

transform_data

Benutzerhandbuch

Jira-Daten aufbereiten fuer Metriken und Berichte

Inhalt

Inhalt	2
1 Was ist transform_data?	4
2 Voraussetzungen und Installation	5
2.1 Was muss installiert sein?	5
2.2 Programm starten	5
3 Eingabedateien	6
3.1 Jira-JSON-Export	6
3.2 Workflow-Definitionsdatei	6
4 Die Workflow-Definitionsdatei	7
4.1 Aufbau	7
4.2 Reihenfolge der Stages	7
4.3 Nicht gemappte Jira-Status	7
5 Die grafische Oberflaeche (GUI)	8
5.1 Dateien auswaehlen	8
5.2 Ausgabe konfigurieren	8
5.3 Verarbeitung starten	8
6 Die Ausgabedateien	9
6.1 IssueTimes.xlsx	9
6.2 Transitions.xlsx	9
6.3 CFD.xlsx	9
7 Wie werden Datum und Zeiten berechnet?	11
7.1 Stage-Zeiten	11
7.2 First Date	11
7.3 Closed Date	11
7.4 Beispiele	12
8 Haeufige Fragen und Tipps	13
<i>F: Ein Issue hat kein First Date -- warum?</i>	13
<i>F: Ein Issue hat kein Closed Date -- warum?</i>	13
<i>F: Im Log erscheint eine Warnung ueber nicht gemappte Status.</i>	13
<i>F: Die Ausgabedateien werden nicht erstellt.</i>	13

F: Welche Datei verwende ich fuer build_reports? 13

F: Kann ich die Ausgabedateien manuell bearbeiten? 13

F: Wie oft muss ich transform_data ausfuehren? 13

F: Was bedeutet 'Carry-forward'? 14

9 Glossar 15

1 Was ist transform_data?

transform_data ist das erste Modul in der situation-report-Werkzeugkette. Es liest einen Rohdaten-Export aus Ihrem Ticketsystem (Jira) und bereitet die Daten fuer die weitere Analyse auf. Das Ergebnis sind drei **Excel-Dateien**, die zeigen, wie lange Issues in welchen Workflow-Schritten verbracht haben und wie sich der Bestand ueber die Zeit entwickelt hat.

Das Programm besitzt eine einfache grafische Oberflaeche (GUI): keine Programmierkenntnisse erforderlich. Sie waehlen zwei Dateien aus, klicken auf 'Verarbeiten' und erhalten die Excel-Dateien automatisch.

Was transform_data liefert:

- **IssueTimes.xlsx**: Alle Issues mit Meilensteinzeitpunkten und Minuten pro Workflow-Stage.
- **Transitions.xlsx**: Vollstaendige Statushistorie aller Issues.
- **CFD.xlsx**: Taeglich Eintrittszaehlungen je Stage fuer das Cumulative Flow Diagram.

Die erzeugten Excel-Dateien werden anschliessend vom Modul **build_reports** verwendet, um Diagramme und Berichte zu erstellen.

2 Voraussetzungen und Installation

2.1 Was muss installiert sein?

Damit transform_data funktioniert, muss auf dem Rechner **Python 3.11 oder neuer** installiert sein. Ausserdem muessen einige Python-Pakete vorhanden sein. Wer das Programm fuer Sie eingerichtet hat, sollte dies bereits erledigt haben.

2.2 Programm starten

Es gibt zwei Moeglichkeiten, transform_data zu starten:

- **Doppelklick** auf die Datei **transform_data_gui.pyw** im Projektordner -- oeffnet die grafische Oberflaeche ohne ein Konsolenfenster.
- **Terminal / Eingabeaufforderung**: Ins Projektverzeichnis wechseln und `python -m transform_data` eingeben.

Tipp: Wer die GUI regelmaessig nutzt, kann eine Verknuepfung zur Datei **transform_data_gui.pyw** auf dem Desktop erstellen.

3 Eingabedateien

transform_data benötigt genau zwei Dateien: den Jira-JSON-Export und die Workflow-Definitionsdatei. Beide werden in der GUI per Datei-Dialog ausgewählt.

3.1 Jira-JSON-Export

Dies ist ein Export aller Issues eines Jira-Projekts im JSON-Format. Er enthält für jedes Ticket die komplette Statushistorie (Changelog), Metadaten wie Issuetyp, Komponenten und Erstellungsdatum sowie den aktuellen Status.

