LineMetrics Offline Sync Tool - Data Monk

```
Version:            0.1 **BETA**
Datum:              30.4.2015
Verantwortlicher:   Thomas Pillmayr <t.pillmayr@linemetrics.com>
```

Data Monk ist ein Tool von LineMetrics, dass OpenSource zur Verfügung gestellt wird und es Kunden erlaubt, Daten aus der LineMetrics Cloud in ein kundenspezifisches lokales Format zu synchronisieren.

Funktionalität

- Frei definierbare zeitliche Synchronisierung (ähnlich Cronjobs)
- Hinterlegen von Meta Attributen zur Erweiterung/Anreicherung der Daten
- Benutzerdefinierte Komprimierung von Daten (Beispielweise: Summe/Durchschnitt pro 15 Minuten)
- Aktuell unterstützte Speicherformate: CSV

Anforderungen

- Java JRE 1.7+
- Internetverbindung
- gültiges LineMetrics Konto mit konfigurierter sicherer API im LineMetrics Cloud Device Manager

TODO

- Abfangen von Fehlern, wenn Sync Vorgang Fehler aufweist
- JDBC Storage Plugin
- grafische Oberfläche zur Konfiguration der Synchronisierungs Vorgänge

Erste Schritte

Es ist vorgesehen in einer späteren Version eine grafische Oberfläche zur Konfiguration der Synchronisierungsvorgänge anzubieten. Aktuell muss die Konfiguration in einem Konfigurationsfile `system.properties` erfolgen.

Aufbau `system.properties`

1. API Zugangsdaten
2. Synchronisierungseinstellungen
3. Aktivierte Synchronisierungsvorgänge
4. Meta Informationen

API Zugangsdaten

```
api.endpoint=http://bapi.linemetrics.com:8002
api.hash=ABCDEFG...
```

`api.endpoint` definiert den API Server Endpunkt und sollte eigentlich immer gleich sein.

`api.hash` ist der eigentliche API Zugang und verifiziert den Zugriff über die API. Den Hash findest du im LineMetrics Cloud Device Manager, wenn du auf deine konfigurierte API Schnittstelle und dann auf den Button "Zugangsdaten" klickst.

Synchronisierungseinstellungen

Die Synchronisierungseinstellungen unterteilen sich in 4 Bereiche:

- Schedulereinstellungen / Wie oft soll synchronisiert werden?
- Welche Daten sollen synchronisiert werden?
- Sollen die Daten vor dem Speichern verarbeitet werden?
- Wie sollen die Daten gespeichert werden?

Schedulereinstellungen

```
# Einstellungen haben immer folgendes Format
# job.[JOB ID].info.[EINSTELLUNG]=[WERT]

job.1.info.scheduler_mask=0 0 8-17 ? * MON-SAT
job.1.info.timezone=Europe/Vienna
job.1.info.batch_size=PT1m
job.1.info.duration=PT1H
```

Alle Einstellungen die zu einem Sync Vorgang gehören, müssen die selbe `JOB ID` beinhalten. Die `JOB ID` ist ein ganzzahliger numerischer Wert, der selbst definiert werden kann und lediglich in der Konfiguration konsistent sein muss.

Mit der Einstellung `scheduler_mask` wird konfiguriert, wie oft die Daten synchronisiert werden sollen.

In der oben angeführten Einstellung wird der Synchronisierungsvorgang Montag-Samstag zwischen 8:00 und 17:00 zu jeder vollen Stunde ausgeführt. Weitere mögliche Einstellungen findest du hier.

`timezone` regelt das Zeitformat der gespeicherten Daten. In der LineMetrics Cloud werden alle Daten im UTC Format gespeichert. Daten werden erst zur Laufzeit (zum Abfragezeitpunkt) in das für den Benutzer passende Format umgewandelt.

`batch_size` legt fest, mit welcher Granularität die Daten aus der API geladen werden sollen. Angaben im ISO_8601 Standard. Mögliche Werte: `PT1M`, `PT1H`, `P1D`

`duration` legt fest, was für ein Zeitraum geladen werden soll. Angaben im ISO_8601.

