

Behandlung von Klimadaten in IBK Simulationssoftware

Andreas Nicolai

7. Dezember 2016

Zusammenfassung

Dieser Artikel beschreibt Dateiformate und Regeln für die Interpretation von Klimadaten in den IBK Simulationsprogrammen THERAKLES, NANDRAD und DELPHIN, welche intern die Funktionalität der CCM-Bibliothek (Climate Calculation Module) verwenden. Es wird die Verwendung zyklischer jährlicher Daten, aber auch die Behandlung nicht-zyklischer Daten, z.B. aus Messreihen beschrieben.

Inhaltsverzeichnis

1 Grundlagen und Berechnungsregeln	1
1.1 Lineare Interpolation	1
1.2 Absolute Zeitpunkte und relative Simulationszeit	2
1.3 Zyklische Daten für mehrjährige Simulationen	3
1.4 Kontinuierliche Datenreihen	4
2 Klimadatenquellen	4
2.1 CCD Format	5
2.1.1 Verwendung von CCD Dateien für zyklische Berechnungen	6
2.1.2 Komprimiertes Format	7
2.2 CLI Format	7
2.3 EPW-Dateien	7
2.4 C6B-Dateien	7
2.5 CSV-Dateien	8
2.6 D6O Dateien	9

1 Grundlagen und Berechnungsregeln

1.1 Lineare Interpolation

IBK Simulationsprogramme rechnen mit variablen Zeitschritten und müssen daher zu verschiedenen Zeitpunkten auch unterhalb von Stunden Klimadaten

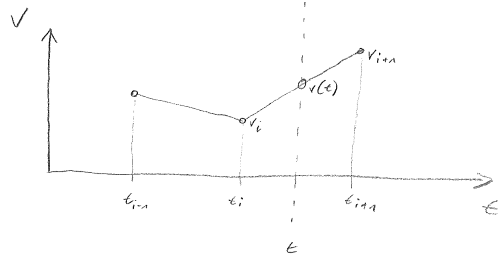


Abbildung 1: Lineare Interpolation des Wertes $v(t)$ aus einer Datentabelle

berechnen. Dafür werden Klimadaten als Zeitpunkt-Werte-Paare definiert und zwischen diesen linear interpoliert (Abbildung 1).

Es findet die folgende Berechnungsvorschrift Anwendung:

$$\alpha = \frac{t - t_i}{t_{i+1} - t_i}$$

$$v(t) = (1 - \alpha) v_i + \alpha v_{i+1}$$

1.2 Absolute Zeitpunkte und relative Simulationszeit

Die *Simulationszeit* beginnt bei Simulationsstart immer bei 0. Bei einer fortgesetzten Simulation (restart) wird auch die Simulationszeit fortgesetzt. Sie ist als *relative Zeit zum Beginn der Simulation* definiert.

Der Bezug zur *absoluten Zeit* bzw. lokalen Zeit wird durch Addition der Simulationszeit auf den in lokaler Zeit angegebenen *Startzeitpunkt* herstellt. Bei Berechnungen mit zyklischem Klima ist das meist Mitternacht am 1. Januar. Dies ist aber nicht verpflichtend so. Absolute Zeitpunkte werden durch zwei Variablen ausgedrückt, dem Jahr (Integerzahl) und dem Zeitabstand seit Beginn des Jahres (time within year), welcher als Abstand zu Mitternacht am 1. Januar des *Startjahres* definiert ist.

```
// Beispielhafte Berechnung aus DELPHIN

t                // Simulationszeit in [s]
t_startOffset    // Startdatum und -zeit in [s]
startYear        // Jahr, in dem die Simulation startet

// Berechne Offset in [s] zum 1. Januar 0 Uhr (Mitternacht)
t_year = t + t_startOffset;

// Berechne Klimadaten, wobei t_year > 365*24*3600 werden kann,
// startYear wird aber nie verändert!
ccm->setTime(startYear, t_year)
```

