# Table of Contents

Publisher .................................................................................................................................................................... 3

Logging In ................................................................................................................................................................. 3

Publisher Configuration ............................................................................................................................................ 3
  Create a New Publish List..................................................................................................................................... 4
  Edit a List .............................................................................................................................................................. 7
  Mapping ............................................................................................................................................................... 8
  Dynamic Mapping ................................................................................................................................................ 9
  Output ................................................................................................................................................................ 14
  Run a list ............................................................................................................................................................. 16
  Error Codes ......................................................................................................................................................... 17
  Delete a list ........................................................................................................................................................ 17

Scheduler Configuration .......................................................................................................................................... 20

Mapping Upload ..................................................................................................................................................... 22
  Create a New Mapping Table ............................................................................................................................. 22

Notifications ............................................................................................................................................................ 24

Subscription List .................................................................................................................................................. 27

Scheduler Application .......................................................................................................................................... 28
  Requirements ..................................................................................................................................................... 28
  Desktop application ........................................................................................................................................... 28
  Server application ............................................................................................................................................ 32
Publisher

The Morningstar Marketplace Publisher provides users with direct access to Morningstar Commodity Data. Your applications and tools benefit from the most up-to-date data that Morningstar has to offer. In addition, you can simplify implementation and reduce the amount of time needed to integrate Morningstar data into your tools and applications.

Logging In

To access the Marketplace Publisher, please visit the following link:
http://mp.morningstarcommodity.com/marketplace/login

Once on the page, type in the username and password and click, “Log In”. If you do not have login credentials, please contact your client manager or commoditydata-support@morningstar.com.

Publisher Configuration

The Marketplace Publisher uses a list-based configuration to subscribe to be the relevant data to be generated. The list view is the first page shown when logging into the Publisher. The List View in the Publisher tab allows a user to create, run, copy, rename, and delete Publisher lists.
Create a New Publish List

To create a new Publish List, type in the desired name for the list in the “List Name” field. Please note that the list name must contain only letters, numbers, or an underscore. A Description can also be added, however it is optional.

Once the list is created, a pop-up will appear with list parameters.
Choose Vendor and Feed
Select the Vendor and Feed from the drop down menu provided. Only Vendors and Feeds which are permissioned to the account being used will be shown.

Select a List
“Select a list” allows a user to add an existing list to their new Publisher List.

Filters
Filters can be applied to any feed’s keys in order to narrow down the list of symbols that a user is able to choose. The filter tool helps users find symbols faster.

- Exact Symbol - Finds an exact match for the key that a user enters.
Example: A user is searching for a key that begins with “3EM,” so they type “3EM” in the Filter tool. The tool filters the list of available Keys that begin with “3EM” and then lists them in the Results section. The user can then check the box for all the keys that they would like to add to their new Publish List.

- **Regex Pattern** – Allows a user to create a regular expression that dynamically pulls symbols based off the criteria in the expression. Regular expressions are very useful for adding new futures contracts. The expressions allow a user to receive data for new contracts as they become available without having to add each new contract individually to their Publish List.

Example: If a user wanted to add natural gas futures contracts from CME to their publish list, but didn’t want to have to go back and add each new monthly contract, they could use a regular expression. In Marketplace, CME contracts are listed like “NG0F” for the current January contract. If the user wanted all the natural gas contracts in their Marketplace output, they should use the regular expression below.

```
^NG[0-9]?[0-9][FGHJMNQUVXZ]
```

This expression will pull all the symbols that start with NG then have one or two numbers between 0-9 followed by the contract letter.

**Add Columns**

Multiple data columns (ex. High or Low) can be added to the output. Select from the list of columns that are available for the keys. At least one column must be selected.
Add Meta Data Columns

Meta Data columns provide additional information about the key. These are optional columns.

Preview Publish List

The preview will display the list of keys and columns which you have selected based on the interaction with the various drop downs and check boxes. Click the “Save” button after reviewing the preview.