Wie der Export aus Jira erzeugt wird, beschreibt das Modul **get_data**. Die JSON-Datei sollte nicht manuell bearbeitet werden.

Typischer Dateiname: ART_A.json, MYPROJECT.json o.äe.

Groesse: Abhängig von der Anzahl der Issues; häufig einige MB.

3.2 Workflow-Definitionsdatei

Die Workflow-Datei ist eine einfache Textdatei (.txt), die beschreibt, wie die Jira-Status Ihres Projekts auf logische Prozessschritte (Stages) abgebildet werden. Sie definiert ausserdem, welche Stage den Beginn der Entwicklung (First Date), den Entwicklungsstart (Implementation Date) und den Abschluss (Closed Date) markiert.

Typischer Dateiname: workflow_ART_A.txt o.äe.

Erstellt von: Ihrem technischen Ansprechpartner oder Scrum Master -- einmalig pro Projekt, selten geändert.

4 Die Workflow-Definitionsdatei

Die Workflow-Datei steuert, wie transform_data die Jira-Status Ihres Projekts interpretiert. Sie muessen diese Datei in der Regel nicht selbst erstellen -- dieses Kapitel erklart jedoch ihren Aufbau, damit Sie sie lesen und bei Bedarf anpassen koennen.

4.1 Aufbau

Jede Zeile der Datei beschreibt entweder eine **Stage** (einen Prozessschritt) oder einen **Marker** (einen Meilenstein). Leerzeilen werden ignoriert.

Beispiel:

```
Funnel:New:Open:To Do
Analysis:In Analysis:Estimated
Implementation:In Implementation:In Progress
Done:Canceled
<First>Analysis
<InProgress>Implementation
<Closed>Done
```

Format	Bedeutung
Stage	Kanonischer Name eines Prozessschritts. Alle Issues, die sich in dieser Stage befinden, werden hier g
Stage:Alias1:Alias2	Stage mit Jira-Statusnamen, die darauf gemappt werden. Alle genannten Jira-Status werden als diese
<First>Stage	Diese Stage markiert den Beginn der aktiven Bearbeitung (First Date). Typisch: der erste Schritt nach
<InProgress>Stage	Diese Stage setzt das Implementation Date. Standard: Stage namens 'Implementation', wenn vorhand
<Closed>Stage	Diese Stage markiert den Abschluss (Closed Date). Typisch: Releasing oder Done.

4.2 Reihenfolge der Stages

Die Stages werden in der Reihenfolge aufgefuehrt, in der sie im Prozess durchlaufen werden. Diese Reihenfolge wird fuer das Cumulative Flow Diagram und fuer die Berechnung des Closed Date bei uebersprungenen Stages verwendet.

4.3 Nicht gemappte Jira-Status

Enthaelt der Jira-Export Status, die in der Workflow-Datei nicht definiert sind, gibt transform_data eine Warnung aus:

```
WARNUNG: 2 Status in den Daten nicht in der Workflow-Datei gemappt:
- To Do
- Unknown
> Zeit dieser Status wird der letzten bekannten Stage zugerechnet.
```

Die Zeit in nicht gemappten Status geht nicht verloren -- sie wird der **letzten bekannten Stage** zugerechnet (Carry-forward). Issues bleiben vollstaendig in allen Ausgaben erhalten.

5 Die grafische Oberflaeche (GUI)

Nach dem Start oeffnet sich ein Fenster mit Datei-Auswahl-Feldern, Optionen und einem Log-Bereich fuer Statusmeldungen.

5.1 Dateien auswaehlen

Laden Sie die beiden Eingabedateien:

- **JSON-Datei** - Klicken Sie auf den Ordner-Button rechts neben dem JSON-Feld und waehlen Sie Ihren Jira-Export aus. Ausgabeordner und Praefix werden automatisch aus dem Dateinamen vorbelegt.
- **Workflow-Datei** - Waehlen Sie die passende .txt-Workflow-Datei fuer Ihr Projekt aus.

Tipp: Wenn Sie die JSON-Datei zuerst auswaehlen, werden Ausgabeordner und Prafix automatisch vorausgefüllt. Sie muessen diese Felder nur aendern, wenn Sie einen anderen Speicherort oder Namen wuenschen.