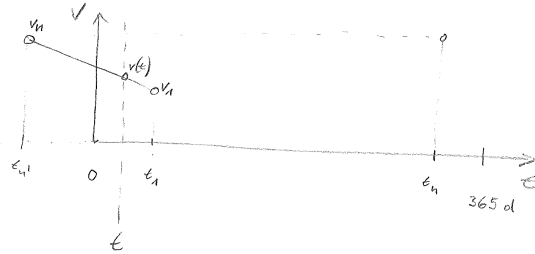


Abbildung 2: Lineare Interpolation des Wertes $v(t)$ zwischen letztem und erstem Datenwert bei zyklischer Behandlung

1.3 Zyklische Daten für mehrjährige Simulationen

Bei mehrjährigen Simulation können Klimadaten, z.B. Testreferenzjahre zyklisch verwendet werden. Dies impliziert einen *jährlichen* Zyklus, sodass Zeitpunkte $\geq 365\text{ d}$ solange um 365 d reduziert werden, bis der Zeitpunkt wieder $< 365\text{ d}$ ist.

```
// Beispielhafte Berechnung aus dem ClimateCalculationModule

void ClimateDataLoader::setTime(int startYear, double t_year) {
    // wenn zyklische Berechnung
    if (m_useCyclicClimateData) {
        // reduziere Zeit solange um 1 Jahr, bis t_year < 1 Jahr ist
        while (t_year >= SECONDS_PER_YEAR) // SECONDS_PER_YEAR = 365*24*3600
            t_year -= SECONDS_PER_YEAR;
    }
    ...
}
```

Diese Prozedure impliziert eine konstanten Jahreslänge von 365 d , d.h. Schaltjahre werden bei zyklischer Betrachtung ignoriert es gibt keinen 29. Februar. Ein Klimadatensatz für mehrjährige Berechnungen sollte den gesamten Zeitraum von $0 \leq t < 365\text{ d}$ definieren. Zeitpunkte $\geq 365\text{ d}$ sind nicht erlaubt. Liegt der zu berechnende Zeitpunkt t vor dem ersten abgelegten Zeitpunkt t_1 , wird zwischen dem ersten und dem letzten Zeitpunkt t_n und den dazugehörigen Werten linear interpoliert (Abbildung 2).

Die Interpolation zwischen erstem und letzten Zeitpunkt erfolgt über die Gleichung:

$$t' = t_n - 365\text{ d}$$

$$\alpha = \begin{cases} \frac{t-t_n}{t_1-t'} & , \text{ wenn } t \geq t_n \\ \frac{t-t'}{t_1-t'} & , \text{ sonst} \end{cases}$$

$$v(t) = (1 - \alpha) v_n + \alpha v_1$$

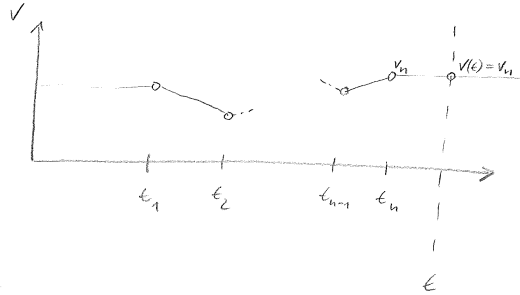


Abbildung 3: Konstante Extrapolation des Wertes $v(t)$ wenn t außerhalb des Datenbereichs liegt, d.h. $v(t) = v_n$ für $t \geq t_n$

1.4 Kontinuierliche Datenreihen

Beim Nachrechnen von eventuell mehrjährigen Messreihen werden keine zyklischen Berechnungsregeln angewendet. D.h. das Simulationsprogramm muss in der Lage sein, zwischen zyklischer (jährlicher) und kontinuierlicher Simulation umzuschalten. Die Verwendung des 29. Januars bei Schaltjahren wäre dann möglich, müsste aber auch konfigurierbar sein (falls in den Klimadaten dieser Tag rausgefiltert wurde).

