

Nachhaltigkeitsanalyse globaler Lieferketten

KONZEPTION EINES TOOLS ZUR AUSWERTUNG MULTIREGIONALER, ÖKOLOGISCH
ERWEITERTER INPUT-OUTPUT-MODELLE ZUR ERFASSUNG VON UMWELTWIRKUNGEN
ENTLANG GLOBALER LIEFERKETTEN

ABSCHLUSSBERICHT DER KOOPERATIONSPHASE 2024/25

27. September 2025

Verfasser:

Lukas HARZBECKER
Jonas LAMBRECHT

Unter Aufsicht von:

Prof. Dr. Hendrik LAMBRECHT

Inhaltsverzeichnis

Abstract	1
1 Einleitung	2
1.1 Motivation	2
1.2 Zielsetzung	3
2 Material und Methoden	4
2.1 Input-Output-Modell	4
2.2 EXIOBASE	7
2.3 Verwendete Python-Bibliotheken	9
2.4 Analyse- und Visualisierungsmethoden	10
3 Ergebnisse	12
3.1 Aufbau des EXIOBASE-Explorers	12
3.2 Funktion der Klasse IOSystem	13
3.2.1 Optimierung des Datenformats und Transformation der Ausgangsdaten .	13
3.2.2 Berechnung der Matrizen zur Analyse sektoraler Umweltauswirkungen .	14
3.2.3 Hierarchische Gruppierung der Daten mittels MultiIndices	15
3.3 Funktion der Klasse SupplyChain	15
3.3.1 Initialisierung und Datenauswahl	16
3.3.2 Berechnung von Umweltauswirkungen entlang von Lieferketten	16
3.3.3 Visualisierung der Ergebnisse	17
3.4 Benutzeroberfläche (GUI)	18
3.4.1 Architektur und Navigation	18
3.4.2 Datenauswahl und -filterung (SelectionTab)	19
3.4.3 Anwendungseinstellungen (SettingsTab)	20
3.4.4 Ergebnisvisualisierung (VisualisationTab)	21
3.4.5 Interaktive Konsole (ConsoleTab)	22
4 Diskussion	24
4.1 Einordnung der Ergebnisse	24
4.2 Anwendungsbereiche der Software	24
4.3 Grenzen und methodische Einschränkungen	25
4.4 Ausblick	26
5 Danksagung	27
Literaturverzeichnis	28
Anhang	29
Selbstständigkeitserklärung	30

Abstract

This documentation presents the EXIOBASE-Explorer¹, a Python-based tool to facilitate the analysis and visualization of multiregional input–output (MRIO) data derived from the EXIOBASE database. As globalization increasingly intertwines production supply chains, their environmental impacts — spanning countries, industries, and multiple production stages — become more difficult to trace and quantify. Urgent challenges such as climate change, biodiversity loss, and resource scarcity require precise tools to inform companies, policymakers, and society.

The EXIOBASE-Explorer addresses this need by combining optimized NumPy routines and Pandas DataFrames to efficiently load and manipulate the 9800×9800 coefficient matrices of the EXIOBASE database. Through the use of a custom *Fast-Load Database* format, data loading times are reduced from approximately two minutes to under three seconds—achieving performance gains of nearly two orders of magnitude compared to existing tools such as `pymrio`. A PyQt5 graphical interface enables interactive exploration of environmental impact indicators such as greenhouse gas emissions or water use across specified supply chains.

The tool facilitates detailed assessments across different regions, sectors, and environmental indicators. For example, it supports the identification of discrepancies between labor input and value generation, as well as the localization of emission hotspots within global production networks. Due to its modular tab-based interface and extensible class architecture, the EXIOBASE Explorer provides a user friendly and easily extendable foundation for advanced sustainability analyses.

¹Available on github (see attachment)

1 Einleitung

1.1 Motivation

Die fortschreitende Globalisierung vernetzt Produktions- und Lieferketten weltweit immer enger. Diese Entwicklung fördert zwar das ökonomische Wachstum, erschwert jedoch gleichzeitig die transparente Erfassung der daraus resultierenden Umweltbelastungen. So entstehen beispielsweise Treibhausgasemissionen, hoher Wasserverbrauch oder Landnutzungsänderungen oft in den frühen Stufen der Wertschöpfungskette oder in anderen Weltregionen, während der Konsum der Endprodukte – und damit der ökonomische Nutzen – in einer ganz anderen Region liegt. Klassische volkswirtschaftliche Input-Output-Analysen sind in erster Linie darauf ausgelegt, wirtschaftliche Zusammenhänge innerhalb einer nationalen Volkswirtschaft abzubilden, etwa um die Auswirkungen von Nachfrageänderungen oder Steueranpassungen auf die Produktion zu untersuchen. Für globale Fragestellungen und die Quantifizierung von Umweltwirkungen sind sie jedoch nur bedingt geeignet, da sie in der Regel keine internationalen Vorleistungsverflechtungen oder umweltbezogenen Daten berücksichtigen.

Angesichts drängender Herausforderungen wie dem Klimawandel, globalen sozialen Disparitäten und Ressourcenknappheit wächst der Bedarf an leistungsfähigen, global ausgerichteten Analysewerkzeugen. Lehrende, Forschende und politische Entscheidungstragende benötigen interaktive und performante Systeme, die komplexe, multilaterale Wirtschafts- und Umweltzusammenhänge nachvollziehbar abbilden.

Mit bestehenden Bibliotheken wie `pymrio` lassen sich umfangreiche MRIO-Datensätze zwar laden und auswerten, jedoch entstehen dabei oft lange Ladezeiten und die Flexibilität bleibt eingeschränkt. Durch die Entwicklung des EXIOBASE-Explorers sollte daher eine benutzerfreundliche und zugleich wissenschaftlich exakte Plattform geschaffen werden, die eine schnelle, intuitive und graphische Analyse globaler Lieferketten ermöglicht.

1.2 Zielsetzung

Das Ziel besteht darin, ein Programm zu entwickeln, mit dem sich die Umweltwirkungen globaler Lieferketten auf Basis des MRIO-Datensatzes EXIOBASE systematisch analysieren und anschaulich darstellen lassen. Die Anwendung soll in Python entstehen und dabei sowohl modular aufgebaut sein als auch eine nutzerfreundliche Bedienung ermöglichen. Die einzelnen Komponenten – etwa für die Auswahl eines Sektors, die Aggregation von Daten oder die grafische Darstellung – sollen unabhängig voneinander arbeiten können und sich flexibel erweitern und anpassen lassen.

Durch einen einfachen Aufbau sollen auch Personen ohne tiefgehende Programmierkenntnisse nachvollziehen können, wie sich beispielsweise der CO₂-Ausstoß oder der Wasserverbrauch entlang der Lieferkette eines bestimmten Produkts verteilt – sowohl strukturell auf die einzelnen Wertschöpfungsstufen als auch räumlich auf die beteiligten Regionen.

Ein zentrales Anliegen besteht darin, wissenschaftlich etablierte Darstellungsformen, wie sie in Unterabschnitt 2.4 erläutert werden, softwaregestützt umzusetzen. Auf diese Weise möchten wir dazu beitragen, komplexe MRIO-Daten besser zugänglich zu machen und ein Werkzeug bereitstellen, das sich sowohl in der Forschung als auch in der Politik und Bildungsarbeit einsetzen lässt.

2 Material und Methoden

Im folgenden Kapitel „Material und Methoden“ werden die für das Projekt verwendeten Grundlagen und Werkzeuge vorgestellt. Zunächst erläutert Unterabschnitt 2.1 das zugrunde liegende Input-Output-Modell. Darauf aufbauend beschreibt Unterabschnitt 2.2 die EXIOBASE-Datenbank als Datengrundlage zur praktische Umsetzung dieses Modells. In Unterabschnitt 2.3 werden die eingesetzten Python-Pakete aufgeführt und ihre Bedeutung für Datenaufbereitung und Analyse erläutert. Abschließend gibt Unterabschnitt 2.4 einen Überblick über etablierte Methoden zur Visualisierung von Lieferketten.

2.1 Input-Output-Modell

Das Input-Output-Modell beschreibt Wirtschaftsprozesse einer Region oder eines Landes als ein Netzwerk verflochtener Sektoren (siehe Vorleistungsmatrix, Abbildung 1). Jeder Bereich produziert Güter oder Dienstleistungen und nutzt zugleich Leistungen anderer Sektoren. Die Input-Output-Tabelle zeigt in den Zeilen, wie viel ein Sektor erzeugt und an andere Sektoren oder die Endnachfrage abgibt (Output bzw. Verwendung). In den Spalten sieht man, in welchem Umfang er Leistungen anderer Sektoren in Anspruch nimmt (Input bzw. Aufkommen). So lässt sich quantifizieren, wie stark einzelne Sektoren voneinander abhängen.

Verwendung Aufkommen		Input der Produktionsbereiche			Letzte Verwendung			Gesamte Verwendung
		PB	SB	TB	Konsum	Investit.	Exporte	
Gütergruppen	PB	Vorleistungsmatrix			Endnachfragematrix			Σ
	SB							
	TB							
Ges. Vorleistungen bzw. Endnachfrage		Σ 30,4						
Komponenten der Wertschöpfung	...	Matrix der Primärinputs						
	...							
	...							
Importe								
Gesamtes Aufkommen		Σ						

Gesamtes Aufkommen gleich gesamte Verwendung

Abbildung 1: Schematische Darstellung einer Input-Output-Tabelle (verändert nach Andreas Kuhn, 2010 [4])

Im dargestellten Beispiel erfolgt die Unterteilung der Produktionsbereiche in drei klassische Wirtschaftssektoren: den primären (PB), den sekundären (SB) und den tertiären Bereich (TB). In realen Systemen wird in der Regel eine deutlich feinere Gliederung verwendet.

Tabellen der volkswirtschaftlichen Input-Output-Analyse stellen prinzipiell sicher, dass das gesamte Aufkommen (Summe aller Inputs) der gesamten Verwendung (Summe aller Outputs) entspricht; dies gilt auch für die hier dargestellte Tabelle.

Im Zentrum des Modells steht die sogenannte technische Koeffizientenmatrix $\mathbf{A}$, auch Vorleistungsmatrix genannt, deren Eintrag a_{ij} angibt, wie viele Einheiten aus Branche i benötigt werden, um eine Einheit Output in Branche j zu erzeugen. Rechts daneben bildet die Endnachfragematrix die Nachfrage von Haushalten, Staat, Investitionen und Export ab. Die untere Matrix der Primärintputs enthält u. a. Importe sowie Löhne und Kapitalentgelte. Eine Veränderung der Endnachfrage wirkt sich somit nicht nur direkt auf einen Sektor, sondern indirekt auch auf alle vor- und nachgelagerten Bereiche aus. Mathematisch beschreibt die Leontief-Gleichung diesen Zusammenhang:

$$\mathbf{x} = (\mathbf{I} - \mathbf{A})^{-1} \mathbf{y}$$

Dabei bezeichnet $\mathbf{x}$ den Vektor der Gesamtproduktion und $\mathbf{y}$ den Vektor der Endnachfrage.

Die Leontief-Inverse

Die Leontief-Inverse $(\mathbf{I} - \mathbf{A})^{-1}$ stellt die mathematische Grundlage eines Input-Output-Modells dar. Sie fasst alle direkten und indirekten Produktionsverflechtungen in einer einzigen Matrix zusammen und zeigt, wie sich eine Änderung der Endnachfrage in einem Sektor über beliebig viele Produktionsstufen hinweg auf die gesamte Volkswirtschaft auswirkt.