Once saved, you will be taken to the edit list. Here you can add or delete items to the list. To go back to list view, click “back to list view”

Edit a List

In the List View, to edit a list, click on the list name.

Edit List

A list of rows will appear with the selected symbols. A user can add more symbols to the current list by clicking on “Add to List”. The same dialog pop-up box which was displayed initially, when adding keys and columns will appear. The user can delete symbols by selecting the row and clicking on “Delete from List”. The changes will auto-save while modifications are being made.
Mapping

Mapping allows a user to create custom fields that are not provided in the source feed. These custom fields can be used to help categorize the output. For example, creating a custom field can also help with mapping your company’s custom symbols to Morningstar symbols. This information is usually relatively static (i.e. not changing with each update).

Add a new column by clicking on the “Add Column” button. The following fields will be prompted:

**Column Name**: name of new column
**Column Value**: the value associated with the new column

**From Row**: each row is numbered. Specify the first row which the value should be filled for under the new column

**End Row**: Specify the last row which the value should be filled for under the new column

Once a custom column is created, values can be added or modified by selecting the checkbox above the column name and clicking “Add Value”.

A custom column can be deleted by selecting the checkbox in front of the column name and clicking Delete Column.

Ensure all changes are saved by clicking “Save List”.

**Dynamic Mapping**

Dynamic mapping allows users to create their own mapping logic for Marketplace. The framework is flexible enough to allow for futures mappings and more complex functionality.

**Uses:**
- A constant that needs to be multiplied/divided by a value column (currency conversions - multiply a value by a multiplier/constant)
- Basic math -
- Value to value - same data/value but with a different header/name
- Create a table to allow for ‘v-lookup’ functionality (UI for the user to add the logic)
- Pull meta data field in (e.g. CL4Z - need to pull in start and end date)
Add new column

Click on “Add Column”.

1) Provide a column name
2) Provide a column value. The column value can be an exact number or an arithmetic operator.
3) Select the rows that should include this column
4) Click “Add”

Operators

All operators should start with an equal “=” sign.

= sum (x, y, ...) Add multiple values (substitution can be done for any x, y, ...)
= subtract(x, y, ...) Subtract from x (substitution can be done for any x, y, ...)
= multiply(x, y, ...) Multiply multiple values (substitution can be done for any x, y, ...)
= divide (x, y) Divide y from x (substitution can be done for any x, y)
= if(<condition>, w, z) IF condition is met when w is executed or else z is executed. A user can substitute can be done for any x, y, w, z) w and z can be any function without “=” predicate. x and y cannot be execution functions. Conditional can be one or a combination of == (equality), != (not equality), > (greater than), < (less than), >= (greater than or equality), <= (less than or equality)
**Simple tests**

Canned Functions Simple Tests

1) \( =\text{sum}(2,${\text{PUBVAL}}}) \)
2) \( =\text{subtract}({\text{PUBVAL}},1) \)
3) \( =\text{multiply}({\text{PUBVAL}},2) \)
4) \( =\text{divide}({\text{PUBVAL}},.5) \)
5) \( =\text{divide}({\text{PUBVAL}},0) \) Failed. Resulted in empty file. Expected behavior should have something in that column such as #Div/0! (like Excel), #ERR, or #NA.
6) \( =\text{if}({\text{PUBVAL}}=1.3567,1,2) \)

**Concatenate**

Concatenate allows a user to join multiple strings of text into one string.

\[=\text{concatenate}("Hello","bye")\]

output:

Hellobye

\[=\text{concatenate}("NG","_","2007")\]

output:

NG 2007

**Average**

Average allows a user to average constants and published values.

\[=\text{average}(2,3,1)\]

output:

2

\[=\text{average}({\text{PUBVAL}},{\text{PUBVAL}},3.33)\]

output:

CAD,Rate,1.3567,2.014466666666667

**Substitution**

Replaces existing text with new text in a text string

\[=\text{substitute}({\text{PUBFEED}}, "Fx", "ForeignExch")\]
output:
in the publisher feed name, replace “Fx” with “ForeignExch”.