Bei der Definition von Klimadaten für kontinuierliche Berechnung muss der absolute Zeitbezug gegeben sein, d.h. der Klimadatensatz muss entweder absolute Zeitstempel enthalten (siehe Abschnitt 2.5 zu CSV-Dateien), oder das Jahr, in dem die Klimadaten beginnen als Zusatzinformation im Header (siehe Abschnitt 2.6 zu D6O-Dateien).

Wenn bei Verwendung nicht-zyklischer Daten der zu berechnende Zeitpunkt t vor dem ersten oder nach dem letzten Datenzeitpunkt liegt, muss eine Fehlerbehandlung erfolgen (eventuell auch konfigurierbar - konstante Extrapolation, siehe Abbildung 3, oder Simulationsabbruch mit Fehler).

2 Klimadatenquellen

Klimadaten müssen nach dem Einlesen als Zeit-Werte-Tabelle entsprechend der oben genannten Regeln interpretierbar sein. Weiterhin muss die Einheit der abgelegten physikalischen Größe bekannt sein, damit der Zahlenwert in die Basis-SI-Einheit zur Verwendung in den Modellgleichungen umgerechnet werden kann. Die physikalische Größe kann je nach Dateiformat über ein Schlüsselwort im Header der Datei, oder den Dateinamen oder durch Spaltenposition gegeben sein.

Grundsätzlich gilt für alle unten aufgeführten Dateiformate, dass ausschließlich **englische Zahlformatierungen** (zum Beispiel 12.43) **ohne Tausendertrennzeichen** unterstützt werden.

2.1 CCD Format

Das CCD Format (Climate Condition Data) speichert stets nur eine Klimakomponenten, d.h. eine Zeitreihe. Es gibt optionale Kommentarzeilen ohne syntaktische Bedeutung. Weiterhin gibt es oberhalb des Datenblocks eine verpflichtende Headerzeile, welche ein Schlüsselwort zur Identifikation der physikalischen Größe und die Werteeinheit (entsprechend der IBK Einheitenliste) beinhaltet.

Zeitpunkte werden über den Tag des Jahres, beginnend mit 0, und der Zeit im Format hh:mm:ss angegeben. Zeitpunkte müssen streng monoton ansteigen. Ein absoluter Zeitbezug ist in CCD-Dateien nicht vorgesehen. Es steckt die implizite Annahme dahinter, dass die Klimadaten **stets im Startjahr der Simulation beginnen**. Dieses deckt auch bei kontinuierlicher Berechnung die meisten Anwendungsfälle ab. Soll bei mehrjährigen Datenaufzeichnungen die Simulation erst zu einem späteren Zeitpunkt (d.h. späteres Jahr) beginnen, muss ein anderes Klimadatenformat (CSV bzw. D6O) gewählt werden.

In der Datensektion entspricht jede Zeile einem Zeit-Wert-Datenpunkt. Tag, Zeit und Wert sind Whitespace-getrennt, d.h. es sind beliebig viele Tabulatorzeichen bzw. Leerzeichen dazwischen. Leerzeilen in der Datei werden grundsätzlich ignoriert.

Beispiel:

```
# Climate Data File
# Potsdam

TEMPER    C

0 1:00:00 -2.6
0 2:00:00 -3.9
0 3:00:00 -4.6
0 4:00:00 -3.9
0 5:00:00 -3.3
0 6:00:00 -2.9
0 7:00:00 -2
0 8:00:00 -1.5
0 9:00:00 -1.2
0 10:00:00 -0.6
...
364 23:00:00 0.7
364 24:00:00 -0.8
```

