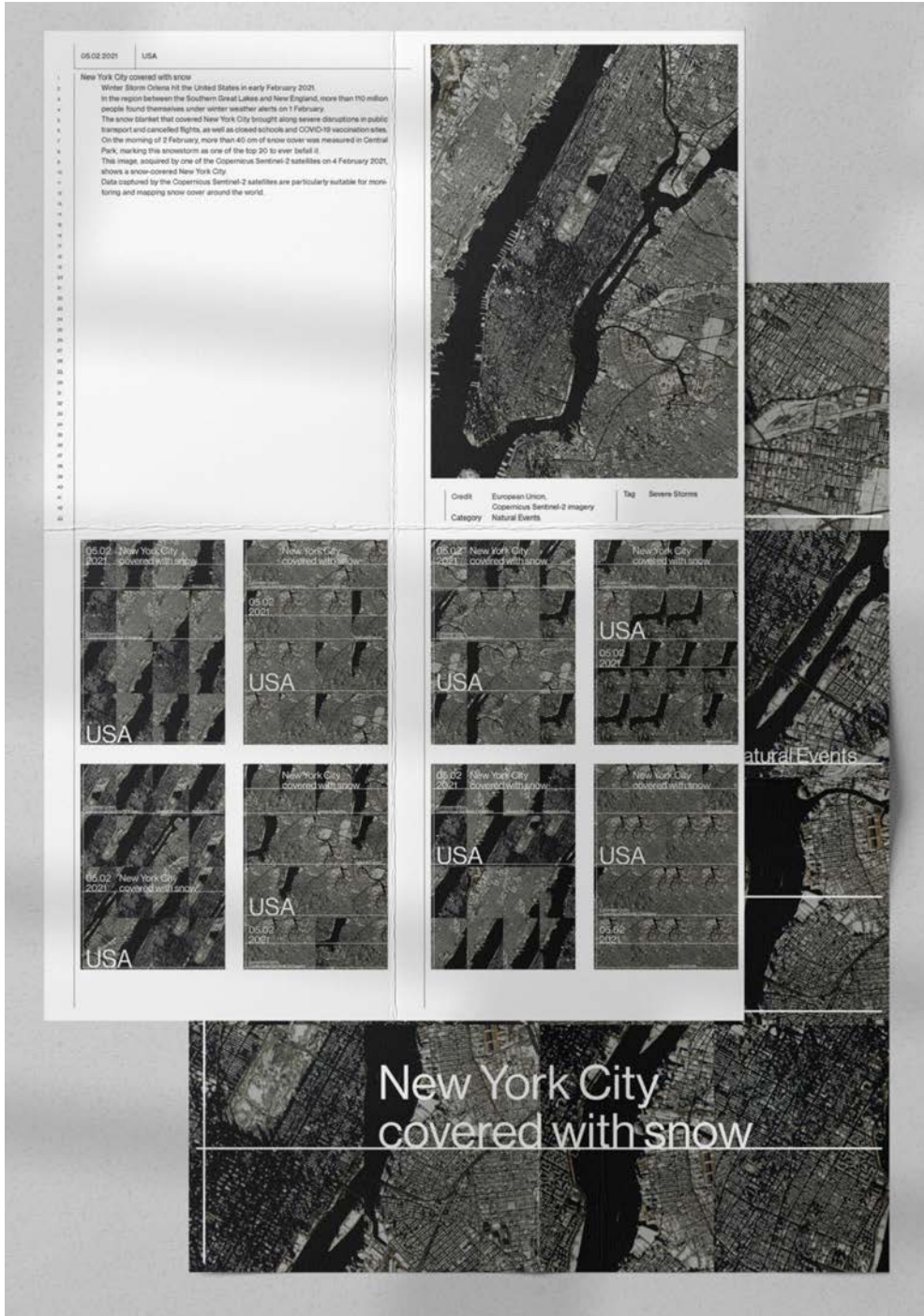
1 rekte Positionierung des Textes ist vor allem dann besonders
2 wichtig, wenn komplexere Befehle ausgeführt werden sollen.
3 Es muss als ständig in Betracht gezogen werden, dass es in
4 *Processing* häufig verschiedene Wege gibt, zu einem Ziel zu
5 gelangen. Es muss demnach eine genau Überlegung darüber
6 angestellt werden, was genau umgesetzt werden soll. Dieses
7 “was” zu verbalisieren und anschließend in eine passende
8 Codeform zu übersetzen, damit es durch *Processing* aus-
9 geführt werden kann, ist demnach einer der elementarsten
10 Bestandteile des Generativen Designs.

11 In meinem Fall stand das Format von Anfang an fest. Um ein
12 abwechslungsreiches Layout zu erstellen, war ein funktionie-
13 rendes Textraster maßgeblich. Hierbei war jedoch zu beachten,
14 dass sich die Elemente pro Durchlauf neu positionieren. Text
15 Hierarchien konnten hingegen nicht durch eine genaue Be-
16 stimmung der Positionierung definiert werden, da diese mög-
17 lichst mannigfaltig sein sollten. Außerdem hatte ich zum Ziel,
18 so wenig Überschneidungspunkte wie möglich zu schaffen,
19 um nochmals die Anzahl an verschiedenen Positionierungs-
20 möglichkeiten des Textes zu erhöhen.

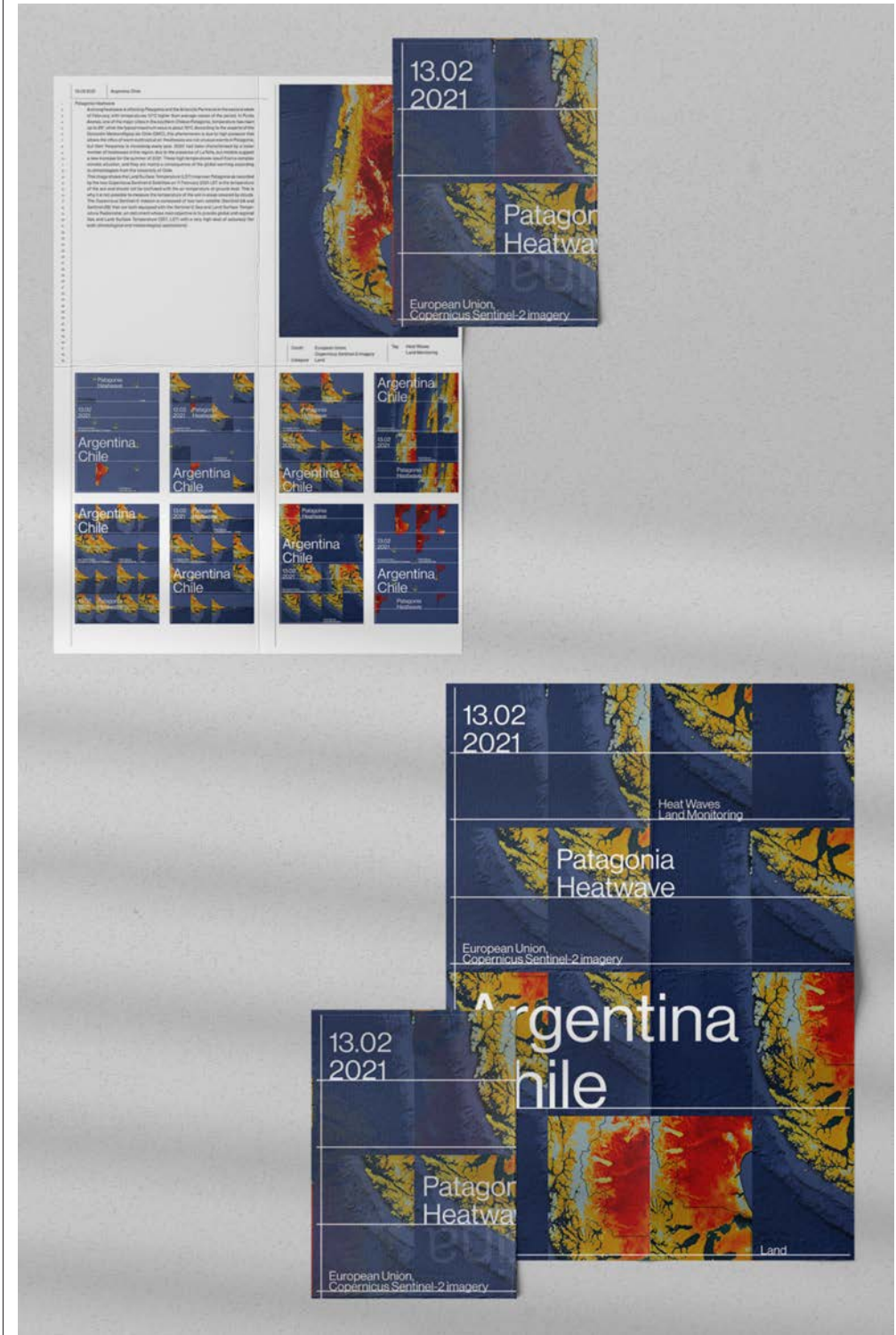
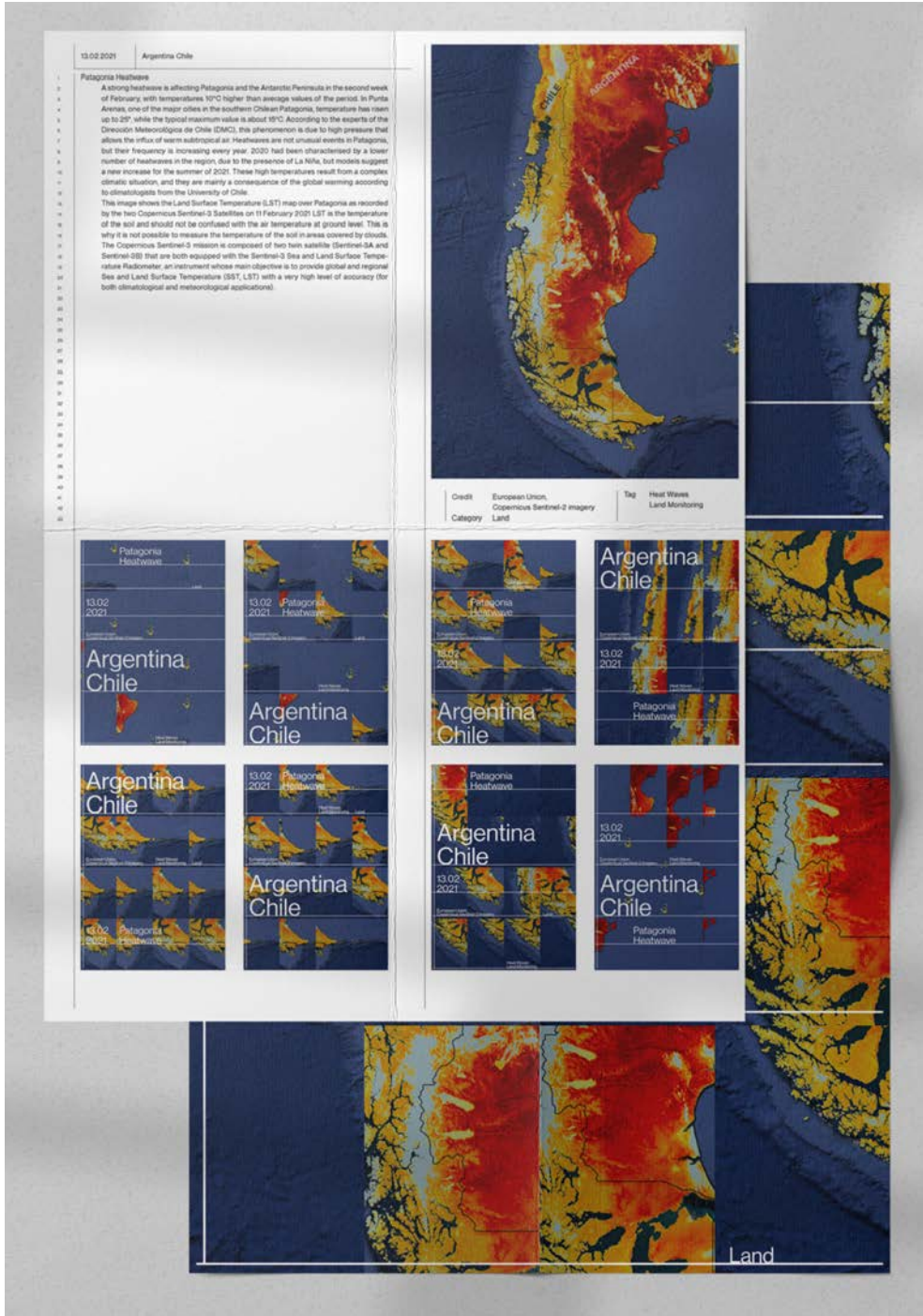
21 Bisher nutze ich zum Bearbeiten eines Motivs für ein Plakat
22 stets *Photoshop*. Dort kann ich aus einer Vielfalt an Filtern
23 und Effekten wählen, um das gewünschte Ergebnis zu er-
24 halten. Diese Effekte und Filter sind im Grunde nichts anderes,
25 als Codes, die bestimmte Befehle ausführen. Da *Processing*
26 selber keine eigenen Template- oder Filter-Bibliotheken an-
27 bietet, stand ich zu diesem Zeitpunkt wieder vor einer ganz
28 neuen Herausforderung. Da keine vordefinierten Filter vor-
29 handen waren, habe ich einen eigenen *Photoshop* Filter in
30 *Processing* erstellt. Mein Ziel war es, einen Effekt zu program-

mieren, der aus einem Motiv, beliebige viele neue Bildwelten
entstehen lässt. Um diesen Output jedoch so selbstständig
und mannigfaltig wie möglich zu generieren, habe ich mit
Random Integer – also zufällige ganze Zahlen – gearbeitet.
Diese bewussten Zufälle erlaubten er mir, meinen Output noch
vielfältiger zu gestalten. Auch an dieser Stelle war das Arbei-
ten mittels eines Rasters vorgesehen. Um eine stringente
Gestaltung mit abwechselnden Resultaten zu erhalten, habe
ich ein System geschrieben, welches zufällige Bereiche eines
Bildes auswählt. Diese Ausschnitte werden anschließend in
einem 4x4 Bildraster auf dem Plakat platziert. Dieses Ver-
fahren ermöglicht es schlussendlich, aus einem einzelnen
Motiv, nahezu unendlich neue Bildwelten zu erschaffen.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Der Bildgenerator besteht im Wesentlichen aus drei Komponenten. Jeder einzelne Teilabschnitt des Verfahrens generiert Informationen, die er an den folgenden Abschnitt weitergibt, bis am Ende das Ergebnis ausgespielt wird.

Z.10

Z.11

Zu Beginn eines *Processing* Dokuments, müssen einige grundlegende Elemente, die man im im späteren Code nutzen möchte, festgelegt werden:

```
import processing.pdf.*;
import java.util.Calendar;
import java.util.Random;
```

Z.12

```
PImage img;
```

Z.14

```
int tileCountX = 4;
int tileCountY = 4;
int tileCount = tileCountX*tileCountY;
PImage[] imageTiles = new PImage[tileCount];
```

Z.16

```
int tileWidth, tileHeight;
```

```
int cropX = 0;
int cropY = 0;
int randomX;
int randomY;
```

Zunächst werden drei bereits bestehende Assits aus der *Processing* Library geladen.

Processing.pdf ist ein Assit, welches das Exportieren einer Druck-PDF ermöglicht.

Java.util.Calendar erlaubt *Processing*, das aktuelle Datum sowie die Uhrzeit zu verwenden. Diese Funktion ist dafür zuständig, die exportierten PDF's mit eben diesen Informationen zu benennen. Dadurch wird immer eine neue Datei erstellt und nicht die aktuelle überschrieben.

java.util.Random ermöglicht es, in diesem Dokument mit zufälligen Zahlen zu arbeiten.

PImage übermittelt *Processing* die Information, dass in diesem Code ein Bild geladen werden soll, sobald man den Befehl *img* beansprucht.

int steht für Integer. Mit Hilfe dieses Befehls können bestimmte Angaben mit einer Zahl beschrieben werden. Im Beispiel Z.16 weiß *Processing* durch den Befehl *tileCountX*, dass es sich hierbei um die Zahl vier handelt.

Der nächste Schritt beim Arbeiten mit *Processing* ist das Festlegen der Datei-Einstellungen. Diese werden in den sogenannten `public void settings` und dem `void setup` vorgenommen:

```

public void settings() {
    size(4961, 7016, "processing.pdf.PGraphicsPDF", ti-
        mESTAMP()+"BILD.pdf");
}

void setup() {
    img = loadImage("i.jpg");
    image(img, 0, 0);
    noCursor();

    tileWidth = width/tileCountY;
    tileHeight = height/tileCountX;
}

```

z. 8

z. 12



An erster Stelle steht die Größe des Dokuments. In diesem Fall beträgt die Größe 4961x7016 Pixel. Die darauf folgenden Befehle sind für den soeben beschriebenen PDF-Export zuständig: `processing.pdf.PGraphicsPDF` bestimmt den Dateitypen der exportierten Datei, also ein PDF. `timestamp()+"BILD.pdf"`, legt die Benennung der Datei fest, `timestamp()` bestimmt Datum und Uhrzeit und `"BILD.pdf"` den frei wählbaren Namen.

Im `void setup` werden noch weitere Faktoren voreingestellt. Hier wird beispielsweise auch das Bild in die Datei geladen. Zuvor wurde in *Processing* definiert, dass `img` für eine Bild-datei steht. Jetzt wird hinzugefügt, dass `img` mit einem bestimmten Bild aus unseren Daten belegt werden soll. In diesem Fall die `"i.jpg"`

An diesem Punkt, dem `void draw`, beginnt – das lässt sich aus der Bezeichnung schon entnehmen – die Zeichnung des Dokuments. In diesem Abschnitt des Codes werden nun die wirklich visuell entscheidenden Dinge eingefügt. Startet man den Prozess, öffnet sich ein weiteres Fenster, in dem der visuelle Output angezeigt wird, der zuvor in `void draw` definiert wurde. Der Code, der in diesem Arbeitsabschnitt geschrieben wird, wiederholt sich so lange, bis man dieses Fenster wieder schließt. Dieser Vorgang wird zu einem späteren Zeitpunkt noch einmal wichtig werden:

```

11 void draw() {
12
13     randomX = r.nextInt(1400);
14     randomY = r.nextInt(1350);
15     cropX = constrain(randomX, 0, 1400);
16     cropY = constrain(randomY, 0, 1350);
17     cropX = randomX;
18     cropY = randomY;
19     image(img, 0, 0);
20     noFill();
21     stroke(255);
22     rect(cropX, cropY, tileWidth, tileHeight);
23 }

```



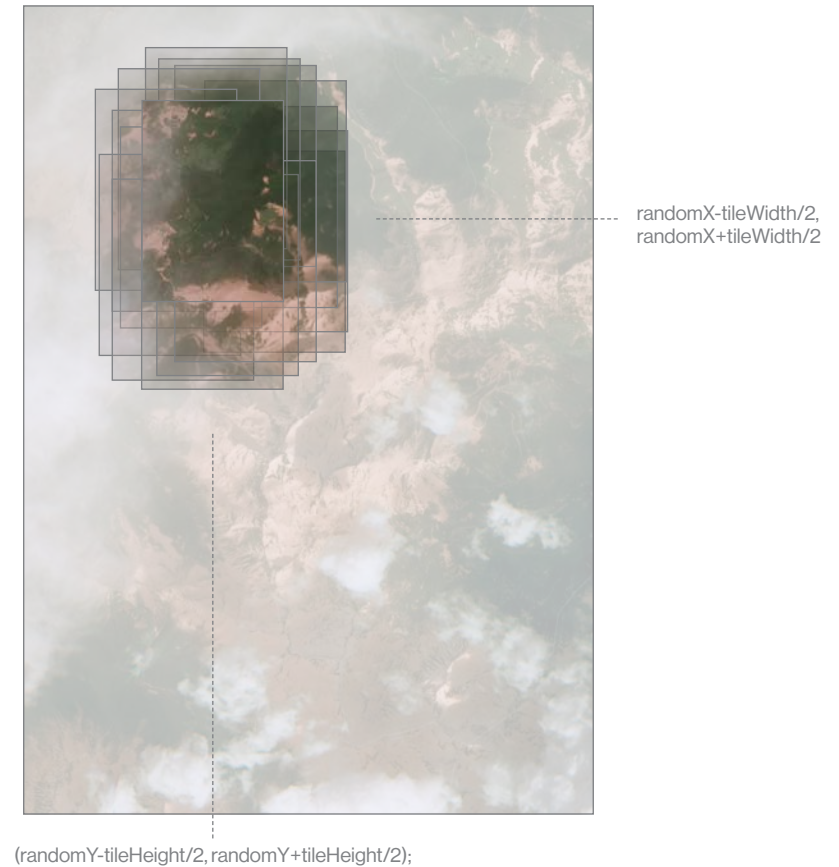
Dieser Teilabschnitt des Codes ist dafür zuständig, einen bestimmten Bereich des Bildes auszuwählen. Dies wird umgesetzt, in dem man sowohl auf der X-Achse als auch auf der Y-Achse ein minimalen sowie einen maximalen Wert angibt. Zwischen diesen beiden Werten soll *Processing* zufällige Koordinaten bestimmen. Diese Koordinaten legen den ausgewählten Bildausschnitt fest, der durch `cropX` und `cropY` definiert wird.

Hierbei handelt es sich um den elementarsten Arbeitsschritt. Um den zuvor ausgewählten Bildausschnitt herum, werden nun weitere 16 Teilausschnitte definiert. Diese Ausschnitte werden anschließend mit Hilfe des Rasters über die gesamte Fläche hinweg platziert:

```

7 int i = 0;
8 for (int gridY = 0; gridY < tileCountY; gridY++){
9     for (int gridX = 0; gridX < tileCountX; gridX++){
10        if (randomMode){
11            cropX = (int) random(randomX-tileWidth/2,
12                                randomX+tileWidth/2);
13            cropY = (int) random(randomY-tileHeight/2,
14                                randomY+tileHeight/2);
15        }
16        cropX = constrain(cropX, 0, width-tileWidth);
17        cropY = constrain(cropY, 0, height-tileHeight);

```



Durch den Befehl `(int) random(randomX-tileWidth/2, randomX+tileWidth/2);` werden 16 weitere, zufällig ausgewählte Ausschnitte rund um `cropX` und `cropY` festgelegt. Durch `tileWidth/2` wird die dafür vorgesehene Positionierung angegeben.

```
1  tileWidth = width/tileCountY;  
2  tileHeight = height/tileCountX;  
3  tileCount = tileCountX * tileCountY;  
4  imageTiles = new PImage[tileCount];  
  
5  
6  imageTiles[i++] = img.get(cropX, cropY, tileWidth,  
7  tileHeight);  
  
8  
9  else {  
10     int i = 0;  
11     for (int gridY = 0; gridY < tileCountY; gridY++){  
12         for (int gridX = 0; gridX < tileCountX; gridX++){  
13             image(imageTiles[i], gridX*tileWidth, gridY*tile-  
14             Height);  
15         i++;  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30
```



Z.1+

Zu Beginn haben wir den Befehl `tileCountY` mit dem Integer (der Zahl) 4 belegt. Durch `tileWidth = width/tileCountY` wird angegeben, dass die Breite der ausgewählten Fläche, ein Viertel der gesamten Breite betragen soll. Das Gleiche wird mit `tileHeight = height/tileCountX` auch bezüglich der Höhe angegeben. Dadurch entsteht das 4x4 Raster, welches bestimmt, dass insgesamt 16 Abschnitte gewählt werden müssen.

1 Abschließend wird angegeben, dass *Processing* all diese
2 Schritte ausführt und nach Abschluss eines Durchlaufs das
3 Ergebnis speichert und das ausführende Fenster wieder
4 schließen soll:

```
5  
6 PGraphicsPDF pdf = (PGraphicsPDF) g;  
7     if (frameCount == 5) {  
8         exit();  
9     } else {  
10        pdf.nextPage();  
11    }  
12 }
```

Z.7

Wie bereits erwähnt, werden alle Vorgänge, die im Bereich `void draw` ablaufen unendlich oft wiederholt. Dieser Codeabschnitt sorgt nun dafür, dass sich kein weiteres Darstellungsfenster öffnet. Mit `if (frameCount == 5)` wird angegeben, dass *Processing* fünf Mal den `draw` Inhalt wiederholen soll. Nach jedem Durchlauf wird eine PDF gespeichert. Wurden alle fünf Durchläufe ausgeführt, wird der Prozess beendet und eine fünfseitige PDF mit den jeweils generierten Motiven ausgegeben.

Der Layoutgenerator ist für den restlichen Content des Plakats zuständig. Dazu gehören der Text, dessen Platzierung, die dazugehörigen Balken sowie der Inhalt, der in Form von Texten umgesetzt wird. Die Erläuterung der Umsetzung, wird in zwei Abschnitte unterteilt. Beginnend mit dem Layout. Darauf folgt das Einbetten der Inhalte.

Wie bereits beim Bildgenerator müssen zu Beginn die Grundelemente festgelegt werden:

```

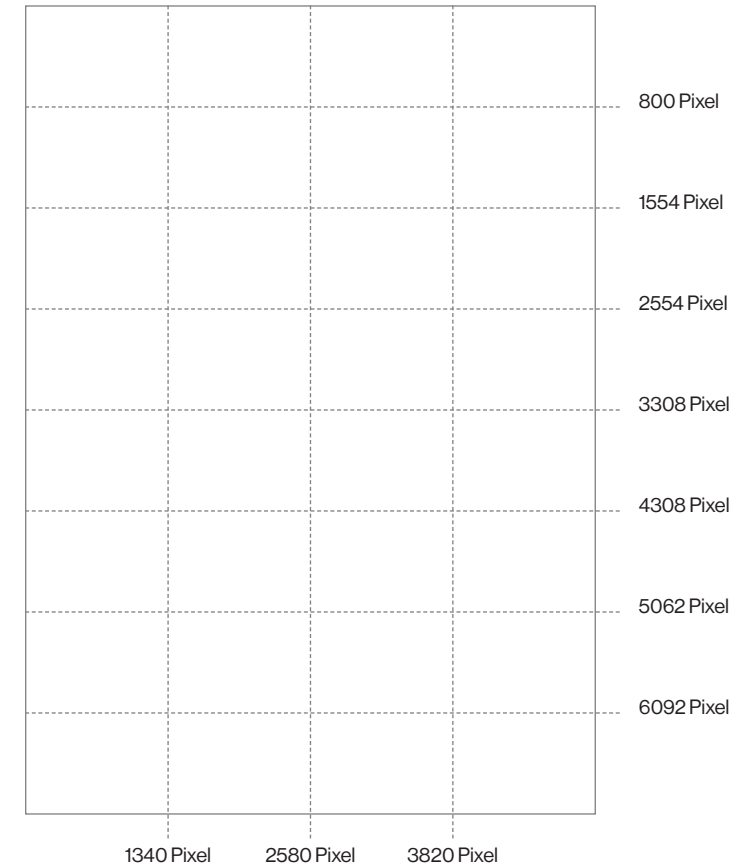
PFont sansmicro;
PFont sanssubhead;
PFont sanshead;
int[] subY = {800, 2554, 4308, 6092};
int[] microY = {1633, 3387, 5141, 6925};
int[] headY = {1554, 3308, 5062, 6846};

int[] headYrect = {1650, 3404, 5158, 6942};
int[] subYrect = {839, 2593, 4347, 6131};
int[] microYrect = {1650, 3404, 5158, 6942};

```

Z.11+

Z.14+



Im Gegensatz zu **PImage** aus dem Bildgenerator, werden hier mit Hilfe von **PFont** die Schriften voreingestellt. In diesem Fall wurden drei verschiedene Schriftgrößen genutzt, sodass drei **PFont** Elemente benötigt wurden.

Als nächstes folgen sechs sogenannte *Arrays*. In diesen *Arrays* wurden die Koordinaten definiert, in denen sich die verschiedenen Texte bewegen sollen. Dazu wurde das gesamte Plakat zunächst vermessen. Dadurch konnten die Positionen genau bestimmt und in einem Raster festgelegt werden.

