# Kalkulationsprogram für Oxygen Not Included

Lastenheft nach DIN 69901-5

### Einführung

Das Spiel Oxygen Not Included (ONI) besitzt mehrere Mechaniken, z.B. Gebäude, aus denen bestimmte Vorgänge hervorgehen. Ein einfacher Vorgang wäre, wie viel kg an Kohle ein Kohle-Generator pro Cycle (in-game Tag = 600 Sekunden) verbraucht, um eine Betriebszeit von 100% aufzuweisen (mit einer Leistung von 600W).

Ein Generator verbraucht 1 kg/s an Kohle bei einer Leistung von 600 W. Die Kalkulation wäre dann:

1 kg/s (Kohle) \* 600 s (ein Tag) = 600 kg (Kohle)

Da ein Generator zusätzlich 20 g/s an Kohlenstoffdioxid erzeugt ist die Rechnung für die Zusatzprodukte wie folgt:

20 g/s (Kohlenstoffdioxid) \* 600 s (ein Tag) = 1200 g (Kohlenstoffdioxid)

Zuletzt werden alle Ergebnisse als Nettosumme angegeben:

Netto (pro Tag):

-600 kg Kohle

+1,2 kg Kohlenstoffdioxid

+600 W

### Beschreibung des Ist-Zustands

Bestimmte Werte die ein Spieler genau wissen möchte, müssen per Hand ausgerechnet werden oder bestehende Software genutzt werden: onicalc.com

### Beschreibung des Soll-Konzepts

Ein Benutzer soll in der Lage sein, mehrere Gebäude/Lebewesen/Ressourcen und die Art des Endergebnis (wie in der Einführung pro Tag oder pro Sekunde) anzugeben. Ergebnisse sollen übersichtlich und detailliert ausgegeben werden.

Zukünftige Features:

* Interaktives Wiki (sehr detailliert; erklärte Mechaniken und deren Produkte)
* zusätzliche Kalkulationen von:
  + Aggregatzustände von Elemente und deren Übergänge

*Zukünftige Features:*

* *Kalkulation von:*
  + Ketten von Elementveränderungen (Manche Elemente verändern ihre Art bei Phasenübergängen e.g. bei dem Erreichen des Siedepunktes von Rohöl entsteht kein gasförmiges Rohöl, sondern flüssiges Petroleum; dies ist keine Verbrennung da Sauerstoff hier unbeteiligt ist (eher zutreffend bei dem Übergang von Holz/ Kohle zu Kohlenstoffdioxid). Dies kann zu Ketten führen bei der ein Element durch mehrere “Veränderungen” geht und zu einem gewollten Produkt wird)
  + Wärmeleitung (Berechnung von Kühlvorrichtungen: Wieviel Kühlung brauche ich damit meine Dampfturbine nicht überhitzt;  
    Berechnung von Wärmekapazität: Wie groß ist die Menge, die meine Vorrichtung an Wärmeenergie pro Temperatur aufnimmt)

### Beschreibung von Schnittstellen

Mögliche Schnittstellen mit den Spieldateien, um Informationen direkt zu erhalten.  
Mögliche Schnittstellen mit dem Online- Wiki, um Informationen bereitzustellen.

### Funktionale Anforderungen

Ein graphisches Interface in einer App für ein Windows-System.

### Nichtfunktionale Anforderungen

#### Zuverlässigkeit

Die Software sollte durch Fehleingaben nicht zum Absturz gebracht werden können.

#### Effizienz

Gegenwärtig keine besonderen Vorgaben.  
Zukünftig effiziente Berechnungen für anspruchsvolle Einsätze.

#### Änderbarkeit

Die Software sollte gut dokumentiert sein, sodass auch andere Teams Änderungen vornehmen können oder die Teile wiederverwenden können.

#### Übertragbarkeit

Keine besonderen Vorgaben.

#### Wartbarkeit

Die Wartbarkeit soll durch gut strukturierten und Kommentierten Code, sowie angemessene

Dokumentation gegeben sein. Tests für die Software sollen umfangreich sein.

### Risikoakzeptanz

Keine besonderen Vorgaben.

### Skizze des Entwicklungszyklus und der Systemarchitektur

Die Software soll nach dem Scrum Verfahren entwickelt werden. Features sollen zur Erweiterung der Software dienen.

### Lieferumfang

Entweder geschrieben in Python oder C# und Projektdokumentation.

### Abnahmekriterien

Vollständiges Projekt inklusive Tests. Ausreichend ist ein implementiertes Feature.