Aus den Einstellungen lässt sich interpretieren, dass Montag-Samstag zwischen 8:00 und 17:00 Uhr zu jeder vollen Stunde die vergangene Stunde als Minuten Datenpunkte abgeholt wird.

Welche Daten sollen synchronisiert werden?

```
job.1.datastream=123
job.1.datastream=456
```

Pro Zeile kann ein Datenstrom zum Sync Vorgang hinzugefügt werden. Die für das Konto verfügbaren Datenströme können über die Datenstrom Übersichtsseite eruiert werden.

Daten-Verarbeitung

Möglicherweise sind die von der LineMetrics Cloud geladenen Daten noch im falschen Format. Über die Datenverarbeitung besteht die Möglichkeit, diese Daten noch zusätzlich zu verarbeiten/aggregieren.

In der aktuellen Version gibt es ein Plugin zum Aggregieren der Daten von der Minuten Granularität in ein 15 Minuten Datenpaket.

```
job.1.processor.type=com.linemetrics.monk.processor.plugins.compress.Compres
sorPlugin
job.1.processor.compression_mode=SUM
job.1.processor.compression_size=PT15M
```

Mit dem Attribut `compression_mode` wird festgelegt, ob die Summe `SUM` oder der Durchschnitt `AVG` gebildet werden soll.

Mit dem Attribut `compression_size` kann festgelegt werden, wieviele Daten zusammengefasst werden sollen. Angaben wieder im Format ISO_8601. In unserem Fall summieren wir die 1 Minuten-Daten-Pakete von der API zu jeweils 15 Minuten-Pakete.

Daten-Speicherung

Am Ende des Sync Vorganges sollen die Daten persistiert werden. Dies passiert im folgenden Abschnitt

```
job.1.store.1.type=com.linemetrics.monk.store.plugins.csv.StorePlugin
job.1.store.1.csv_number_locale=de_AT
job.1.store.1.csv_file_path=exports/
job.1.store.1.csv_file_template=${job.start:YYYY-mm-dd}.csv
job.1.store.1.csv_header_template=Das ist der Header meiner CSV
job.1.store.1.csv_line_template=${item.start:YYYY-mm-dd
HH:mm:ss};${item.end:YYYY-mm-dd HH:mm:ss};${item.value:0.00}
```

Die Konfiguration folgt dabei folgenden Aufbau:

```
job.[JOB ID].store.[STORE ID].[ATTRIBUTE]=[WERT]
```

Die Store ID kann beliebig vergeben werden und ermöglicht das verwenden von mehreren Persistierungsarten innerhalb eines Jobs. In diesem Beispiel wird als Store ID eine ansteigende nummerische Reihenfolge verwendet, wobei Store 1 die Daten in ein CSV persistiert und Daten bei einer bestehenden Datei anhängt werden. Im nächsten Abschnitt wird die Persistierung über das Prefilled Plugin vorgestellt, welche als zweiter Store mit der ID 2 konfiguriert wird.

Das Attribut `csv_number_locale` legt das Locale für den Export Vorgang fest. Dies ist vor allem für die richtige Formatierung von numerischen Werten wichtig.

Das Attribut `csv_file_path` legt den Speicherort (Ordner) der CSV Datei fest.

Das Attribut `csv_file_template` legt den Dateinamen der CSV Datei fest. **Wichtig:** Bei allen Template Attributen können Platzhalter verwendet werden, die erst während des Sync Vorgangs mit Werten befüllt werden. Platzhalter werden im einem späteren Abschnitt genauer behandelt.

Das Attribut `csv_header_template` legt den Kopf (die ersten Zeilen) der CSV Datei fest, wenn die Datei neu angelegt wird. Falls die Datei, in der die Daten geschrieben werden sollen, bereits existiert, werden die Daten (ohne HEADER) lediglich angehängt.

Das Attribut `csv_line_template` legt das Format fest, wie die einzelnen Datenpunkte in das CSV File geschrieben werden sollen.