Im nächsten Abschnitt sind wieder die `public void settings` sowie das `void setup` vertreten:

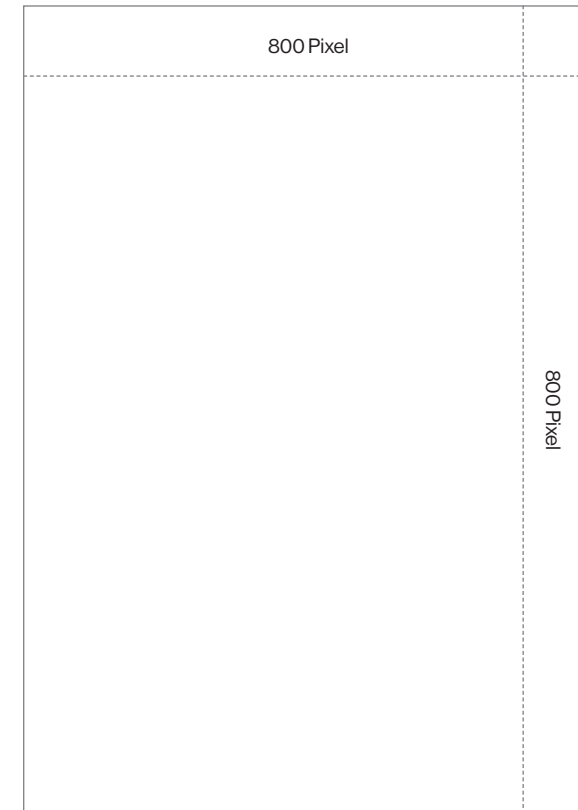
```

1 public void settings() {
2     size(4961, 7016, "processing.pdf.PGraphicsPDF",
3         "210215_183734BILD.pdf");
4 }
5
6 void setup() {
7     sansmicro = createFont("NHaasGroteskDSPro-55Rg.
8         otf",1000);
9     sanssubhead = createFont("NHaasGroteskDSPro-
10        55Rg.otf",1000);
11     sanshead = createFont(„NHaasGroteskDSPro-55Rg.
12        otf",1000);
13 }
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

```

Z. 4+

Z. 9+



Da der Text zum Schluss auf dem gesamten Plakat verteilt werden soll, wird in den `public void settings` die gleiche Einstellung hinterlegt, wie im Bildgenerator.

Im `void setup` werden nun die verschiedenen Fonts in *Processing* geladen.

Der Abschnitt `void draw` beginnt im Layoutgenerator mit einer Reihe von Random Integern, die aus den oben festgelegten *Arrays* eine zufällige Zahl auswählen. Diese Befehle werden von sogenannten *While-Schleifen* gefolgt, die dafür sorgen, dass sich bestimmte Koordinaten nicht überschneiden, um zu verhindern, dass Texte übereinander platziert werden:

```

int r7 = int(random(headY.length));
int r1 = int(random(supY.length));
int r2 = int(random(supY.length));
int r3 = int(random(microY.length));
int r4 = int(random(microY.length));
int r5 = int(random(microY.length));
int r6 = int(random(microY.length));

while (r6 == r7 ){
  r6 = int(random(3));
}
while (r1 == r7){
  r1 = int(random(3));
}
while (r2 == r7){
  r2 = int(random(3));
}
while (r3 == r7){
  r3 = int(random(3));
}
while (r4 == r7 ){
  r4 = int(random(3));
}

```

Z.8

Z.8

Z.16

Z.17

Wie bereits beobachtet, werden normalerweise `int` (Integer) zu Beginn eines Dokumentes festgelegt. In diesem Fall mussten sie jedoch manuell im `void draw` platziert werden. Wie bereits im Bildgenerator erwähnt, wird dieser Abschnitt immer wiederholt. Da diese Integer die Platzierung des Textes bestimmen, müssen diese nach jedem Durchlauf neu gewürfelt werden.

`int r7 = int` definiert, dass die Aussage `r7` eine bestimmte Zahl bedeutet.

`(random(headY.length))` deklariert, um welche Zahl es sich dabei handeln soll. Ausgewählt wird eine zufällige Zahl aus dem oben deklarierten *Array* `headY`

Eine *While-Schleife* ist dafür zuständig, den Kontrollfluss des Codes zu verändern. In diesem Fall sorgt die *While-Schleife* dafür, dass die Random Integer – die zuvor generiert werden – nicht das gleiche Ergebnis erzielen.

`while (r6 == r7)` legt fest, bei welcher Situation diese Schleife ausgeführt werden soll. In diesem Fall, sobald die gewürfelte Zahl bei `r6` und `r7` übereinstimmen.

`r6 = int(random(3))`: Ist die Bedingung (`while (r6 == r7)`) erfüllt, würfelt *Processing* mit diesem Befehl `r6` so oft nochmal neu aus, bis die beiden Zahlen nicht mehr identisch sind.

1 Im nachfolgenden Bereich werden nun die Texte sowie die
2 Balken definiert und platziert. An dieser Stelle der Arbeit wird
3 exemplarisch lediglich ein Teil dieses Codes aufgezeigt und
4 erklärt, da im Grunde alle Textdefinitionen die gleichen sind:

```
5 textFont(sanssuphead);  
6 textAlign(LEFT,BOTTOM);  
7 textSize(340);  
8 textLeading(340);
```

Z.6

Z.7

Z.8

Z.9

An erster Stelle wird hier, am Beispiel der Subheadline, die Schrift definiert.

`textFont(sanssuphead)`: Die zuvor ausgewählte und geladene Schrift, wird eingesetzt.

`textAlign(LEFT,BOTTOM)`: Die Ausrichtung des Textes wird festgelegt.

`textSize(340)`: Definition der Schriftgröße.

`textLeading(340)`: Definition des Zeilenabstands.

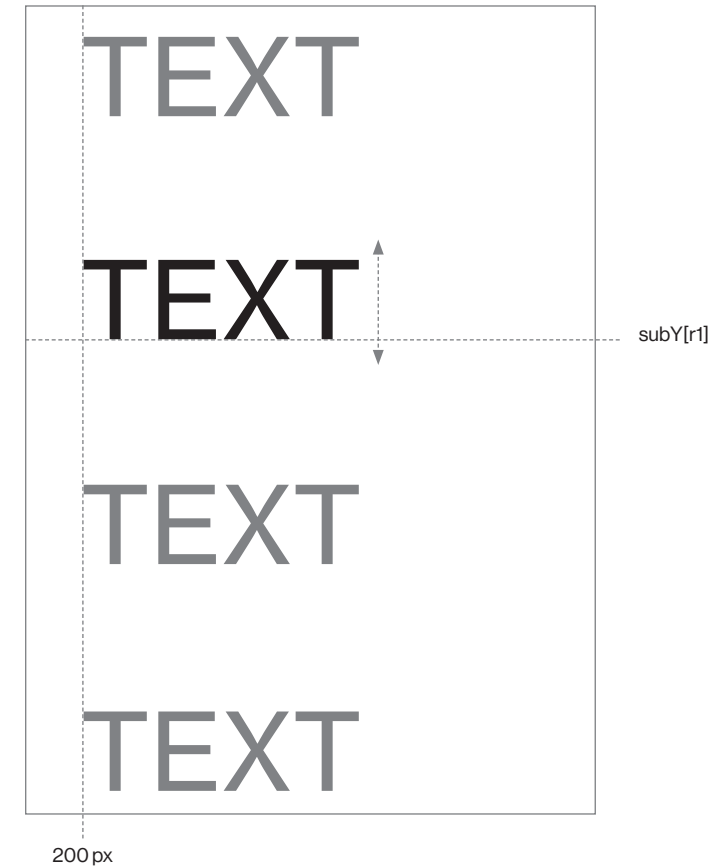
```

1  push();
2  translate(200,subY[r1]);
3  text(date,0,0);
4  pop();
5
6  push();
7  translate(1340,subY[r2]);
8  text(name,0,0);
9  pop();

```

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

z.2



Nun wird der Text positioniert. Durch das Einklammern des Textes mithilfe der `push` und `pop` Matrix, wird festgelegt, dass die Koordinaten die wir in `translate` angeben, ausschließlich für diesen Bereich gelten.

`translate(200,subY[r1])`: Die Angaben in der Klammer stehen für die X sowie Y Koordinate. Für X wird der Pixelwert 200 eingetragen. Für Y wird nun das *Array* eingefügt, aus dem der random Integer `r1` eine zufällige Y Koordinate auswählt und einfügt. Somit ist es möglich, dass der Text bei jedem Durchlauf seine Position auf der Y Koordinate verändert.

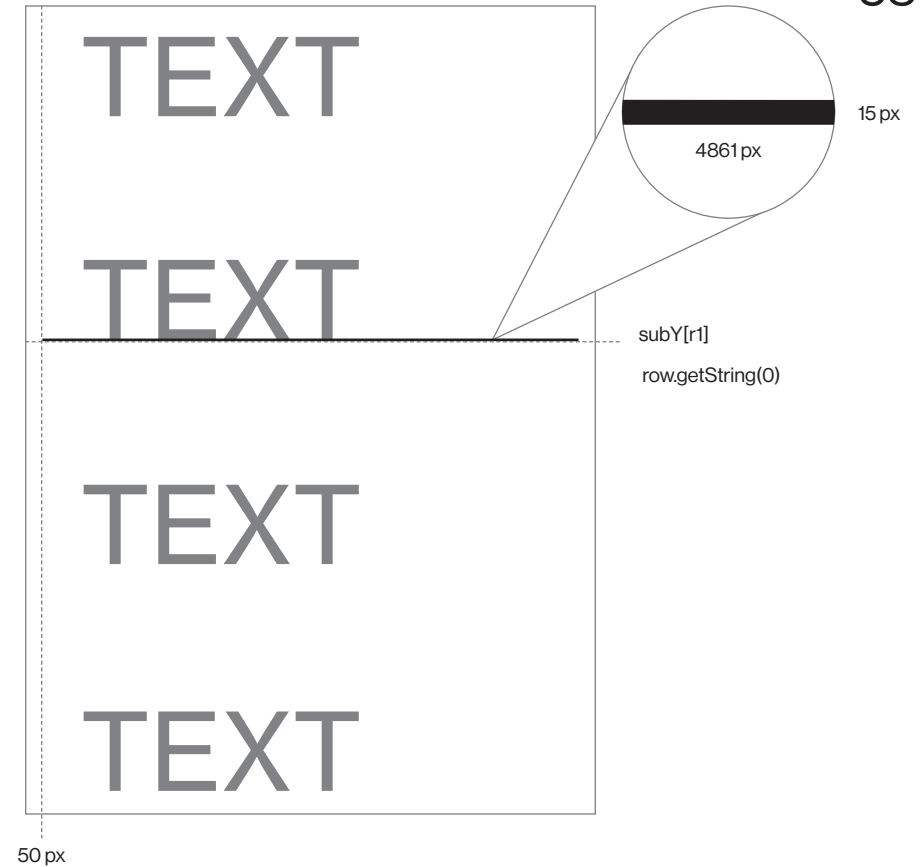
Zum Schluss werden nun die dazugehörigen Balken definiert:

```

1  stroke(255);
2
3  rect(50 ,headYrect[r7], 4861, 15);
4  rect(50 ,subYrect[r1], 4861, 15);
5  rect(50 ,subYrect[r2], 4861, 15);
6  rect(50 ,microYrect[r3], 4861, 15);
7  rect(50 ,microYrect[r5], 4861, 15);
8  rect(50 ,microYrect[r6], 4861, 15);
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

```

Plakat



Z.4

`rect(50 ,headYrect[r7], 4861, 15)`: Mit dem Befehl `rect` wird ein Rechteck gezeichnet. Die ersten beiden Angaben stehen wieder für die Koordinaten. Y soll ebenso wie beim Text, immer neu gewürfelt werden und verfolgt durch das Einbinden von `r7` immer den Headline-Text.

1 Mit dem Abschluss dieses Codes, wurde ein Generator er-
2 stellt, der von alleine Texte platziert und diesen Vorgang so
3 häufig, wie gewünscht, umsetzt. Nun fehlen aber noch die
4 Inhalte, die eingesetzt werden sollen. Diese Inhalte sollen aus
5 einer *Excel* Datei herausgezogen werden. Für dieses Verfahren
6 habe ich mich bewusst entschieden, da es einige Vorteile mit
7 sich bringt. In vielen Fällen wird der Text manuell in den Code
8 eingetragen. Diese Methode ist jedoch relativ umständlich
9 und sorgt dafür, dass bei jeder Änderung wieder in *Processing*
10 einzelne Zeilen umgeschrieben werden müssen. Greift *Pro-*
11 *cessing* jedoch direkt auf eine *Excel* Datei zu, können dort
12 immer neue Inhalte eingetragen werden die wiederum auto-
13 matisiert importiert werden. Diese Herangehensweise vertieft
14 somit auch nochmal die Grundidee des Projektes, möglichst
15 viele Arbeitsschritte automatisiert ablaufen zu lassen.
16 Hierzu wurde eine *Google Spreadsheet/Excel* Datei mit den
17 nötigen Inhalten gefüllt und in eine Reihenfolge gebracht. Nun
18 konnte die Tabelle in *Processing* geladen und mit dem ent-
19 sprechenden Code festgelegt und einsortiert werden. Hierfür
20 wurde die Funktion eines Strings genutzt. Mit Hilfe eines
21 Strings können bestimmte Anweisungen mit geschriebenem
22 Text belegt werden. Dafür werden zu Beginn des Codes die
23 Strings definiert.

```
24  
25 String location;  
26 String name;  
27 String data;  
28 String credit;  
29 String category;  
30 String tag;
```

Mit diesen Befehlen wird zum Beispiel festgelegt, dass der Begriff *location* ein String ist. Durch *location* können so bestimmte Text aufgerufen werden.

Im `void setup` muss nun die gewünschte Tabelle geladen und die bereits eingestellten Strings einer bestimmten Zelle zugeordnet werden:

```
Table table = loadTable("CONTENT5.csv");
TableRow row = table.getRow(1);
```

```
location = row.getString(0);
name = row.getString(1);
data = row.getString(2);
credit = row.getString(3);
category = row.getString(4);
tag = row.getString(5);
```


z.6

In der erste Zeile wird, wie bereits mit Bild und Font geschehen, nun auch die Tabelle in die *Processing* Datei geladen. Mit `TableRow row = table.getRow(1);` wird festgelegt, welche Zeile aus der geladenen Tabelle angesprochen werden soll.

z.8

Nun folgen die Einstellungen der einzelnen Stings. `location = row.getString(0);` sagt beispielsweise aus, dass der String `location` mit dem Inhalt aus der ersten Reihe und der nullten (was in Processing immer für die erste steht) Zeile befüllt werden soll.

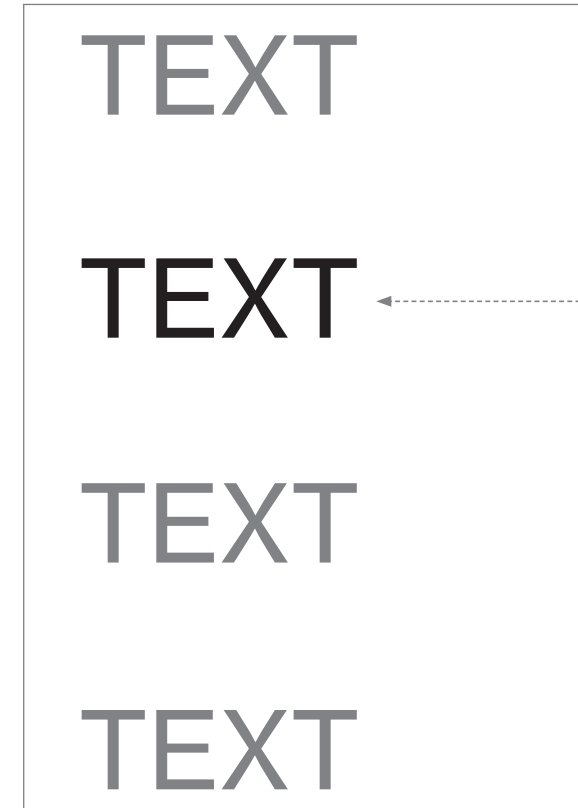
1 Jetzt können, zurück in den Text-Einstellungen, die gefüllten
 2 Strings eintragen werden:

```
3
4 push();
5 translate(200,headY[r7]);
6 text(location,0,0);
7 pop();
```

Tabelle

location

Plakat

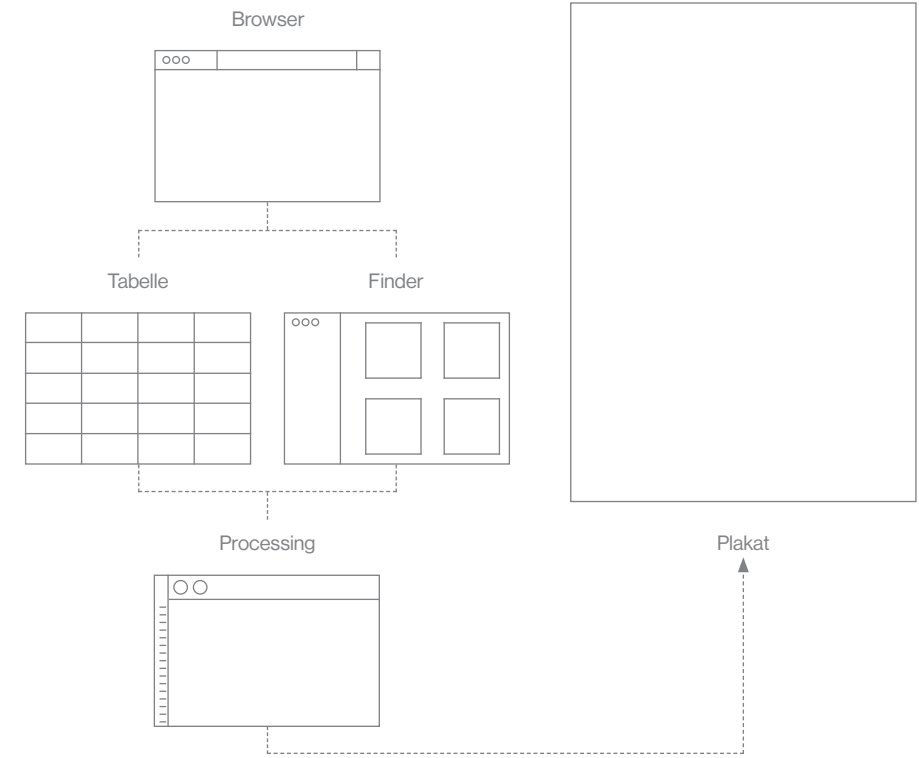


Z.6

Anhand dieses Codes ist nun ersichtlich, dass bei `text` als Inhalt `location` angegeben wurde. Nun weiß *Processing*, dass an dieser Stelle derjenige String `location` eingetragen werden soll, der mit der ersten Reihe und der ersten Zeile aus der importierten Tabelle hinterlegt ist.

1 Nach dem Abschluss dieser beiden Generatoren, lag ein Verfahren
 2 vor, was zum einen ein Bild liefert und zum anderen das dazuge-
 3 hörige Layout erstellt. Um das nachträgliche Zusammenfügen jedoch
 4 zu vermeiden, entschied ich mich dafür, diese beiden Verfahren
 5 in einen gemeinsamen Code zu schreiben. Ein einzelnes, über-
 6 geordnetes Verfahren zu haben, bringt den großen Vorteil mit sich,
 7 die Erstellung des Plakats noch autonomer gestalten zu können.
 8 Denn der damit neu gestaltete Code, wird sowohl das Bild als auch
 9 den Text importieren, wodurch direkt ein fertiges Plakat erstellt
 10 wird.

11
 12 Dementsprechend wurde eine *Processing* Datei angelegt, in
 13 der Stück für Stück die beiden Codes zusammengefügt wur-
 14 den. Hierbei mussten jedoch einige Befehle umgeschrieben
 15 und ersetzt werden. Dieser Prozess muss mit äußerster Vor-
 16 sicht und viel Geduld durchgeführt werden, da Dopplungen
 17 oder ungenaue Zuweisungen der einzelnen Befehle, schnell
 18 zu Fehlern führen, die dafür sorgen, dass der gesamte Code
 19 nicht mehr funktioniert.



Ist dieser Schritt getan, ist eine autonome Systemkette entstanden, die von alleine ein Plakat entwirft. Der Beginn dieser Kette liegt im Browser. Von dort aus werden die Daten heruntergeladen. Die Texte werden in die Tabelle eingefügt und das Bild in den entsprechenden Ordner importiert. Von hier aus nimmt sich *Processing* die Daten, verarbeitet sie weiter und wirft eine PDF mit dem fertigen Plakat aus.

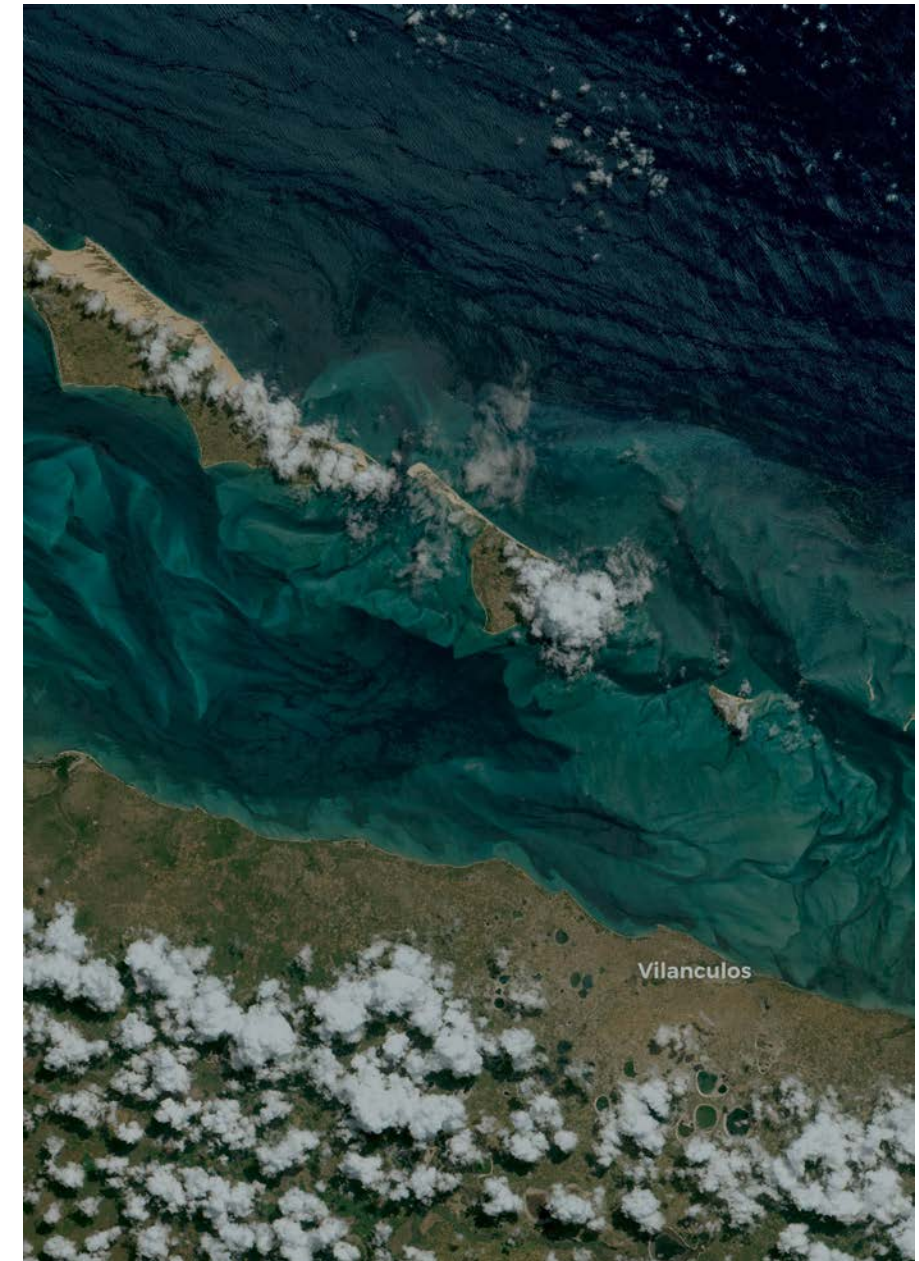
Bazaruto Archipelago National Park

Spanning 1,430 km² of seascape and five islands off the coastline of Mozambique, the Bazaruto Archipelago National Park encompasses terrestrial and marine habitats of unique ecological value

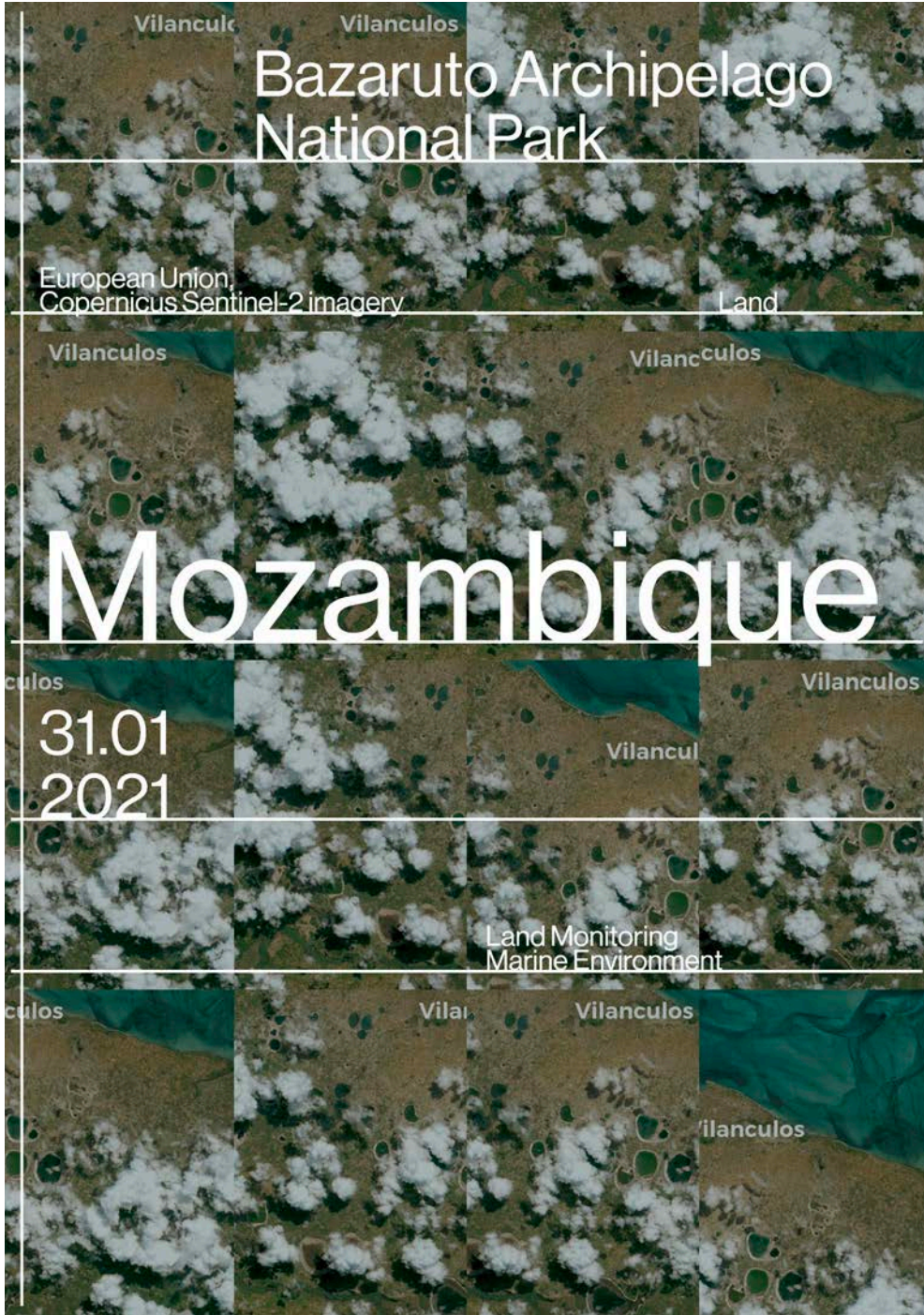
The Park was created in 1971 and is mandated to protect species of high conservation value, including sharks, dolphins, marine turtles, and corals. It is home to the largest and only viable population of dugongs (a marine mammal commonly known as "sea cows") in the Western Indian Ocean.

This true colour image was captured by one of the Copernicus Sentinel-2 satellites on 5 January 2021.