Anschaulich lässt sich diese Matrix-Inverse als Summe aller Produktionsschritte interpretieren: Zunächst wird die direkte Nachfrage an Vorleistungen betrachtet ($\mathbf{A}$), im zweiten Schritt die Vorleistung der Vorleistungen ($\mathbf{A}^2$), und so weiter. Die Potenzen der technischen Koeffizientenmatrix $\mathbf{A}^n$ repräsentieren Effekte, die erst nach n Zwischenstufen auftreten, wobei höhere Potenzen weiter entfernte Wirkungen abbilden.

Die Leontief-Inverse lässt sich mathematisch als geometrische Reihe schreiben:

$$(\mathbf{I} - \mathbf{A})^{-1} = \mathbf{I} + \mathbf{A} + \mathbf{A}^2 + \mathbf{A}^3 + \dots$$

Anstelle einer schrittweisen Berechnung jeder Produktionsrunde wird die Reihe mithilfe der Matrixinversion zusammengefasst. Die genaue Herleitung dieser Inversion bleibt an dieser Stelle außen vor – entscheidend ist, dass sich durch sie sämtliche Vorleistungsverflechtungen mit geringem Rechenaufwand bestimmen lassen. Dies ist besonders vorteilhaft, weil wirtschaftliche Vorleistungsmatrizen in der Regel Rückkopplungen enthalten ($a_{kl} \neq 0$ und gleichzeitig $a_{lk} \neq 0$). Solche Zirkelbezüge machen die zugrunde liegende geometrische Reihe tatsächlich unendlich, sodass die Matrixinversion nicht nur der Effizienz dient, sondern überhaupt erst eine geschlossene, berechenbare Lösung ermöglicht.

Multiregionale Input-Output-Modelle (MRIO)

Multiregionale Input-Output-Modelle (MRIO) erweitern das klassische Modell für einzelne Volkswirtschaften auf globale Wirtschaftsbeziehungen. Sie bilden nicht nur die sektorale Struktur einzelner Volkswirtschaften ab, sondern verknüpfen diese Regionen explizit über internationale Handelsströme. Auf diese Weise entsteht ein geschlossenes Modell, das die gesamte Weltwirtschaft in einem einzigen, umfassenden Produktions- und Konsumkreislauf darstellt.

Im Unterschied zu vielen nationalen IO-Modellen sind bei MRIO-Systemen in der Regel keine externen Import- oder Exportgrößen erforderlich: Sämtliche Vorleistungsflüsse und Endnachfragen sind innerhalb des Modells verortet. Es gilt das Prinzip der geschlossenen Bilanz: Was an einer Stelle produziert wird, wird an anderer Stelle im System nachgefragt. Nichts geht verloren, nichts kommt von außen hinzu. Deswegen besitzen MRIO-Modelle im Gegensatz zu normalen IO-System-Modellen weder eine Primärinput- noch eine Endnachfragematrix (siehe Abb. 1).

Die technische Koeffizientenmatrix $\mathbf{A}$ eines MRIO ist entsprechend blockstrukturiert. Jeder Block $\mathbf{A}^{(rs)}$ beschreibt, wie viele Vorleistungen Region r aus Sektor s bezieht. Die Struktur lässt sich vereinfacht wie folgt darstellen:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}^{(11)} & \mathbf{A}^{(12)} & \dots & \mathbf{A}^{(1R)} \\ \mathbf{A}^{(21)} & \mathbf{A}^{(22)} & \dots & \mathbf{A}^{(2R)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{(R1)} & \mathbf{A}^{(R2)} & \dots & \mathbf{A}^{(RR)} \end{pmatrix}$$

Jeder Block $\mathbf{A}^{(rs)}$ hat dabei die Dimension $N \times N$, wobei N die Anzahl der Sektoren pro Region ist. Die vollständige Matrix $\mathbf{A}$ umfasst somit $R \cdot N$ Zeilen und Spalten. Dieses Modell ermöglicht eine präzise Nachverfolgung globaler Lieferketten bis in ihre kleinsten Bestandteile – sowohl innerhalb einzelner Länder als auch über Kontinente hinweg.

Umweltauswirkungen als Erweiterung

Ein wesentliches Merkmal moderner MRIO-Modelle besteht in der Möglichkeit, sozioökonomische Daten mit Umwelt- oder Ressourceninformationen zu verknüpfen. So lassen sich beispielsweise CO₂-Emissionen, Landnutzung oder Arbeitszeit sektor- und regionsspezifisch erfassen und in das Modell integrieren. Diese sogenannten *Impacts* werden als Matrix $\mathbf{F}$ gespeichert, die angibt, wie viel einer bestimmten Umweltwirkung pro Einheit Gesamtproduktion in einem Sektor entsteht.

Aus der Kombination dieser Matrix mit der Leontief-Inverse ergibt sich die sogenannte $\mathbf{D}_{\text{cba}}$ -Matrix. Sie fasst für jede Kombination aus Region r und Sektor s zusammen, welche direkten und indirekten Umweltwirkungen durch den Konsum einer zusätzlichen Einheit entstehen. Sie wird gebildet, indem die Leontief-Inverse $(\mathbf{I} - \mathbf{A})^{-1}$ mit der Impact-Matrix $\mathbf{F}$ multipliziert wird.

Die $\mathbf{D}_{\text{cba}}$ -Matrix bildet somit die Grundlage vieler Analysen: Sie erlaubt es, die Gesamtwirkung beliebiger Konsumvektoren auf Umweltindikatoren zu berechnen, ohne die komplexen Lieferketten erneut vollständig durchlaufen zu müssen. Für jede Umweltkategorie (z. B. Treibhausgase, Landnutzung, Feinstaub) wird ersichtlich, wie stark ein Sektor einer Region zur Belastung beiträgt, wenn seine Produktion um eine Einheit erhöht wird.

2.2 EXIOBASE

EXIOBASE [1] ist eine multiregionale Input-Output-Datenbank (MRIO), die von der Europäischen Kommission initiiert wurde und im Rahmen mehrerer Forschungsprojekte kontinuierlich gepflegt wird. Sie liefert ein global konsistentes Abbild ökonomischer Transaktionen und ihrer Umweltwirkungen und dient als zentrale Datenquelle für Lieferketten- und Nachhaltigkeitsanalysen.

Versionen, Regionen und Sektoren

Von 1995 bis 2022 sind jährliche Versionen von EXIOBASE erschienen. Jede Version enthält

- $R = 49$ Regionen (44 Länder + 5 Restweltregionen),
- $N = 200$ Wirtschaftssektoren pro Region,
- $R \times N = 9\,800$ Kombinationen aus Region und Sektor.

Da es sich um ein MRIO handelt (vgl. Abschnitt 2.1), sind alle internationalen Vorleistungsflüsse in einem geschlossenen System modelliert – Importe und Exporte werden nicht separat ausgewiesen, sondern als direkte Verflechtungen zwischen den Regionen.

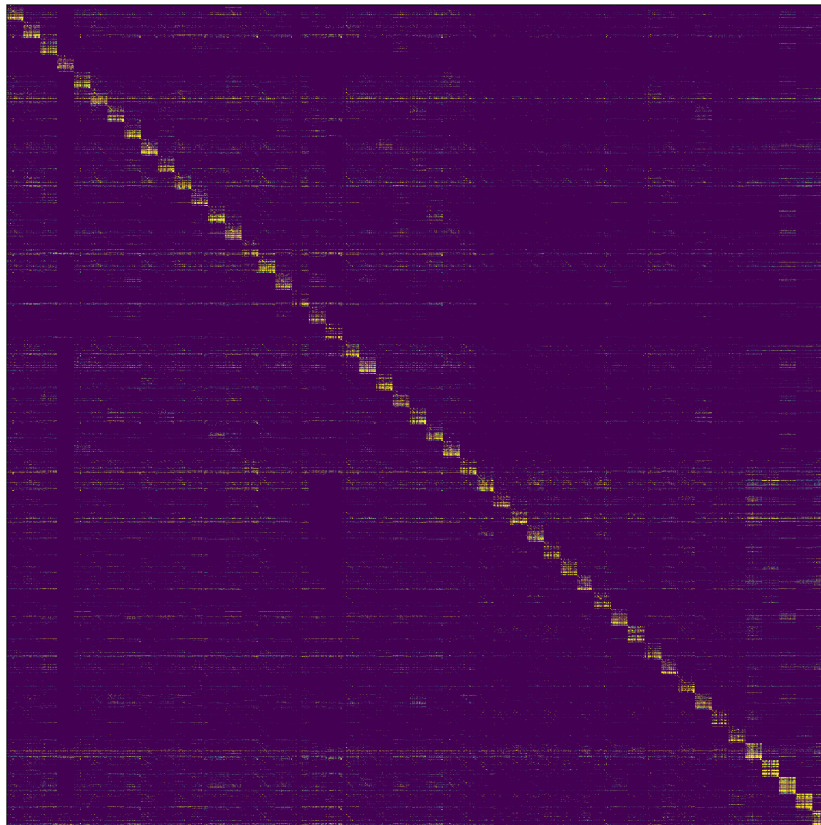


Abbildung 2: Visualisierung der technischen Koeffizientenmatrix A des Jahres 2022

Abbildung 2 verdeutlicht, dass jede Region-Sektor-Kombination vor allem innerhalb derselben Region umfangreiche Vorleistungen nachfragt (helle Diagonale). Die verblassenden Off-

Diagonalelemente stehen für die zum Teil geringeren, aber zahlreich vorhandenen globalen Handelsbeziehungen.

Zentrale Matrizen

Im Folgenden sind die wichtigsten EXIOBASE-Matrizen im Detail aufgelistet:

- **A** (technische Koeffizienten, Größe $9\,800 \times 9\,800$) – zeigt die Intermediärverflechtungen zwischen allen Region-Sektor-Einheiten.
- **Y** (Endnachfrage, Größe $9\,800 \times M$) – unterteilt nach Haushalten, Staat, Investitionen und Exporten; M ist die Anzahl der Endnachfragetypen (7 pro Region, 343 insgesamt).
- **F** (Umweltauswirkungen, Größe $E \times 9\,800$) – $E = 126$ Umweltindikatoren (z. B. Treibhausgasemissionen, Wasserverbrauch), pro Spalte werden die direkten Umweltwirkungen pro Produktionseinheit hinterlegt.

Weitere Matrizen, wie beispielsweise die Leontief-Inverse **L**, lassen sich aus diesen Matrizen ableiten.

EXIOBASE ermöglicht durch diese breite Auswahl an Matrizen detaillierte Analysen von globalen Lieferkettenstrukturen sowie die Abschätzung direkter und indirekter Umweltwirkungen verschiedener Nachfrageszenarien. Damit stellt die Datenbank eine wichtige Grundlage für wirtschafts- und umweltwissenschaftliche Forschung sowie politische Entscheidungsprozesse dar.