5.2 Ausgabe konfigurieren

Feld	Bedeutung
Ausgabeordner	Verzeichnis, in das die drei Excel-Dateien gespeichert werden. Standard: Verzeichnis der JSON-Dateien
Praefix	Namens-Praefix fuer die Ausgabedateien. Aus 'ART_A' werden z.B. ART_A_IssueTimes.xlsx, ART_A_...

5.3 Verarbeitung starten

Klicken Sie auf '**Verarbeiten**'. Das Programm liest die Daten ein, berechnet alle Werte und speichert die drei Excel-Dateien. Der Fortschritt und eventuelle Warnungen erscheinen im Log-Bereich.

Warnungen im Log beachten: Eine Warnung ueber nicht gemappte Status bedeutet nicht, dass die Verarbeitung fehlgeschlagen ist -- sie ist ein Hinweis, dass die Workflow-Datei moeglicherweise unvollstaendig ist. Die Ausgabedateien werden trotzdem erstellt.

6 Die Ausgabedateien

transform_data erzeugt drei Excel-Dateien. Diese Dateien dienen als Eingabe fuer build_reports und sollten nicht manuell bearbeitet werden.

6.1 IssueTimes.xlsx

Die wichtigste Ausgabedatei. Sie enthaelt eine Zeile pro Issue mit allen Zeitangaben. Diese Datei wird von build_reports fuer alle Metriken ausser dem Cumulative Flow Diagram benoetigt.

Spalte	Inhalt
Project	Projektschlüssel (z.B. ART_A)
Key	Issue-Schlüssel (z.B. ART_A-123)
Issuetype	Typ des Issues (z.B. Feature, Bug, Story)
Status	Aktueller Jira-Status
Created Date	Erstellungsdatum des Issues in Jira
First Date	Erster Eintritt in die <First>-Stage (Beginn der Bearbeitung)
Implementation Date	Erster Eintritt in die <InProgress>-Stage
Closed Date	Zeitpunkt des Abschlusses (leer = noch offen)
Stage-Spalten	Je eine Spalte pro Workflow-Stage: Minuten in dieser Stage
Resolution	Abschlussart aus Jira (z.B. Fixed, Duplicate, Canceled)

6.2 Transitions.xlsx

Enthaelt die vollstaendige Statushistorie aller Issues -- einen Eintrag pro Statuswechsel. Nuetzlich fuer detaillierte Analysen einzelner Issues.

Spalte	Inhalt
Key	Issue-Schlüssel
Transition	Stage-Name (oder 'Created' fuer den Erstellungszeitpunkt)
Timestamp	Datum und Uhrzeit des Statuswechsels

6.3 CFD.xlsx

Enthaelt fuer jeden Kalendertag die Anzahl der Issues, die an diesem Tag in die jeweilige Stage **eingetreten** sind (taeglich Eintrittszaehlungen, keine Snapshots). build_reports akkumuliert diese Werte kumulativ zu einem laufenden Gesamtwert und erzeugt daraus das Cumulative Flow Diagram.

Spalte	Inhalt
Day	Datum (YYYY-MM-DD)

Spalte	Inhalt
Stage-Spalten	Je eine Spalte pro Stage: Anzahl neuer Eintritte an diesem Tag

7 Wie werden Datum und Zeiten berechnet?

Dieses Kapitel erklart, nach welchen Regeln transform_data die Meilenstein-Daten (First Date, Closed Date) und die Stage-Zeiten ermittelt. Das Verstaendnis dieser Regeln hilft bei der Interpretation der Ergebnisse.

7.1 Stage-Zeiten

Fuer jedes Issue wird gemessen, wie viele Minuten es in jeder Stage verbracht hat. Dabei gelten folgende Regeln:

Situation	Regel
Issue erstellt, aber noch kein Statuswechsel	Die Zeit von der Erstellung bis zum ersten Statuswechsel wird der initialen Stage zugerechnet.
Statuswechsel zu einer nicht gemachten Stage	Die Zeit geht in der letzten bekannten Stage weiter (Carry-forward). Es gibt keinen Zeitverlust.
Aktueller Status	Die letzte bekannte Stage akkumuliert Zeit bis zum Zeitpunkt der Verarbeitung.
Kein Statuswechsel	Alle Stage-Spalten zeigen 0 Minuten.

7.2 First Date

Das First Date wird gesetzt, sobald ein Issue zum ersten Mal in die in der Workflow-Datei mit **<First>** markierte Stage wechselt. Es repraesentiert den Beginn der aktiven Bearbeitung.