Open data from the Sentinel satellites are used to monitor fragile ecosystems around the world and for bathymetric applications.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Natural Events

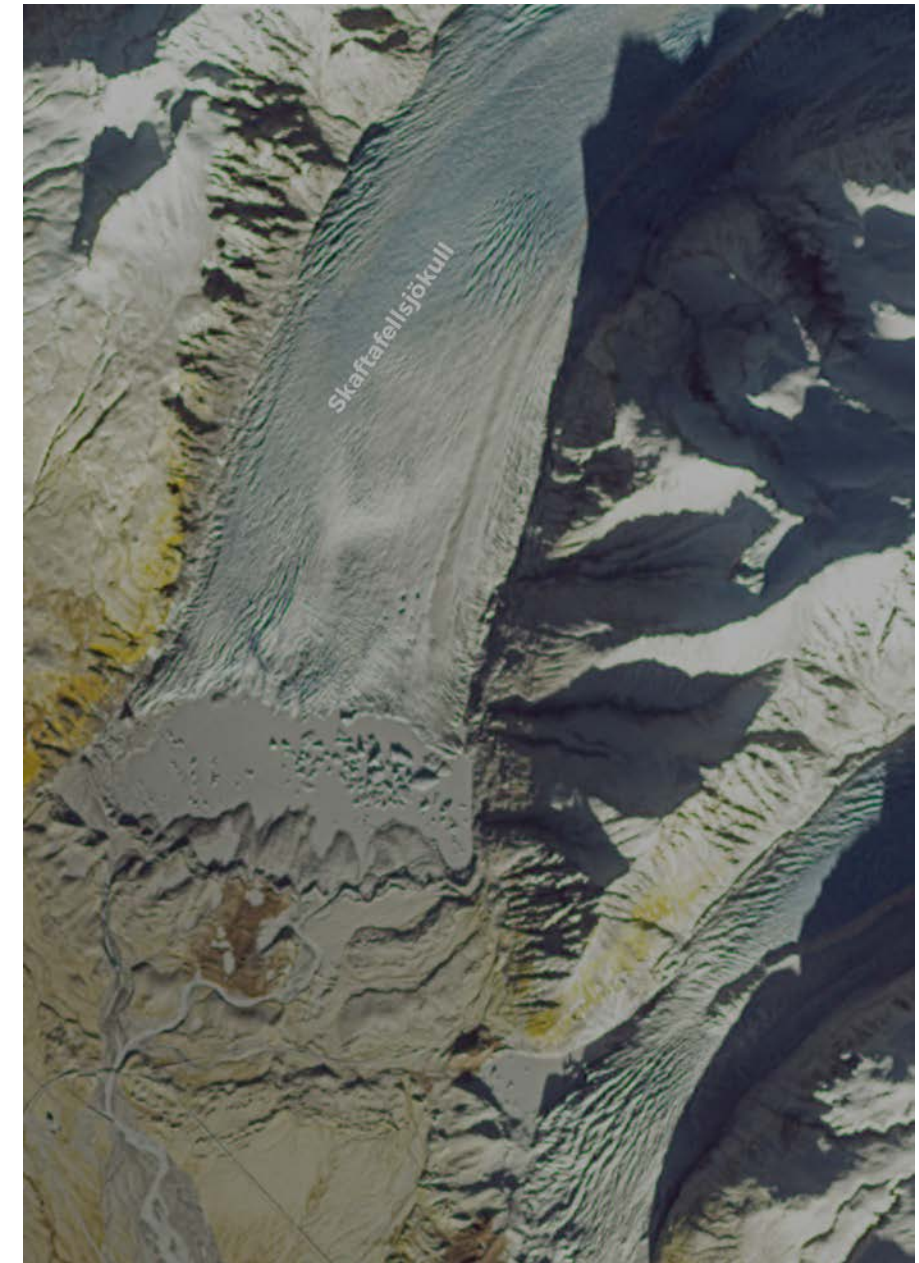
Tag Coastal Erosion
 Floods
 Land Monitoring

Skaftafellsjökull glacier

Located in the southeast of Iceland, Skaftafell is a preservation area within the Vatnajökull National Park, which measures approximately 14% of the total surface area of Iceland. Due to its stunning variety of landscape features, created by an interplay of fluvial, glacial, volcanic and geothermal activity, the region has also been designated a UNESCO World Heritage Site. Its glaciers stem from the Vatnajökull ice cap, Europe's largest glacier outside of the Arctic Circle. According to recent research, the ice cap has retreated dramatically in the past 30 years as a consequence of climate change, by a total of about 400 km².

This image, which was acquired by one of the two Copernicus Sentinel-2 satellites on 31 January 2021, is the first cloudless image to capture the Icelandic glaciers since the 2020-2021 darkness period ended. It shows two Skaftafell glaciers, Skaftafellsjökull in the middle of the image and Svínafellsjökull on the right.

The Copernicus Sentinel-2 mission, which includes two twin satellites, provides multi-spectral data over 13 different spectral bands and at a maximum spatial resolution of 10 metres. The granular images it helps create can then be used to detect changes in land surface in high resolution even at northern latitudes, helping to monitor the melting of glaciers.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Cryosphere

Tag Climate Change Glaciers

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

Tag Floods Severe Storms

Western Cape

Africa is amongst the continents most vulnerable to climate change. Aside from the resultant environmental damage, the temperature rise by itself could negatively affect economic productivity in the coming years.

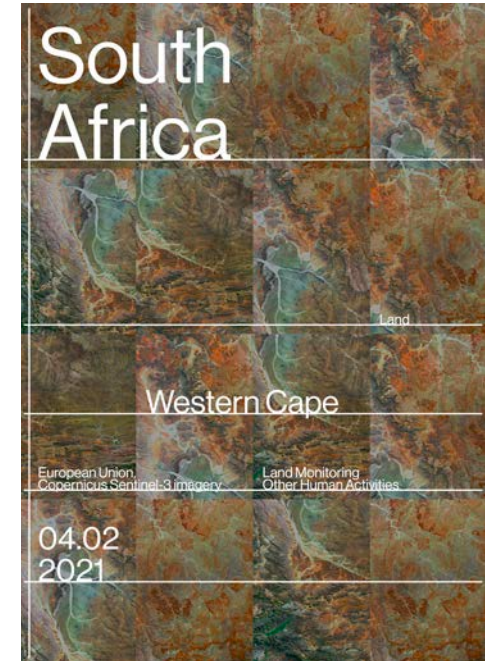
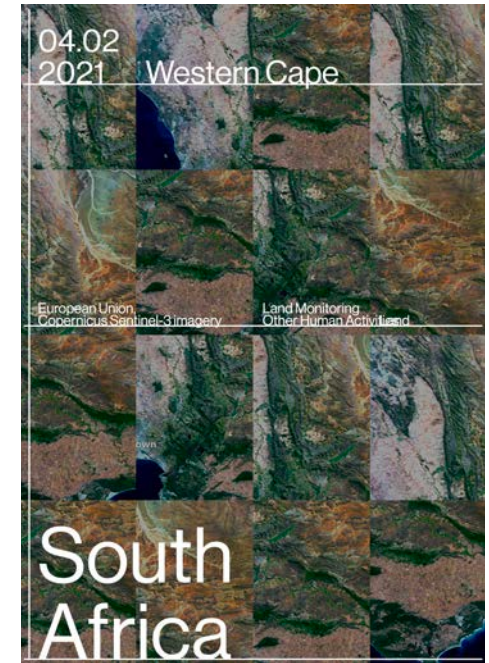
According to a recent study coordinated by the CMCC Foundation and the European Institute on Economics and the Environment (RFF-CMCC), in South Africa climate change could lead to a reduction in GDP per capita of up to 20% by the end of this century, as compared to an idealised scenario without any of its impacts.

This image, acquired by one of the Copernicus Sentinel-3 satellites on 2 February 2021, shows the Western Cape province of South Africa. Between 2015 and 2018, this area faced one of the worst droughts to have impacted the country. Scientists attribute the water scarcity in the region to the increasingly severe effects of climate change.

Open Data supplied by the Copernicus Sentinel satellites and services allow for the monitoring of the impacts of climate change around the world.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-3 imagery
 Category Land

Tag Land Monitoring Other Human Activities

1 New York City covered with snow

2 Winter Storm Orlena hit the United States in early February
3 2021.

4 In the region between the Southern Great Lakes and New
5 England, more than 110 million people found themselves under
6 winter weather alerts on 1 February.

7 The snow blanket that covered New York City brought along
8 severe disruptions in public transport and cancelled flights,
9 as well as closed schools and COVID-19 vaccination sites.

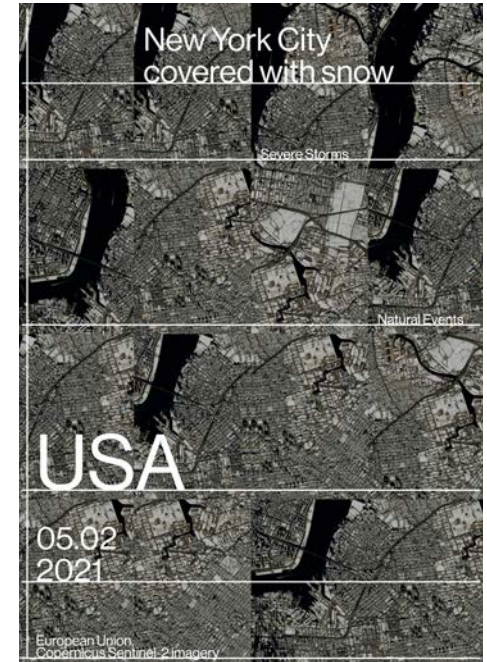
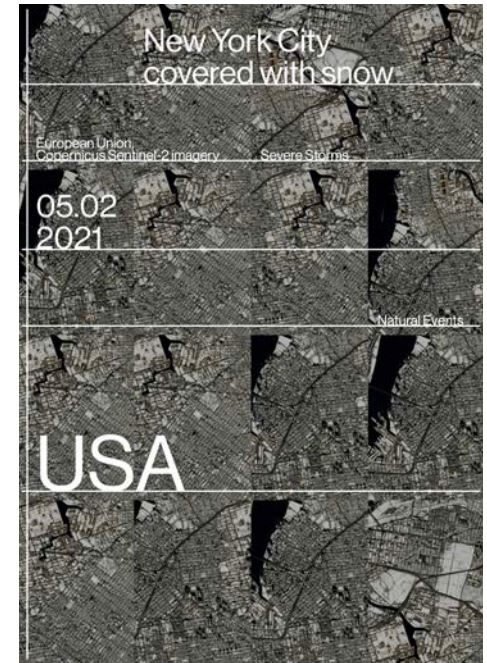
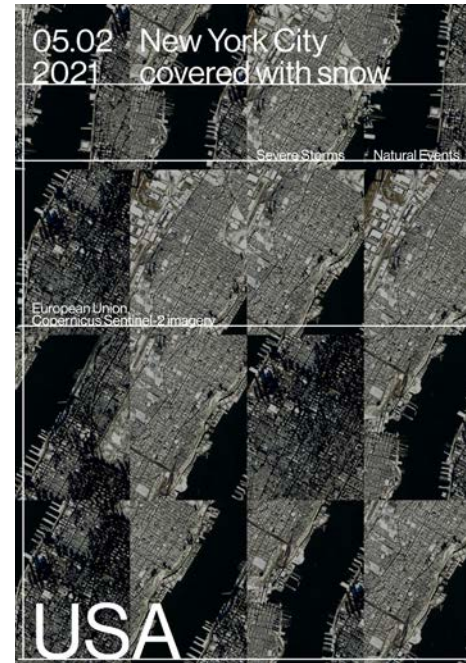
10 On the morning of 2 February, more than 40 cm of snow cover
11 was measured in Central Park, marking this snowstorm as
12 one of the top 20 to ever befall it.

13 This image, acquired by one of the Copernicus Sentinel-2
14 satellites on 4 February 2021, shows a snow-covered New
15 York City.

16 Data captured by the Copernicus Sentinel-2 satellites are
17 particularly suitable for monitoring and mapping snow cover
18 around the world.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Natural Events

Tag Severe Storms

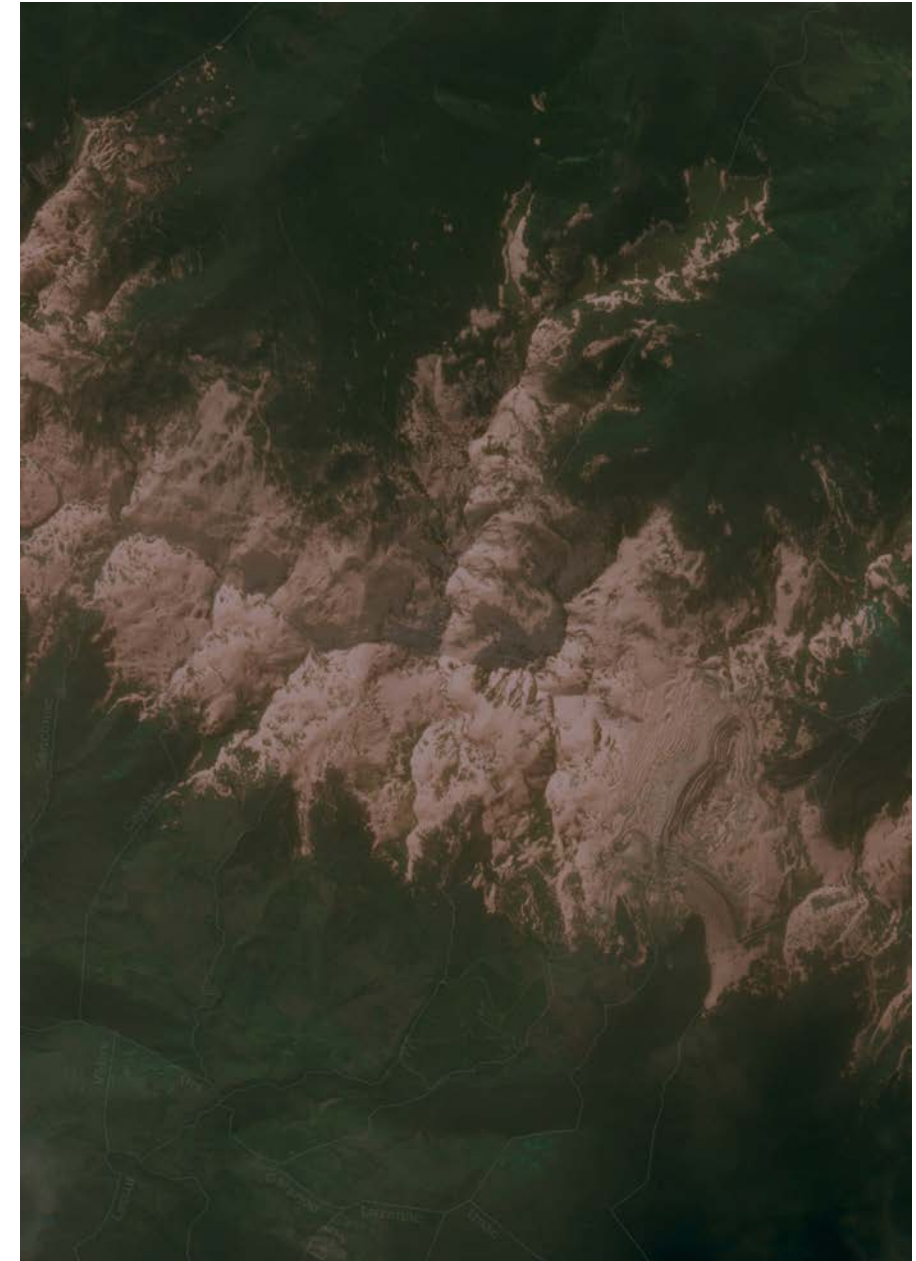
Saharan dust turns Pyrénées

A massive Saharan dust plume with high values of dust aerosol and surface PM10 is sweeping across southwestern Europe. It has painted the snow in the Pyrénées mountain range (and also in the Alps) in brownish colours and turned the skies orange. This episode had been forecasted by the Copernicus Atmosphere Monitoring Service.

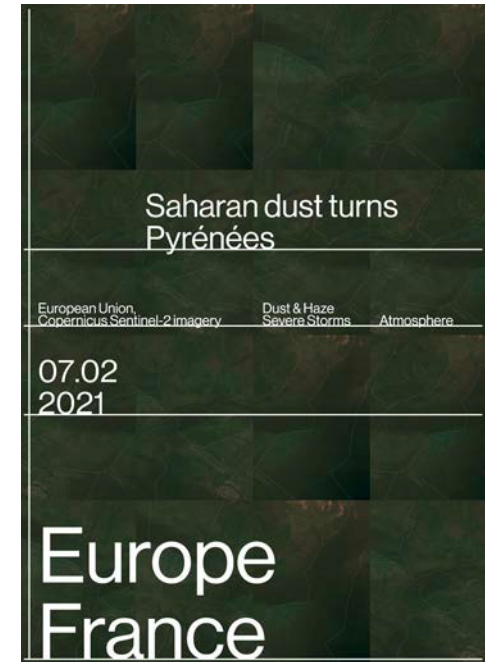
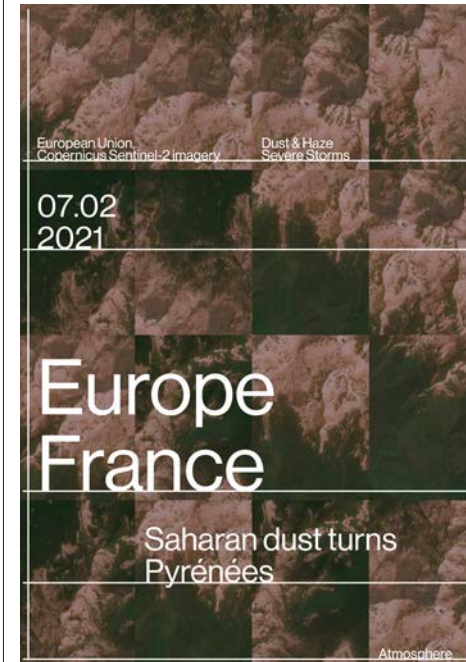
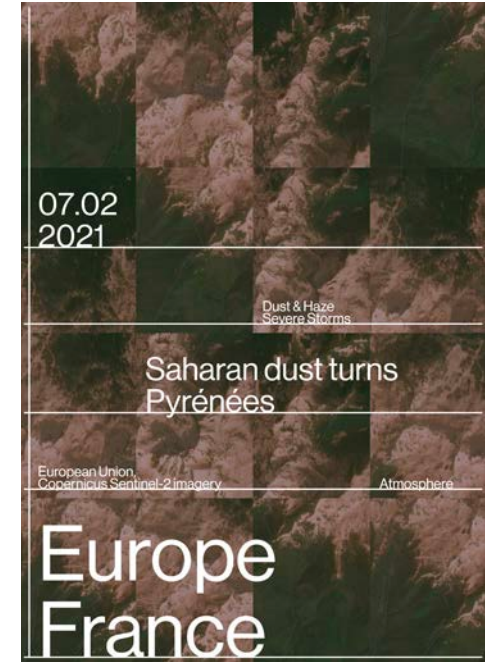
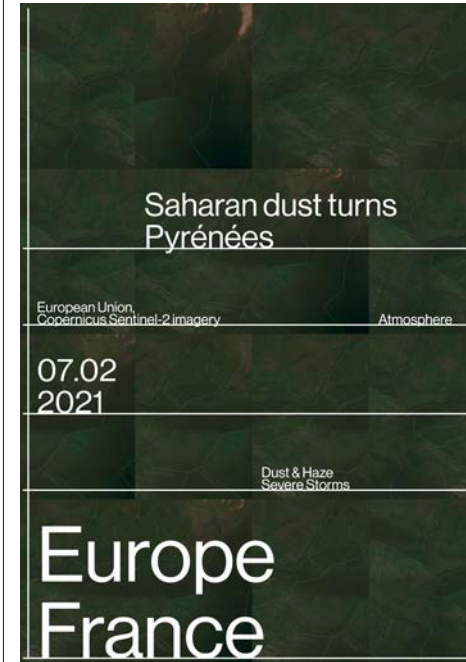
Such events have been documented in the scientific literature. For example, a 2019 study entitled, “Saharan dust events in the European Alps: role in snowmelt and geochemical characterization” was published in *The Cryosphere*. The paper concluded that Saharan dust reduces snow cover duration in Europe. Through a process called the snow-albedo feedback, dust darkens snow making it less reflective, which leads to increased snowmelt over time.

These dust episodes also represent a major environmental and health hazard.

This true colours image, acquired by one of the Copernicus Sentinel-2 satellites on 6 February 2021, shows dust-covered snow in the Monts d'Olmes Pyrénées mountain resort in southeast France.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Atmosphere

Tag Dust & Haze
Severe Storms

1 Saharan dust turns Pyrénées 2

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

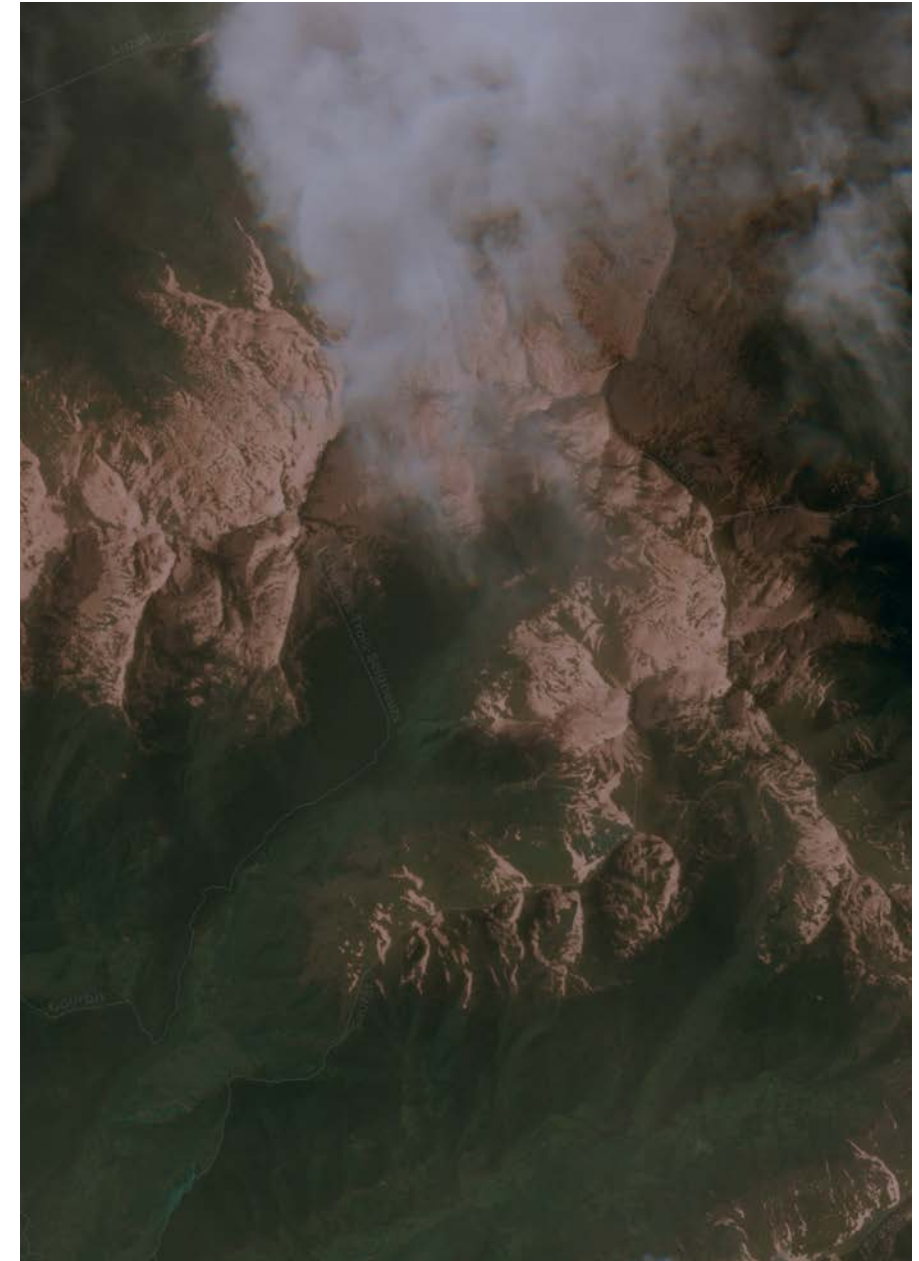
26

27

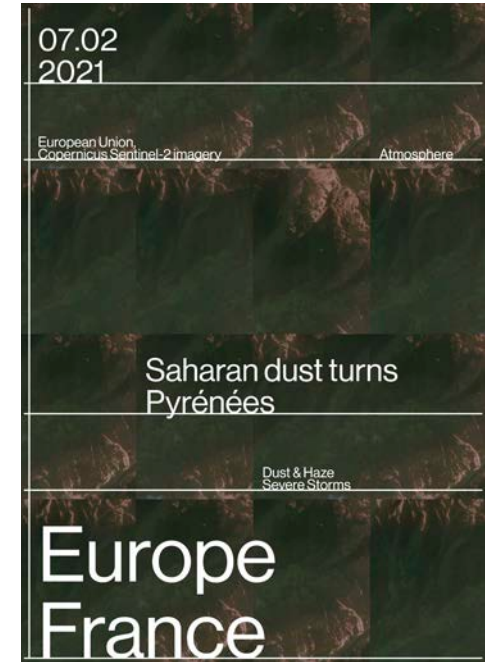
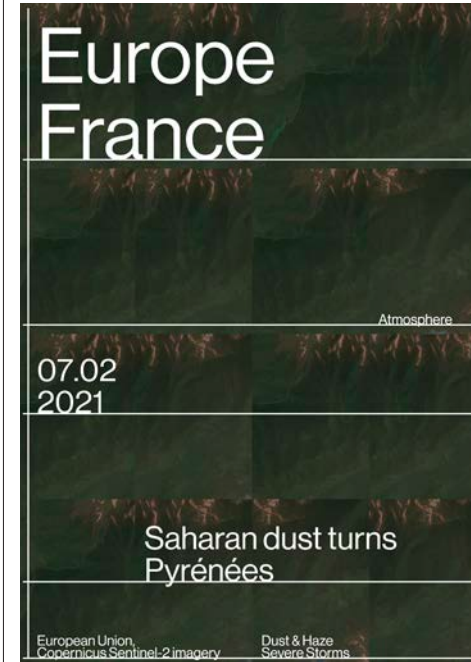
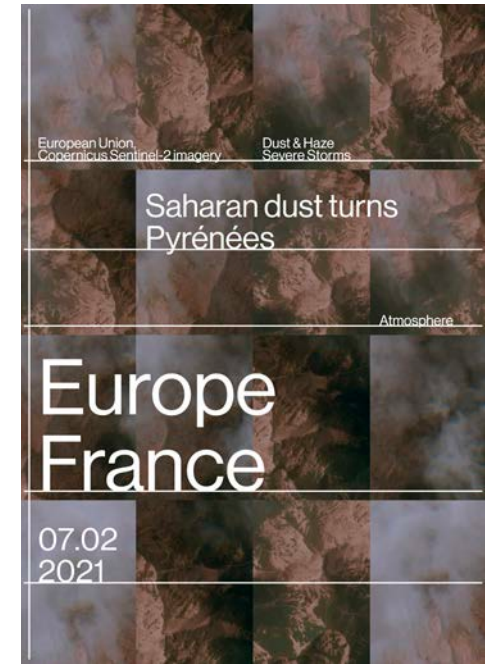
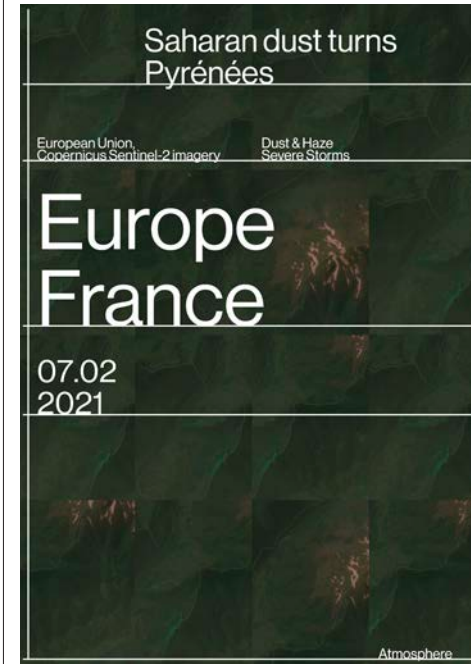
28