2.3 Verwendete Python-Bibliotheken

Die Entwicklung des EXIOBASE-Explorers basiert maßgeblich auf vier Python-Bibliotheken:

pymrio Die Bibliothek **pymrio** [7] bietet eine umfassende Schnittstelle zum Laden und Analysieren von EXIOBASE-Daten und diente zunächst als funktionaler Ausgangspunkt. Erste Tests zeigten jedoch, dass das Einlesen umfangreicher MRIO-Datensätze (EXIOBASE 3.8, 9800×9800) mit erheblichen Ladezeiten verbunden war. Daher wurde, basierend auf **pymrio**, eine eigene, auf Geschwindigkeit optimierte Lösung zur Datenverarbeitung implementiert.

numpy Die Bibliothek **numpy** [2] ermöglicht eine hocheffiziente Ausführung von Vektor- und Matrizenoperationen. Die im Projekt verwendeten sogenannten *Fast-Load-Databases* speichern die Daten in kompakten **.npy**-Dateien, die besonders schnelle Lese- und Schreiboperationen erlauben. Speziell bei der Berechnung der Leontief-Inverse und bei umfangreichen Matrizenmultiplikationen zeigen die optimierten C-Routinen von **numpy** ihre Vorteile.

pandas Mit **pandas** [6] lassen sich sämtliche Daten in **DataFrames** mit hierarchischem **MultiIndex** organisieren. Dies ermöglicht eine flexible Strukturierung nach Region, Wirtschaftssektor und weiteren Klassifikationsebenen. Dadurch wird etwa die gezielte Analyse spezifischer Wirtschaftsabteilungen oder Ländergruppen erleichtert.

PyQt5 Für die Implementierung der grafischen Benutzeroberfläche wird **PyQt5** [5] eingesetzt. Durch variable Tabs, ausklappbare Menüs und flexible Eingabefelder wird eine intuitive Nutzung des Explorers ermöglicht – auch für Anwender:innen ohne Programmierkenntnisse.

2.4 Analyse- und Visualisierungsmethoden

Nachdem die mathematischen und programmiertechnischen Grundlagen erläutert wurden, stellt sich die zentrale Frage, wie sich Ergebnisse aus einem globalen MRIO-Datensatz wie EXIOBASE systematisch auswerten und grafisch darstellen lassen. Als inhaltliche Referenz dient der „Umweltatlas Lieferketten“ von Jungmichel [3], in dem exemplarisch für den deutschen Textilsektor zwei Visualisierungsansätze präsentiert werden: eine tabellarische Darstellung der Umweltindikatoren entlang der Wertschöpfungsstufen sowie eine Weltkarte, die die geografische Verteilung der Umweltwirkungen zeigt. Ziel ist es, diese beiden Darstellungsformen zu verallgemeinern und so zu abstrahieren und zu implementieren, dass sie sich flexibel auf beliebige Sektoren, Regionen und Umweltindikatoren übertragen lassen.

Verteilung auf Wertschöpfungsstufen

Die erste Darstellungsform gliedert die Umweltwirkungen nach vier Stufen der Wertschöpfungskette:

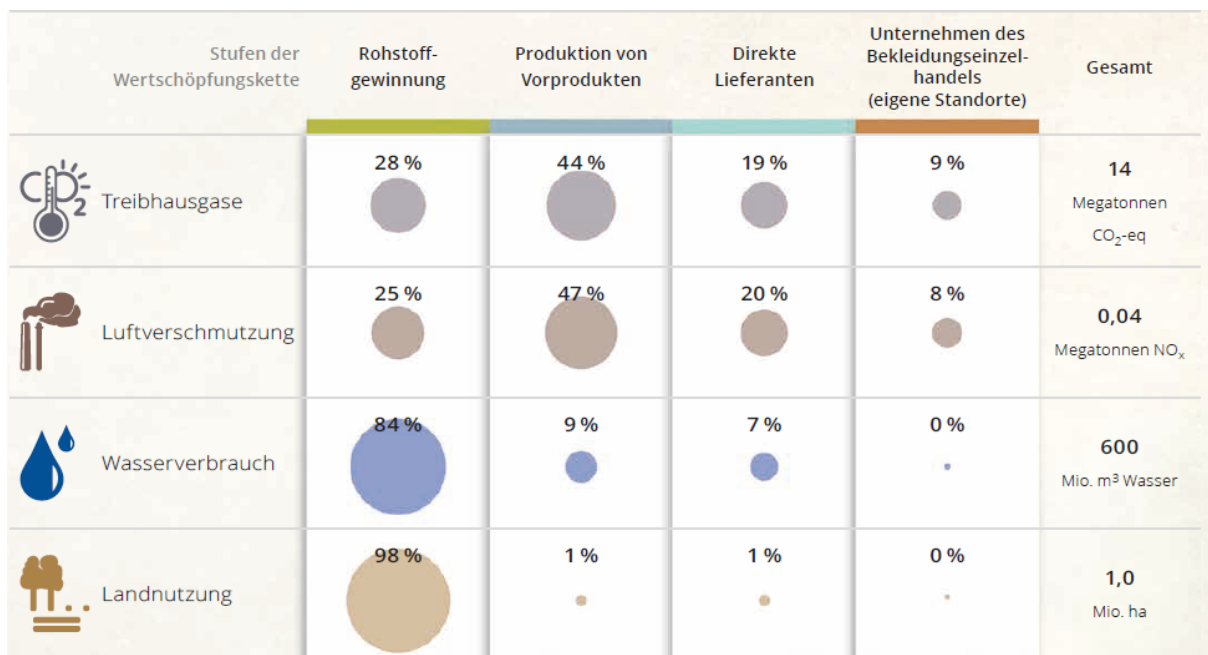


Abbildung 3: Beispielhafte Verteilung von Umweltindikatoren auf Wertschöpfungsstufen (Quelle: Jungmichel [3]).

Die tabellarische Darstellung (Abbildung 3) erleichtert die Identifikation derjenigen Stufen der Wertschöpfungskette, die besonders stark zu Umweltwirkungen beitragen. Die klare Zuordnung – etwa zu den Abschnitten „Rohstoffgewinnung“ oder „Produktion von Vorprodukten“ – verdeutlicht, in welchem Abschnitt ein bestimmter Umweltindikator in besonderem Maße auftritt. Im Fall des Textilsektors zeigt sich beispielsweise, dass der Wasserverbrauch überwiegend bereits während der Rohstoffgewinnung anfällt – ein Hinweis auf mögliche Handlungsfelder.

Globale Verortung auf der Weltkarte

Die zweite Darstellungsform projiziert dieselben Umweltwirkungen auf eine Weltkarte:

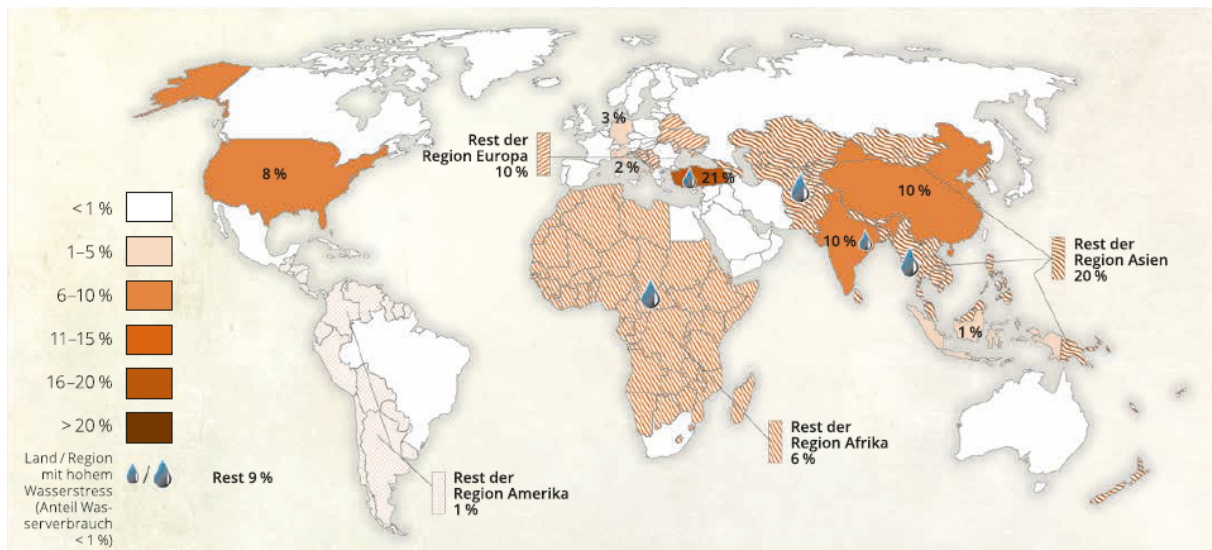


Abbildung 4: Räumliche Verteilung von Umweltwirkungen entlang der Lieferkette am Beispiel des Textilsektors Deutschland (Quelle: Jungmichel [3]).

Die Projektion auf eine Weltkarte (Abbildung 4) veranschaulicht die geografische Dimension der Umweltfolgen globaler Produktion. Anstelle aufwendiger Tabellenanalysen ermöglicht sie eine direkte visuelle Zuordnung des Wasserverbrauchs in der textilen Lieferkette zu konkreten Ländern und Regionen. In diesem Beispiel zeigt die Färbung der Länderoberflächen, in welchen Staaten welcher Anteil des weltweiten Wasserverbrauchs in der Textilproduktion entsteht. Tropfensymbole kennzeichnen Länder mit hohem Wasserstress – also Regionen, in denen die Wassernutzung besonders kritisch ist, weil nur begrenzte Ressourcen zur Verfügung stehen. Dabei fällt auf, dass ein erheblicher Teil des Wasserverbrauchs auf Länder mit geringem Wasservorkommen entfällt, insbesondere in Asien und Afrika. Diese räumliche Entkopplung von Konsum und ökologischen Folgen verdeutlicht, dass Umweltwirkungen im Bezug auf den Faktor Wasser häufig in frühen Stufen der Lieferkette auftreten – oftmals in Regionen, die am Ende nur begrenzt in den Konsum der Endprodukte eingebunden sind.

Mögliche Erweiterungen

Tabellarische und kartografische Darstellungen bilden eine solide Grundlage, um MRIO-Ergebnisse zu analysieren und anschaulich darzustellen. Darüber hinaus lassen sich vielfältige Erweiterungen umsetzen – etwa durch Zeitreihen, Sankey-Diagramme zur Visualisierung von Stoff- und Geldflüssen oder interaktive Dashboards, die eine dynamische Kombination von Indikatoren und Regionen ermöglichen. Die modulare Architektur des entwickelten EXIOBASE-Explorers bietet die Möglichkeit solche Visualisierungstypen künftig problemlos zu integrieren.

3 Ergebnisse

Das entwickelte Tool transformiert die Struktur der öffentlich verfügbaren EXIOBASE-Daten in ein für die weitergehende Analyse optimiertes Format, das gezielt die Leistungsfähigkeit der `numpy`-Bibliothek nutzt. Besonderes Augenmerk lag dabei auf der vollständigen Erhaltung sowie der Erweiterung der Multiindex-Funktionalität. So wurde unter anderem die flexible Anpassung der sektoralen Hierarchien und deren freie Übersetzbarkeit in verschiedene Sprachen ermöglicht – stets orientiert an den spezifischen Anforderungen der Nutzer:innen.

Durch diese datenstrukturelle Optimierung konnte die Ladezeit im Vergleich zur Standardnutzung mit `pymrio` deutlich reduziert werden. Das Tool wird über eine grafische Benutzeroberfläche (GUI) auf Basis von `PyQt5` bedient, die in logisch strukturierte, intuitiv nutzbare Tabs gegliedert ist und so eine benutzerfreundliche Interaktion erlaubt.