Uebersprungene First-Stage: Betritt ein Issue eine Stage, die im Workflow **nach** der **<First>**-Stage, aber **vor** der **<Closed>**-Stage liegt, ohne die First-Stage selbst zu betreten, wird der Eintrittszeitpunkt dieser spaeter erreichten Stage als First Date verwendet. Gleiches gilt fuer die **<InProgress>**-Stage: wird sie uebersprungen, aber eine Stage danach (vor Closed) betreten, gilt deren Zeitstempel als Implementation Date -- sofern ein First Date vorhanden ist.

Issues, die ausschliesslich die Closed-Stage oder spaetere Stages erreichen ohne vorherige Entwicklungsschritte, erhalten kein First Date.

7.3 Closed Date

Das Closed Date wird beim Eintritt in die mit **<Closed>** markierte Stage gesetzt. Bei Issues, die mehrfach geoeffnet und geschlossen wurden, zaehlt der **letzte** Schliessungszeitpunkt.

Uebersprungene Closed-Stage: Hat ein Issue ein First Date, aber die **<Closed>**-Stage wurde im Prozess uebersprungen (z.B. direkter Wechsel von 'Implementation' nach 'Done'), gilt die **erste Stage chronologisch nach der Closed-Stage** im Workflow als Abschlusszeitpunkt. Issues ohne First Date erhalten kein Closed Date.

Wiedergeoeffnete Issues: Befindet sich ein Issue aktuell in einer Stage **vor** der **<Closed>**-Stage, wird kein Closed Date gesetzt -- auch wenn die Stage oder eine spaetere Stage zuvor schon einmal erreicht wurde. Das Issue gilt als noch offen.

Nie bearbeitete Issues: Issues, die direkt von einer 'To Do'-Stage in die <Closed>-Stage gesprungen sind, ohne je in Bearbeitung zu sein (kein First Date), erhalten **kein Closed Date**. Sie erscheinen in keiner Metrik als abgeschlossen und zaehlen auch nicht in der Flow Velocity. Typisches Muster: Issue wurde im Ticketsystem manuell in wenigen Sekunden durch alle Status geklickt ohne echte Entwicklungsarbeit.

7.4 Beispiele

Beispiel 1 -- Uebersprungene Closed-Stage (ART_A-615):

Datum	Statuswechsel	Ergebnis
13.10.2025	Funnel -> Analysis	First Date = 13.10.2025 (Analysis ist <First>-Stage)
07.11.2025	Analysis -> Program Backlog	
07.11.2025	Program Backlog -> Implementation	Implementation Date = 07.11.2025
21.11.2025	Implementation -> Done	Done liegt nach Releasing (<Closed>-Stage) -> Closed Date = 21.11.2025

Beispiel 2 -- Uebersprungene First-Stage (ART_A-583):

Datum	Statuswechsel	Ergebnis
08.09.2025	Funnel -> Program Backlog	First Date = 08.09.2025 (Program Backlog liegt nach Analysis im Word)
13.10.2025	Program Backlog -> Implementation	Implementation Date = 13.10.2025
02.11.2025	Implementation -> Releasing	Closed Date = 02.11.2025 (Releasing ist <Closed>-Stage)

Beispiel 3 -- Wiedergeoeffnetes Issue (ART_A-2):

Datum	Statuswechsel	Ergebnis
10.01.2025	Funnel -> Implementation	First Date = 10.01.2025 (Fallback: Implementation nach Analysis)
16.01.2025	Implementation -> Done	Done liegt nach Releasing -> vorl. Closed Date
17.01.2025	Done -> Program Backlog	Issue wiedergeoeffnet -- Program Backlog liegt VOR Releasing
17.01.2025	... -> Program Backlog (aktueller Status)	Closed Date = leer (Issue ist aktuell noch offen)

8 Haeufige Fragen und Tipps

F: Ein Issue hat kein First Date -- warum?

A: Das Issue hat weder die -Stage noch eine Stage danach (vor Closed) erreicht. Es ist moeglicherweise direkt aus dem Eingang storniert oder ausschliesslich in Stages vor der -Stage verblieben. In build_reports wird es als 'To Do' gezaehlt.

F: Ein Issue hat kein Closed Date -- warum?