Die Berechnung des Zeitpunktes erfolgt aus der Zeitangabe mittels folgender Gleichung:

```
std::string line = "364 23:00:00 0.7";

// ersetze : mit Leerzeichen
std::replace(line.begin(), line.end(), ':', ' ');

int day, hour, min, sec;
double value;
// Lesen der Zahlen aus der Zeile
std::stringstream strm(line);
// Lesen und auf Fehler _und_ Vollständigkeit prüfen (wegen deutscher
// Zahlformate "12,7")
if (!(strm >> day >> hour >> min >> sec >> value) || !strm.eof()) {
    // Fehlerbehandlung
    ...
}

// Berechnung der Sekunden des Jahres
double sec = day * 24 * 3600 + hour * 3600 + min * 60 + sec;
```

Dadurch werden Zeitangaben von 00:00:00 bis 23:59:59 als auch Zeitangaben von 01:00:00 bis 24:59:59 korrekt eingelesen. Letztlich sind dadurch die Zeitstempel „0 24:00:00“ und „1 00:00:00“ identisch (und dürfen daher nicht gleichzeitig in einer CCD Datei vorkommen).

2.1.1 Verwendung von CCD Dateien für zyklische Berechnungen

Bei der zyklischen jährlichen Berechnung dürfen Zeitpunkte $t \geq 365$ d nicht definiert sein. Es ist aber möglich, den Zeitpunkt $t = 365$ d bzw. „364 24:00:00“ oder alternativ „365 00:00:00“ in der CCD-Datei zu verwenden, **solange nicht gleichzeitig** „0 00:00:00“ angegeben ist. Bei der Berechnung wird der Zeitpunkt $t = 365$ d als Zeitpunkt $t = 0$ d interpretiert. Beispiele für gültige Klimadateien mit stündlichem Raster sind:

```
0    00:00:00 -10
0    01:00:00 -10.5
...
364  23:00:00 -8
```

oder beginnend mit dem Ende der ersten Stunde

```
0    01:00:00 -10.5
...
364  23:00:00 -8
365  00:00:00 -10
```

oder alternativ mit 24 Stundenformatierung.

```
0    01:00:00 -10.5
...
364  23:00:00 -8
364  24:00:00 -10
```

Inhaltlich sind diese Angaben im Kontext der mehrjährigen zyklischen Simulation identisch.

2.1.2 Komprimiertes Format

Bei Klimadaten im CCD-Format müssen Bei jährlichen Klimadaten

2.2 CLI Format

Das in DELPHIN 4 verwendete CLI Format wird derzeit nicht vom CCM unterstützt. CLI-Dateien können (nahezu verlustfrei) in CCD-Dateien konvertiert werden. Verlustbehaftet ist die Konvertierung nur für Zeitstempel mit Auflösungen unterhalb einer Sekunde. Falls derartige Eingaben benötigt werden, muss das DATAIO-Format verwendet werden (siehe D6O Formatbeschreibung in Abschnitt 2.6). Die Verwendungsinformation (zyklisch oder nicht, interpolieren oder nicht) sind in den neueren Simulationsprogrammen nicht mehr Eigenschaft der Klimadateien selbst, sondern der Datenstrukturen (z.B. Climate Condition Definition), welche die Verwendung definieren.

2.3 EPW-Dateien

Das CCM unterstützt EPW-Dateien mit jährlichen, stündlichen Daten. Es müssen exakt 8760 Datensätze beginnend mit dem Zeitpunkt $t = 1 \text{ h}$ (bzw. „001:00:00“ in CCD-Notation) definiert sein. Ausgelesen werden alle Spalten bzw. physikalischen Größen, die aktuell vom CCM unterstützt werden. EPW-Dateien sind Klimadatenkontainer und liefern mehrere Klimadatenätze pro Datei. Das Startjahr bzw. der absolute Zeitstempel wird bei EPW-Dateien ignoriert. Bei nicht-zyklischer Anwendung (kontinuierliche Berechnung) wird das Simulationsstartjahr verwendet.