Im Vergleich zu bestehenden Lösungen bietet das entwickelte System folgende Vorteile:

- **Anpassbare Sektorenhierarchie und Sprachoptionen:** Nutzer:innen können eigene sektorale Klassifikationen definieren und diese beliebig sprachlich anpassen.
- **Deutliche Effizienzsteigerung:** In unseren Tests verringerte die optimierte Datenstruktur die Ladezeit um etwa 2–3 Minuten pro Ladevorgang.
- **Konsistente Lieferkettenvisualisierung:** Komplexe Lieferketten lassen sich in einheitlicher und klarer Darstellung abbilden.
- **Niedrige Einstiegshürde:** Die GUI ermöglicht eine einfache Bedienung; Systemkenntnisse sind nicht mehr für die gesamte Visualisierung, sondern lediglich für die Installation und Einrichtung nach der Anleitung auf Github erforderlich.
- **Hohe Flexibilität:** Die Matrizen des gewünschten Jahres sowie die Sprache der Benutzeroberfläche und der Metadaten, unter anderem die der Labels, können direkt über die Benutzeroberfläche geändert werden.
- **Modulare Erweiterbarkeit:** Die tab-basierte Architektur der GUI erlaubt eine unkomplizierte Integration zusätzlicher Funktionen.

3.1 Aufbau des EXIOBASE-Explorers

Das Tool ist in zwei zentralen Klassen organisiert: Die Klasse `IOSystem` übernimmt die Transformation der EXIOBASE-Daten in das optimierte, speichereffiziente Format und stellt zugleich die mathematische Grundlage für alle nachgelagerten Berechnungen bereit. Darauf aufbauend analysiert und berechnet die Klasse `SupplyChain` die Lieferketten auf Basis der zuvor aufbereiteten Daten. Gesteuert wird die Anwendung über das Skript `main.py`, welches die grafische Benutzeroberfläche initialisiert und deren Zustand laufend aktualisiert.

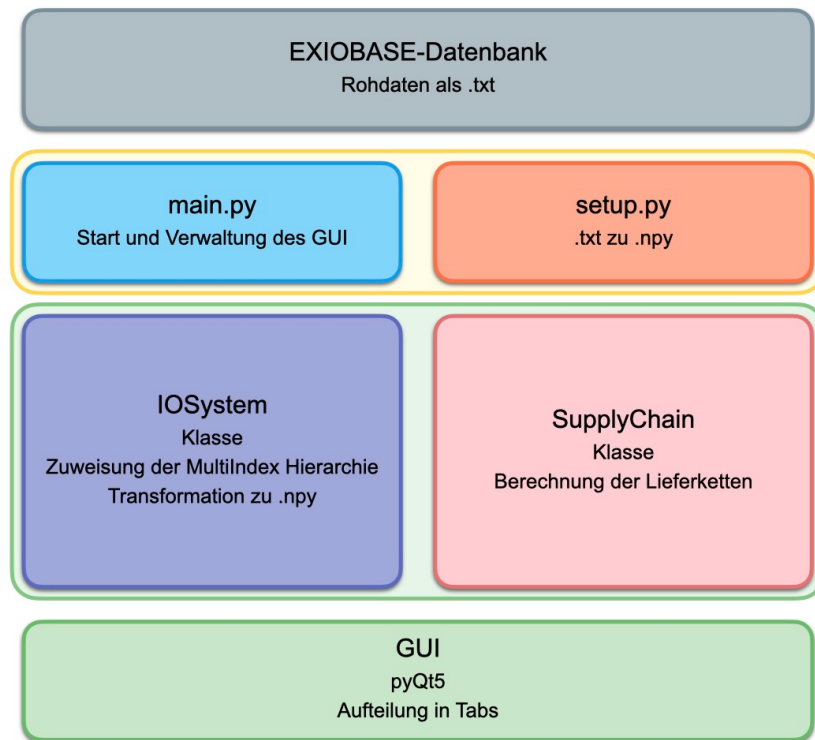


Abbildung 5: Schematische Darstellung des Tools EXIOBASE-Explorer

3.2 Funktion der Klasse IOSystem

3.2.1 Optimierung des Datenformats und Transformation der Ausgangsdaten

Das ursprünglich verwendete Datenformat speichert lediglich die grundlegenden Matrizen in Form von `.txt`-Dateien, wobei sowohl die numerischen Inhalte als auch die zugehörigen Indizes gesondert abgelegt werden. Diese Methode bringt mehrere Nachteile mit sich. Einerseits fehlen relevante Matrizen wie beispielsweise die Leontief-Inverse, sodass diese vor jeder Analyse erneut berechnet werden müssen. Andererseits erfordert das Einlesen der Textdateien eine zusätzliche Trennung von Daten und Indizes, was zu erhöhtem Zeitaufwand führt und die flexible Festlegung eigener Sektorenhierarchien erschwert.

Aus diesen Gründen werden die Daten in ein optimiertes Format überführt. Dabei kommen `.npz`-Dateien zum Einsatz, die zudem direkt für Berechnungen und damit zur Analyse von Lieferketten verwendet werden können. Darüber hinaus werden alle für das Tool relevanten Matrizen einmalig berechnet und in diesem Format dauerhaft gespeichert. Die Sektorhierarchie wird dabei beibehalten und sogar erweitert, indem sie durch Excel-Dateien auf benutzerfreundliche Weise angepasst werden kann.

Die Transformation des Datenformats verläuft in mehreren Schritten. Zuerst liest das System aus mehreren `file_parameters.json`-Dateien innerhalb des unveränderten komprimierten Archivs die benötigten Metadaten aus. Dabei erfasst es insbesondere, wie viele Kopfzeilen und Indexspalten vorliegen, um die Matrixstrukturen korrekt zu interpretieren.

Anschließend wird ein neues Verzeichnis für die sogenannten `fast_load_databases` erstellt, sofern es nicht bereits existiert. In diesem Schritt werden wichtige Matrizen – darunter `A`, `L`, `Y`, `I`, `S`, `D_cba`, `total`, `retail`, `direct_suppliers`, `resource_extraction` und `preliminary_products` – entweder aus den vorhandenen `.txt`-Dateien des ursprünglichen Archivs geladen oder, sofern notwendig, mithilfe der Methoden, die aus `pyMRIIO` übernommen wurden, neu berechnet. Diese Matrizen werden anschließend im `.npy`-Format gespeichert.

Zusätzlich werden begleitende Konfigurationsdateien, die Informationen zu Sektoren, Regionen, Umweltauswirkungen, Einheiten sowie allgemeine Metadaten enthalten (z. B. `sectors.xlsx`, `regions.xlsx`, `impacts.xlsx`, `units.xlsx` und `general.xlsx`), in das neue Datenverzeichnis übernommen. Nach erfolgreicher Umwandlung und Speicherung aller Matrizen und Konfigurationsdateien erfolgt eine Aktualisierung sämtlicher Multi-Indizes innerhalb des Systems. Diese Maßnahme stellt sicher, dass alle Matrizen korrekt beschriftet und für nachgelagerte Analysen unmittelbar einsetzbar sind.

3.2.2 Berechnung der Matrizen zur Analyse sektoraler Umweltauswirkungen

Neben den unmittelbar aus dem Archiv extrahierten Grundmatrizen werden im Rahmen der Analyse weitere, nicht standardmäßig bereitgestellte Matrizen berechnet. Diese dienen der differenzierten Erfassung und Interpretation sektor- und regionsspezifischer Umweltauswirkungen.

Die Input-Output-Koeffizientenmatrix **A** sowie die Endnachfragematrix **Y** werden direkt aus den Quelldateien innerhalb des ZIP-Archivs geladen. Gleiches gilt für die Umweltintensitätsmatrix **S**. Ergänzend dazu wird eine Identitätsmatrix **I** generiert, die für spätere Matrixoperationen erforderlich ist.

Zur Ermittlung der indirekten Effekte wird die Leontief-Inverse (siehe Abschnitt 2.1) **L** über $(\mathbf{I} - \mathbf{A})^{-1}$ berechnet, welche den gesamten Produktionsoutput (direkt und indirekt) angibt, der zur Deckung einer zusätzlichen Einheit Endnachfrage erforderlich ist. Aufgrund des rechenintensiven Inversionsverfahrens erfolgt die Berechnung einmalig vor Beginn der eigentlichen Analyse.

Die Matrix der Gesamtauswirkungen `total` ergibt sich aus dem Produkt $\mathbf{S} \cdot (\mathbf{L} \cdot \mathbf{Y})$. Dabei bezeichnet der Ausdruck $\mathbf{L} \cdot \mathbf{Y}$ die gesamte Produktionsmenge, die durch die Endnachfrage ausgelöst wird. Die resultierenden Umweltwirkungen werden anschließend regional gruppiert und nach Region sowie Wirkungskategorie sortiert, um eine übersichtliche Darstellung zu ermöglichen.

Zur Ermittlung der direkten Umweltauswirkungen der Endnachfrage wird die Matrix `retail` berechnet, indem $\mathbf{S} \cdot \mathbf{Y}$ gebildet wird. Diese Matrix enthält ausschließlich direkte Emissionen, die auf die Endnachfrage zurückzuführen sind, ohne vorgelagerte Lieferketten zu berücksichtigen. Die Zeilenanordnung erfolgt analog zu `total`.

Die Auswirkungen direkter Zulieferer werden in der Matrix `direct_suppliers` abgebildet. Sie wird durch das Produkt $\mathbf{S} \cdot (\mathbf{A} \cdot \mathbf{Y})$ bestimmt. Um direkte Zulieferungen aus dem Rohstoffsektor an dieser Stelle auszuschließen, da sie bereits bei der Rohstoffgewinnung selbst bilanziert werden, werden vor der Berechnung sämtliche Zeilen von **A**, die Rohmaterialien betreffen, auf null gesetzt.

Die Matrix `resource_extraction` dient der Isolierung von Umweltauswirkungen, die auf die Gewinnung von Rohstoffen zurückzuführen sind. Die Berechnung erfolgt mithilfe des Ausdrucks $(\mathbf{L} - \mathbf{I}) \cdot \mathbf{Y}$, wobei nun alle nicht rohstoffrelevanten Zeilen von $(\mathbf{L} - \mathbf{I})$ auf null gesetzt werden.

Das Ergebnis wird wie zuvor nach Region und Wirkungskategorie gegliedert.

Schließlich wird die Matrix `preliminary_products` gebildet, um die Umweltauswirkungen von Vorprodukten zu quantifizieren. Hierfür wird die Formel $(\mathbf{L} - \mathbf{I} - \mathbf{A}) \cdot \mathbf{Y}$ verwendet. Auch in diesem Fall werden alle Zeilen von $(\mathbf{L} - \mathbf{I} - \mathbf{A})$, die Rohmaterialien betreffen, auf null gesetzt, sodass ausschließlich Effekte durch vorgelagerte Zwischenprodukte berücksichtigt werden. Die Darstellung erfolgt in gleicher Weise sortiert und gruppiert wie bei den übrigen Matrizen.

3.2.3 Hierarchische Gruppierung der Daten mittels MultiIndices

Zur strukturierten Verarbeitung umfangreicher sektoraler und regionaler Daten nutzt das entwickelte System hierarchische `MultiIndex`-Strukturen. Diese ermöglichen eine klare und konsistente Abbildung der Beziehungen zwischen ökonomischen Aktivitäten und Umweltauswirkungen über verschiedene Hierarchieebenen der Sektoren hinweg. Dieses Datenmodell wurde gewählt, weil es notwendig ist, große und heterogene Datensätze effizient zu organisieren, auszuwerten und interaktiv analysieren zu können.