29

30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Atmosphere

Tag Dust & Haze
Severe Storms

1 Saharan dust turns Pyrénées 3

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

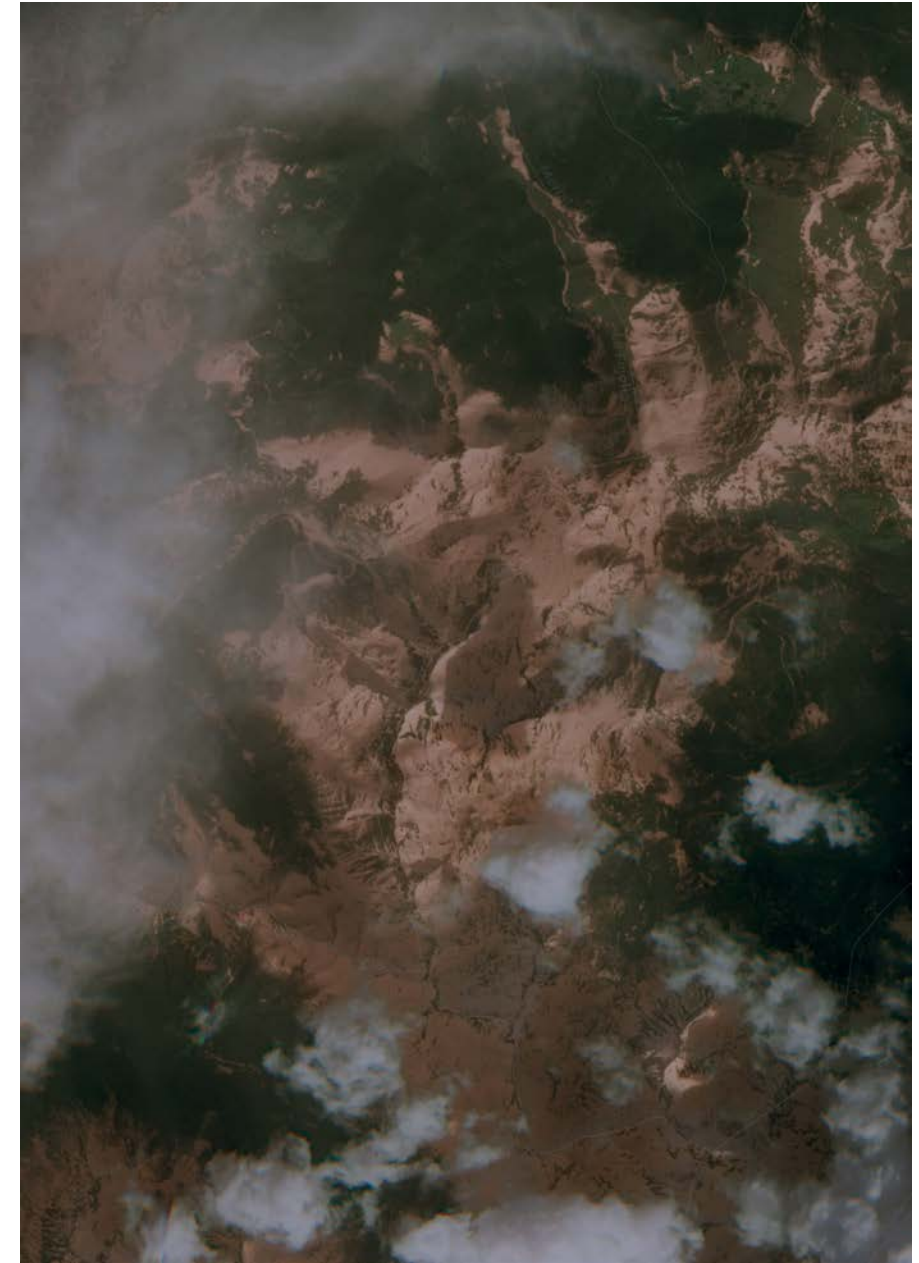
26

27

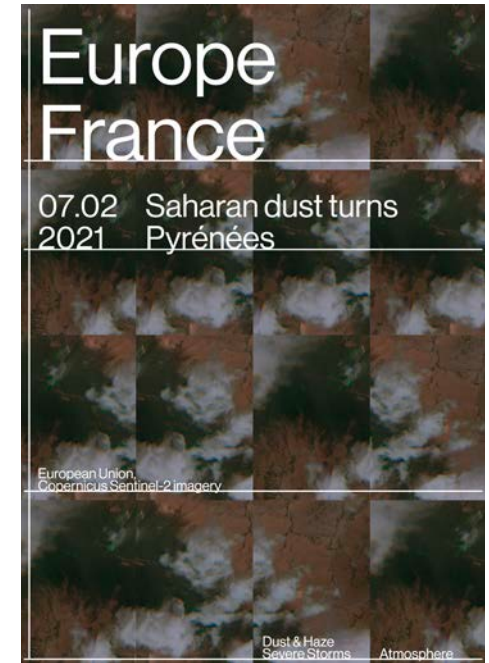
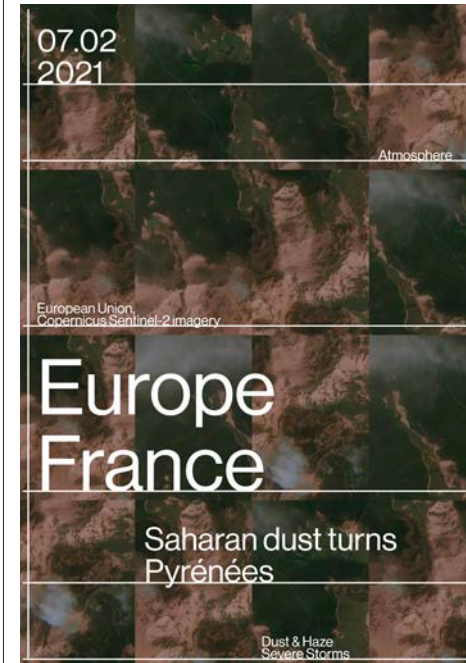
28

29

30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Atmosphere

Tag Dust & Haze
Severe Storms

Sicily and Saharan dust from space

At the end of the first week of February, three spring-like days raised daily temperature highs on the island of Sicily to over 20°C, with Palermo recording values of up to 29.5°C. On 6 February, the almost cloudless sky over the region allowed one of the two Copernicus Sentinel-3 satellites to acquire a beautiful image of Sicily and its surrounding waters, showing multiple interesting natural phenomena.

The left side of the image shows a partial view of the massive cloud of Saharan dust, which has been affecting southwestern European countries.

Along the southern coast of Sicily, the presence of a phytoplankton bloom often observed in this area can be seen from space. This impressive bloom is fuelled by specific environmental conditions: an abundance of sediment released into the water from sandy beaches and the high concentration of nutrients that can be found in these coastal waters.

On the eastern side of the island, this Sentinel-3 acquisition also shows a cloud-free view of the Mount Etna volcano, whose four craters have been particularly active during the first week of February. An ash plume can be seen east of the summit snow-covered sector.

The Copernicus Sentinel-3 mission, which is composed of two twin satellites (Sentinel-3A and Sentinel-3B), provides multispectral data with over 21 spectral bands that allow the development of innovative applications for both marine and land monitoring activities.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Atmosphere

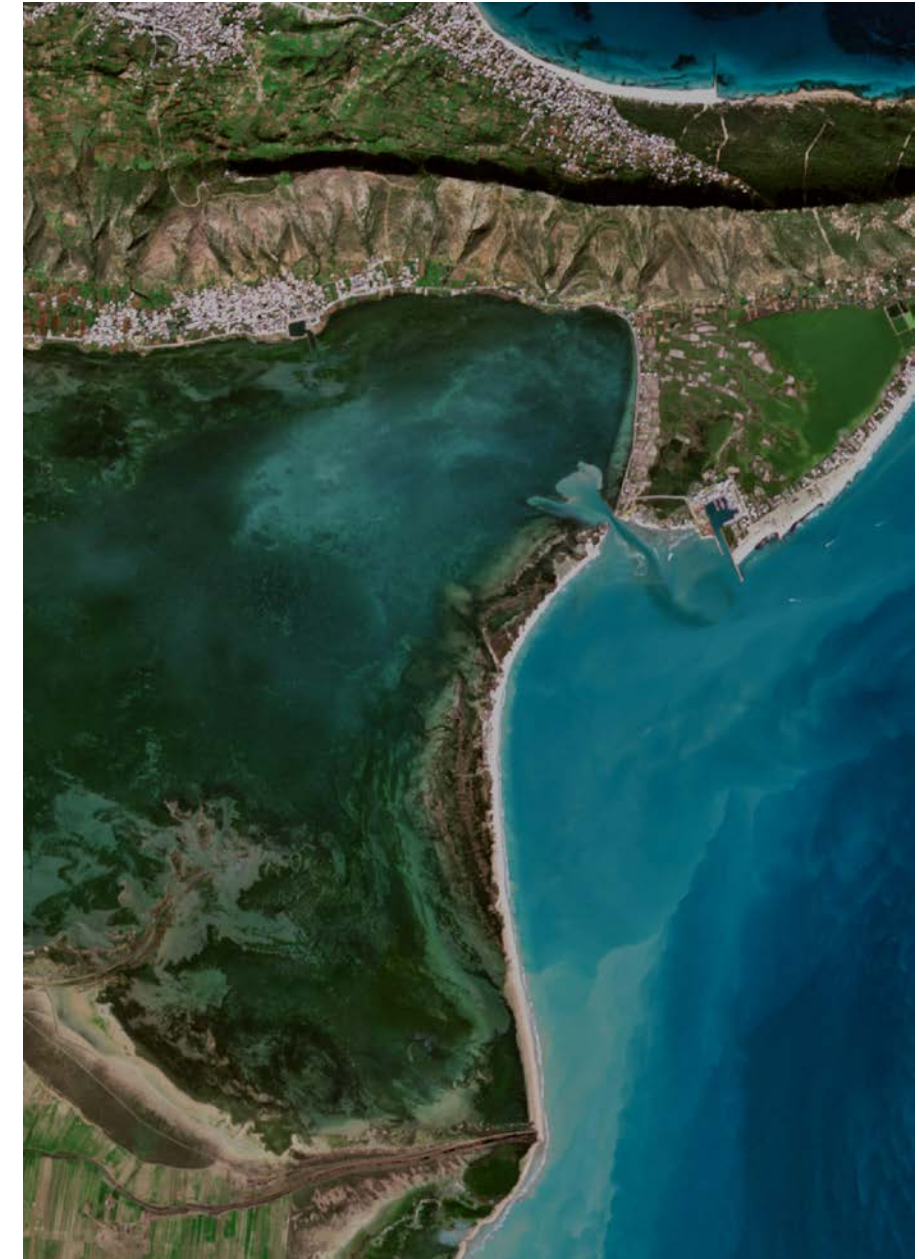
Tag Dust & Haze Land Monitoring Water Colou

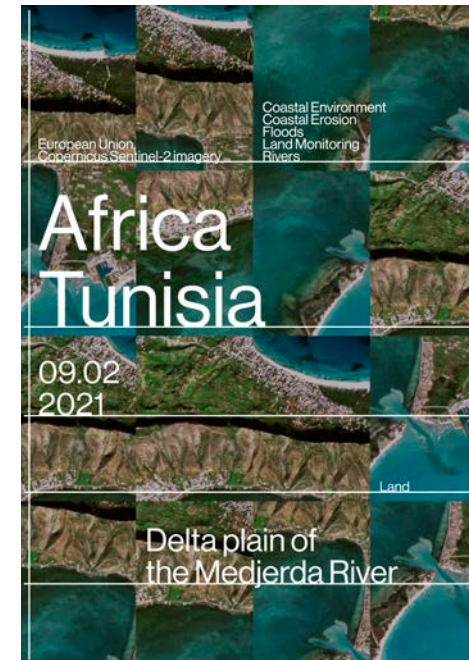
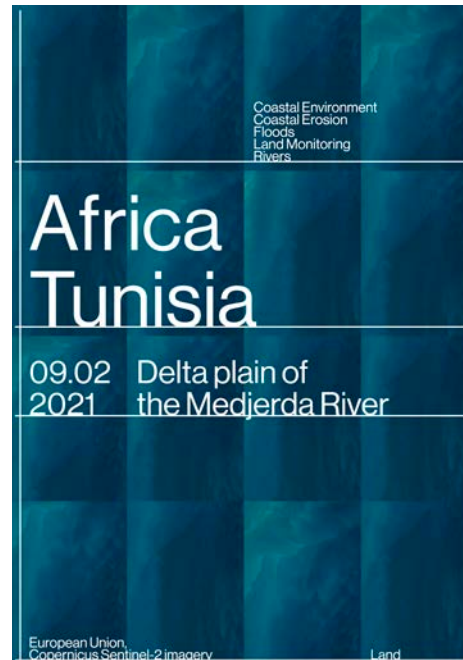
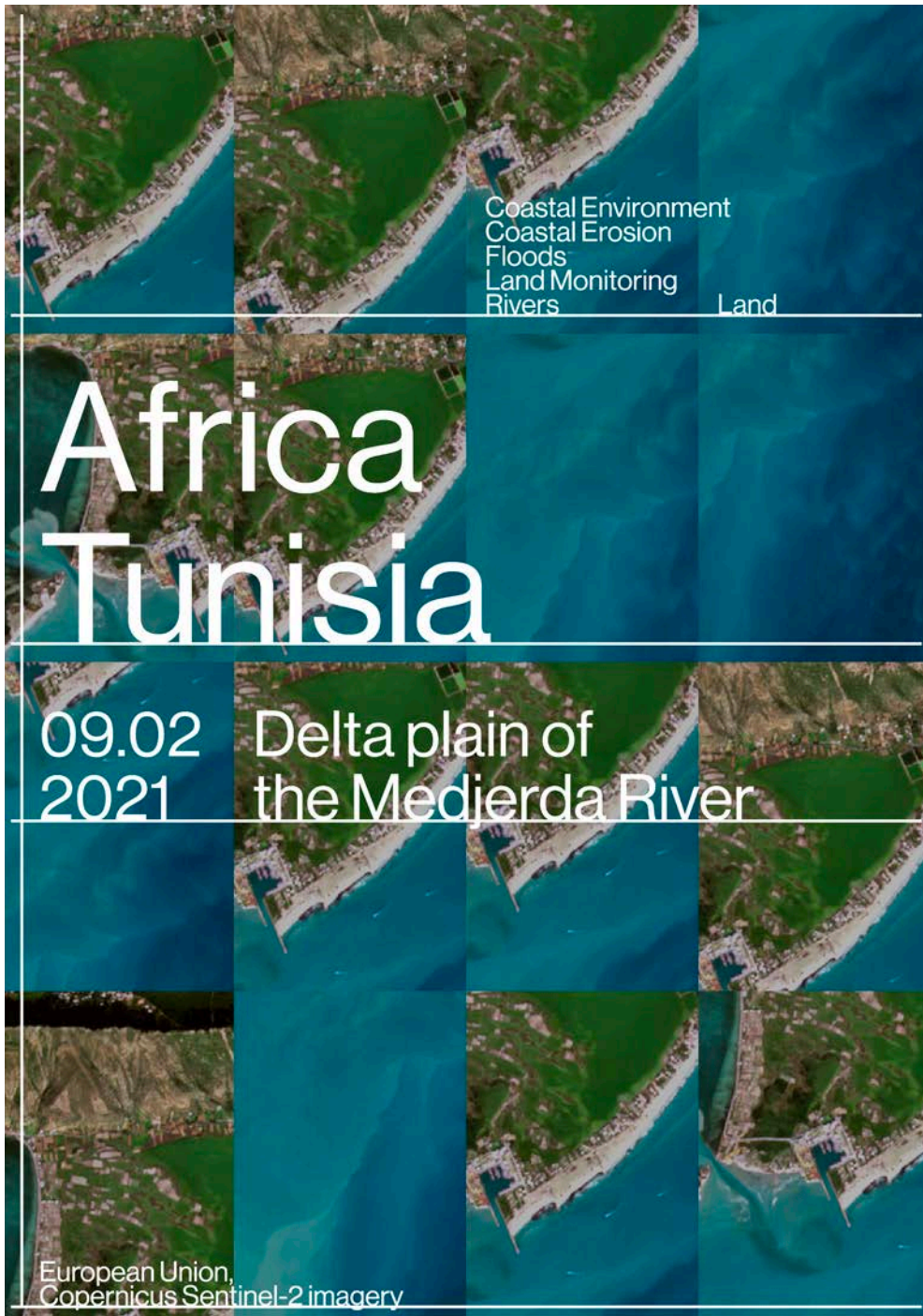
Delta plain of the Medjerda River

As a consequence of climate change and the ongoing rise of the global sea level, the arid coasts of North Africa are experiencing pronounced shoreline retreats and coastal flooding. Along the 4,633 km coastline that extends from the Gulf of Tunis to the Nile Delta, deltaic and sandy beaches exhibit some particularly interesting dynamics due to anthropogenic and climate change-related factors that increase their vulnerability to coastal hazards. According to a study recently published in the Scientific Reports journal, 47% of North African coasts have a high or very high vulnerability as a result of anthropogenic drivers such as damming and the rapid urban growth of the past decades, and not (only) global warming. The densely populated river deltas in Tunisia and Egypt are particularly vulnerable to coastal hazards, the environmental consequences of which affect crop yields and result in increased migrations from the affected regions.

This image, acquired by one of the two Copernicus Sentinel-2 satellites on 20 January 2021, shows the northern side of the Medjerda River delta plain in Tunisia, which is one of the regions analysed in the newly published study. To better observe the shoreline retreat and the deterioration of the river's coasts, the study authors used Sentinel-2 images captured in 2019 and compared them to aerial photographs and Landsat-5 imagery of the same region, taken in the 1970s.

The Copernicus Sentinel-2 mission, which is composed of two twin satellites, Sentinel-2A and Sentinel-2B, allows users to monitor and study coastal zones and wetlands globally, with a maximal spatial resolution of 10 metres and an average revisit time of five days at the equator.





Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

Tag Floods Land Monitoring Rivers

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

Bruxelles blanketed in snow

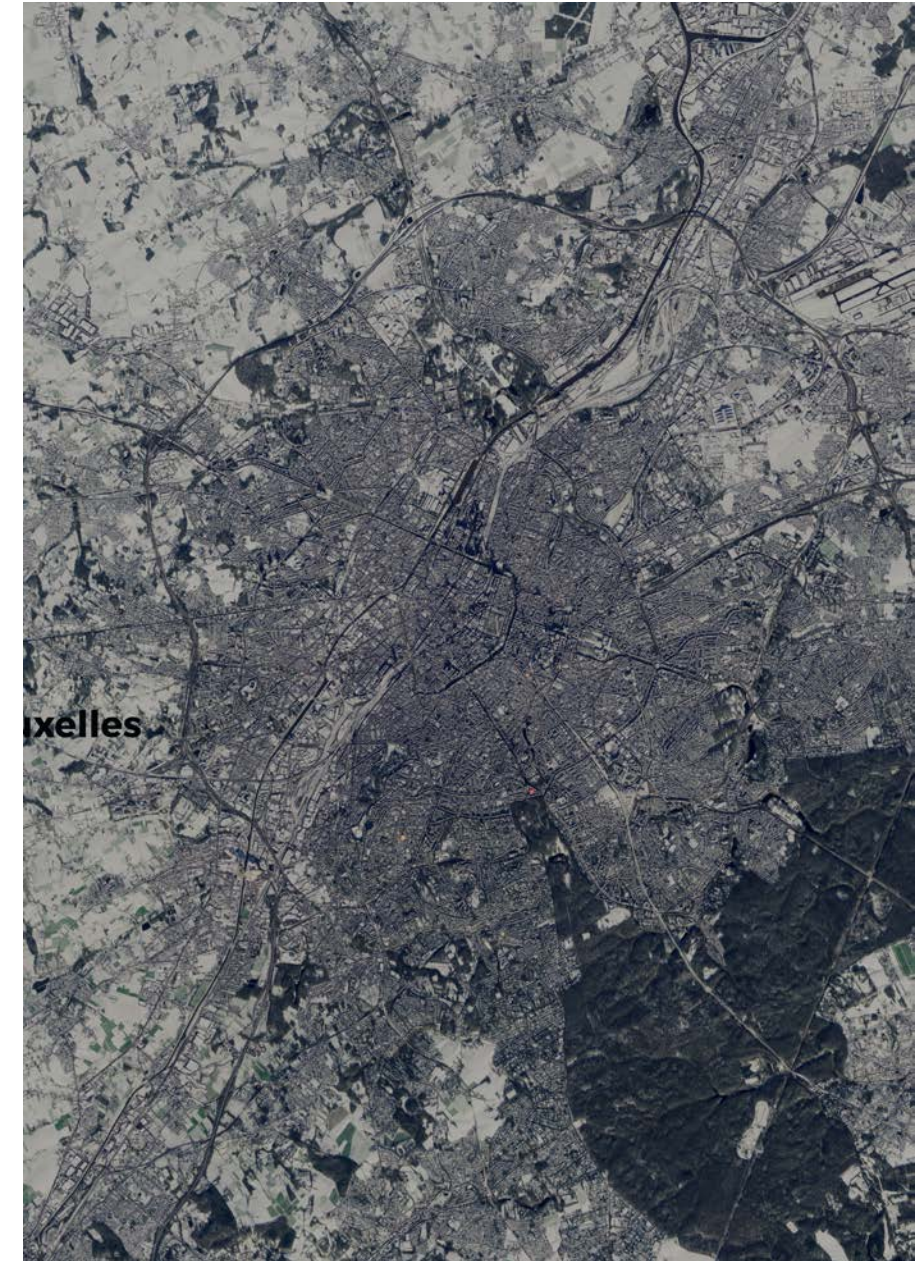
In the aftermath of Storm Darcy that hit Europe in the second week of February 2021, regions that rarely experience heavy snowfall have been blanketed in snow. The storm, which originated in Russia, has affected Northern European countries the most, causing major disruptions in daily life.

The Netherlands has experienced its first large snowstorm in the past ten years, with disrupted air and rail traffic, and disruptions due to the snow were also recorded throughout the entire rail and road traffic network in Germany. In Brussels, the three largest train stations are temporarily being kept open for 24 hours a day to shelter the homeless from the snow and freezing temperatures.

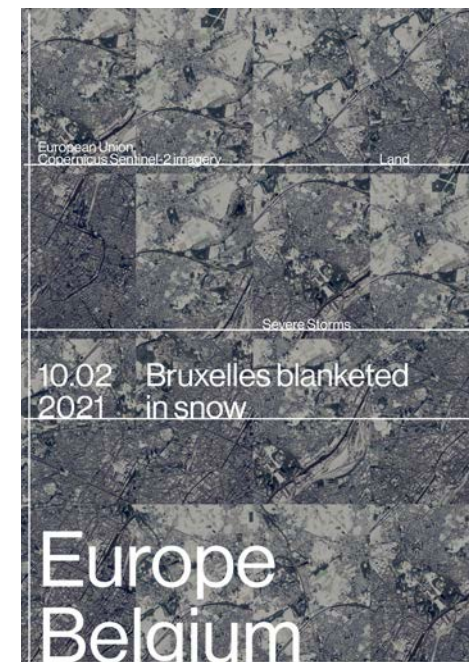
Storm Darcy is the latest extreme weather phenomenon in a particularly cold winter in Northern Europe. According to a report published by the Copernicus Climate Change Service on 9 February, the entire month of January 2021 was colder than average.

This image, acquired by one of the Copernicus Sentinel-2 satellites on 9 February 2021, shows Brussels covered with snow.

The Copernicus Sentinel-2 mission provides high-resolution data that can be used to produce detailed snow and ice maps of Europe.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



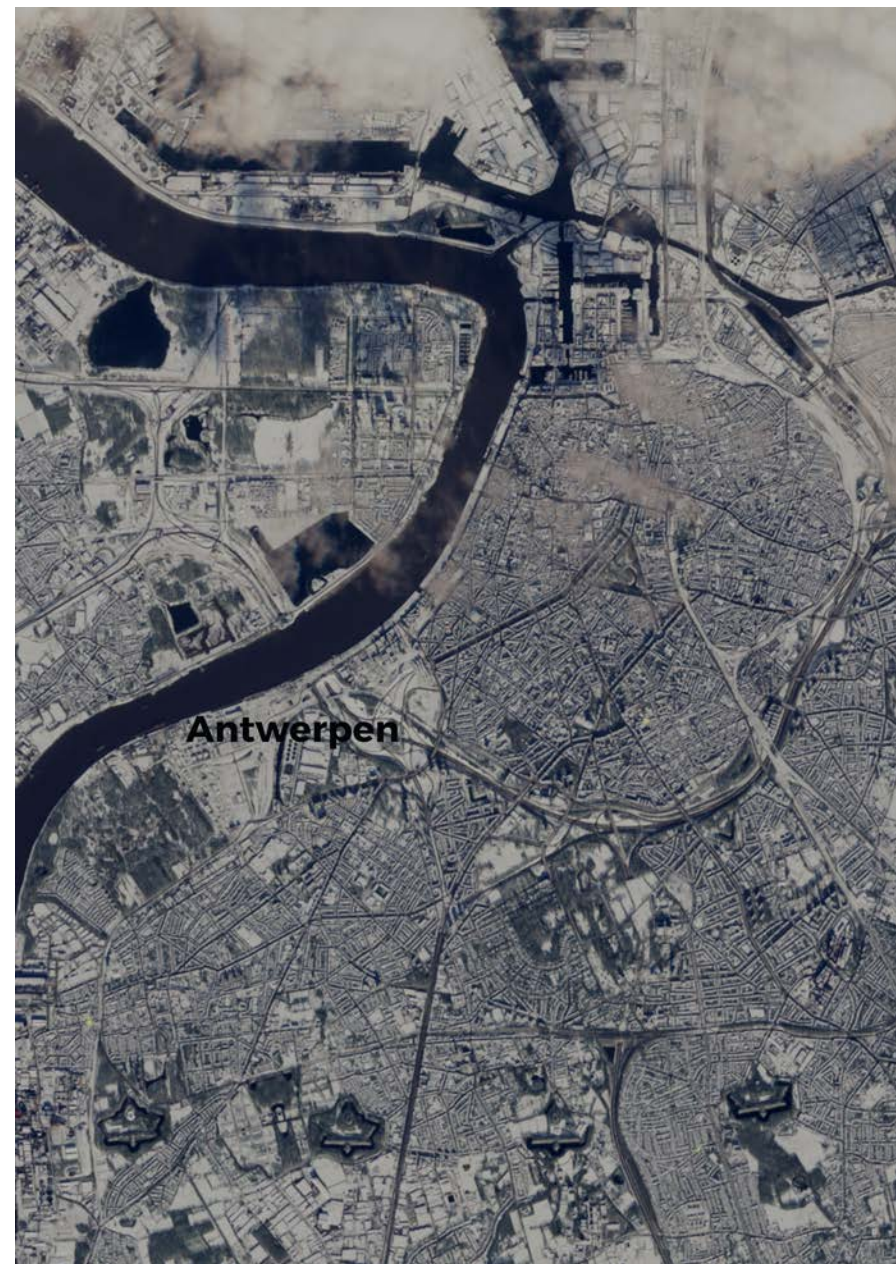
Credit European Union, Copernicus Sentinel-2 imagery

Category Land

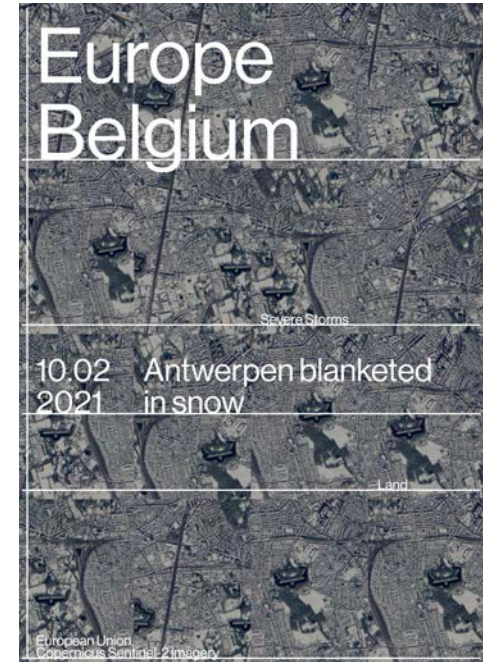
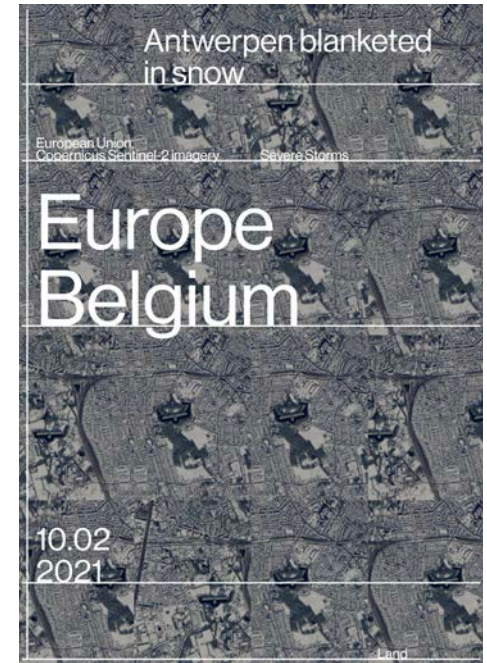
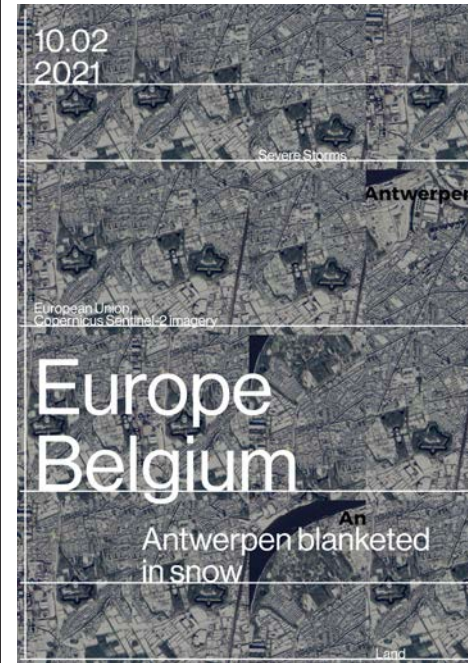
Tag Severe Storms