2.4 C6B-Dateien

Das C6B-Dateiformat wurde aus lizenzrechtlichen Gründen eingeführt, sodass wir Klimadaten „nicht im Klartext“ weitergeben können. Es enthält einen Header mit Standortinformationen und Beschreibungen und Datenblock mit allen vom CCM unterstützten Klimakomponenten. C6B-Dateien sind wie EPW-Dateien Klimadatenkontainer und liefern mehrere Klimadatenätze pro Datei. Bis auf die Header-Informationen der EPW-Datei, können die Daten aus einer EPW-Datei direkt in eine C6B-Datei konvertiert werden. Der Datenblock enthält ebenso 8760 Dateireihen, beginnt mit dem Zeitpunkt $t = 1 \text{ h}$ und endet mit dem letzten Zeitpunkt $t = 365 \text{ d}$ (welcher bei der Berechnung wiederum als Zeitpunkt $t = 0 \text{ d}$ verwendet wird).

2.5 CSV-Dateien

Das ungewöhnliche Zeitstempelformat der CCD-Dateien (geeignet für Laborversuche/synthetische Klimadateien/Zeitreihen für Simulationsentwicklung) ist für die praktische Anwendung ungeeignet. Daher wird ein einfaches CSV-Format unterstützt, wobei der Zeitstempel ein Format hat, welches durch Tabellenkalkulationsprogramme (Excel, LibreOffice, ...) oder Skriptsprachen (Python) direkt und ohne umständliche Konvertierung lesbar und erstellbar ist.

Beispiel für die Verwendung von CSV für eine einzelne Klimakomponente:

```
Comment,"Auch Kommentare werden als CSV-Zeichenketten abgelegt"
Unit,C
Quantity,Temperature

2016-01-01 01:00:00, -10.5
...
2016-31-12 23:00:00, -8
2016-31-12 24:00:00, -10
```

Beispiel für die Verwendung von CSV für mehrere Klimakomponenten (Klimacontainer):

```
Comment,"Auch Kommentare werden als CSV-Zeichenketten abgelegt"
Unit,C
Quantity,Temperature, RelativeHumidity, DirectRadiationNormal, DiffuseRadiationHorizontal, WindDirection, ←
WindVelocity, LongWaveCounterRadiation, AirPressure, Rain

2000-01-01 01:00:00, -3.6, 90, 0, 0, 200, 3.1, 206, 96910, 0
2000-01-01 02:00:00, -3.6, 90, 0, 0, 200, 3.1, 206, 96910, 0
2000-01-01 03:00:00, -4.8, 84, 0, 0, 160, 2.4, 206, 96660, 0
2000-01-01 04:00:00, -5.8, 81, 0, 0, 160, 2.4, 204, 96450, 0
2000-01-01 05:00:00, -5.7, 76, 0, 0, 190, 2.2, 207, 96380, 0
2000-01-01 06:00:00, -5.7, 76, 0, 0, 130, 2.5, 202, 96300, 0
2000-01-01 07:00:00, -5, 63, 0, 0, 170, 2, 216, 96220, 0
2000-01-01 08:00:00, -5.2, 59, 0, 0, 160, 3, 221, 96150, 0
2000-01-01 09:00:00, -3.5, 57, 0, 0, 170, 3, 223, 96100, 0
2000-01-01 10:00:00, -2.7, 58, 15.9979, 36, 180, 5, 249, 96050, 0
2000-01-01 11:00:00, -0.9, 46, 22.1223, 75, 190, 5, 239, 95990, 0
2000-01-01 12:00:00, -0.3, 46, 169.561, 78, 160, 7, 238, 95920, 0
2000-01-01 13:00:00, 1.2, 36, 178.319, 97, 170, 6, 249, 95810, 0
...
```

Headereinträge werden durch das erste CSV-Element (erste Spalte) definiert:

Comment jedwede Kommentare, müssen in Anführungszeichen eingeschlossen sein, wenn sie mehrzeilig sind oder Kommata enthalten, mehrere Zeilen mit Comment-Einträgen sind mögliche

Unit gefolgt in der zweiten Spalte von der Werteeinheit (entsprechend IBK Einheiten-Liste)

Quantity gefolgt von Bezeichnern der physikalischen Größe (sollte eine aus der IBK Quantity-Liste sein)

Nur die Angabe der Unit ist verpflichtend.