Entscheidend ist hierbei die Verwendung einer speziell implementierten `Index`-Klasse, welche über die Methode `read_configs()` relevante Konfigurationsdateien einliest. Diese liegen im `.xlsx`-Format vor und enthalten definitorische Informationen zu Sektoren, Regionen, Wirkungskategorien, Einheiten und weiteren Metadaten. Basierend auf diesen Konfigurationen erzeugt die Methode `create_multiindices()` eine Vielzahl an `pandas.MultiIndex`-Objekten. Dazu zählen unter anderem ein hierarchischer Index für Sektoren-Regionen-Kombinationen (`self.sector_multiindex`), ein separater Index für verschiedene Umweltauswirkungskategorien (`self.impact_multiindex`) sowie kombinierte Indizes wie `self.impact_per_region_multiindex`, die speziell für die Analyse regional differenzierter Auswirkungen vorgesehen sind.

Um die Konsistenz innerhalb des gesamten Systems zu gewährleisten, aktualisiert die Methode `update_multiindices()` sämtliche relevanten Matrizen – darunter die Input-Output-Matrix A , die Leontief-Inverse L , die Endnachfragematrix Y , die Umweltintensitätsmatrix S sowie alle berechneten Wirkungsdaten wie `total`, `retail`, `direct_suppliers`, `resource_extraction` und `preliminary_products`. Durch diese einheitliche Indexierung wird eine kohärente Datenstruktur geschaffen, die sowohl für einfache Filteroperationen als auch für komplexe mehrdimensionale Auswertungen genutzt werden kann.

Die Vorteile dieses Ansatzes liegen in der eindeutigen Zuordenbarkeit einzelner Datenpunkte, der Möglichkeit zur Aggregation oder Dekomposition entlang definierter Hierarchien und der direkten Integration in Analysepipelines. Insgesamt trägt die Nutzung von `MultiIndex`-Strukturen entscheidend zur Transparenz, Reproduzierbarkeit und analytischen Tiefe der Umweltwirkungsanalyse entlang globaler Lieferketten bei.

3.3 Funktion der Klasse SupplyChain

Die Klasse `SupplyChain` bildet das zentrale Element des entwickelten Analyse-Tools zur detaillierten Untersuchung und Visualisierung von Umweltauswirkungen entlang spezifischer Lieferketten. Sie fungiert als Schnittstelle zwischen den aufbereiteten Datensätzen des `IOSystem` und den darauf aufbauenden analytischen sowie visuellen Ausgabefunktionen.

Ihre Hauptaufgabe besteht darin, auf Grundlage benutzerdefinierter Auswahlparameter – etwa in Bezug auf Regionen, Sektoren oder Wirkungskategorien – die relevanten Teilmengen der Daten zu extrahieren, geeignete Wirkungsindikatoren zu berechnen und die Ergebnisse für die weitere Interpretation aufzubereiten. Dabei greift die Klasse sowohl auf vorverarbeitete Matrizen als auch auf dynamisch generierte Ergebnisdaten zurück.

Darüber hinaus stellt **SupplyChain** Funktionen zur Verfügung, mit denen ausgewählte Lieferkettenpfade analysiert und ihre Umweltauswirkungen strukturiert dargestellt werden können. Dies umfasst insbesondere die Möglichkeit, die Ergebnisse in interaktive Grafiken zu überführen, um komplexe Zusammenhänge zwischen vorgelagerten Produktionsprozessen und resultierenden Umweltwirkungen nachvollziehbar zu machen.

3.3.1 Initialisierung und Datenauswahl

Die Initialisierung der **SupplyChain**-Klasse erfolgt durch Übergabe einer referenzierten **IOSystem**-Instanz, welche die Grundlage für sämtliche Analysen bildet. Die Auswahl der zu untersuchenden Lieferkettenteile kann auf zwei verschiedenen Wegen erfolgen: Entweder wird eine Liste von Indizes über den Parameter **indices** bereitgestellt, oder die Auswahl erfolgt hierarchisch anhand von Schlüsselwortargumenten (**kwargs**), die den in der Datenbank definierten Klassifikationen für Regionen und Sektoren entsprechen.

Im letztgenannten Fall unterstützt die Methode **get_multiindex_selection()** die nutzerseitige Auswahl durch eine interaktive grafische Benutzeroberfläche, die auf dem **PyQt5**-Framework basiert. Diese Oberfläche erlaubt eine intuitive Navigation innerhalb der komplexen **MultiIndex**-Strukturen und ermöglicht es, spezifische Kombinationen von Regionen und Sektoren gezielt auszuwählen. Diese flexible Herangehensweise eröffnet sowohl die Analyse auf Basis vordefinierter Klassifikationen als auch die Untersuchung individueller Fragestellungen durch manuelle Selektion bestimmter Indizes.

Die finale Auswahl wird intern als Liste von gültigen Indexkombinationen gespeichert und bildet die Grundlage für sämtliche nachfolgenden Berechnungen und Visualisierungen innerhalb der **SupplyChain**-Instanz.

3.3.2 Berechnung von Umweltauswirkungen entlang von Lieferketten

Die Klasse **SupplyChain** erweitert das **IOSystem** um eine benutzerorientierte Zugriffsebene. Sie führt keine erneuten Matrixoperationen durch, sondern greift auf die vom **IOSystem** vorberechneten Matrizen zurück. Damit liegt der Schwerpunkt auf der gezielten Auswahl und Aggregation relevanter Teilausschnitte.

Die Selektion erfolgt über Attribute wie **self.indices** und **self.hierarchy_levels**, die die vom Nutzer gewählten Regionen und Sektoren abbilden. So können aus voll dimensionierten Matrizen wie der Leontief-Inversen **L** oder der Gesamtauswirkungsmatrix **total** exakt die Einträge extrahiert werden, die für die Analyse benötigt werden.

Die Methode **total(impact)** illustriert dieses Vorgehen. Sie verwendet nicht erneut den Ausdruck $\mathbf{S} \cdot (\mathbf{L} \cdot \mathbf{Y})$, sondern greift auf die bereits berechnete Matrix **total** zurück und filtert diese über die gespeicherten Indizes. Auf diese Weise entsteht eine Submatrix aus ausgewählten

Zeilen, die die direkten und indirekten Umweltauswirkungen im jeweiligen Lieferkettenschritt darstellt.

Das Verfahren ist für sämtliche Lieferkettenschritte einheitlich: Jede Methode greift auf die entsprechende, im `IOSystem` hinterlegte Matrix (z. B. `total`, `direct_suppliers`, `preliminary_products`) zu, filtert anhand der Auswahlattribute und schließt rohstoffbezogene Zeilen dort aus, wo nur Zwischenprodukte oder nachgelagerte Effekte betrachtet werden.

Die Methode `calculate_all()` führt alle beschriebenen Extraktionen und Filterungen automatisiert für eine definierte Liste von Umweltauswirkungen durch und liefert die Ergebnisse wahlweise in absoluten Werten oder als relative Anteile am Gesamteinfluss. Anschließend konvertiert `transform_unit()` die Einheiten in die vom Nutzer gewählten Standards und rundet die Werte gemäß den Vorgaben in `units.xlsx`.

3.3.3 Visualisierung der Ergebnisse

Zur anschaulichen Darstellung der berechneten Umweltauswirkungen bietet die Klasse `SupplyChain` Möglichkeiten zur Visualisierung, die sowohl die sektorale als auch die räumliche Dimension der Analyse abdecken.

Die Methode `plot_supplychain_diagram()` (siehe Abbildung 6) dient der grafischen Aufbereitung der Auswirkungen entlang der verschiedenen Stufen der Lieferkette – von der Rohstoffgewinnung über Vorprodukte und direkte Zulieferer bis hin zum Einzelhandel. Für die jeweils ausgewählten Umweltindikatoren wird eine tabellarische Darstellung erzeugt, in der jede Zelle durch eine Kreisfläche repräsentiert wird. Die Größe dieser Kreisflächen ist proportional zum Anteil der jeweiligen Stufe am gesamten Umwelteinfluss. Dadurch lassen sich Unterschiede zwischen Wirkungskategorien sowie zwischen Lieferkettenabschnitten unmittelbar visuell erfassen.

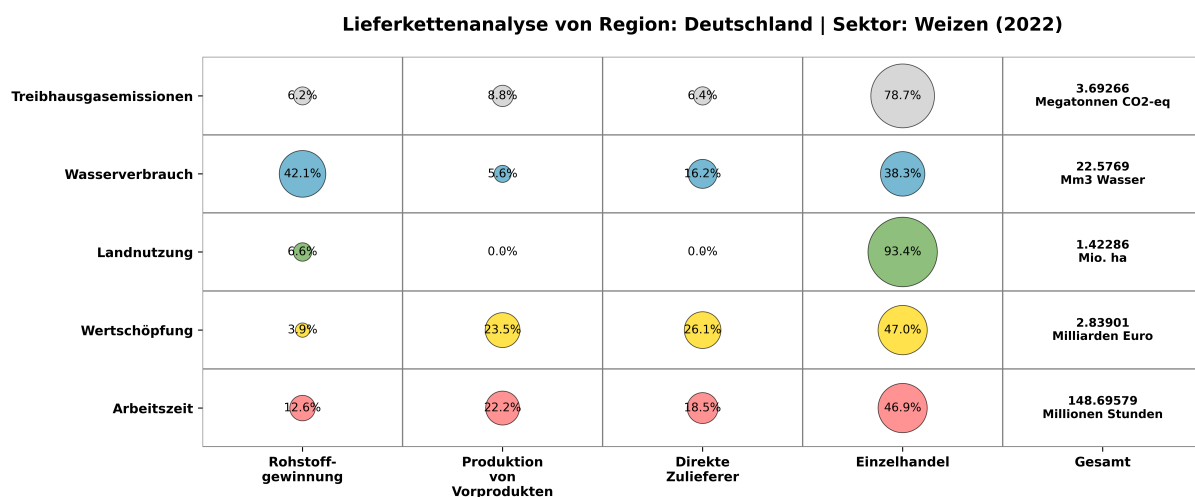


Abbildung 6: Ausgabe der Funktion `plot_supplychain_diagram()`

Für die räumliche Darstellung stehen die Methoden `plot_worldmap_by_subcontractors()` (siehe Abbildung 7) und `plot_worldmap_by_impact()` zur Verfügung. Beide erzeugen thematische Weltkarten in Form von Choroplethen, die entweder die geographische Verteilung der Umweltauswirkungen oder die Bedeutung einzelner Subunternehmerregionen visualisieren. Die

Darstellungen können sowohl absolute Werte als auch relative Beiträge abbilden. Beim Überfahren mit dem Mauszeiger werden jeweils der Name der Region sowie ihr Beitrag zum gewählten Impact im Tool angezeigt.