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30

Antwerpen blanketed in snow



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



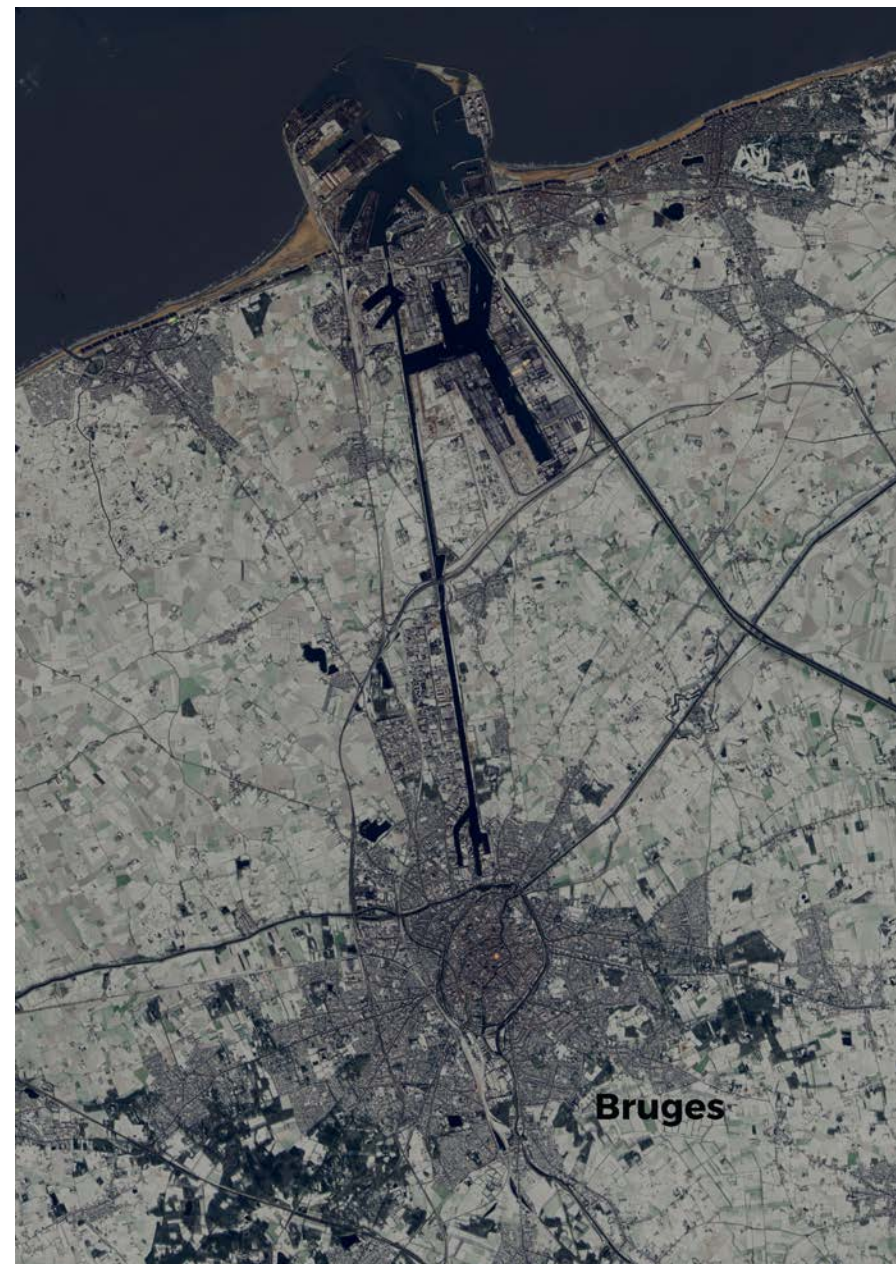
Credit European Union, Copernicus Sentinel-2 imagery

Category Land

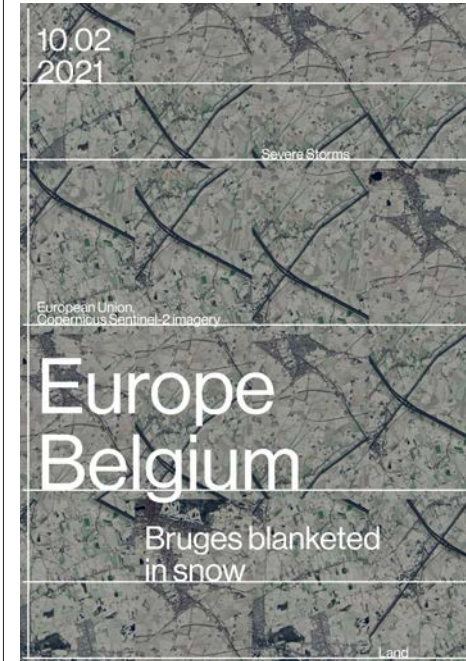
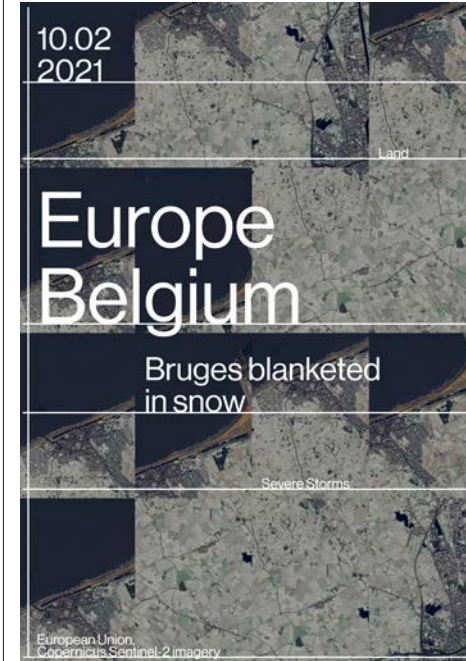
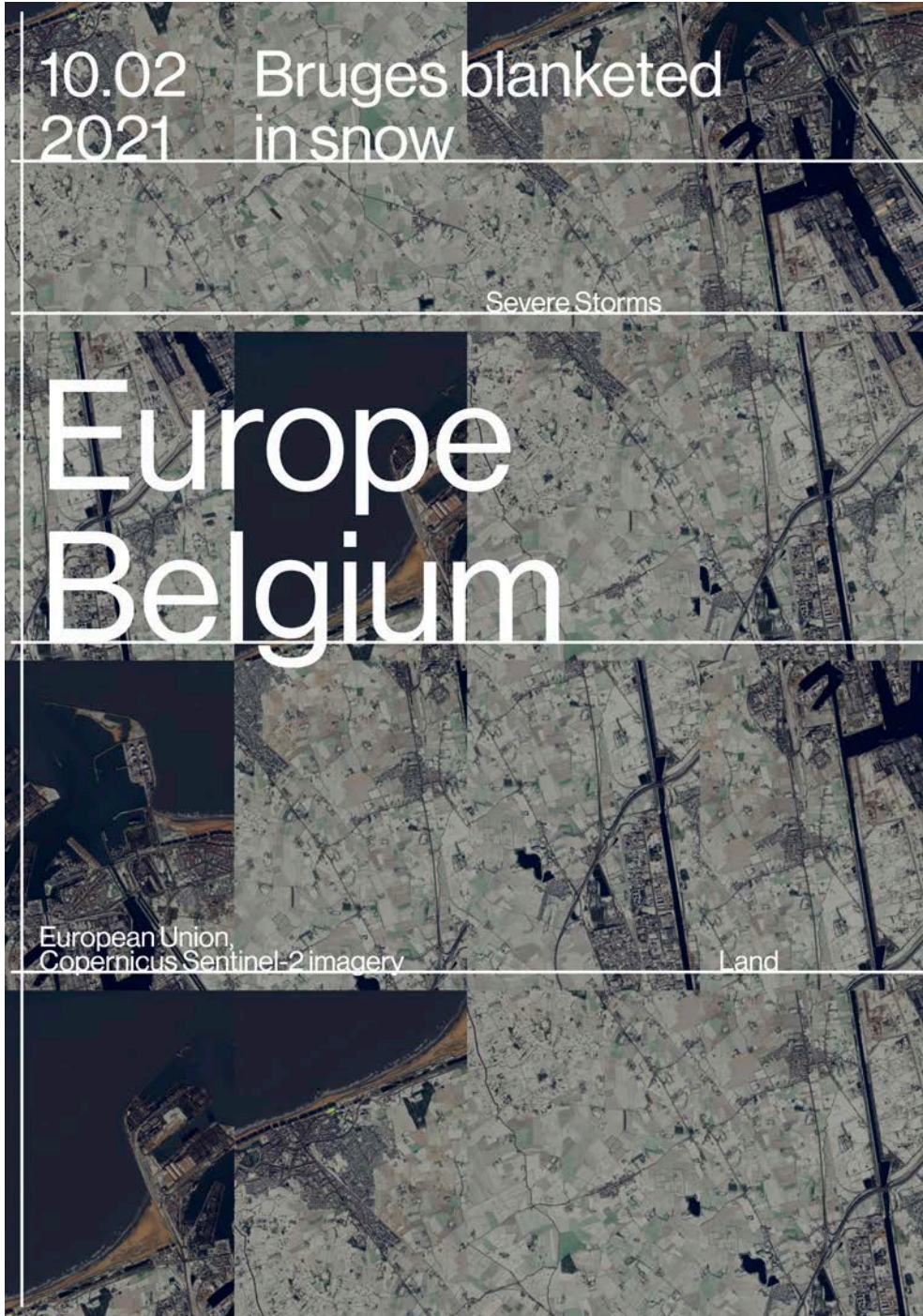
Tag Severe Storms

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30

Bruges blanketed in snow



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



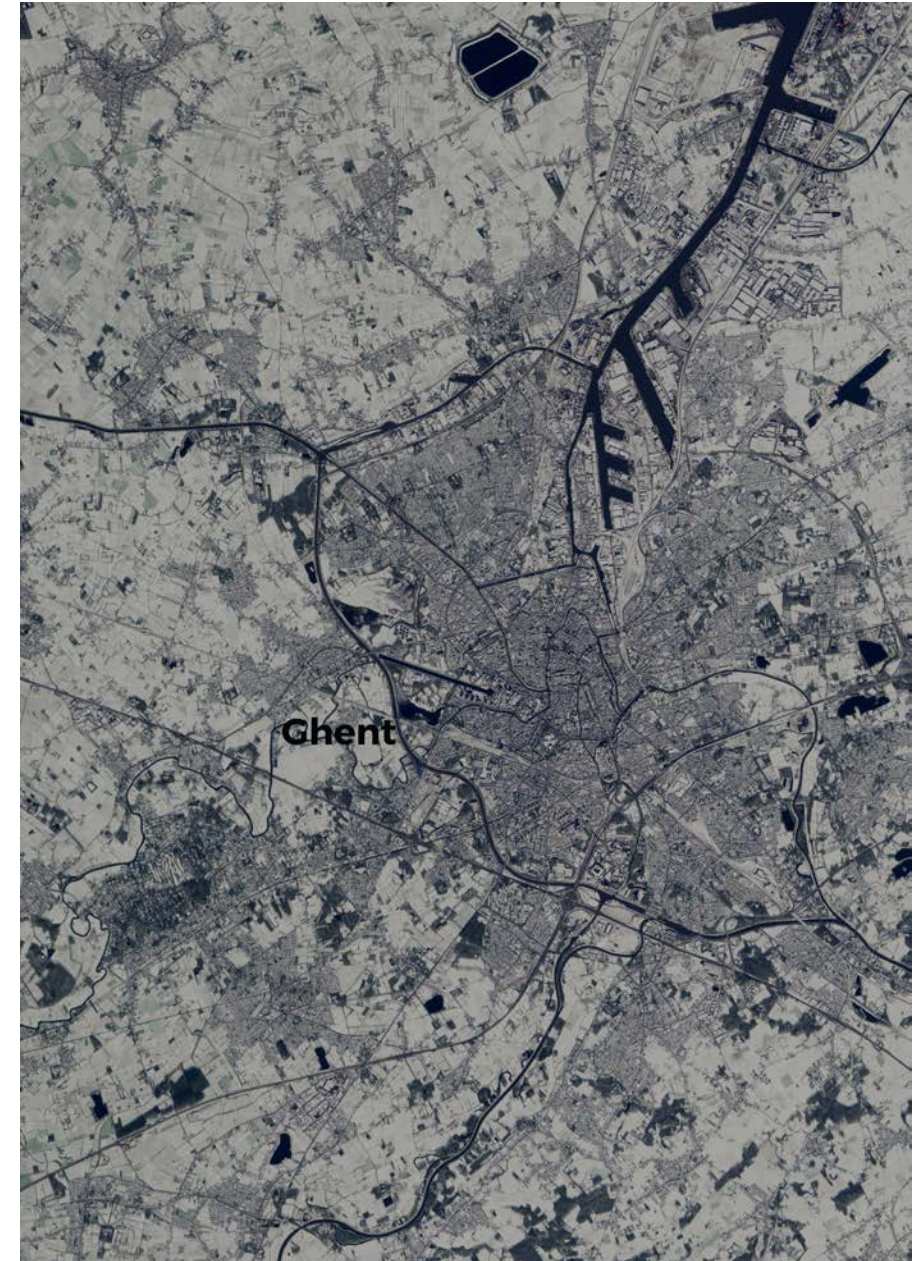
Credit European Union,
Copernicus Sentinel-2 imagery

Category Land

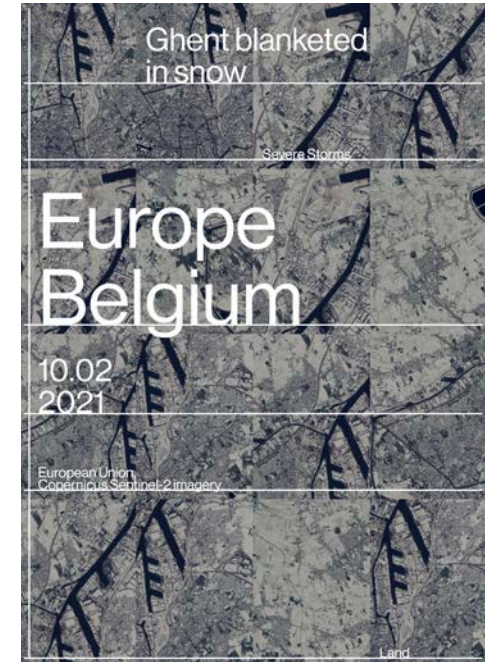
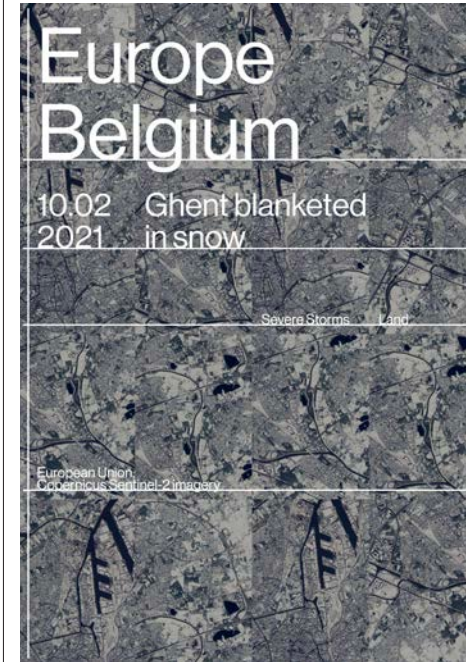
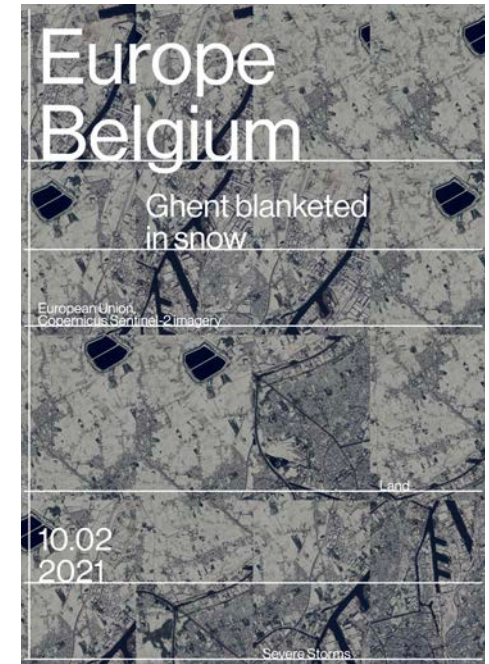
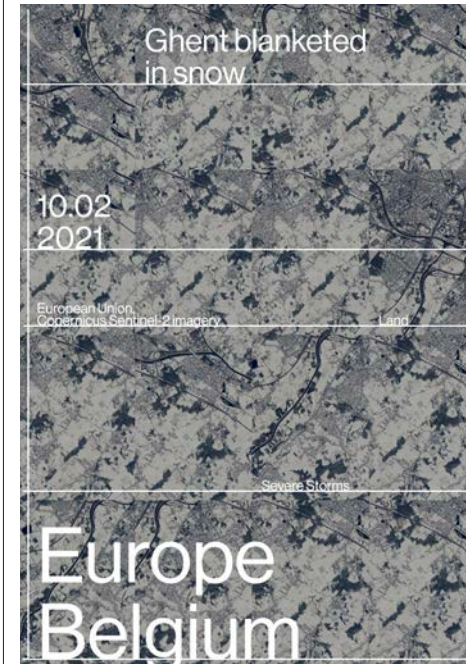
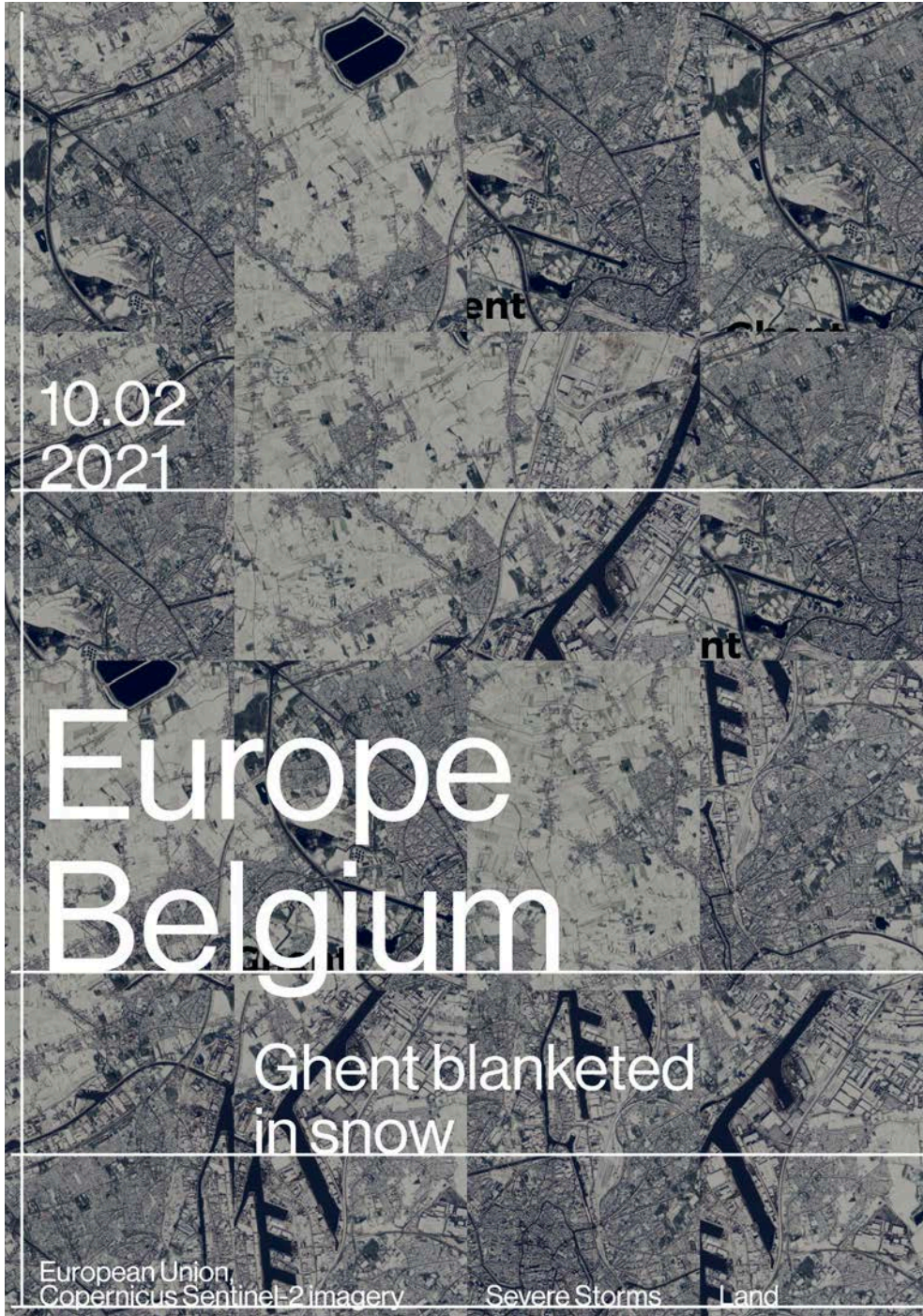
Tag Severe Storms

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30

Ghent blanketed in snow



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Land

Tag Severe Storms

1 Leiden blanketed in snow

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

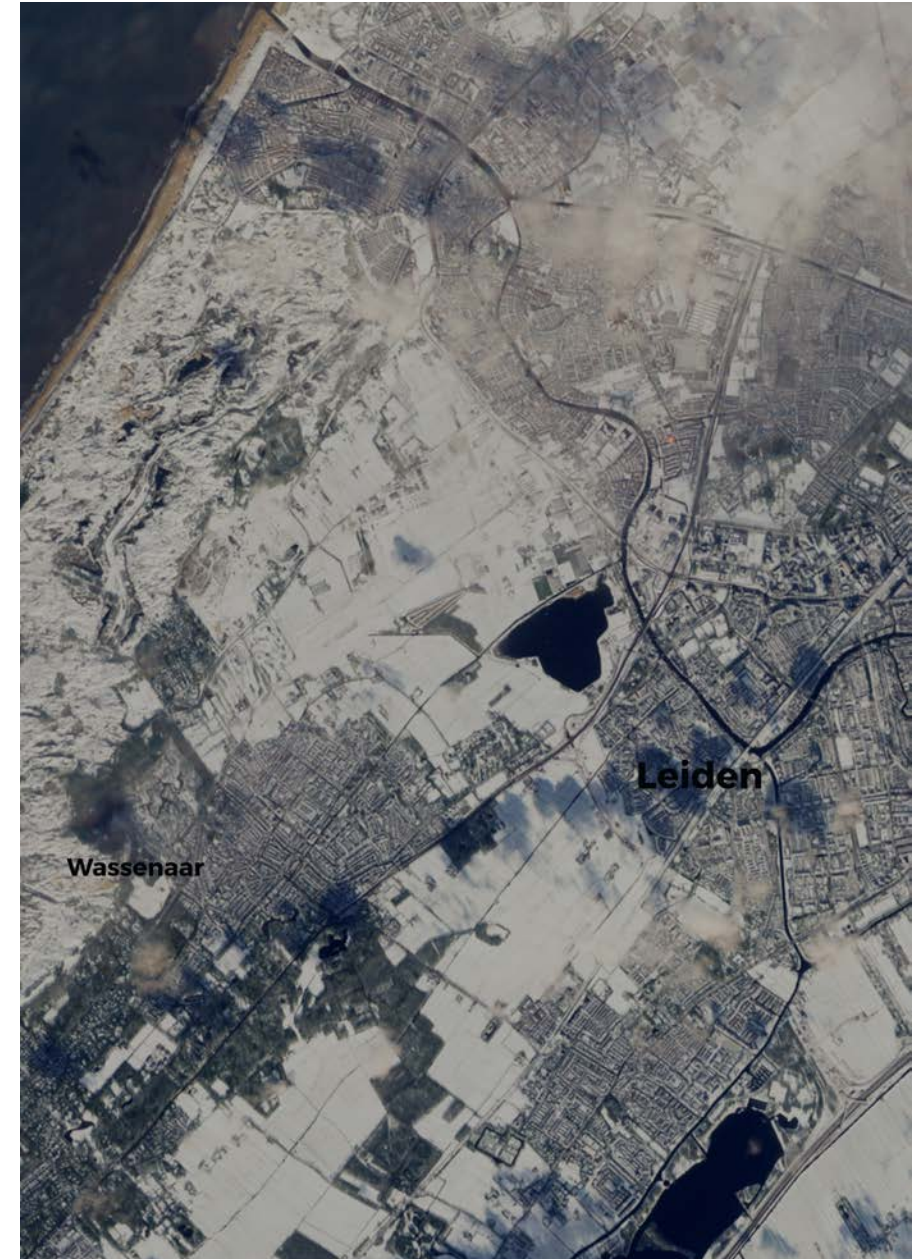
26

27

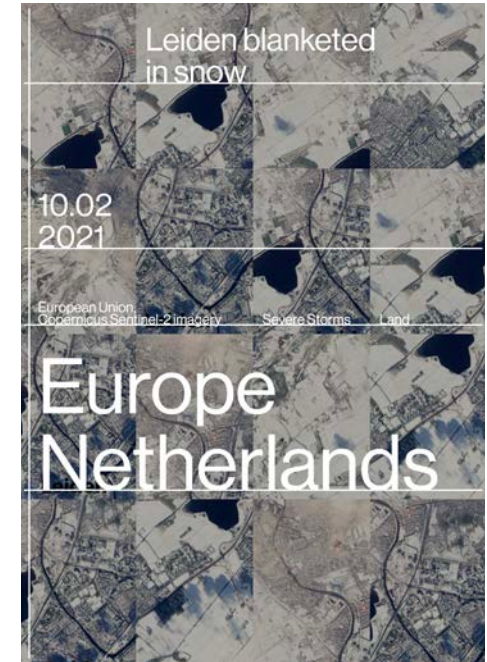
28

29

30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

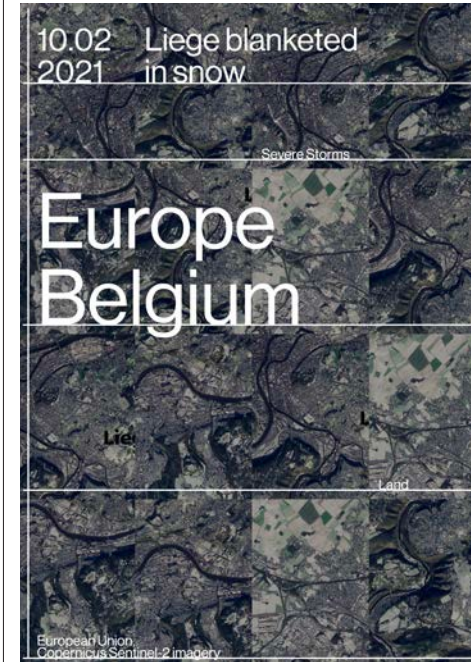
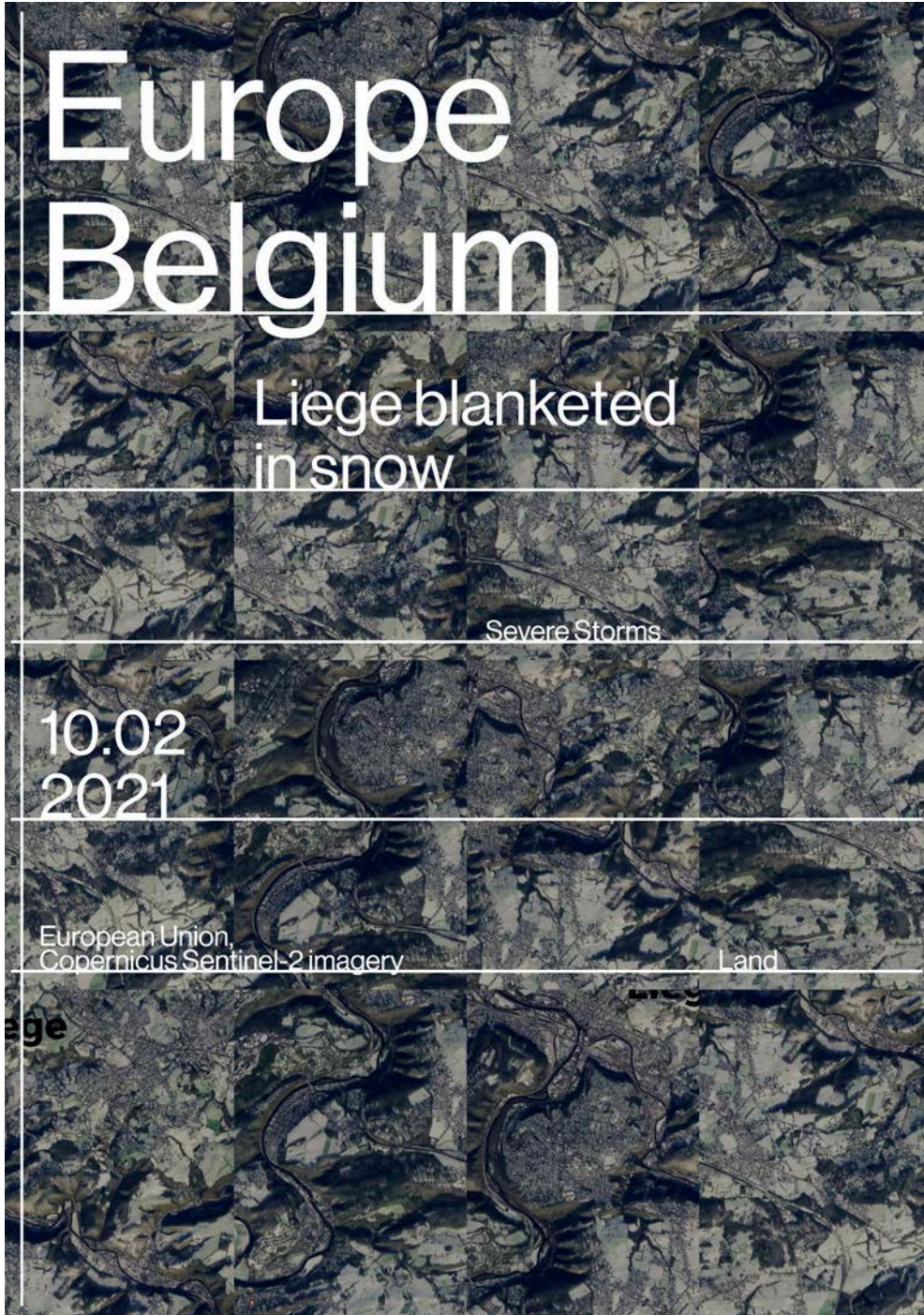
Tag Severe Storms