Daten-Speicherung mit Prefilled Plugin

Das Prefilled Plugin ermöglicht die Persistierung der Daten ähnlich dem CSV Plugin, allerdings mit dem Unterschied, dass die CSV Datei beim Anlegen bereits mit allen konfigurierten Zeitstempeln gefüllt wird. Die Daten werden dann erst Schritt für Schritt in die Datei eingefügt.

Beispiel Konfiguration für das Prefilled Plugin

```
job.1.store.2.type=com.linemetrics.monk.store.plugins.csv.PrefilledPlugin
job.1.store.2.csv_time_scope=PTD
job.1.store.2.csv_time_slice=PT15M
job.1.store.2.csv_number_locale=de_AT
job.1.store.2.csv_file_path=exports/
job.1.store.2.csv_file_template=${job.start:YYYY-mm-dd}.csv
job.1.store.2.csv_header_template=Das ist der Header meiner CSV
job.1.store.2.csv_empty_line_template=${item.start:YYYY-mm-dd
HH:mm:ss};${item.end:YYYY-mm-dd HH:mm:ss};
job.1.store.2.csv_line_template=${item.start:YYYY-mm-dd
HH:mm:ss};${item.end:YYYY-mm-dd HH:mm:ss};${item.value:0.00}
```

Neben den Attributen des CSV Plugin sind noch folgende weitere Attribute zu konfigurieren:

Das Attribut `csv_time_scope` legt fest, über welchen Zeitraum sich eine einzelne Datei erstreckt. Diese Angabe ist notwendig, da das System nicht automatisch über den Dateinamen den Zeitraum extrahieren kann. Angaben im ISO_8601 Format. P1D = 1 Tag

Das Attribut `csv_time_slice` legt fest, über welchen Zeitraum sich ein einzelner Datenpunkt erstreckt. Angaben im ISO_8601 Format. PT15M = 15 Minute

Das Attribut `csv_empty_line_template` legt das Format fest, wie die leeren Datenpunkte, die beim Anlegen einer neuen Datei für den gesamten Zeitraum geschrieben werden, aussehen sollen. Diese Zeilen werden später durch die tatsächlichen Werte bzw. durch die Konfiguration von `csv_line_template` ersetzt.

Aktivierte Synchronisierungsvorgänge

Es müssten nicht immer zwingend alle Synchronisierungsvorgänge gestartet werden. Durch das Attribut `activated_jobs` kann genau festgelegt werden, welcher Vorgang berücksichtigt und welcher ignoriert werden soll. **Wichtig:** Pro aktiviertem Vorgang eine eigene Zeile. Das Attribut wird nicht überschrieben.

Meta Informationen

Oft macht es Sinn die exportierten Daten mit zusätzlichen semantischen Daten zu versehen, um bspw. Daten später besser mit anderen Stammdaten verknüpfen zu können.

```
#Meta Informationen haben immer folgenden Aufbau
meta.(job oder datastream).[ID].[KEY]=[VALUE]

meta.job.1.data_type=Energy Consumption
meta.datastream.8782.customer_id=1234
```

Hier wird zuerst eine Meta Information für den Synchronisierungsvorgang mit der ID 1 angelegt und dann ein Info für den Datenstrom 8782. Meta Informationen können bei den Templates in Form von Platzhaltern verwendet werden. Platzhalter werden im nächsten

Abschnitt behandelt.

Platzhalter

Arten von Platzhalter

Art	**Beschreibung**	**Formatierung**
Meta	Meta Informationen die über Konfiguration spezifiziert worden sind	NEIN
Job	Daten die im Context des Synchronisationsvorgang stehen	JA
Item	Daten im Context eines einzelnen Datenpunkts	NEIN

