### **Projektdokumentation Log-Datei-Visualisierung**

Das Log-Datei-Visualisierungsprogramm (im Folgenden Plotter oder Script genannt) dient der Visualisierung des zeitlichen Verlaufs des Verbrauchs eines Jobs auf dem Lichtenberg-Hochleistungsrechner. Es generiert ein Bild, auf dem abgesehen von der Belastung auch eine abschnittsweise farbliche Bewertung in vier Kategorien abzulesen ist.

***Inhalte*:** „Plotter.zip“, Anleitung

Das Projekt nutzt Python (Version 3.5, mindestens jedoch Version 3) sowie das Numpy, das Pyplot Modul der Matplotlib-Bibliothek und insbesondere die datetime-Bibliothek.

***Anwendung****:*  
*Schritt 1*: Entpacken Sie das Zip Archiv in einen Ordner Ihrer Wahl (z.B. „C:\Folder“).

*Schritt 2*: Geben Sie in das Terminal Ihrer Wahl folgenden Befehl ein:

|  |
| --- |
| python plotter.py Source\_Datei Ziel\_Datei q=Jahresquota –s=Startpunkt –p=Projektname |

hierbei ersetzen Sie bitte „Source\_Datei“ mit dem Namen der Source-Datei, inklusive des Pfads, den Sie in Schritt 1 ausgewählt haben, das „Ziel\_Datei“ mit dem Namen der Ziel-Datei, ebenfalls inklusive eines Pfades, sowie das „Jahresquota“ mit einer Angabe einer Jahresquote in Corestunden (z.B „

|  |
| --- |
| python C:\Folder\plotter.py C:\Folder\Mueller.log-example Abbildung3.png –q=10000.-s=2017-11-07 p=005 |

). Der Begriff Startpunkt soll mit einem Datum oder einem Datum inklusive sekundengenaue Uhrzeit in folgendem Format angegeben werden: „YY-mm-dd-HH-MM-SS“, der Projektname soll innerhalb des Accountnamens vorkommen.

Die Parameter Jahresquota, Startpunkt und Projektname sind optional.

Das Script generiert nun, unter dem von Ihnen angegebenen Namen, eine Grafik, die den Rechenverbrauch des Projekts (oder, falls Sie keinen Filter mit „p=“ gesetzt haben, aller Projekte) visualisiert und das Verhalten in Relation zur Jahresquota farblich markiert (rot: deutlich über Veranschlagung, gelb: über Veranschlagung, dunkelgrün: Innerhalb des Rahmens, hellgrün: deutlich unter dem zugeteilten Verbrauch).  
Für Monatsabschnitte sind 109575 Sekunden, also 30,4375 Tage vorgesehen.