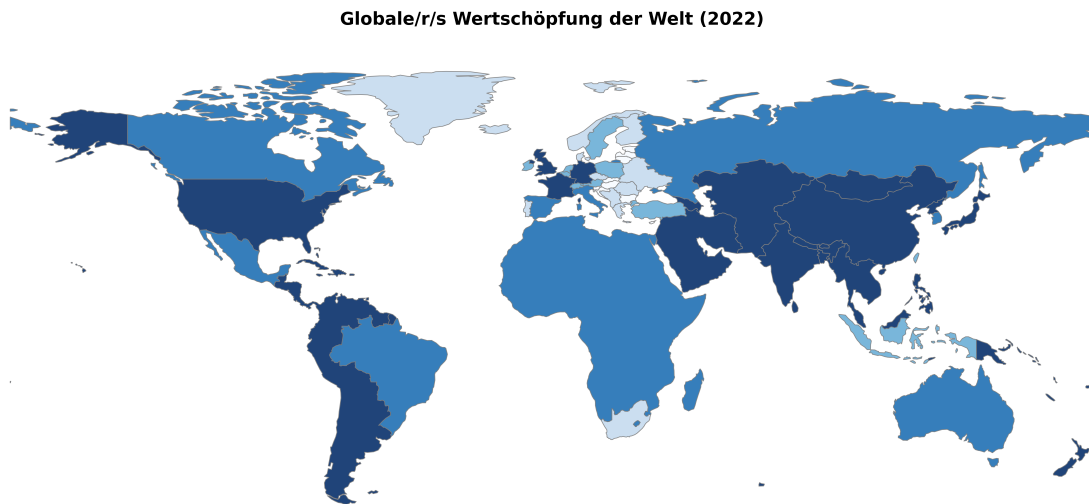


Abbildung 7: Ausgabe der Funktion `plot_worldmap_by_subcontractors()`

3.4 Benutzeroberfläche (GUI)

Die grafische Benutzeroberfläche des Analyse-Tools wurde mit dem Ziel entwickelt, eine intuitive, modulare und zugleich leistungsfähige Interaktion mit den komplexen EXIOBASE-Daten sowie den darauf aufbauenden Analysefunktionen zu ermöglichen. Sie basiert auf dem `PyQt5`-Framework und orientiert sich an einer klar strukturierten Tab-Architektur, die die einzelnen Funktionalitäten logisch voneinander trennt.

Jeder Tab innerhalb der Anwendung deckt einen spezifischen Aspekt der Datenverarbeitung oder Analyse ab – von der allgemeinen Konfiguration über die Auswahl relevanter Regionen, Sektoren und Umweltauswirkungen bis hin zur Visualisierung der Ergebnisse. Die Oberfläche erlaubt es Nutzer:innen, sowohl standardisierte als auch individuelle Analysen durchzuführen, indem sie flexibel zwischen verschiedenen Auswahl- und Darstellungsebenen navigieren.

Durch die Trennung in klar definierte Funktionsbereiche und die Nutzung interaktiver Steuerelemente (z. B. Dropdown-Menüs, Kontrollkästchen und Auswahlfenster) soll die Nutzerfreundlichkeit gewährleistet werden. Gleichzeitig kann die Oberfläche durch die modulare Struktur leicht erweitert und an neue Anforderungen oder Datenquellen angepasst werden.

3.4.1 Architektur und Navigation

Die grafische Anwendung wird über die `UserInterface`-Klasse im Hauptskript `main.py` gesteuert. Diese Klasse initialisiert das zentrale Anwendungsfenster, das auf `QMainWindow` basiert, und integriert ein `QTabWidget`, über das die einzelnen Funktionsbereiche der Software als separate Tabs organisiert werden.

Zu den implementierten Tabs zählen der **SelectionTab** zur Auswahl relevanter Regionen und Sektoren, der **VisualisationTab** für die grafische Darstellung der Analyseergebnisse, der **SettingsTab** zur Konfiguration anwendungsspezifischer Parameter sowie der **ConsoleTab**, der eine direkte Interaktion mit dem Python-Backend ermöglicht und insbesondere für fortgeschrittene Nutzer:innen von Interesse ist.

Die zentralen Daten werden über Instanzen der Klassen **IOSystem** und **SupplyChain** verwaltet, die in der **UserInterface**-Klasse zentral instanziiert werden. Diese Architektur gewährleistet, dass alle Tabs auf eine einheitliche, konsistente Datenquelle zugreifen und somit eine synchronisierte Bearbeitung und Visualisierung möglich ist.

Funktionen wie `reload_tabs()` und `update_supplychain()` stellen sicher, dass nach jeder relevanten Benutzeraktion – etwa einer Jahresauswahl oder einer neuen Kombination von Parametern – alle Anzeigeelemente automatisch aktualisiert werden. Änderungen an den Daten werden zentral verarbeitet und an alle abhängigen UI-Komponenten propagiert, wodurch Inkonsistenzen vermieden werden.

3.4.2 Datenauswahl und -filterung (SelectionTab)

Der **SelectionTab** dient der benutzerfreundlichen Auswahl spezifischer Sektoren und Regionen aus dem EXIOBASE-Datensatz und stellt damit den Einstiegspunkt für die datenbasierte Analyse dar. Die Interaktion erfolgt primär über hierarchisch strukturierte Baumansichten, die mit dem Qt-Element **QTreeWidget** realisiert wurden. Diese visualisieren die zugrundeliegenden **MultiIndex**-Strukturen in einer intuitiv navigierbaren Form.

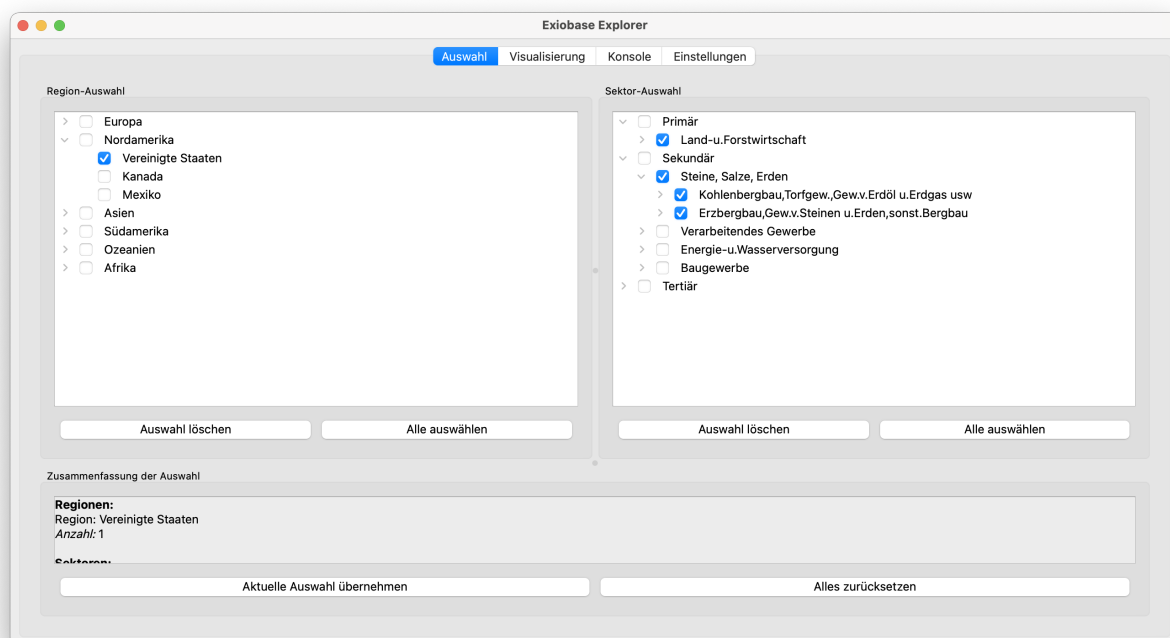


Abbildung 8: Der Selection-Tab des EXIOBASE Explorers

Um die komplexen **MultiIndex**-Daten in den Baumstrukturen darstellen zu können, wird

die Hilfsfunktion `multiindex_to_nested_dict()` verwendet, welche die hierarchischen Indizes in verschachtelte Python-dict-Objekte überführt. Dadurch kann die sektorale und regionale Struktur der Daten in einer für Nutzer:innen verständlichen Weise visualisiert werden.

Zudem bietet der Tab eine `reset_selection()`-Funktion, mit der die gesamte Auswahl mit einem Klick zurückgesetzt werden kann. Dadurch wird eine schnelle Neuausrichtung der Analyse ohne manuelles Entfernen einzelner Auswahlpunkte ermöglicht.

3.4.3 Anwendungseinstellungen (SettingsTab)

Der **SettingsTab** stellt die Steuereinheit für globale Anwendungsconfigurationen bereit und ermöglicht eine flexible Anpassung des Verhaltens der Software an die Bedürfnisse der Nutzer:innen. Über eine Dropdown-Auswahl (`QComboBox`) kann die Sprache der gesamten Benutzeroberfläche gewechselt werden. Die entsprechenden UI-Texte werden dabei aus einem Wörterbuch im `IOSystem` (genannt `general_dict`) geladen, um eine mehrsprachige Nutzung zu gewährleisten.

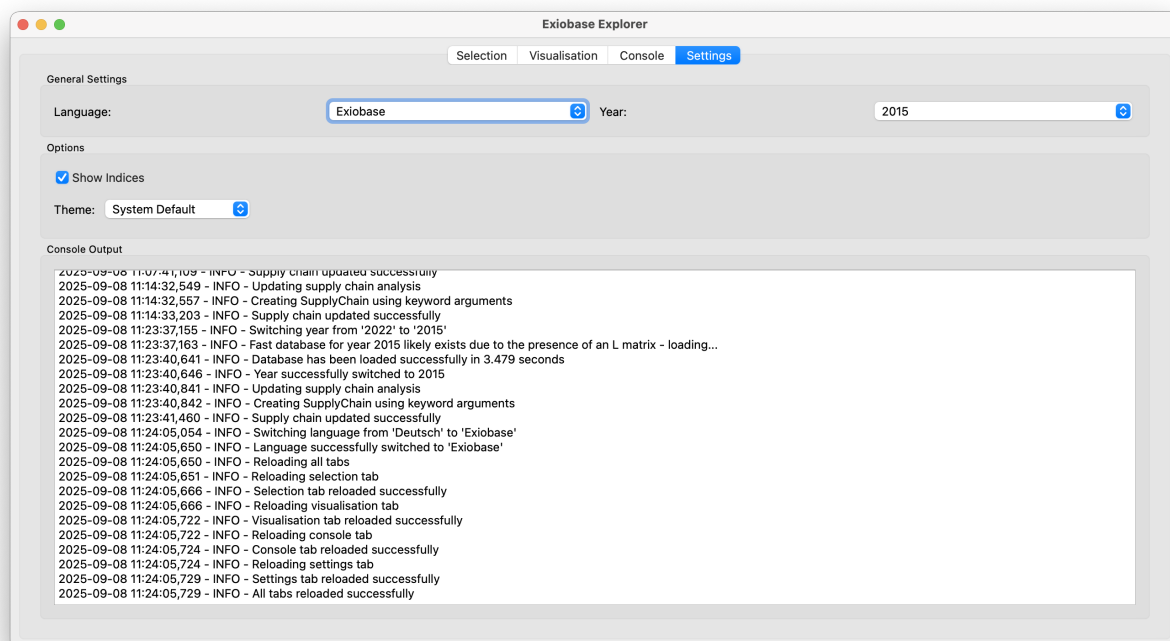


Abbildung 9: Der Settings-Tab des EXIOBASE Explorers

Ein weiteres Steuerelement ist die Auswahl des Analysejahres, die ebenfalls über ein Dropdown-Menü getroffen werden kann. Eine Änderung dieser Auswahl veranlasst das `IOSystem`, die relevanten Datensätze neu zu laden und das gesamte Analysesystem entsprechend zu aktualisieren.

Außerdem kann mit einer Checkbox zwischen hellem und dunklem Erscheinungsbild (Darkmode) gewechselt werden.