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30

Liege blanketed in snow



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

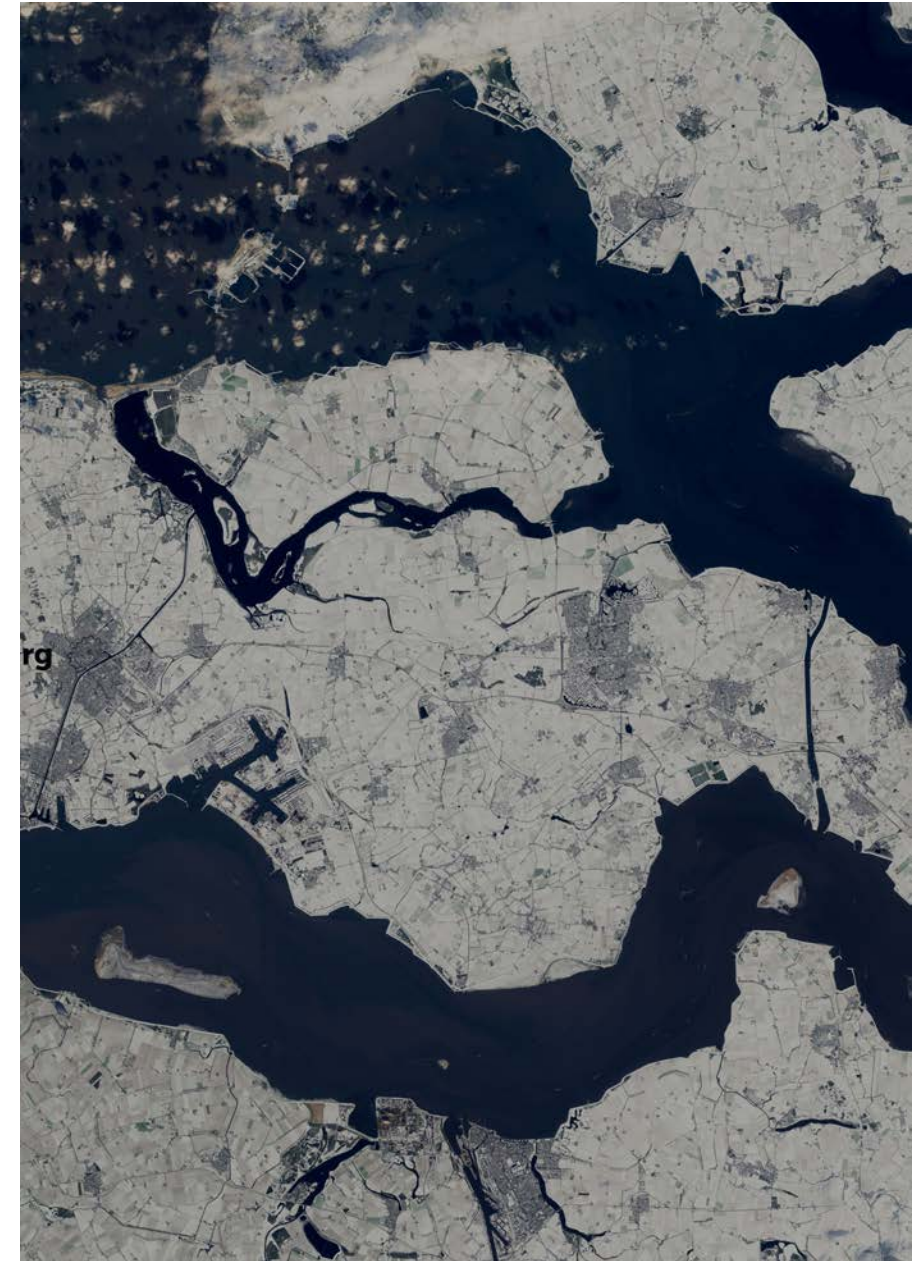


Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

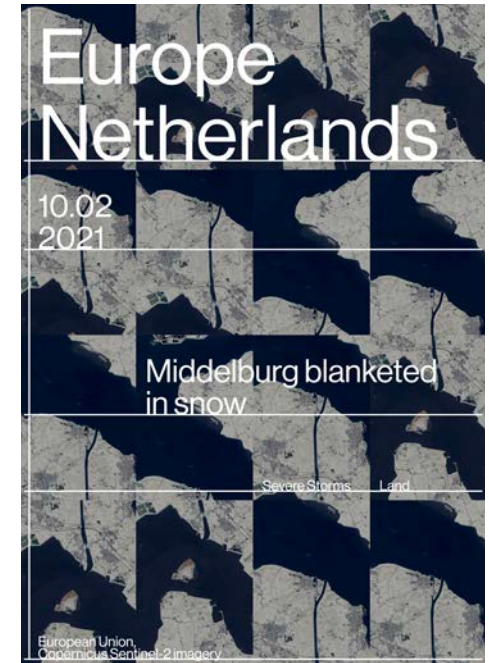
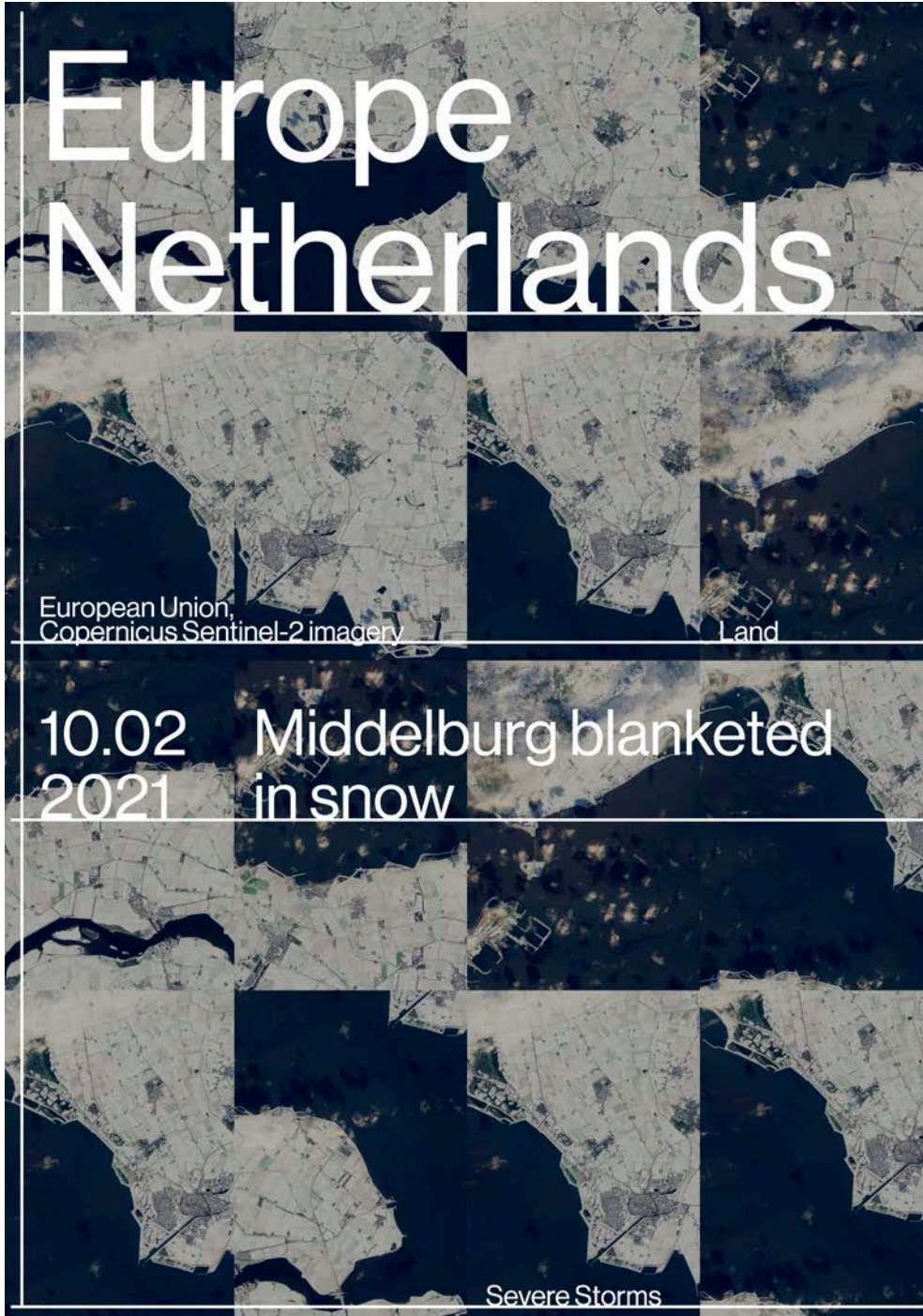
Tag Severe Storms

1 Middelburg blanketed in snow

- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

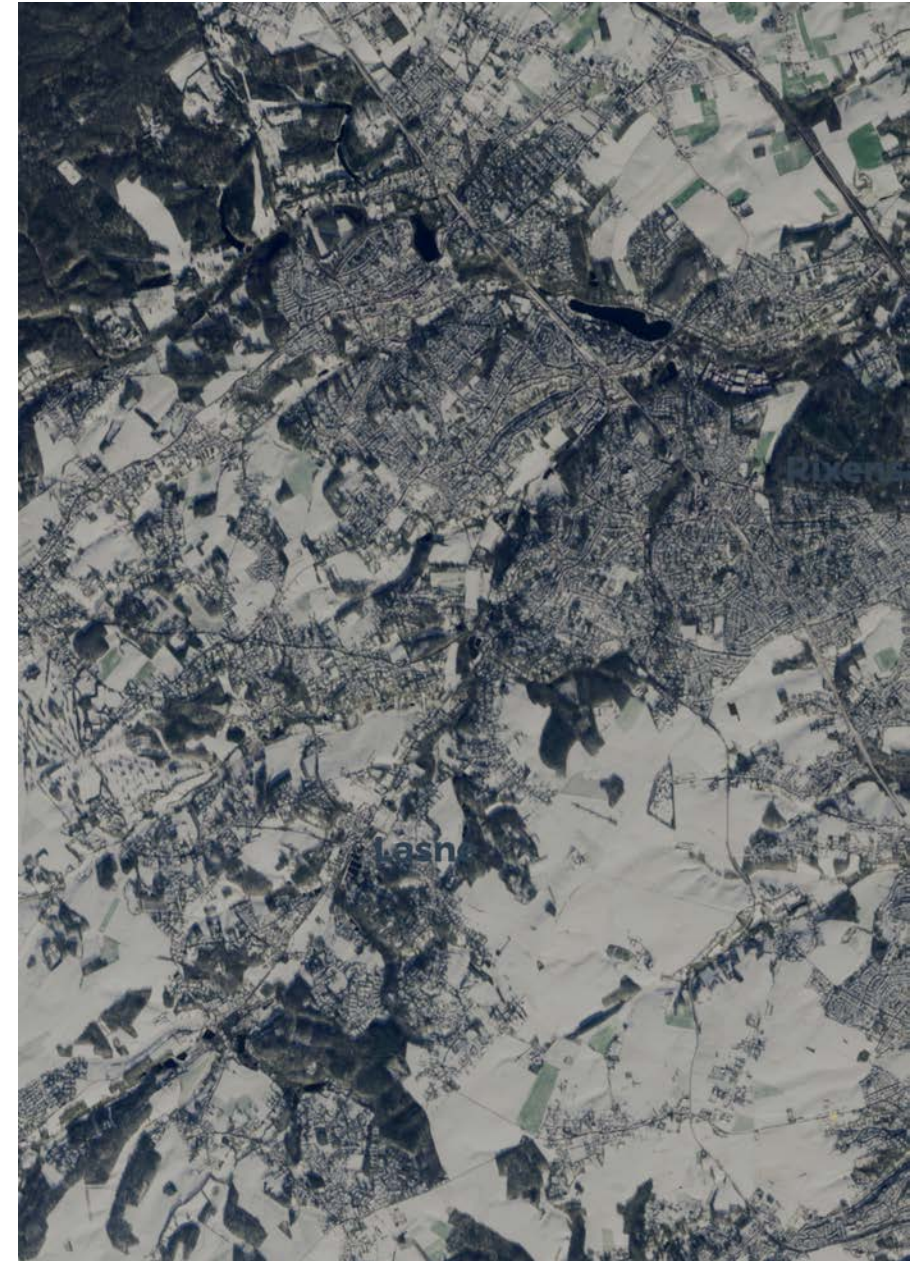


Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

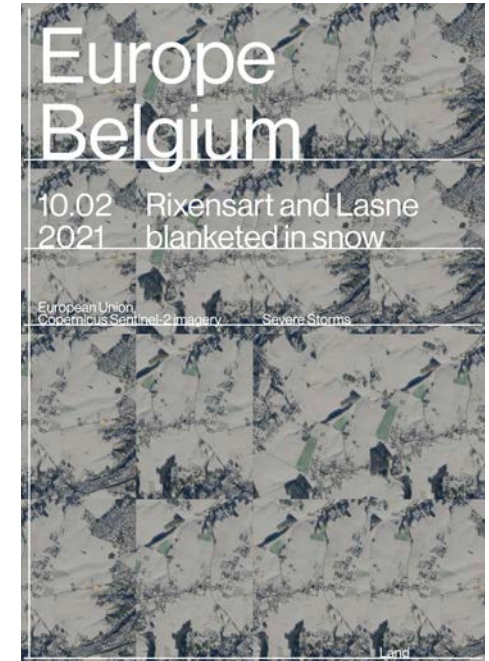
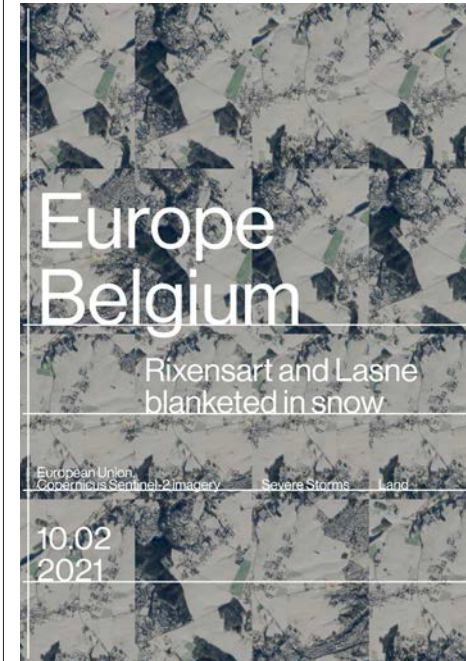
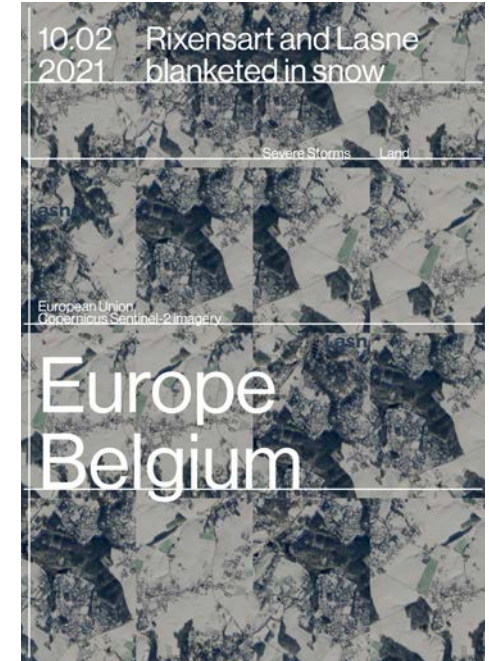
Tag Severe Storms

1 Rixensart and Lasne blanketed in snow

- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26
- 27
- 28
- 29
- 30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

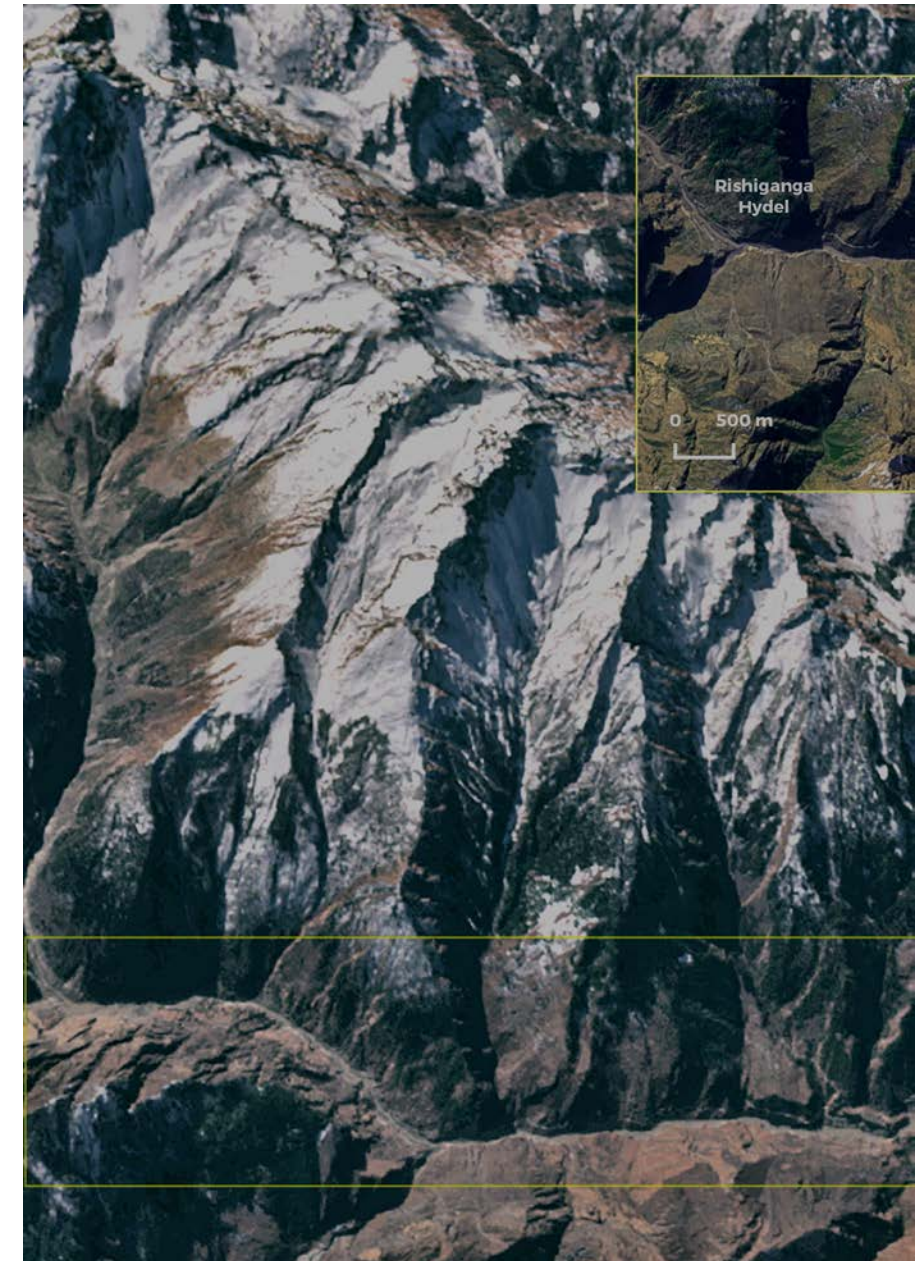
Tag Severe Storms

Uttarakhand disaster

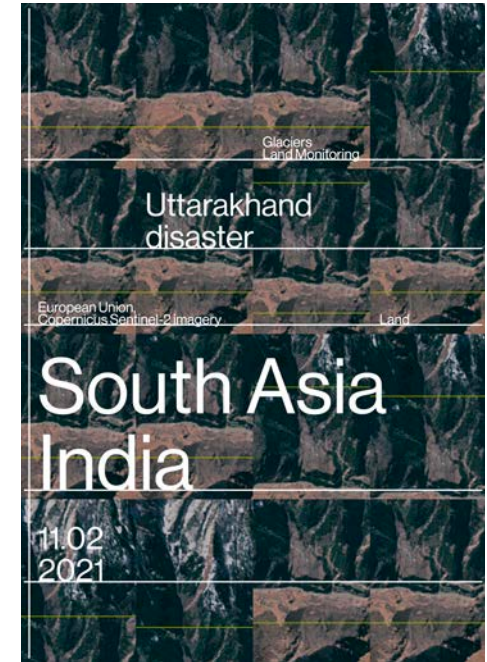
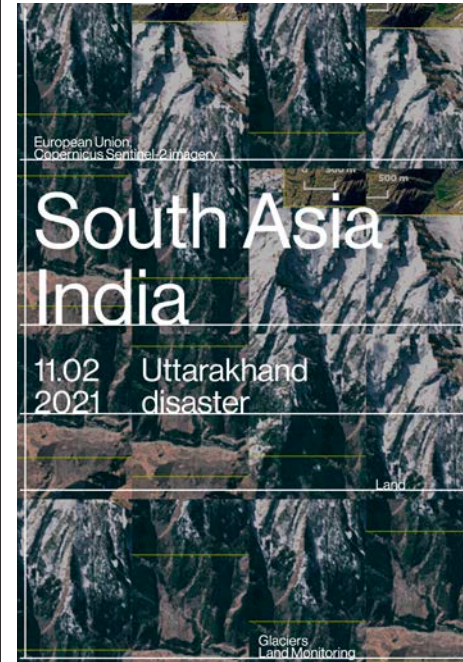
A devastating flash flood that happened on 7 February 2021 in India's northern state of Uttarakhand caused the deaths of at least 32 people, while several hundred workers of the two hydropower plants construction sites that were destroyed on the rivers Rishi Ganga and Dhauri Ganga are reported missing. The Chamoli district has suffered the brunt of the flood, with massive infrastructural damage, loss of life and widespread environmental damage in the ecologically fragile area of Uttarakhand that lies on the southern slope of the Himalaya range.

Relief efforts are still underway, with emergency crews attempting to rescue workers trapped in underground service tunnels and the inhabitants of nearby villages whose houses were washed away by the flood.

The exact cause of the devastating flood is yet to be ascertained, but early assessments indicate that the event could have started by a landslide that caused a large avalanche of ice and rock between two remote glacier peaks, Raunthi and Mrigthuni, both at above 5,500 metres of elevation. The role of climate change and climate-related factors has not yet been confirmed, but what is clear is that this was not the first recorded avalanche in this region, as a similar but smaller event was recorded in 2016.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

Tag Glaciers Land Monitoring

Uttarakhand disaster

On 10 February 2021 heavy rain hit Southern Italy, in particular the Campania region, which suffered numerous landslides and floods as a result of the rainfall. Disruptions in local road networks isolated some towns in the inner provinces of Avellino and Benevento, and an entire part of the Amalfi Coast was isolated due to landslides in Vietri sul Mare and Ravello. In the Province of Salerno, the severe weather caused a water main to break, leaving thousands of citizens without drinking water. On 11 February 2021, the Italian Civil Protection Department issued a new warning for possible landslides in the region. This image, acquired by one of the two Copernicus Sentinel-2 satellites on 11 February 2021, shows the floods caused by the rivers Calore and Volturno in northern Campania. Along the coastal area, massive sediment plumes generated by the discharge of the rivers Volturno and Garigliano can also be seen.

The Copernicus Sentinel-2 mission allows for flood mapping and damage assessment thanks to its maximum spatial resolution of ten metres and a maximum revisit time of five days.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Land

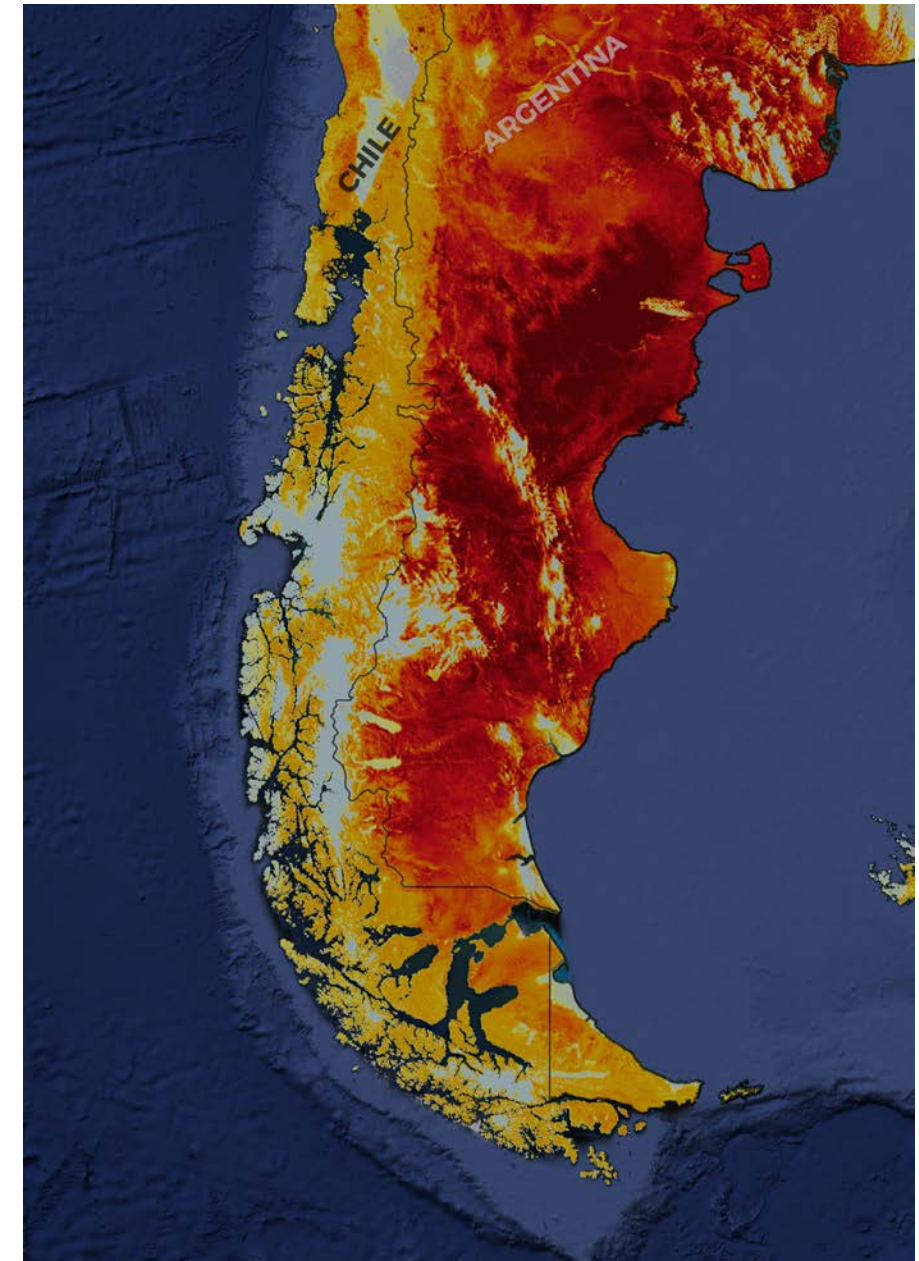
Tag Floods
Land Monitoring
Severe Stormsv

Patagonia Heatwave

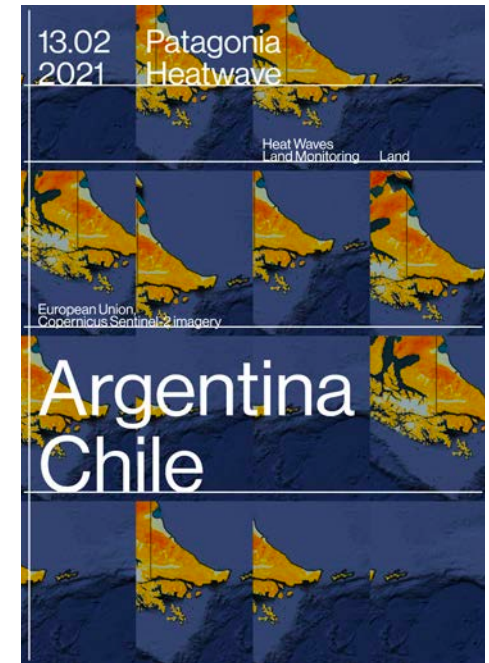
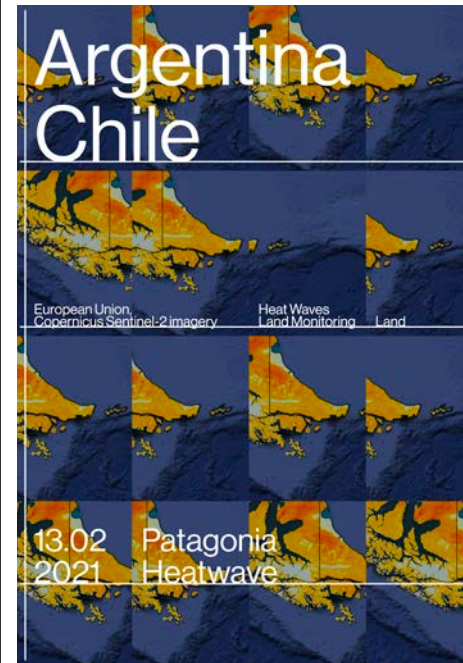
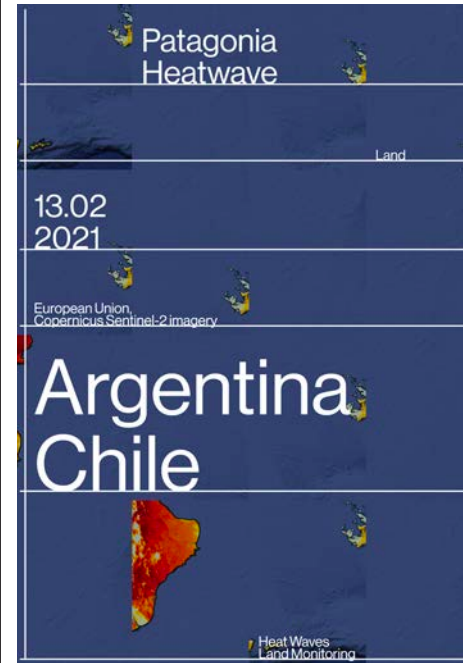
A strong heatwave is affecting Patagonia and the Antarctic Peninsula in the second week of February, with temperatures 10°C higher than average values of the period. In Punta Arenas, one of the major cities in the southern Chilean Patagonia, temperature has risen up to 25°, while the typical maximum value is about 15°C. According to the experts of the Dirección Meteorológica de Chile (DMC), this phenomenon is due to high pressure that allows the influx of warm subtropical air. Heatwaves are not unusual events in Patagonia, but their frequency is increasing every year. 2020 had been characterised by a lower number of heatwaves in the region, due to the presence of La Niña, but models suggest a new increase for the summer of 2021. These high temperatures result from a complex climatic situation, and they are mainly a consequence of the global warming according to climatologists from the University of Chile.