Die Zeitstempel beinhalten das Jahr, welches bei zyklischer Anwendung jedoch ohne Bedeutung ist. Zeitstempel sind **immer in Winterzeit**. Es gelten die gleichen Regeln wie bei CCD-Dateien hinsichtlich der erlaubten Zeitspanne (streng

monoton, keine doppelten Zeitpunkte, Zeitpunkt $t = 0$ d und $t = 365$ d nicht gleichzeitig in einer Datei).

Falls mehr als nur eine Datenspalte in der Datei ist, hängt es vom Kontext ab, wie die Datei interpretiert wird. Wird die CSV Datei z.B. als einzelner Klimadatensatz referenziert, so wird ausschließlich die erste Spalte verwendet. Die physikalische Größe aus der Quantity-Kopfzeile kann zur Konsistenzprüfung herangezogen werden.

Wird eine CSV-Datei jedoch alternativ zu c6b oder epw als Klimacontainer verwendet, so muss sie **alle vom CCM unterstützten Spalten in der richtigen Reihenfolge** enthalten (siehe Beispiel oben und Dokumentation des CCM). Damit ist das CSV-Format das einzige Format, welches für die Definition von mehrjährigen Klimadaten, bzw. detaillierten Klimadaten mit sub-stündlichen Intervallen mehrere Klimakomponenten in einer Datei vereint.

2.6 D6O Dateien

Ergebnisdateien der Simulation können als Eingangsdaten für die Klimasimulation verwendet werden. Gleichzeitig können nur mit diesem Format Zeitpunkt mit Abständen < 1 Sekunde definiert werden. Das grundlegende Dateiformat ist in der DATAIO-Spezifikation beschrieben. Für die Verwendung im CCM ist jedoch nur eine Untermenge an Daten notwendig. Es gelten weiterhin folgende Anforderungen:

- QUANTITY_KW Schlüsselwort sollte Bezeichnung der physikalischen Größe sein (eine aus der IBK Quantity-Liste)
- VALUE_UNIT muss die Werteeinheit angeben (entsprechend IBK Einheiten-Liste)
- TIME_UNIT muss die Zeiteinheit angeben (entsprechend der IBK Einheiten-Liste)
- START_YEAR gibt das Jahr an, in dem die Klimadaten beginnen (bei zyklischer Anwendung wird das ignoriert, bei kontinuierlicher Anwendung muss das ein gültiger Wert sein oder ein negativer Wert als ungültiges Startjahr. In diesem Fall wird das Startjahr der Simulation verwendet.)
- Ausschließlich die erste Spalte (erster Wert pro Datenzeile) wird eingelesen.

Alle anderen Header-Daten sind optional und werden beim Einlesen durch das CCM ignoriert.

Ein minimalistisches D6O Beispiel wäre:

```
VALUE_UNIT = C
TIME_UNIT = d

0.04167 -2.6
0.08333 -3.9
0.12500 -4.6
0.16667 -3.9
0.20833 -3.3
0.25000 -2.9
0.29167 -2
0.33333 -1.5
0.37500 -1.2
0.41667 -0.6
0.45833 0.1
0.50000 0.5
0.54167 0.4
```

Es können jedoch auch beliebige andere D6O-Dateien (z.B. Simulationsergebnisse) verwendet werden.

Sollte das D6O-Format für zyklische Klimadaten verwendet werden, muss beim Generieren der Zeitpunkte **auf Rundungsfehler geachtet werden**, sodass keine Zeitpunkten $> 365\text{d}$ entstehen (Beispiel: „365.00001“ bei Verwendung von Tagen bzw. „1.00000012“ bei Verwendung von Jahren als Zeiteinheit).