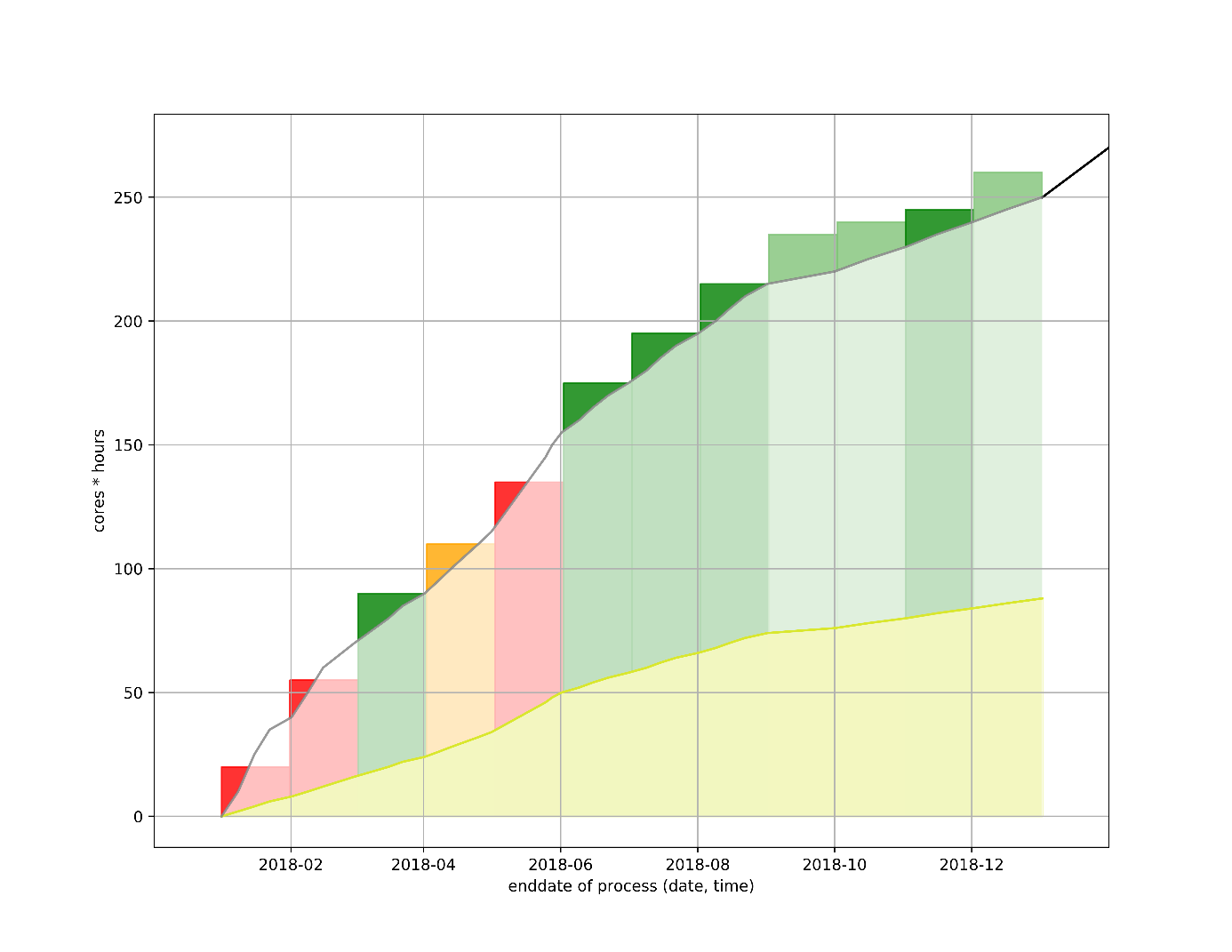


Abbildung 1: Eine Visualisierung einer Log-Datei

Das Script erstellt einen Graphen ähnlich dem im Beispielbild. Hier ist in Grau die verwendete Rechenzeit in Corestunden gegenüber dem Endzeitpunkt des Prozesses dargestellt. Somit lässt sich der zeitliche Verlauf des Rechenleistungsverbrauchs interpretieren, hier durch ein farbiges Highlight der möglichen Quotas.

Die Farbliche Einteilung der Gebiete unterteilt die Abschnitte in Bereiche, in denen der Verbrauch unter 70% des Vergleichswertes liegt, solche, in denen er sich zwischen 70% und 110% befindet, sowie den Gelben Bereichen, welche einen Verbrauch zwischen 110% und 150% signalisieren. Darüber hinaus existieren rote Markierungen, welche eine Corerechenzeit von mehr als 150% des Monatsquotas entspricht.  
Als Monatsquota gilt hier ca. ein zwölftel des als Parameter eingegebenen Jahresquotas.

*Interna*:  
*Funktionen*:

**translate\_date\_to\_sec**; Diese Funktion dient dem umwandeln von datetime-formaten in Timestamps, die pure integer sind. Darüber hinaus dient die Funktion als Filter für nicht auswertbare Daten, speziell „Unknown“, üblicherweise bei Aktionen die noch im Gang sind.

**Colorisation**; nimmt als Parameter zwei Vergleichswerte entgegen, und gibt für das Verhältnis der beiden eine Farbe zurück.

**Essentialpar**: greift die verwendeten Parameter auf, und interpretiert jeden Parameter, der mit „-o“ anfängt als Output, sowie jeden, der mit „-src“ anfängt als Quelldatei für Daten, darüber hinaus alle Parameter, die nicht mit einem „-“ anfangen ebenfalls als Datenquelle, es sei denn, es existiert kein Parameter, der mit „-o“ anfängt. In diesem Fall wird der letzte, vorzeichenlose Parameter als Outputdatei angesehen.

**Translate\_time\_to\_sec:** dient der Umwandlung von Zeiträumen, die in einem Format angegeben werden, das aus optionalen Tagen, Stunden, sowie Sekunden besteht. Die Funktion liefert die Anzahl der Sekunden zurück.

*Globale Variablen:*

**seconds\_per\_instances** bezeichnet die Sekundenlänge jeder Instanz.

**number\_of\_instances** errechnet hieraus die Anzahl der Instanzen.

**x\_start** bezeichnet den Zeitpunkt zu dem der erste Prozess beendet wurde.

**y\_end** bezeichnet entsprechend den Endzeitpunkt des letzten Prozesses.

**plot\_array** beinhaltet die einzelnen Datenpunkte der eingelesenen Datei; Zeitpunkt, Rechenzeit und akkumulierte Rechenzeit

**tmp\_y** beinhaltet die y-Werte dieser Datenpunkte.

**tmp\_x** beinhaltet die x-Werte dieser Datenpunkte

**tmp\_y2** und **tmp\_x3** beinhalten x- und y-Werte für die Quotenvisualisierungen.

**col** bezeichnet die aktuell verwendete Farbe.