Um Monitoring und Fehlerdiagnose zu unterstützen, zeigt der **SettingsTab** Systemmeldungen und Statusinformationen in Echtzeit in einem integrierten Log-Bereich an. Dieser basiert auf einer `QTextEditLogger`-Komponente und erleichtert die Nachverfolgung laufender Prozesse direkt innerhalb der Benutzeroberfläche.

Schließlich bietet der Tab eine spezielle Option zur Anzeige der detaillierten Indexinformationen der im **SelectionTab** getroffenen Auswahl als Liste der Einzelindizes. Diese Funktion ermöglicht es, die MultiIndex-Strukturen in der Zusammenfassung der Auswahl bei Bedarf ein- oder auszublenden und trägt damit zur Übersichtlichkeit bei.

3.4.4 Ergebnisvisualisierung (VisualisationTab)

Der **VisualisationTab** übernimmt die Darstellung der Analyseergebnisse und ist in ein **QTabWidget** eingebettet, das die gleichzeitige Öffnung und Verwaltung mehrerer Visualisierungen in separaten Unter-Tabs ermöglicht. Neue Visualisierungs-Tabs können über die Methode `add_map_tab()` erzeugt werden.

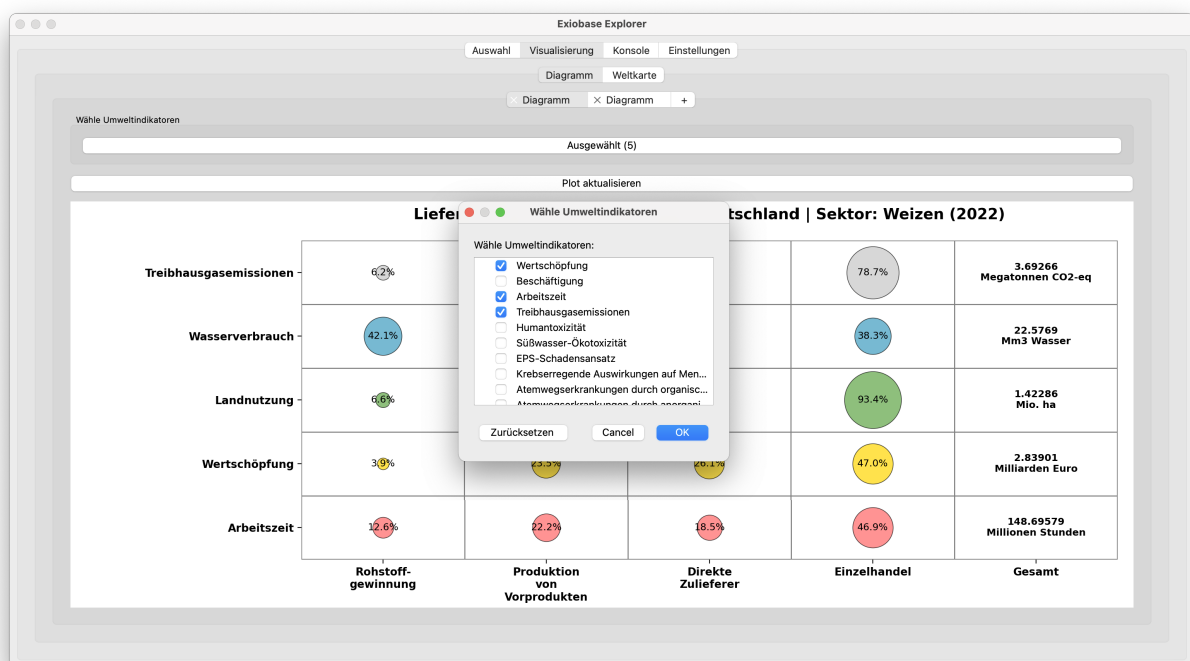


Abbildung 10: Die Visualisierung der Lieferkette nach Wertschöpfungsstufen

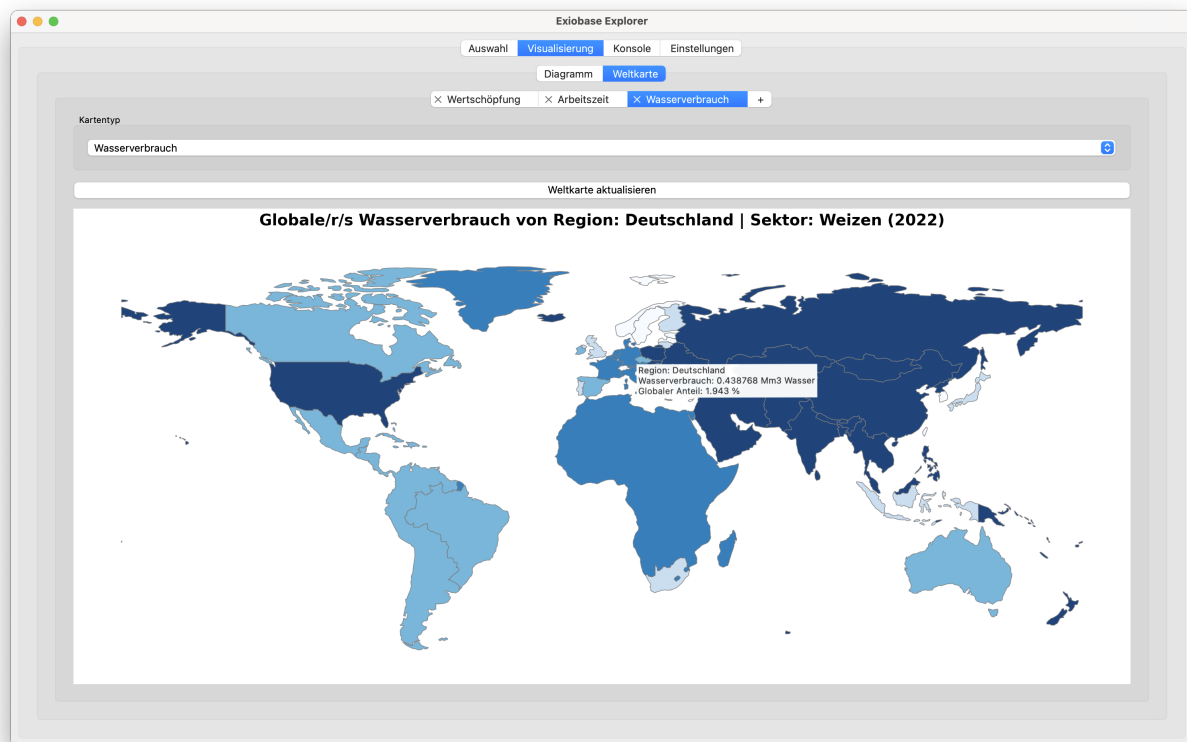


Abbildung 11: Die Visualisierung der Lieferkette auf einer Weltkarte

Die grafische Umsetzung der Diagramme (siehe Abbildung 10) und Karten (siehe Abbildung 11) erfolgt mit der Bibliothek `matplotlib`, deren Darstellungen mittels `FigureCanvasQTAgg` in die PyQt5-Oberfläche integriert werden. Dadurch können Choroplethenkarten zur räumlichen Verteilung der Umweltauswirkungen ebenso realisiert werden wie Balkendiagramme, die eine differenzierte Aufschlüsselung der Effekte entlang der verschiedenen Stufen der Lieferkette erlauben.

3.4.5 Interaktive Konsole (`ConsoleTab`)

Der `ConsoleTab` (siehe Abbildung 12) bietet eine interaktive Python-Konsole innerhalb der Anwendung. Er basiert auf `code.InteractiveConsole` aus der Standardbibliothek. Nutzer:innen mit Erfahrung können hier beliebige Python-Befehle ausführen. So greifen sie direkt auf die geladenen Datenstrukturen zu und arbeiten mit Instanzen der Kernklassen wie `self.IOSystem` und `self.SupplyChain`. Auch deren Attribute und Methoden stehen zur Verfügung.

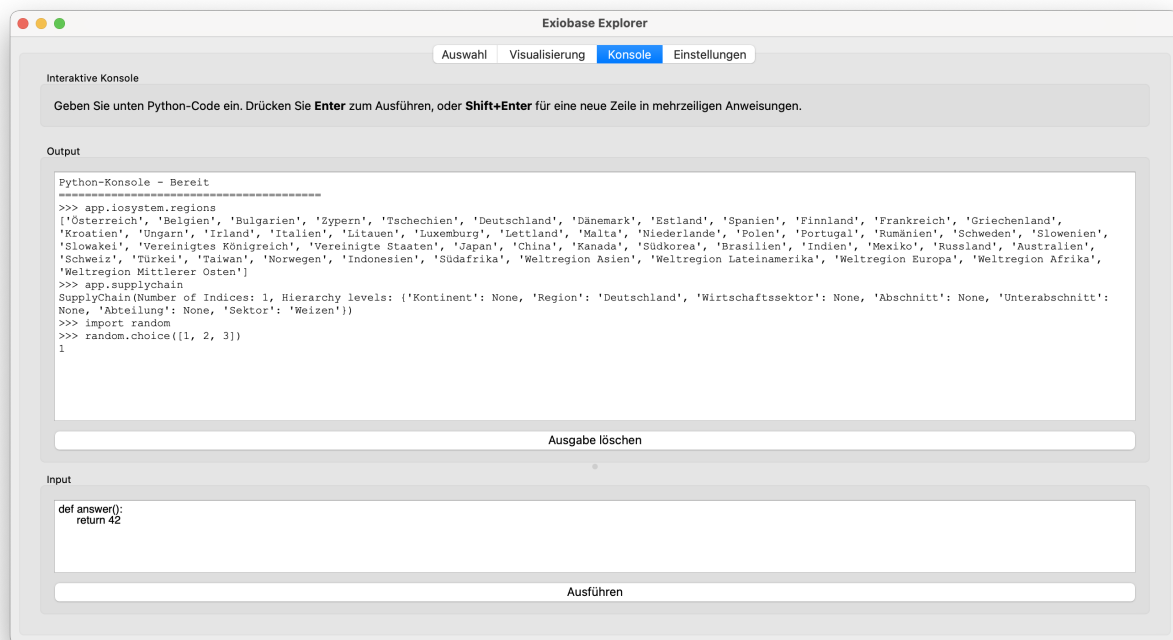


Abbildung 12: Der Console-Tab des EXIOBASE Explorers

Zur Integration der Konsolenausgaben in die grafische Benutzeroberfläche wird die Standardausgabe (`stdout`) sowie die Fehlerausgabe (`stderr`) intern mittels einer `StreamCatcher`-Klasse umgeleitet. Dadurch erscheinen alle Ausgaben und Fehlermeldungen unmittelbar in einem dedizierten Textfeld des `ConsoleTab`.

4 Diskussion

Die vorangegangenen Kapitel haben Zielsetzung, Konzeption und technische Umsetzung des EXIOBASE-Explorers erläutert. In diesem Abschnitt wird nun untersucht, inwiefern das entwickelte Tool die selbst definierten Anforderungen erfüllt und welchen Beitrag es zur Analyse globaler Umweltwirkungen leisten kann.

4.1 Einordnung der Ergebnisse

Die in Kapitel 3 dargestellten Anwendungsmöglichkeiten – exemplarisch dargestellt in Abbildung 13 für den Textilsektor Deutschlands im Jahr 2022 – veranschaulichen die Funktionsfähigkeit und Flexibilität des EXIOBASE-Explorers. So lassen sich mithilfe des Programms für beliebige Kombinationen von Region, Sektor und Umweltindikator detaillierte Analysen durchführen.