**IF-THEN-ELSE statements**

=if(<condition>,<true action>, <fail action>)

=if(date($\{PUBDATE\}) < 08/01/2013, 1, 0)

output:
If the published date is before 8/1/2013 a value of 1 will be shown in the output. If the published date is after 8/1/2013 a 0 will be shown.

=if(date($\{PUBDATE\}) < 08/01/2013, "Old", "New") Success

output:
If the published date is before 8/1/2013 the word “Old” will be shown in the output. If the published date is after 8/1/2013 the word “New” will be shown.

**Regular expression substitution**

=re_substitute(input_string, match_pattern, result_pattern)

Example:
=re_substitute("N7_2009F", "([A-Z0-9])_[0-9][0-9][0-9]([0-9])([FGHJ][K])", "$1$3$2")

Received expected result of N7F9.

**Date Addition**

Add either day (d), month (m), year (y), or quarter (q) to the current provided date.

Below will add 1 day to the MCD delivery end date:

=dateadd("d", 1, Date($\{MCD_DeliveryEndDate\}))

Below will add 6 months to current published date:

=dateadd("m", 6, Date($\{PUBDATE\}))
Date Format

Takes the standard published date and converts it to a string value in the specified configuration format, otherwise the same output format will be used.

=format({KEY},"format style")
=format(${mcd_deliverystartdate},"yyyy-MM-dd")
=format(dateadd("d", 6, Date(${PUBDATE})), "yyyy-dd-MM")

Supported formats:
"yyyy-MM-dd'T'HH:mm:ssZ",
"yyyy-MM-dd'T'HH:mm:ss.SSSZ",
"yyyy-MM-dd",
"MM/dd/yyyy",
"yyyyMMdd"

Vlookup

‘Vlookup’ in Marketplace works similar to vlookup in Microsoft Excel. Specify a column in the table and ‘vlookup’ will provide the information for all the subsequent columns. For example, Column 1 contains all the Marketplace Keys. Column 2 contains the alternative values for each Marketplace key.

=vlookup(${Keys},"Mapping File Name")

getKey

g getKey allows a user to retrieve the key name.

=getKey("string keyname")
=getKey("Currency") => "JPY"
=getKey("code_id") => 1352
Calendar
Determine if the date is a weekday, weekend, or holiday. This function will dynamically populate a field with one of the three options (“Weekend”, “Weekday”, “Holiday”) based on the PUBDATE. If a given day is both a weekday/weekend and a holiday, the output will display “Holiday”. As a default, the holiday calendar is currently using NERC holidays.

=calendar(<date string>)
=calendar(${PUBDATE})

Output

The Output tab allows a user to prioritize and categorize the order of the list they will output. Select the columns to be displayed in order from left to right by selecting the column name and clicking on “Add”. To remove a column, click on “Remove”. A user can also move a column up or down by selecting the column and clicking on the “up” or “down” buttons.