Variablen

Art	Variable	Beschreibung	Formattierung
Meta	${meta.[KEY]}	Gibt je nach Context den Wert des Meta Info Schlüssels zurück. Wenn der Schlüssel nicht existiert wird UNDEFINED ausgegeben.	
Job	${job.start.[FORMAT]}	Sync Zeitbereich Startzeitpunkt	Formatierung Zeit
	${job.end.[FORMAT]}	Sync Zeitbereich Endzeitpunkt	Formatierung Zeit
	${job.timezone.[FORMAT]}	Zeitzone	
Item	${item.start.[FORMAT]}	Datenpunkt Zeitbereich Start	Formatierung Zeit
	${item.end.[FORMAT]}	Datenpunkt Zeitbereich Ende	Formatierung Zeit
	${item.min.[FORMAT]}	Datenpunkt Min Wert (falls verfügbar)	Formatierung Zahl
	${item.max.[FORMAT]}	Datenpunkt Max Wert (falls verfügbar)	Formatierung Zahl
	${item.value.[FORMAT]}	Datenpunkt Wert	Formatierung Zahl

Formatierungen

Zeit

Definition laut ISO8601:

```
YYYY                  (z.B. 1997)
YYYY-MM               (z.B. 1997-07)
YYYY-MM-DD            (z.B. 1997-07-16)
YYYY-MM-DDThh:mmTZD   (z.B. 1997-07-16T19:20+01:00)
YYYY-MM-DDThh:mm:ssTZD (z.B. 1997-07-16T19:20:30+01:00)
```

Zahl

```
    #,##            12,675       =>       12,67
00000,00            12,6         =>    00012,60
0.000,##          1212,6         =>    1.212,6
0.000,00          1212,6         =>    1.212,60
    #                2,3         =>        2
```

Vollständige Konfiguration

```
api.endpoint=http://bapi.linemetrics.com:8002
api.hash=ABCDEFG...

job.1.info.scheduler_mask=0 0 8-17 ? * MON-SAT
job.1.info.timezone=Europe/Vienna
job.1.info.batch_size=PT1m
job.1.info.duration=PT1H

job.1.datastream=123
job.1.datastream=456

job.1.processor.type=com.linemetrics.monk.processor.plugins.compress.Compres
sorPlugin
job.1.processor.compression_mode=SUM
job.1.processor.compression_size=PT15M

job.1.store.1.type=com.linemetrics.monk.store.plugins.csv.StorePlugin
job.1.store.1.csv_number_locale=de_AT
job.1.store.1.csv_file_path=exports/
job.1.store.1.csv_file_template=${job.start:YYYY-mm-dd}.csv
job.1.store.1.csv_header_template=Das ist der Header meiner CSV
job.1.store.1.csv_line_template=${item.start:YYYY-mm-dd
HH:mm:ss};${item.end:YYYY-mm-dd HH:mm:ss};${item.value:0.00}

job.1.store.2.type=com.linemetrics.monk.store.plugins.csv.PrefilledPlugin
job.1.store.2.csv_time_scope=P1D
job.1.store.2.csv_time_slice=PT15M
job.1.store.2.csv_number_locale=de_AT
job.1.store.2.csv_file_path=exports/
job.1.store.2.csv_file_template=${job.start:YYYY-mm-dd}.csv
job.1.store.2.csv_header_template=Das ist der Header meiner CSV
job.1.store.2.csv_empty_line_template=${item.start:YYYY-mm-dd
HH:mm:ss};${item.end:YYYY-mm-dd HH:mm:ss};
job.1.store.2.csv_line_template=${item.start:YYYY-mm-dd
HH:mm:ss};${item.end:YYYY-mm-dd HH:mm:ss};${item.value:0.00}

activated_jobs=1

meta.job.1.data_type=Energy Consumption
meta.datastream.8782.customer_id=1234
```

Starten / Ausführen des Programms

Nach erfolgreicher Konfiguration kann das Programm über die `start.cmd` Datei gestartet werden. Sollte das Programm nicht ohnehin über das Terminal gestartet werden, öffnet sich beim Aufruf der Datei ein Terminal Fenster. Solange das Fenster **nicht geschlossen** wird, wird das Programm ausgeführt. Das Programm beinhaltet einen Scheduler, der je nach Konfiguration die Sync Jobs automatisch und periodisch startet. Dieser Scheduler

verhindert es, dass das Programm terminiert. Wird das Programm trotzdem unerwartet beendet, ist das ein Hinweis auf einen Fehler. Bitte senden Sie in diesem Fall den Inhalt der Datei `service.log` an ticket@linemetrics.com.