This image shows the Land Surface Temperature (LST) map over Patagonia as recorded by the two Copernicus Sentinel-3 Satellites on 11 February 2021. LST is the temperature of the soil and should not be confused with the air temperature at ground level. This is why it is not possible to measure the temperature of the soil in areas covered by clouds.

The Copernicus Sentinel-3 mission is composed of two twin satellite (Sentinel-3A and Sentinel-3B) that are both equipped with the Sentinel-3 Sea and Land Surface Temperature Radiometer, an instrument whose main objective is to provide global and regional Sea and Land Surface Temperature (SST, LST) with a very high level of accuracy (for both climatological and meteorological applications).



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

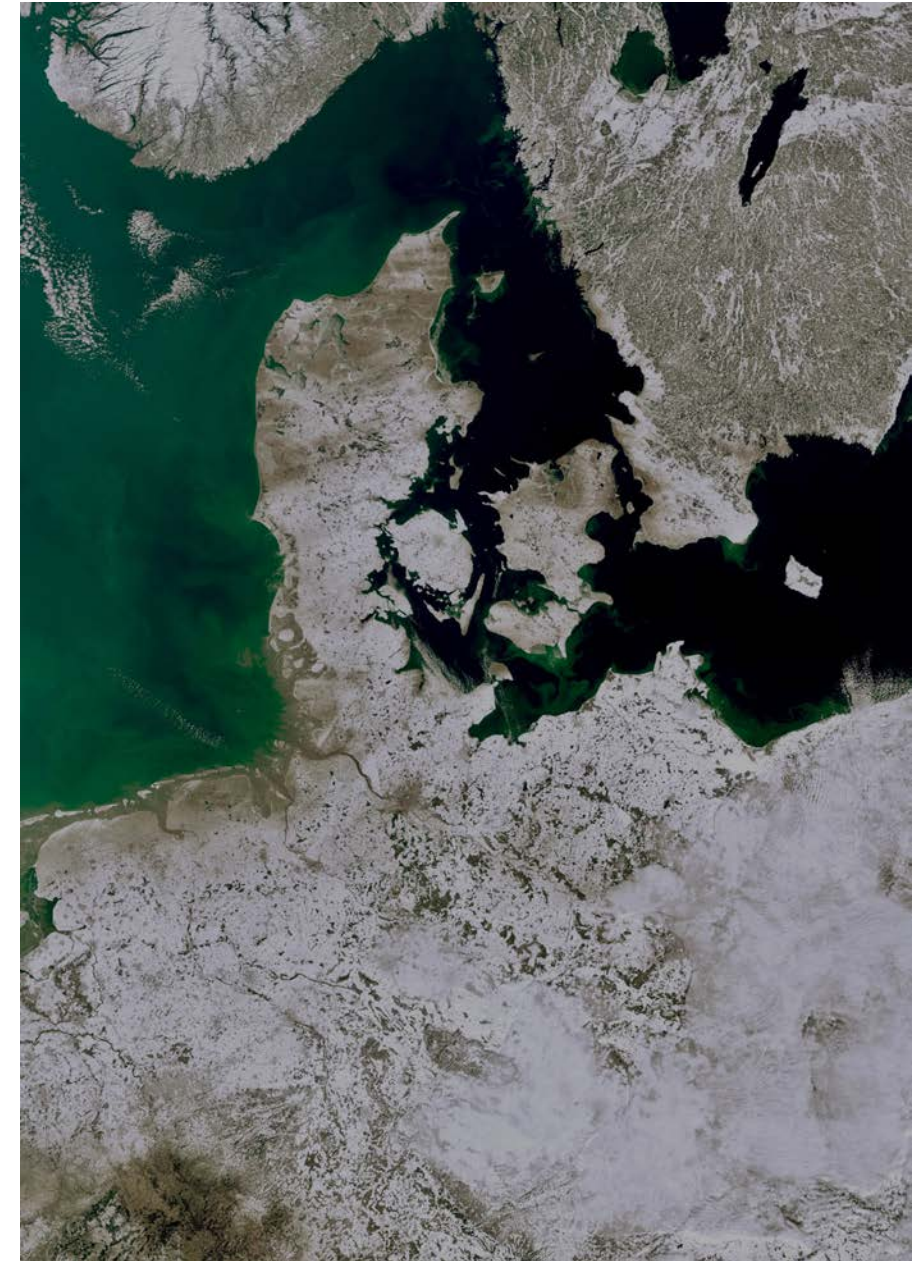
Tag Heat Waves Land Monitoring

Snowy Denmark and surrounding countries

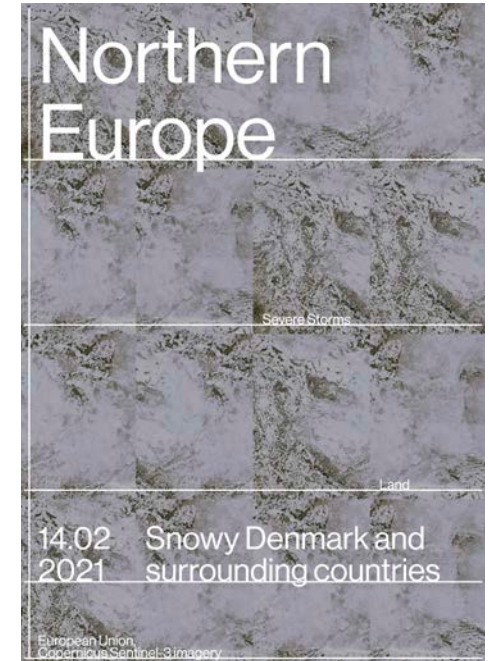
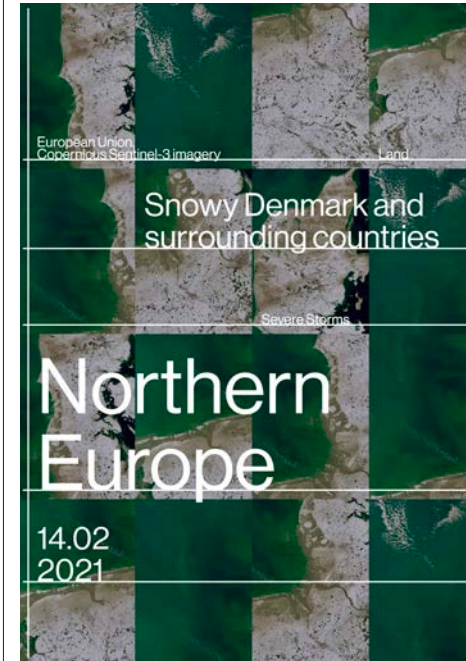
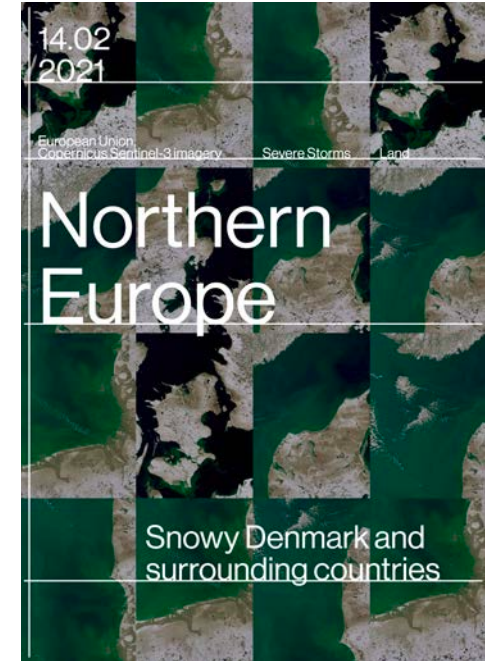
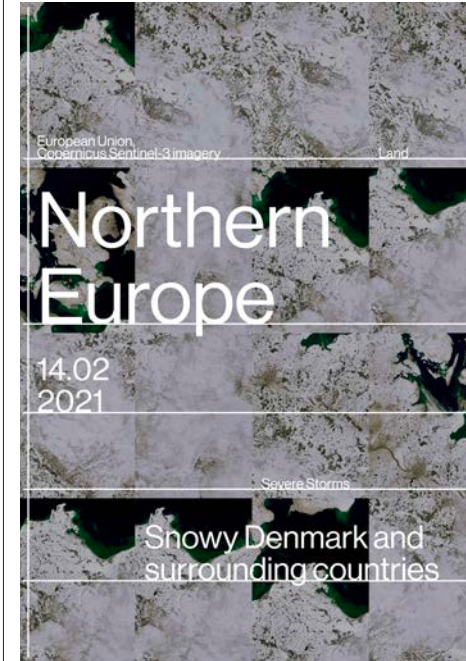
Polar weather has gripped Europe in the last few days, with up to 40 cm of snow blanketing parts of Europe, sparking widespread disruption. Blizzard and sub-zero temperatures are being recorded from northern to southern Europe. After storm Darcy caused heavy snowfall in Northern Europe, heavy snow has hit northern and eastern Italy and could also reach Greece. At the same time, Moscow has seen the worst snowfall in decades, with freak weather disrupting air travel and bringing traffic to a halt. Several people have been injured as a result of roofs collapsing in the Moscow Region.

This image, captured by one of the two Copernicus Sentinel-3 satellites on 13 of February, shows the snow still covering the whole of Denmark and its neighboring states after the passage of storm Darcy. One can also notice a strong presence of suspended sediments in the North Sea, especially along the western coastal waters of Denmark.

The Copernicus Sentinel-3 mission, that is composed of two twin satellites (Sentinel-3A and Sentinel-3B), provides multi-spectral imagery with two different optical sensors, allowing to implement innovative application for ocean colour measurements and snow cover monitoring activities.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-3 imagery

Category Land

Tag Severe Storms

1 Vehicles in flames after fuel tanker exploded

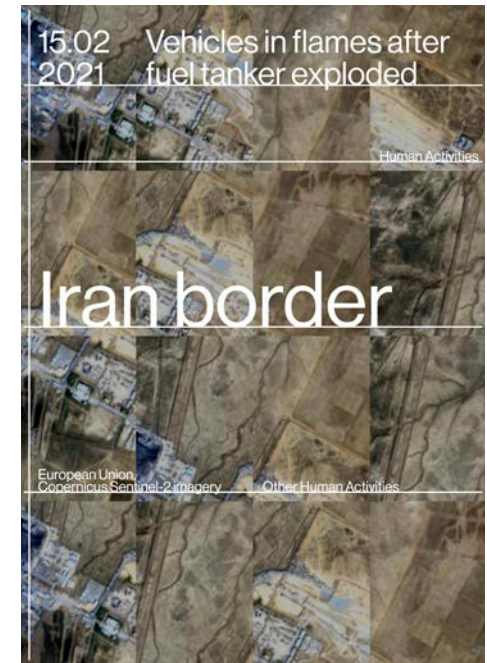
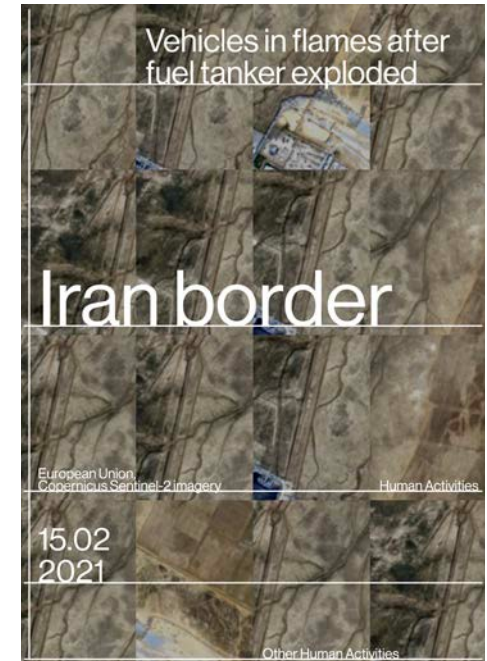
2 On 13 February 2021, a gas-carrying truck exploded in Islam
3 Qala, in Afghanistan's western province of Herat, at a border
4 post with Iran. It is still not known what caused the tanker to
5 explode and set fire to more than 100 other nearby fuel tan-
6 kers. A total of 500 vehicles ended up being destroyed by
7 flames, while at least 60 people were injured, and power
8 supply was disrupted. Local authorities estimate that the in-
9 cident caused 50 millions of dollars in damages.

10 This image, acquired by one of the Copernicus Sentinel-2 on
11 14 February 2021, shows that, one day after the explosion,
12 the fires (red hot spots) continued to burn.

13 With a revisit time of 5 days on average at the equator, the
14 Copernicus Sentinel-2 satellites deliver free and open data
15 that can be used to monitor human activities around the world.
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery-
Category Human Activities

Tag Other Human Activities

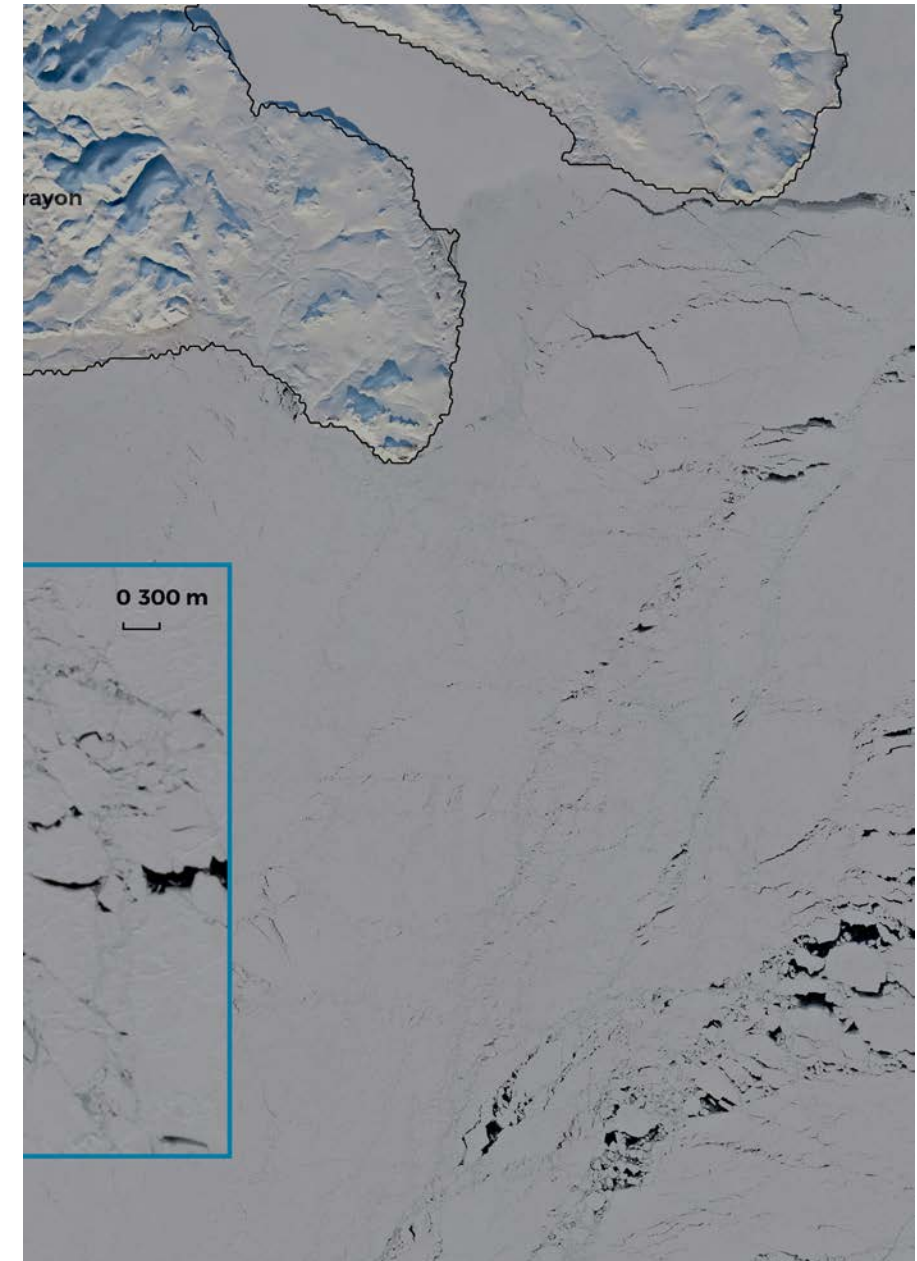
LNG carrier sailing across the Bering Strait

In recent years, climate change has been transforming the Arctic and opening up maritime routes that used to be closed in the winter. The altered seasonality of the sea ice floe in this fast-changing environment is attracting more shipping traffic than ever before.

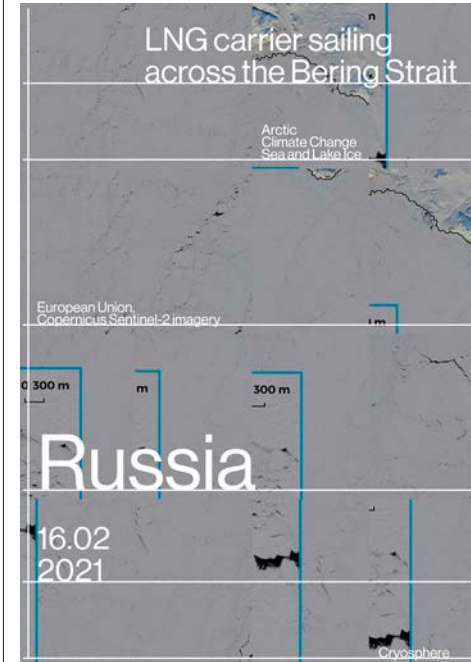
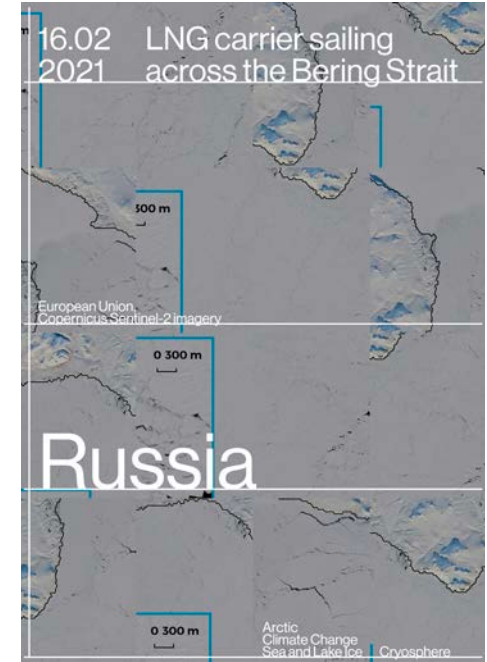
This image, acquired by one of the Copernicus Sentinel-2 satellites on 6 February, shows the Christophe de Margerie, a Russian icebreaking liquefied natural gas (LNG) carrier, travelling through the Bering Sea back to Sabetta, an LNG plant located in the Russian Arctic Ocean territory. This crossing was the first smooth crossing by a commercial vessel in February.

According to Rick Thoman, a climate specialist of the Alaska Center for Climate Assessment, a mere twenty years ago this route would not have been accessible due to the thickness of the ice that was making navigation impossible.

Open data supplied by the Copernicus Sentinel satellites are used for maritime surveillance around the world, in particular for developing sea ice charts.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery-

Category Cryosphere

Tag Arctic Climate Change Sea and Lake Ice

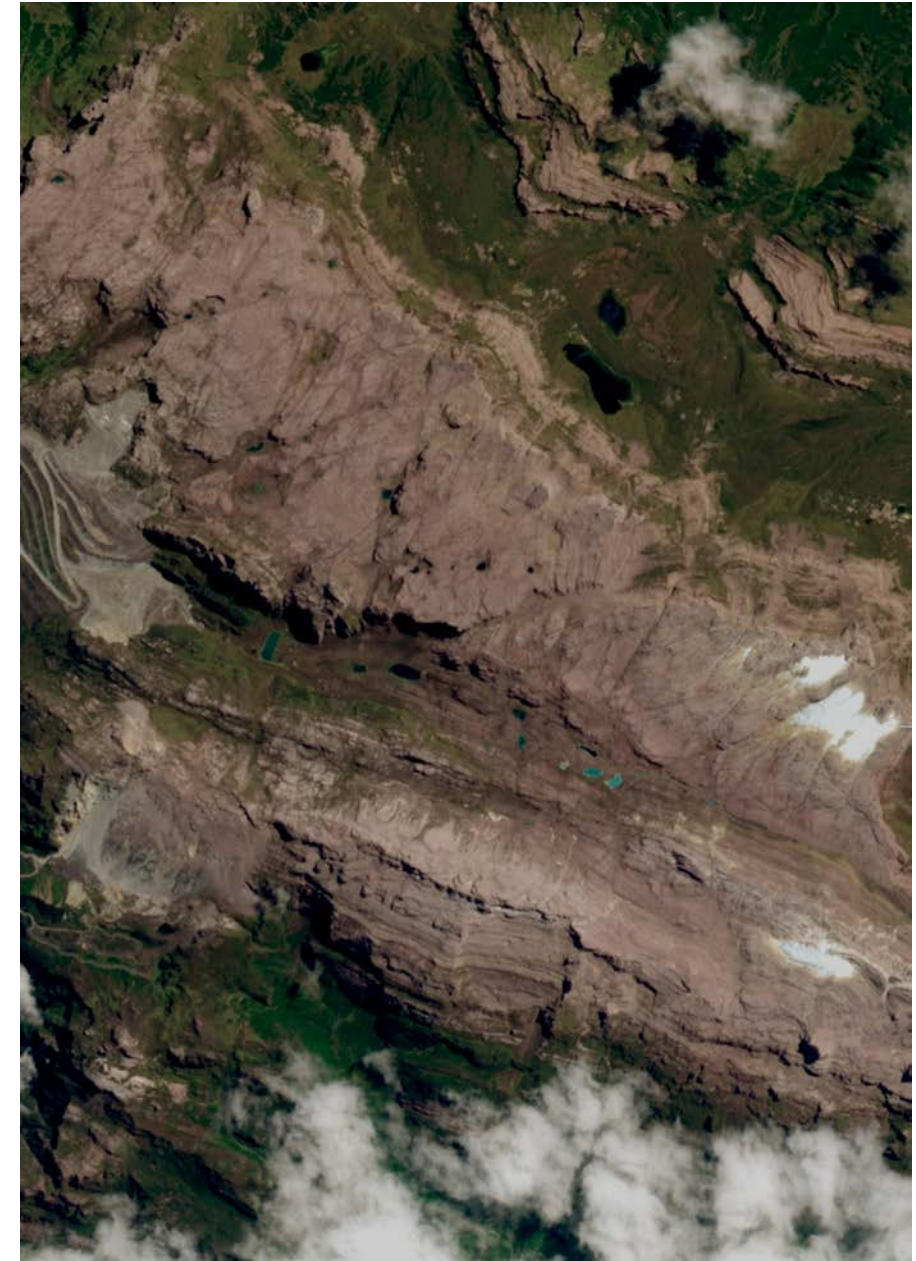
1 The Carstenz Glacier is poised to disappear

2 One of the most obvious signs of climate change is the speed
3 at which glaciers and ice sheets are disappearing – and not
4 only beyond the Arctic or the Antarctic Circle.

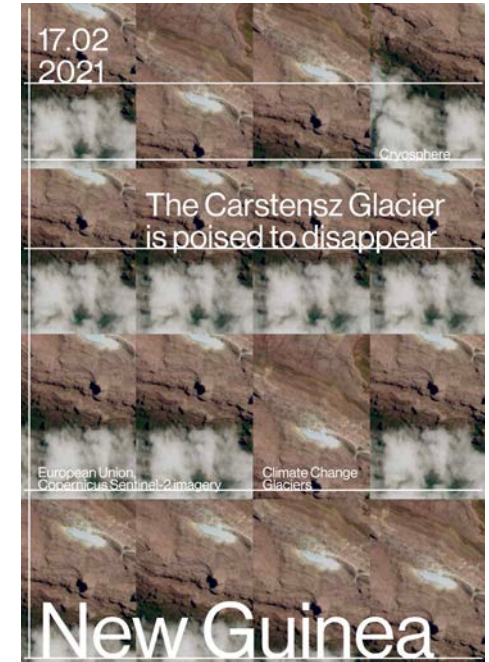
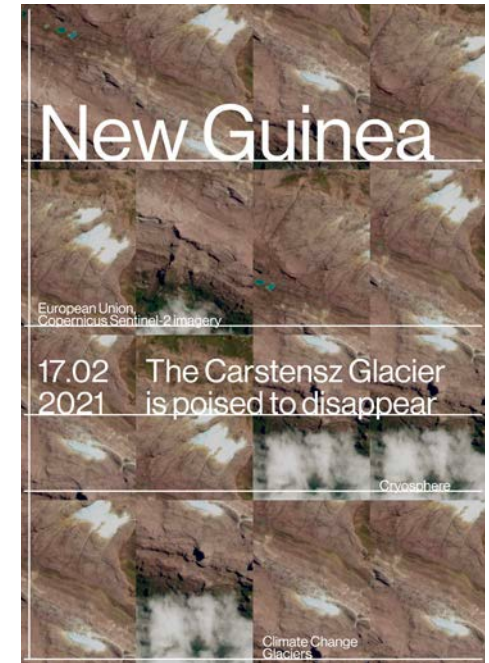
5 One of these is the Carstenz Glacier, near the peak of the
6 Puncak Jaya mountain on the South Pacific island of New
7 Guinea. According to Donald Sukma Permana, researcher at
8 the Indonesian Meteorological, Climatological, and Geophysical
9 Agency (BMKG), the Carstenz Glacier is likely to completely
10 disappear by 2023.