Choose Output Style:
- Options for CSV or CSV – no header. (Header will be omitted)
- Delimiter – choose from comma, pipe, colon separators
- Date Format – select the desired date format from the drop down
- Number format – select from decimal or comma denotation
The table below discusses the output data point fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Type of delta file - delete, insert, update</td>
</tr>
<tr>
<td>TSID</td>
<td>The time series ID for the given key values for the line in the output. This is for tracking in the time series database only.</td>
</tr>
<tr>
<td>Keys</td>
<td>The value for each key (e.g. NG5F)</td>
</tr>
<tr>
<td>KEY_NAMES</td>
<td>The name for each key (e.g. PNodeID)</td>
</tr>
<tr>
<td>COL_NAMES</td>
<td>All the available column values for the feed</td>
</tr>
<tr>
<td>NVInsertionDT</td>
<td>All new data insertion date and time</td>
</tr>
<tr>
<td>NewValues</td>
<td>All new data values</td>
</tr>
<tr>
<td>PVInsertionDT</td>
<td>Previous data insertion date and time before a corrections</td>
</tr>
<tr>
<td>PreviousValues</td>
<td>Previous data values before a correction</td>
</tr>
<tr>
<td>PUBREASON</td>
<td>Reason the point was published</td>
</tr>
<tr>
<td>PUBCOL</td>
<td>The name of the column for the PUBVAL ie: Settlement Price</td>
</tr>
<tr>
<td>PUBVAL</td>
<td>The publication value</td>
</tr>
<tr>
<td>PUBDATE</td>
<td>The publication date</td>
</tr>
<tr>
<td>PUBMD</td>
<td>The publication metadata requested</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PUBCMD</td>
<td>The publication company metadata requested</td>
</tr>
<tr>
<td>PUBLIST</td>
<td>Name of the list that contained the key or regular expression</td>
</tr>
<tr>
<td>PUBFEED</td>
<td>The feed name for a key</td>
</tr>
<tr>
<td>MATCHPATTERN</td>
<td>If the PUBREASON was because of a pattern/regular expression, this will contain that expression</td>
</tr>
<tr>
<td>PUBSOURCE</td>
<td>The name of the original source of the data</td>
</tr>
<tr>
<td>PUBDATE_TZ</td>
<td>Published date time zone</td>
</tr>
</tbody>
</table>

“Show Preview”- will display a preview of the options you have selected with the header names.- actual data is not shown at this time.

Example:, Keys, KEY_NAMES, COL_NAMES, PUBDATE, Type

Ensure all changes are saved by clicking “Save Output”.

**Run a list**
To run a current list, click on the “Run” button.
Sample output:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#CIRCREF</td>
<td>circular reference has to be detected</td>
</tr>
<tr>
<td>#BADREF</td>
<td>bad reference or reference was not found</td>
</tr>
<tr>
<td>#DIVBY0</td>
<td>arithmetic divide by 0 detected</td>
</tr>
<tr>
<td>#EXPR</td>
<td>unknown expression provided</td>
</tr>
<tr>
<td>#DATE</td>
<td>unknown date format</td>
</tr>
<tr>
<td>#CONST</td>
<td>unknown type provided</td>
</tr>
<tr>
<td>#COND</td>
<td>invalid condition detected</td>
</tr>
<tr>
<td>#FORMAT</td>
<td>invalid value format</td>
</tr>
<tr>
<td>#SYNTAX</td>
<td>syntax (pre compile) error detected</td>
</tr>
<tr>
<td>#NORESULT</td>
<td>no result from getkey</td>
</tr>
<tr>
<td>#BADENTRY</td>
<td>bad table or key for vlookup</td>
</tr>
<tr>
<td>#DBASE</td>
<td>no result from vlookup</td>
</tr>
<tr>
<td>#ERROR</td>
<td>generic error detected</td>
</tr>
</tbody>
</table>

Delete a list

To delete a current list, click on the “Delete” button.
Set optional parameters

If you have configured a subscription list via the Marketplace web interface, you can use REST calls to view the subscription list configuration or to pull the data associated with the subscription list.

List name

To examine a subscription list's details, use the list resource. The list name is the name of the subscription list that was initially created.

Example:
https://mp.morningstarcommodity.com/lds/lists/{listName}

There are no optional query parameters for the list resource.

List Content

The “list content” resource allows you to pull the data associated with the subscription list.

Example:
http://mp.morningstarcommodity.com/lds/lists/{listName}/content

Optional Parameters

When pulling the list, you can specify a time range of points so that you can see any new points or corrections that have arrived during that window of time. After the “?”, add the parameter required. Add additional parameters by separating them with the “&” symbol.

<table>
<thead>
<tr>
<th>Optional Query Parameters for List Content Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>fromDateTime</td>
</tr>
<tr>
<td>toDateTime</td>
</tr>
<tr>
<td>delim</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>daysBack</td>
</tr>
<tr>
<td>backDateTime</td>
</tr>
</tbody>
</table>

Example:
http://mp.morningstarcommodity.com/lds/lists/{listName}/content?fromDateTime=2012-10-20&toDateTime=2012-10-25
Scheduler Configuration

Create a new schedule by providing a schedule name and then click “Create”.