A: Vier moegliche Gruende: (1) Das Issue ist noch offen und hat den Abschlusspunkt im Prozess noch nicht erreicht. (2) Es hat kein First Date -- es war nie in Bearbeitung (z. B. direkt storniert oder von To Do nach Closed gesprungen ohne Entwicklungsarbeit). Ohne First Date wird kein Closed Date gesetzt. (3) Das Issue wurde nach dem Abschluss wieder geoeffnet -- sein aktueller Status liegt vor der -Stage. (4) Die -Stage wurde uebersprungen und es existiert auch keine spaeteren Stage danach in der Workflow-Reihenfolge. Pruefen Sie den aktuellen Status in der IssueTimes.xlsx.

F: Im Log erscheint eine Warnung ueber nicht gemappte Status.

A: Einige Jira-Status im Export sind nicht in der Workflow-Datei definiert. Die Verarbeitung wird trotzdem abgeschlossen. Die Zeit dieser Status wird der letzten bekannten Stage zugerechnet. Wenden Sie sich an Ihren technischen Ansprechpartner, um die Workflow-Datei zu ergaenzen.

F: Die Ausgabedateien werden nicht erstellt.

A: Pruefen Sie, ob der Ausgabeordner existiert und ob Sie Schreibrechte haben. Beachten Sie auch, ob die Dateien moeglicherweise noch in Excel geoeffnet sind -- Excel sperrt Dateien beim Oeffnen, was das Ueberschreiben verhindert.

F: Welche Datei verwende ich fuer build_reports?

A: IssueTimes.xlsx ist die Pflichtdatei fuer alle Metriken ausser dem Cumulative Flow Diagram. CFD.xlsx benoetigen Sie zusaetzlich fuer das CFD. Transitions.xlsx wird von build_reports nicht direkt verwendet, steht aber fuer eigene Analysen bereit.

F: Kann ich die Ausgabedateien manuell bearbeiten?

A: Das sollten Sie vermeiden. build_reports erwartet ein genaues Format. Wenn Sie Daten filtern oder ergaenzen moechten, nutzen Sie die Filter-Funktion in build_reports, anstatt die Quelldateien zu veraendern.

F: Wie oft muss ich transform_data ausfuehren?

A: Immer dann, wenn Sie aktualisierte Daten aus Jira benoetigen. Ein neuer Export (get_data) gefolgt von einer neuen Verarbeitung (transform_data) liefert stets den aktuellen Stand.

F: Was bedeutet 'Carry-forward'?

A: Wenn ein Issue in einen Status wechselt, der nicht in der Workflow-Datei gemappt ist, laeuft die Zeit weiterhin in der letzten bekannten Stage. Es gibt also keinen 'Zeitverlust' -- die Minuten werden immer einer Stage zugeordnet.

9 Glossar

Begriff	Erklärung
Carry-forward	Zeit in einem nicht gemappten Status wird der letzten bekannten Stage zugerechnet -- kein Zeitverlust
CFD	Cumulative Flow Diagram -- zeigt die Entwicklung des Bestands ueber die Zeit aufgeteilt nach Stage
Closed Date	Datum des Abschlusses eines Issues. Gesetzt beim Eintritt in die <Closed>-Stage oder, falls diese nicht existiert, in die <Final>-Stage.
First Date	Datum der ersten aktiven Bearbeitung. Gesetzt beim ersten Eintritt in die <First>-Stage.
Implementation Date	Datum des Entwicklungsbeginns. Gesetzt beim ersten Eintritt in die <InProgress>-Stage.
Issue	Ein Ticket im Ticketsystem (z.B. eine Jira-Karte).
Issuetyp	Kategorie eines Issues, z.B. Feature, Bug, Story, Task.
JSON	Einfaches Textformat fuer strukturierte Daten. Der Jira-Export liegt in diesem Format vor.
Kanonischer Stage-Name	Der erste Name in einer Stage-Zeile der Workflow-Datei. Alle Aliase werden intern auf diesen Namen abgebildet.
Marker	Zeilen in der Workflow-Datei, die eine Stage als Meilenstein auszeichnen: <First>, <InProgress>, <Closed>.
Praefix	Namens-Vorsatz fuer die Ausgabedateien (z.B. 'ART_A' ergibt ART_A_IssueTimes.xlsx).
Stage	Ein logischer Prozessschritt, dem ein oder mehrere Jira-Status zugeordnet sind (z.B. 'Analysis').
Workflow-Datei	Textdatei, die die Stages und Marker eines Projekts beschreibt. Einmalig pro Projekt angelegt.