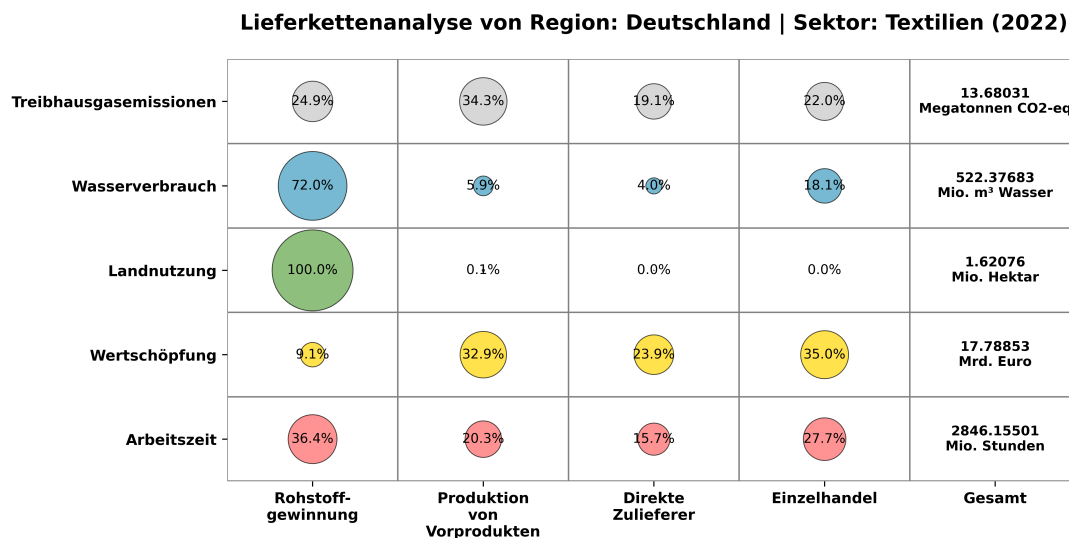


Abbildung 13: Lieferkettenanalyse für Deutschland | Textilien – Verteilung von Treibhausgasemissionen, Wasserverbrauch, Landnutzung, Wertschöpfung und Arbeitszeit nach Wertschöpfungsstufen.

Abbildung 13 zeigt, dass anhand der beiden grundlegenden Darstellungsformen – der tabellari-schen Aufschlüsselung nach Wertschöpfungsstufen und der Weltkarte (siehe Unterabschnitt 2.4) – wichtige Erkenntnisse gewonnen werden können. So lassen sich beispielsweise Diskrepanzen zwischen Arbeitsaufwand und wirtschaftlichem Ertrag sichtbar machen: Im Textilsektor entfällt 36 % der Arbeitszeit auf die Rohstoffgewinnung, während dieser Abschnitt lediglich 9 % der gesamten Wertschöpfung erzeugt. Solche Befunde eröffnen die Möglichkeit, strukturelle Unterschiede und Ungleichgewichte innerhalb globaler Wertschöpfungsketten näher zu untersuchen.

4.2 Anwendungsbereiche der Software

Aufgrund seiner flexiblen Struktur eröffnet der EXIOBASE-Explorer vielfältige Einsatzmöglichkeiten:

- **Wissenschaftliche Forschung:** Ökonomische und ökologische Analysen auf Sektor- und

Regionsebene ermöglichen Untersuchungen zu globalen Wertschöpfungsketten, dynamischen Systemzusammenhängen und Fragen der Nachhaltigkeitsbewertung.“

- **Unternehmensstrategie und Lieferkettenmanagement:** Unternehmen können ökologische Hotspots innerhalb ihrer Beschaffungsnetzwerke identifizieren und gezielt Maßnahmen zur Emissions- und Ressourceneinsparung entlang der gesamten Lieferkette entwickeln.
- **Bildungseinrichtungen und Öffentlichkeitsarbeit:** Lehrende und NGOs können die Darstellungsmöglichkeiten nutzen, um komplexe Zusammenhänge anschaulich zu vermitteln und das Bewusstsein für globale Umwelt- und Sozialfragen zu schärfen.

4.3 Grenzen und methodische Einschränkungen

Trotz der beschriebenen Stärken sind einige Einschränkungen zu berücksichtigen, die die Aussagekraft der Analysen begrenzen:

- **Datenqualität und Aggregationsgrad:** EXIOBASE bietet eine hohe sektorale und regionale Auflösung (200 Sektoren, 49 Regionen). Dennoch können durch die Aggregation auf Sektorebene Verzerrungen entstehen. Regionale Besonderheiten oder untergeordnete Cluster bleiben teilweise unberücksichtigt.
- **Input-Output-Modellannahmen:** Das Leontief-Modell geht von fixen technischen Koeffizienten und streng proportionalen Input-Output-Beziehungen aus. Substitutionseffekte – also die Möglichkeit, einen Produktionsfaktor durch einen anderen zu ersetzen – werden ebenso wenig berücksichtigt wie Kapazitätsgrenzen oder Mengenreaktionen auf Preisänderungen.
- **Umwelt-Extension-Unsicherheiten:** Die Umweltwirkungsfaktoren (**F**) basieren auf Sekundärliteratur und statistischen Durchschnittswerten, die für jeden Sektor der betrachteten Regionen von EXIOBASE erhoben wurden. Unterschiede in Produktionsprozessen einzelner Unternehmen gehen in den globalen Koeffizienten nicht ein.
- **Rechenzeit und Speicherbedarf:** Trotz optimierter Ladevorgänge beansprucht die Verarbeitung großer Matrizen (9800×9800) immer noch mehrere Sekunden pro Abfrage. Interaktive Analysen sind prinzipiell möglich, erfordern jedoch leistungsfähige Hardware.
- **Statische Jahresdaten:** Die EXIOBASE-Versionen liegen für einzelne Jahre vor. Schwankungen innerhalb eines Jahres und saisonale Effekte werden nicht erfasst. Zeitreihenanalysen stützen sich somit auf diskrete Datenpunkte und bilden kurzfristige Entwicklungen nicht ab.

Diese aktuell noch bestehenden methodischen und technischen Einschränkungen sind nicht unüberwindbar, stellen derzeit aber noch wichtige Aspekte bei Interpretation der Ergebnisse dar. Eine kritische Auseinandersetzung mit den Daten- und Modellannahmen ist daher unabdingbar, um aussagekräftige Schlussfolgerungen abzuleiten.

4.4 Ausblick

Mit den bislang implementierten Auswertungsmethoden - der tabellarischen Verteilung nach Wertschöpfungsstufen (Abbildung 3) und der räumlichen Projektion auf einer Weltkarte (Abbildung 4) - wurde nur ein Teil der Analysepotenziale ausgeschöpft, die durch die zugrunde liegenden Klassen `IOSystem` und `SupplyChain` bereitgestellt werden.

Die modulare Architektur der Benutzeroberfläche – insbesondere die flexibel gestalteten Tabs – erleichtert die Integration zusätzlicher Darstellungsformen. Auch in der `SupplyChain`-Klasse lassen sich neue Analysemethoden ergänzen, ohne bestehende Komponenten anpassen zu müssen. Mögliche Erweiterungen umfassen beispielsweise Sankey-Diagramme zur Visualisierung von Material- und Geldflüssen, Zeitreihenanalysen auf Basis mehrerer EXIOBASE-Versionen (1995 bis 2022) oder interaktive Netzwerkgrafiken, die komplexe Lieferverflechtungen abbilden.

Dank der offen angelegten Struktur können je nach verfügbaren Ressourcen und analytischen Anforderungen ohne großen Aufwand weitere Funktionen hinzugefügt werden. Der bestehende Code bietet dafür eine skalierbare Plattform, die sich flexibel erweitern und auf neue Fragestellungen zuschneiden lässt.

5 Danksagung

Wir bedanken uns herzlich bei Prof. Dr. Hendrik Lambrecht, der als Kooperationspartner am Institut für Industrial Ecology (INEC) der Hochschule Pforzheim das Projekt angeboten und mit wertvollen Anregungen und Kritik unterstützt hat.

Unser besonderer Dank gilt auch Anke Richert, die uns im Rahmen des Hector-Seminars betreut und insbesondere bei der Dokumentation und Präsentation des Projekts maßgeblich unterstützt hat. Ebenso danken wir Paul Bischof für seine inhaltliche Korrektur der Dokumentation.

Ermöglicht wurde dieses Vorhaben durch die großzügige Förderung von Herrn Dr. h. c. Hans-Werner Hector, Frau Josefine Hector und der Hector-Stiftungen.

Literaturverzeichnis

- [1] EXIOBASE-Konsortium. *EXIOBASE (Versionen 1–3, 1995–2022)*. Datenbank. Zugänglich unter <https://www.exibase.eu/>. 2022.
- [2] Charles R. Harris u. a. *Array programming with NumPy*. 2020. DOI: 10.1038/s41586-020-2649-2. URL: <https://numpy.org/> (besucht am 06.07.2025).
- [3] Norbert Jungmichel, Christina Schampel und Daniel Weiss. *Umweltatlas Lieferketten – Umweltwirkungen und Hot-Spots in der Lieferkette*. Mit Beiträgen von Moritz Nill. Berlin / Hamburg: adelphi / Systain Consulting GmbH, 2017.
- [4] Andreas Kuhn. *Input-Output-Rechnung im Überblick*. Artikelnummer: 5815116099004. Wiesbaden: Statistisches Bundesamt Deutschland, Apr. 2010. URL: <https://www.destatis.de/DE/Publikationen/Thematisch/VolkswirtschaftlicheGesamtrechnungen/InputOutputRechnung/InputOutputRechnung.pdf>.
- [5] Riverbank Computing Limited. *PyQt5 – Python bindings for the Qt application framework*. Python library. 2023. URL: <https://www.riverbankcomputing.com/software/pyqt/> (besucht am 06.07.2025).
- [6] Wes McKinney. *pandas: powerful Python data analysis toolkit*. Python library. 2022. URL: <https://pandas.pydata.org/> (besucht am 06.07.2025).
- [7] Konstantin Stadler. *pymrio – Python library for EXIOBASE and IO analysis*. Python library. 2021. URL: <https://github.com/konstantinstadler/pymrio> (besucht am 06.07.2025).

Anhang

Der Quellcode des EXIOBASE-Explorers ist öffentlich zugänglich über das folgende GitHub-Repository:

https://github.com/JonasLmbt/exiobase_explorer

Das Repository enthält den EXIOBASE-Explorer (`main.py`) als zentrales Werkzeug zur interaktiven Analyse von Umweltdaten. Die grafische Benutzeroberfläche ermöglicht eine intuitive Bedienung, auch ohne tiefere Programmierkenntnisse. Für weitergehende oder individuelle Auswertungen stehen zusätzlich die beiden zentralen Module `IOSystem` und `SupplyChain` zur Verfügung, mit denen sich EXIOBASE-Datenbanken direkt laden, strukturieren und manuell analysieren lassen. Eine begleitende Anleitung erläutert die Installation sowie die eigenständige Nutzung des Tools.

Selbstständigkeitserklärung

Hiermit versichern wir, dass wir die vorliegende Dokumentation selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet haben. Alle wörtlich oder sinngemäß übernommenen Stellen sind als solche kenntlich gemacht.

Karlsruhe, den 27. September 2025

Lukas Harzbecker

Jonas Lambrecht