A schedule template will appear which will allow you to set Machine ID, File names, directories, check frequency, and time zones.
- **Machine ID** - Specify the unique name for the profile where the data is to be sent. This must match the Machine ID that was created during the Publisher Scheduler Application installation.

- **Add One of more lists** - Click the "+" button to add the lists to be run in this schedule. One or multiple lists can be selected.

- **Save File Name** - Specify the file name along with the date and file type.

- **Directory to save file** - Each list will be saved individually to the specified location

- **Check Data** - Only check for new data from the last scheduled run, or offset the schedule by a certain number of days

- **Frequency** – Specify which days of the week the schedule should run.

- **Time** - Select which hours the schedule should run.
- **Increments** - Set increments to schedule a re-occurring run.

  - **Time Zone** – Select the time zone for the schedule.

**Mapping Upload**

Mapping Upload allows users to upload their own proprietary mappings to Marketplace. By utilizing ‘vlookup’-type functionality, users can more easily map Marketplace keys to their own proprietary keys.

```
vlookup(\{KEYS\},"StartMap")
```

Create a New Mapping Table

To create a new Mapping Table, type in the desired name for the mapping table in the “Mapping Name” field. Next, click the Choose File button in the File Location section. This button will allow you to choose a file for upload from your computer. After selecting the desired file, click the Upload button and the mapping file will be uploaded to Marketplace. Please see below for Mapping File Specifications.

After uploading the new Mapping Table, the mapping will be available for use in the Mapping section of a list. In order to utilize the mapping file, navigate to the Mapping section and click the “Add Column” button. For “Enter Column Name,” select a name for the column that will contain the mapped symbols. For “Enter Column Value,” enter the ‘vlookup’ syntax:

```
vlookup(\{KEYS\},"Name of Mapping File")
```

For example: `=vlookup(\{KEYS\},"ERCOT Lmps")`
Then, select the rows that the mapping should apply to and click “Add”. Once the column for the mapped symbols has been created, it can be added to the output document in the Output tab. The column will be available in the Choose Output Data Points section.

**Mapping File Specifications**

Mapping Table files can only be in .csv or .txt format. The file must contain the Marketplace keys in the first column of the upload file and the corresponding proprietary keys/symbols must be in the second column.

For example: In the image below you will see that Column A contains Marketplace symbols and Column B contains a user’s proprietary symbols. The Marketplace symbols in the mapping file must be specific symbols and not a regular expression. Marketplace mappings can only map one Marketplace key to one proprietary symbol.
Changing a Mapping File

If a mapping file needs to be changed or updated, the original file must be deleted and a new replacement file must be uploaded.

Click “Download” to download the current mapping file. Click “Delete” to delete the current mapping file.

Notifications

Notifications allow a user to check and see when the data within a particular Publisher list last updated.

The Notifications tab contains the Publish Lists found under the Publisher tab. In order to see when the data in a specific Publisher List last updated, select the check box beside the appropriate List Name and click “Run Selected Lists”. After Marketplace finishes checking on the status of the data, a link indicating the date and time will appear beside the selected List Name in the Created section of the page. For each new update of the notification report, a new report will need to be created. The notification tool does not automatically refresh the report.

After a notification has been created, click on the blue hyperlink in the Created section to see the details of the notification. The notification will contain the Data Availability Report as of the time that the notification was created. The two sections of the report will be Data Available and Data Not Available. In both sections, the list of information will include the Source of the data, Marketplace
Feed name, Marketplace Keys, Frequency of the data for the corresponding Marketplace symbol/key, and the time the data was last updated.

If the data for a particular symbol updated between 12:00:00 AM and the time the report was created of the same day, then the row for that symbol will be located in the Data Available section. If no data was received for a symbol, then the row for that symbol will remain in the Data Not Available section.