11 This image, acquired by one of the Copernicus Sentinel-2
12 satellites on 26 August 2020, shows the Carstenz Glacier,
13 whose surface area at the time of acquisition was less than
14 2 ha while it stood at 180 ha in 2005!

15 Open data supplied by the Copernicus Sentinel satellites are
16 used to monitor the retreat of glaciers around the world.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery

Category Cryosphere

Tag Climate Change Glaciers

Heaviest snowfall in 40 years

On 15 February 2021, the cold front Medea caused the heaviest snowfall recorded in Greece in the past four decades. The severe snowfall affected especially Central Greece and the region of Attica, including Athens, causing disruptions in transport, damage to water supply systems and power cuts for more than 70,000 people. Due to the devastating effects of the snowfall, a state of emergency was declared in several areas.

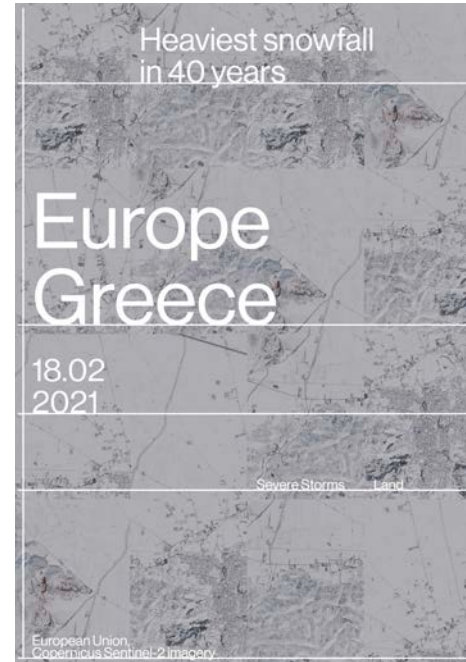
According to media reports, three deaths have already been reported as a result of Medea.

This image, acquired by one of the Copernicus Sentinel-2 satellites on 17 February 2021, shows the Athens–Thessaloniki motorway near Thiva, in Central Greece. The enormous amount of snow that made the driving conditions extremely hazardous led Greek authorities to close the motorway from Kifissia to Lamia during the passage of Medea. At the time of acquisition of this image, the motorway had reopened and traffic continued despite severe difficulties.

The Copernicus Land Monitoring Service product portfolio includes the High Resolution Snow and Ice Monitoring (HR-S&I) products that provide high-resolution (20m × 20m) snow and ice data for the 39 Member States of the European Environment Agency, in near real-time.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

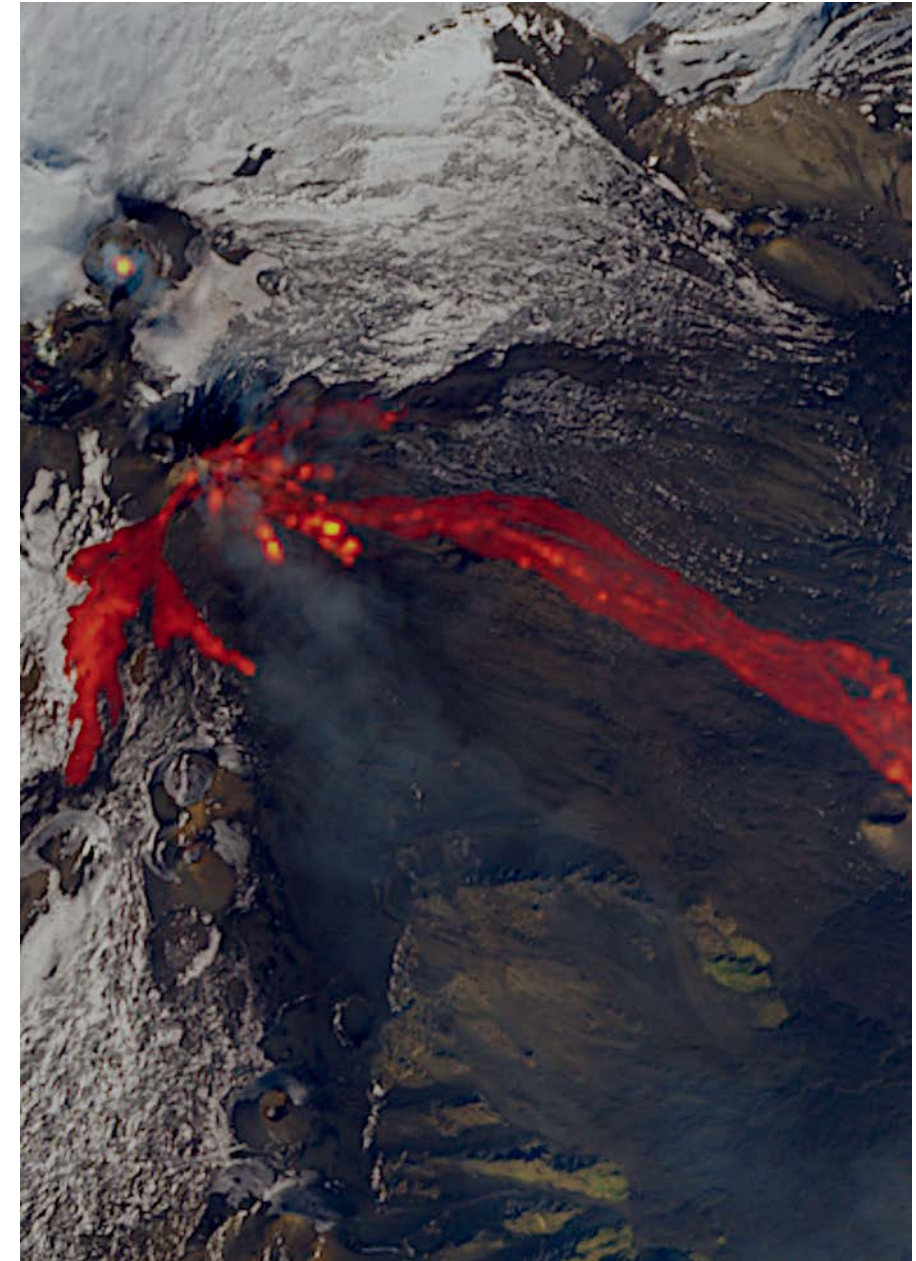
Tag Severe Storms

New eruptive phase of Mount Etna

A massive volcanic eruption of the Italian volcano Mount Etna began in the early hours of 18 February 2021, a mere 32 hours after the spectacular eruptive episode that occurred on 16 February.

This image, acquired by one of the Copernicus Sentinel-2 satellites on 18 February, shows the slowly cooling lava flowing into the Valle del Bove basin via the volcano's southeast flank. The eruption has been accompanied by lava fountains that reached heights of up to 700 m and a strong emission of ash, which rose to 9.1 km above sea level. At the pinnacle of the episode, the Aviation Color Code in the region was raised to Red, but lowered back to Orange after few hours. Ash and lapilli ejected from Mount Etna have been recorded southeast of the volcano in the municipalities of Zafferana, Fleri and Acireale, about 20 km away.

The Copernicus Sentinel-2 mission plays an important role in the monitoring of volcanic activity: its high spatial resolution and an average revisit time of five days gives insights into various volcanic processes.^a



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Land

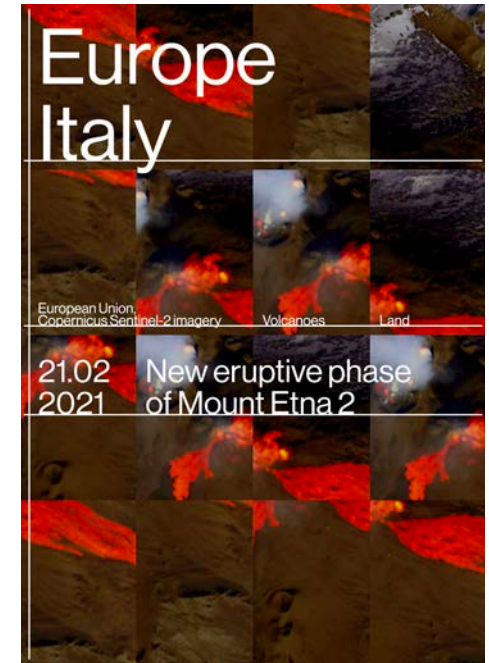
Tag Volcanoes

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30

New eruptive phase of Mount Etna 2



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery
 Category Land

Tag Volcanoes

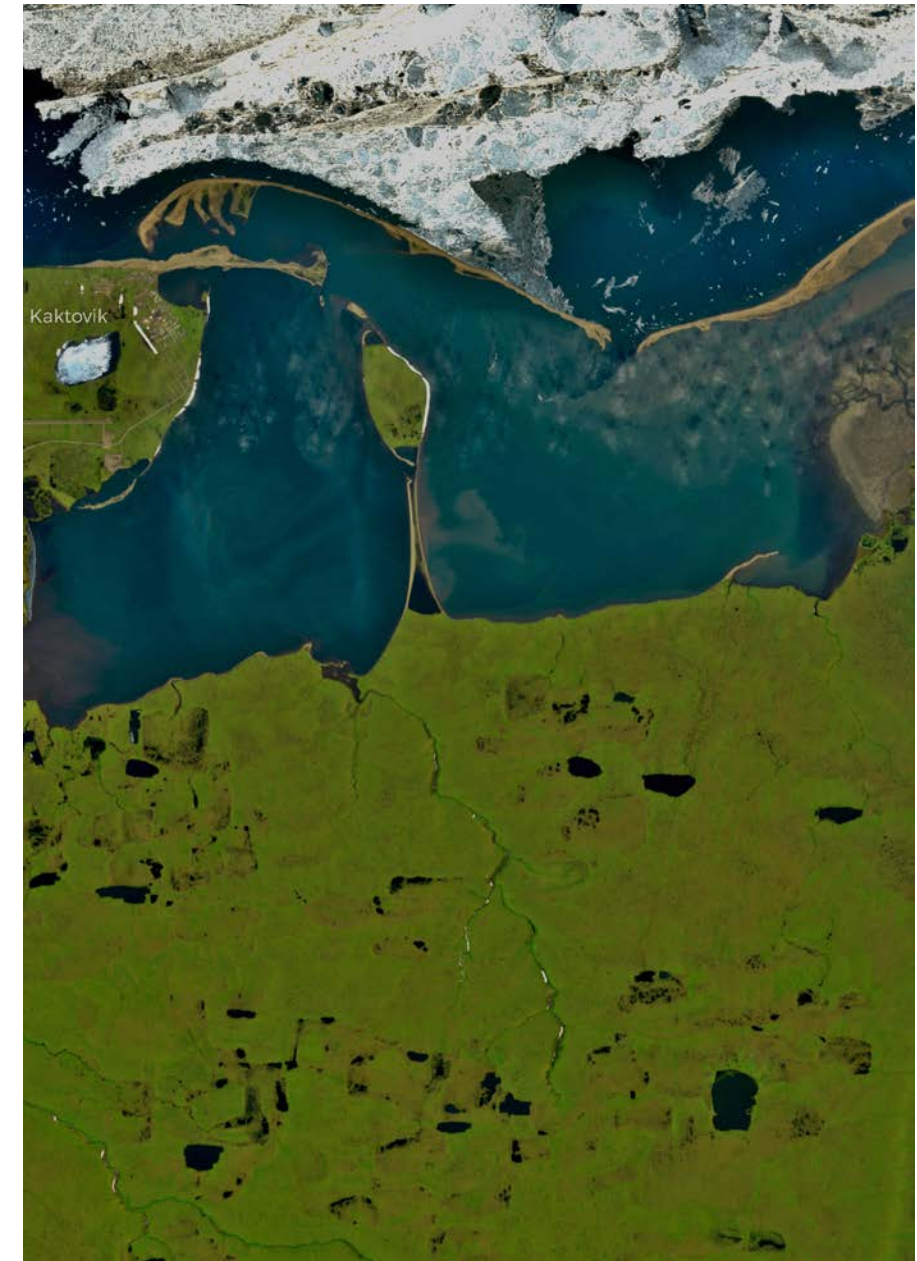
Oil and Gas exploration halted

The Arctic National Wildlife Refuge (ANWR), located in the very Northeast of Alaska, is the largest national wildlife refuge in the United States. It is home to various species of fauna, including polar bears, moose, and wolves, as well as countless migratory birds from around the world that visit it each year. The region first became a federal protected area in 1960, and the ANWR was established in two decades later, via the Alaska National Interest Lands Conservation Act. However, the United States Congress deferred the decision regarding management of oil and gas exploration in a coastal plain discussed in section 1002 (referred to as the 1002 area). This area spans about 600,000 ha and – despite its importance as a wildlife habitat – has been a subject of ongoing discussion due to the enormous oil and gas resources it is suspected to contain.

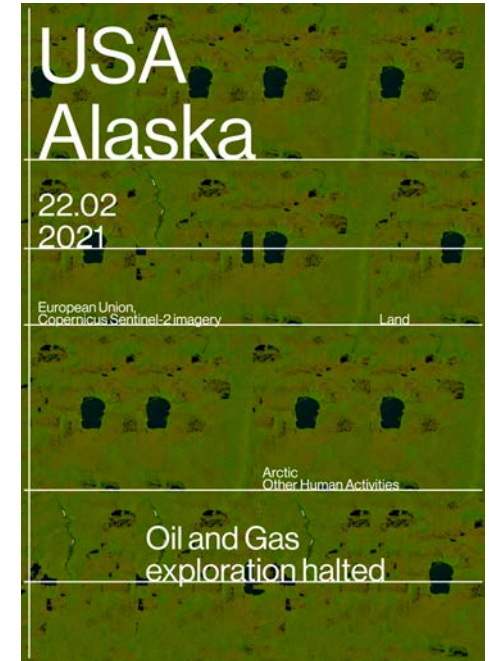
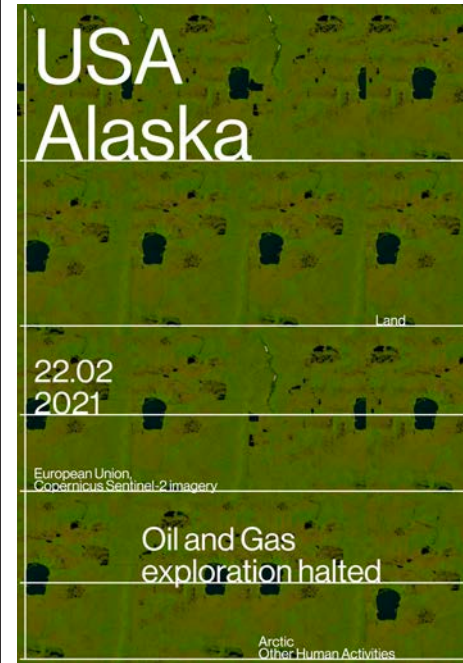
No decisions were made regarding oil exploration in the 1002 area until 2017, when a tax legislation that would allow for oil and gas drilling was signed into law. Numerous disputes followed this decision, but in February 2021, all exploration plans are reported to have finally fizzled out due to their lack of measures to ensure protection of the polar bear habitats of the ANWR.

This image, acquired by one of the Copernicus Sentinel-2 satellites on 7 July 2020, shows the city of Kaktovik in the 1002 area, around which seismic surveys were planned.

Data acquired by Copernicus Sentinel satellites are used operationally to monitor the impact of climate change on the fragile ecosystems in the Arctic.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union,
Copernicus Sentinel-2 imagery

Category Land

Tag Arctic
Other Human Activities

Salt Lake at risk of lead poisoning

This image, acquired by one of the Copernicus Sentinel-2 satellites on 2 February 2021, shows the Larnaca Salt Lake, one of the most significant protected wetlands in Cyprus.

The Larnaca Salt Lake comprises four salt lakes, three of which – Aliki, Orphani and Soros – are interconnected, with the smallest, so-called Airport Lake, having been cut off from Orphani after the construction of the runway of the Larnaca International Airport.

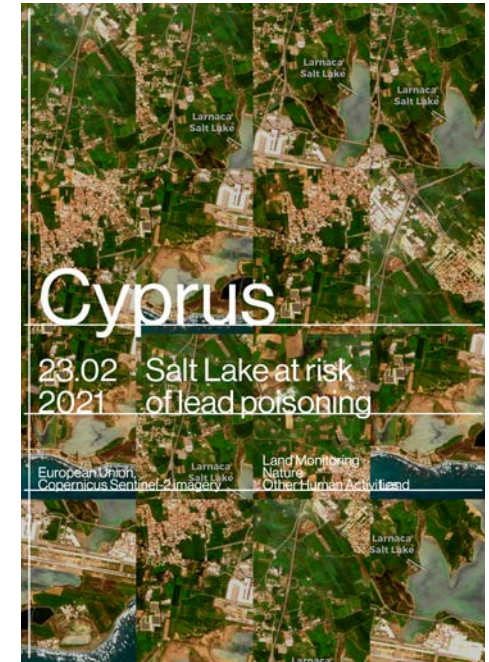
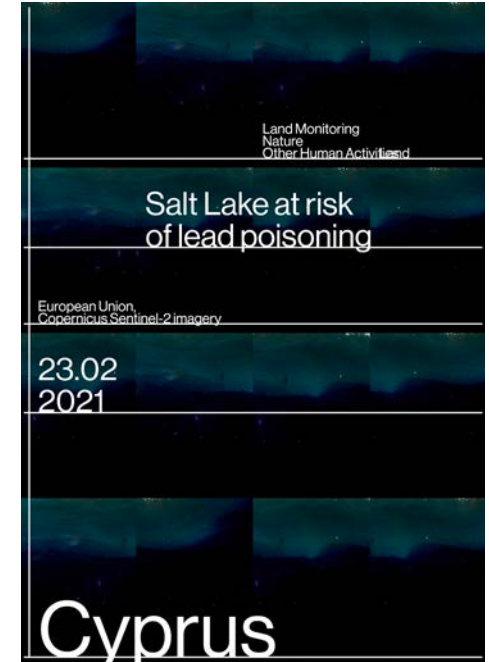
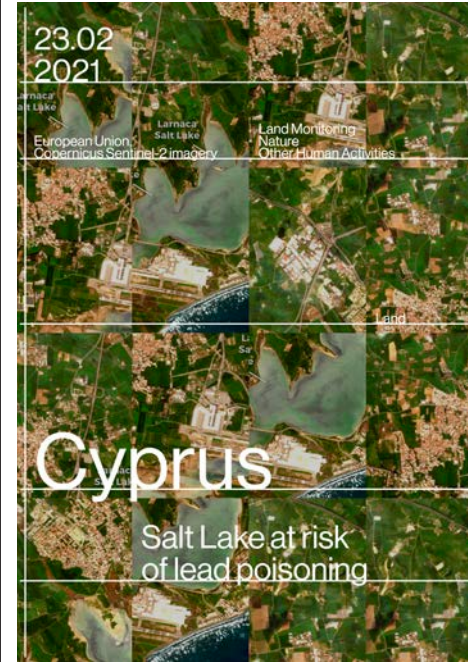
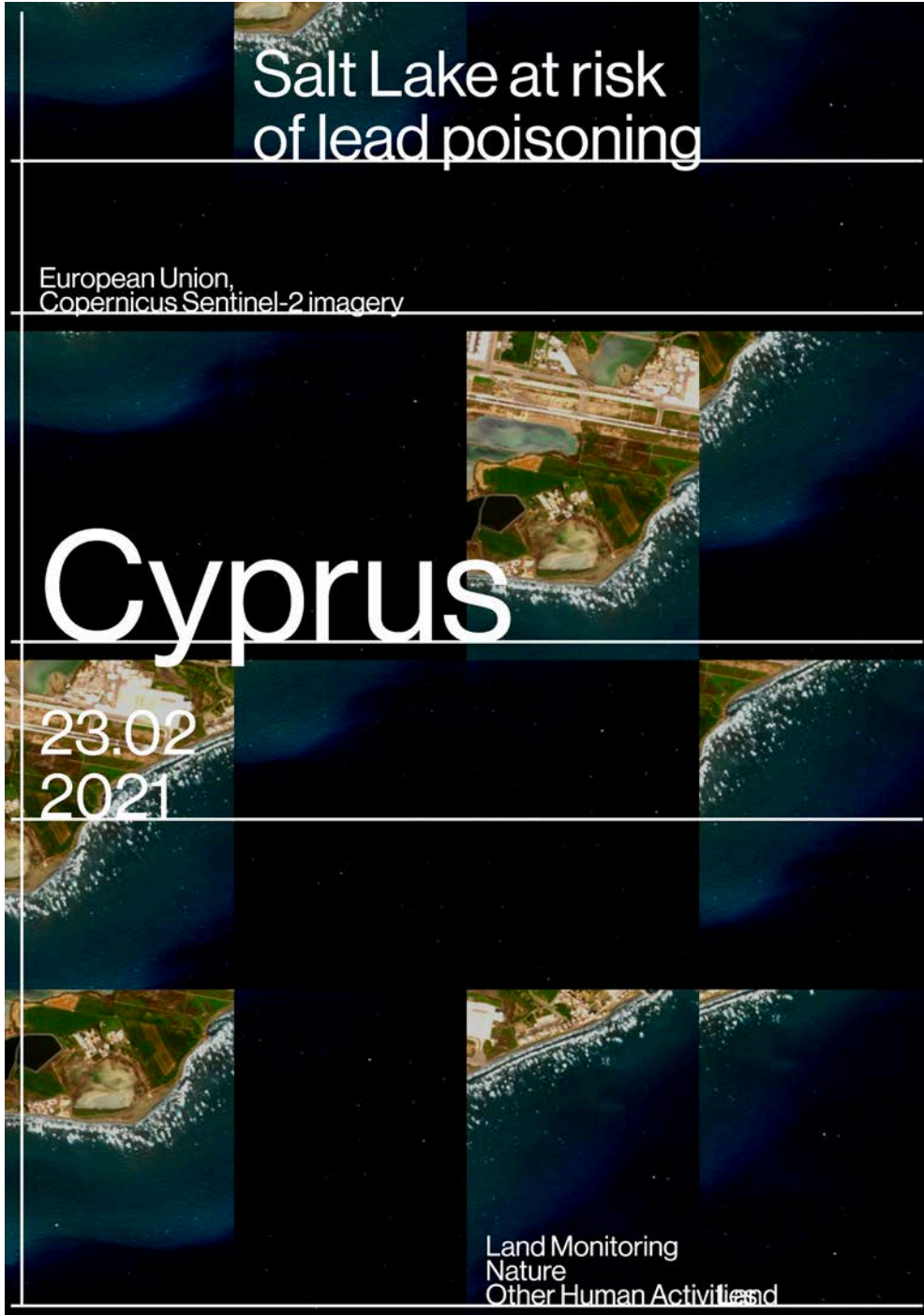
This area is a site of great international ecological significance that has been declared a Ramsar site, a Natura 2000 site, as well as a Special Protected Area under the Barcelona Convention. It is a crucial stop for many species of migratory birds that fly between Africa and Europe, welcoming as many as 15,000 flamingos each winter.

In 2020, 96 flamingos were found dead at the Larnaca Salt Lake. According to Martin Hellicar, the director of Birdlife Cyprus, they had ingested lethal quantities of lead shotgun pellets as they fed. Hellicar is urging authorities to ban hunting at the coast and protect flamingos from the risk of lead poisoning by shotgun pellets by extending the national and EU-wide bans on the use of lead pellets near wetlands, which may not be enforced successfully.

Every day the Copernicus Sentinel missions and Copernicus Services deliver petabytes of open data that can be used to monitor natural reserves worldwide.



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30



Credit European Union, Copernicus Sentinel-2 imagery

Category Land

Tag Land Monitoring Nature Other Human Activities



31.01.2021 S.62

Bazaruto Archipelago
National Park

04.02.2021 S.74

Western Cape



07.02.2021 S.86

Saharan dust turns
Pyrénées 2

02.02.2021 S.66

Skaptafellsjökull
glacier

05.02.2021 S.78

New York City
covered with snow

07.02.2021 S.90

Saharan dust turns
Pyrénées 3

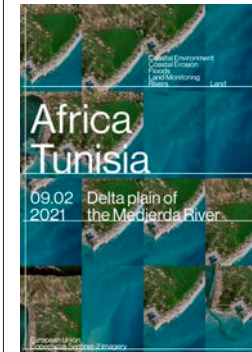
03.02.2021 S.70

Floods in Landes
department

07.02.2021 S.82

Saharan dust turns
Pyrénées

08.02.2021 S.94

Sicily and Saharan
dust from space

09.02.2021 S.98

Delta plain of
the Medjerda River

10.02.2021 S.110

Bruges blanketed in snow



10.02.2021 S.122

Liege blanketed in snow



10.02.2021 S.102

Bruxelles blanketed
in snow

10.02.2021 S.114

Ghent blanketed in snow



10.02.2021 S.126

Middelburg blanketed
in snow

10.02.2021 S.106

Antwerpen blanketed
in snow

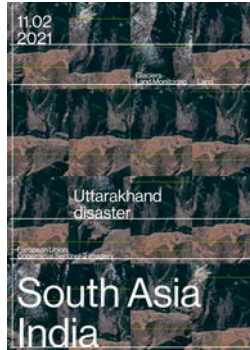
10.02.2021 S.118

Leiden blanketed in snow



10.02.2021 S.130

Rixensart and Lasne
blanketed in snow



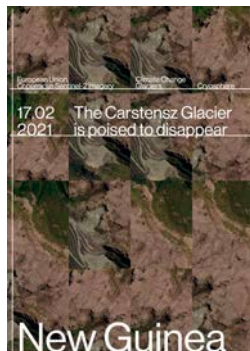
11.02.2021 S.134

Uttarakhand disaster



14.02.2021 S.146

Snowy Denmark and surrounding countries



17.02.2021 S.158

The Carstensz Glacier is poised to disappear



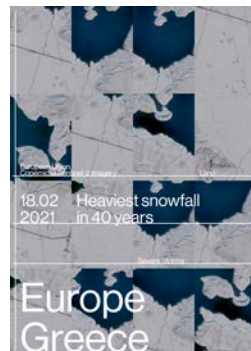
12.02.2021 S.138

Floods in Campania



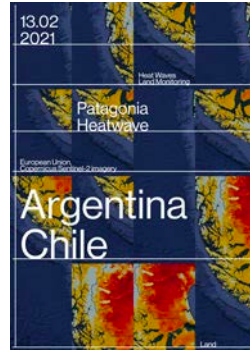
15.02.2021 S.150

Vehicles in flames after fuel tanker exploded



18.02.2021 S.162

Heaviest snowfall in 40 years



13.02.2021 S.142

Patagonia Heatwave



16.02.2021 S.154

LNG carrier sailing across the Bering Strait



19.02.2021 S.166

New eruptive phase of Mount Etna



21.02.2021 S.170

New eruptive phase of Mount Etna 2



22.02.2021 S.174

Oil and Gas exploration halted



23.02.2021 S.178

Salt Lake at risk of lead poisoning

Quellen

Textquellen

Schaffrinna, A. S. (2018, Juni 16). Generatives Design – Co-Kreation dank künstlicher Intelligenz. Designtagebuch. <https://www.designtagebuch.de/generatives-design-co-kreation-dank-kuenstlicher-intelligenz/>

Lorenz, M. (o. J.). Flexible Visuelle Identitäten. Design made in Germany. Abgerufen 20. August 2020, von <https://www.designmadeingermany.de/2013/140646/>

Was ist generatives Design à Tools und Software | Autodesk. (o. J.). autodesk. Abgerufen 24. August 2020, von <https://www.autodesk.de/solutions/generative-design>

About Copernicus | Copernicus. (o. D.). Copernicus. Abgerufen am 7. Februar 2021, von <https://www.copernicus.eu/en/about-copernicus>

copernicus-detail. (o. D.). Copernicus. Abgerufen am 7. Februar 2021, von <https://www.copernicus.eu/en/about-copernicus/copernicus-detail>

Copernicus. (o. D.-b). Image of the day | Copernicus. Image of the day. Abgerufen am 21. Februar 2021, von <https://www.copernicus.eu/en/media/image-day>

Bildquellen

Balland, L. (o. J.). Weltformat 2018 [Plakat]. Slanted. <https://www.slanted.de/10-weltformat-graphic-design-festival/>

Copernicus. Image of the day [Fotografie]. Copernicus. <https://www.copernicus.eu/en/media/image-day>

Impressum

Text, Bild und Gestaltung

Leon Wahlefeld

Schrift

Neue Haas Grotesk – Linotype

www.linotype.com

Papier

–

Druck

buch.one – Offsetdruckerei Grammlich

www.buch.one

Generative Gestaltung

Wie eine interdisziplinäre Herangehensweise die Zukunft des Designs revolutionieren könnte.