For Example: Since the PJM Keys below updated between June 24, 2014 12:00:00 AM and when the report was run June 24, 2014 at 9:59:00 AM, the PJM Keys are located in the Data Available section. A more specific time stamp of when each PJM Key updated can be found in the last update time. Since all the symbols in the PJM_LMP list updated, there are no rows in the Data Not Available section.
Subscription List

The subscription List tab provides a list of all subscriptions that the current logged in user is subscribed to. The information is separated by Feed Name, Data source, and Type.

<table>
<thead>
<tr>
<th>Feed Name</th>
<th>Data source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADW_Taq_data</td>
<td>ADW</td>
<td>Site</td>
</tr>
<tr>
<td>ADW_Taq_area</td>
<td>ADW</td>
<td>Site</td>
</tr>
<tr>
<td>ADW_Taq_map</td>
<td>ADW</td>
<td>Site</td>
</tr>
<tr>
<td>AESO_CurrentSupplyDemandBio</td>
<td>Aeso</td>
<td>Site</td>
</tr>
<tr>
<td>AESO_CurrentSupplyDemandCoal</td>
<td>Aeso</td>
<td>Site</td>
</tr>
</tbody>
</table>
Scheduler Application

The Marketplace Scheduler is part of the Marketplace Publisher.

The Scheduler User Interface application adds a Morningstar icon to the system tray of the user’s desktop running Windows, Mac, or Linux. The user can access the Scheduler UI by clicking on the icon in the system tray and choosing the "Show Scheduler" menu option. After the UI is initialized it creates a Scheduler and displays the status of the jobs in the Scheduler window.

There are two other components to the Scheduler which can be run on the server side.

1. Downloader
2. Scheduler

Once the schedules are created in the Marketplace, the Downloader component can be used to pull the publisher list or “publist” to the server. The Downloader only downloads a specified publist. Publists are created in Marketplace.

The Scheduler downloads schedules that were created in the Marketplace Scheduler Configuration and generates a job for each schedule. When a trigger is executed based on the schedule criteria set in the Marketplace, the application calls the Downloader for each publist.

Requirements

- Java 7 update 6 or later
- The Scheduler file
  - v0.0.4-SNAPSHOT.zip (with executable jar)

Desktop application

Scheduler User Interface

The Scheduler is a Windows application that loads existing schedules from Marketplace, runs the schedules accordingly and downloads Publisher lists specified in the schedules.

To install the Scheduler:
1. Download the zip file. Inside it you will find a pubclient-<version> directory. (i.e. pubclient-0.0.4-SNAPSHOT)
2. Open the zip file in Windows Explorer.
3. Extract all files (or copy the pubclient-<version> directory) to a directory of your choice.
4. Open the extracted directory.
5. Double click the jar file.

The application can also run from the command line to provide optional command line arguments. Open a command prompt and cd to the extracted directory. The following shows an example running the app and connecting to a specific host via the --h command line option.

```
C:\>cd pubclient-0.0.4-SNAPSHOT
C:\pubclient-0.0.4-SNAPSHOT>java -jar pubclient-0.0.4-SNAPSHOT.jar --h=http://localhost:8080
```

The host is determined in the following order:

1. As a command line option. (--h=http://somehost:port)
2. As a JVM parameter. (-Dscheduler.host=http://somehost:port)
3. In the config file. (host=http://somehost:port)
4. Undefined. When no host is defined as a command line option, JVM parameter or in the config file the default is http://mp.morningstarcommodity.com
Installation

**Morningstar® Marketplace™
Configuration Scheduler**

- **Username** – Utilize the same username and password combination as the one used to log into the Marketplace Publisher to create the Publish Lists and Schedules.
- **Password** - Utilize the same username and password combination as the one used to log into the Marketplace Publisher to create the Publish Lists and Schedules.
- **Machine ID** – Create a unique Machine ID which will be used as an identifier for each schedule.

**Scheduler User Interface**

The Scheduler UI displays the schedules created via the Marketplace Scheduler Configuration browser page. The Schedule Name and Description columns match the information entered when creating the schedule in Marketplace. The Status/Next Run column shows the next scheduled run or the current status of a schedule: Running or Error. The right hand side pane displays the Publisher List names that are included in the schedule. The File Information section displays the Directory and File Format. The Schedule Details section displays the frequency selection for each schedule.

To modify any of the schedule inputs users need to access the Marketplace Scheduler Configuration page. The application Refresh button allows users to refresh the list of schedules or the information
associated with each schedule once it has been modified in the Marketplace Scheduler Configuration page.

Logging

In the Scheduler UI some messages are displayed as a notification near the system tray. All log messages are written to a log file regardless which application is ran (UI, Scheduler or Downloader). The log files can be found in: \<UserHome>/Marketplace/Scheduler.
**Server application**

**Downloader**

The Downloader can be run on the server as a standalone Java application:

```java
com.morningstar.publisher.dl.Downloader
```

The Downloader requires the following arguments:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cfgFile</td>
<td>The configurations file with username, password and host. The host is optional and can be specified as a JVM parameter (-Dscheduler.host=<a href="https://host:port">https://host:port</a>) or undefined to use as default.</td>
</tr>
<tr>
<td>saveToDir</td>
<td>The directory where downloaded publist files are saved to</td>
</tr>
<tr>
<td>datePattern</td>
<td>Java SimpleDatePattern to append to file names. (yyyyMMddHHmmss, MMddyyyy, yyyy-MM-dd)</td>
</tr>
<tr>
<td>fromDate</td>
<td>The starting date for new data. Format must be yyyyMMdd</td>
</tr>
<tr>
<td>pubListNames</td>
<td>List of all publist names to be downloaded, separated by space (pubList1 pubList2 pubList3)</td>
</tr>
</tbody>
</table>

Example java command line:

```java
java -cp Pubclient-0.0.3-SNAPSHOT-jar-with-dependencies.jar com.morningstar.publisher.dl.Downloader scheduler.cfg F:\temp\mpdownloads yyyyMMddHHmmss 2013-07-01 Y_List
```

**Scheduler**

The Scheduler can be run on the server as a standalone Java application:

```java
com.morningstar.publisher.sched.PubScheduler
```

Note: The JavaFX library must be referenced on the classpath. Jar files in the library directory need to be included on the classpath.

Example Java command line:

```java
java -cp F:\java\jdk1.7.0_21\jre\lib\jfxrt.jar;Pubclient-0.0.3-SNAPSHOT-jar-with-dependencies.jar com.morningstar.publisher.sched.PubScheduler scheduler.cfg
```

Example of Jar file included in classpath:

```java
java -cp `ls lib/*.jar | tr \n \;`;"pubclient-0.0.4.jar
```
The `ls lib/*.jar | tr \n \;` adds all the jar files in the lib directory to the classpath (ie. `-cp`)

The host is determined in the following order:

1. As a JVM parameter. `-Dscheduler.host=http://somehost:port`
2. In the config file. `host=http://somehost:port`
3. Undefined. When no host is defined as a command line option, JVM parameter or in the config file the default is `http://mp.morningstarcommodity.com`

**Configuration file**

All three components (User Interface, Downloader, and Scheduler) use a configuration file for reading the username, password and machine id.

The configuration file should contain the following properties:

- **username** - The user's login name for Marketplace
- **password** - The user's password. May be un-encrypted. Will be encrypted the first time one of the applications reads it.
- **encrypted** - A Boolean flag stated if the password is encrypted or not. (true | false)
- **machine_id** - The ID of the machine the application is to be run on. (i.e. dev01, prod02)
- **host** - (Optional) Specifies the Marketplace host. If not defined then the default `https://mp.morningstarcommodity.com` is used unless specified as a JVM or command line option.

The configurations file example below will only download schedules for `john.doe@globe.com` with the machine ID “dev01”:

```plaintext
username=john.doe@mycompany.com
password=FiMNBF6MLMB9d/5ba07onQUtGFBUgIgC4Uq06aBU7U=
encrypted=true
machine_id=dev01
```