

# Operating Instructions

---

**Fronius Push Service**

**Fronius Push Service**

**DE** | Bedienungsanleitung

**EN** | Operating Instructions





# Inhaltsverzeichnis

Bedienungsanleitung .....	4
Allgemeines .....	4
Übersicht .....	5
Details .....	7
Allgemeines .....	7
Protokolle .....	7
Formate .....	7
SunSpec Datalogger v1.0b - inverter float model .....	9
SunSpec Datalogger v1.2 - meter model .....	10
SolarAPI v1 - CurrentData - Inverter .....	11
SolarAPI v1 - CurrentData - SensorCard .....	13
SolarAPI v1 - CurrentData - StringControl .....	14
SolarAPI v1 - CurrentData - Meter .....	15
SolarAPI v1 - CurrentData - Storages .....	17
SolarAPI v1 - CurrentData - Powerflow .....	20
SolarAPI v1 - Logdata - Data .....	21
SolarAPI v1 - Logdata - Errors and Events .....	23
Datamanager IO States .....	25
Demo Content .....	27
Intervalle .....	28

# Bedienungsanleitung

---

## Allgemeines

Mit Hilfe der Funktion Push Service können Aktuell- und Log-Daten des Wechselrichters in unterschiedlichen Formaten oder mit unterschiedlichen Protokollen auf einen externen Server exportiert werden. Voraussetzung dafür ist ein Fronius Datamanager oder Fronius Datamanager 2.0.

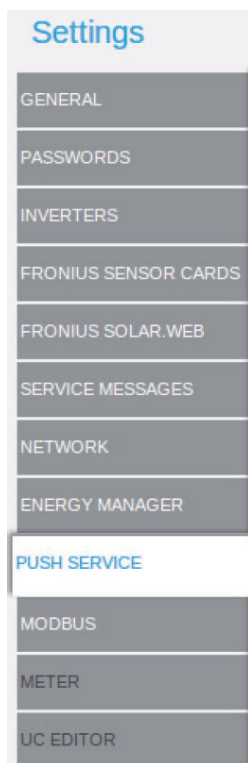
Formate definieren, wie die Informationen gesendet werden.

Protokolle definieren die Art der Datenübertragung.

Werksseitig sind keine Jobs vorgegeben.

Mit dieser Funktion können max. 10 Jobs installiert werden. Für jeden einzelnen Job können unterschiedliche Protokolle, Formate oder externe Server verwendet werden.

Die Funktion Push Service ist auf der Website des Fronius Datamanagers im Menü Einstellungen zu finden:



## Push Service

☐ ✓ (1)    ☐ ✕ (2)

+ Hinzufügen (3)
▼ **New FTP Service 0** (5) (4) Status: ---

Bezeichnung:

Datenformat:  per (6)

Intervall:  ☒ aktiviert

---

Server:Port:

Upload  
Dateiname:  (7)

☐ Anmeldung:

---

☒ Proxy

Server:Port:  (8)

Benutzer:

Passwort:

---

(9)

(6)

(7)

(8)

- (1) Schaltfläche Übernehmen / Speichern  
speichert die Konfiguration und startet einen sofortigen Testlauf aller installierten und aktivierten Funktionen
- (2) Schaltfläche Abbrechen / Eingaben verwerfen  
verwirft sämtliche Änderungen und lädt die ursprüngliche Konfiguration
- (3) Schaltfläche Hinzufügen  
Klicken der Schaltfläche fügt einen neuen Push Service Job hinzu. Der neue Job wird durch Klicken der Schaltfläche 'Übernehmen / Speichern' (1) gespeichert.
- (4) Status  
zeigt den aktuellen Stand des jeweiligen Push Service Jobs an;  
Bei einem Fehler können detaillierte Informationen angezeigt werden, indem der Mauszeiger über den Ursachen-Text gebracht wird. Die Detailinformationen werden nur in Englisch angezeigt.
- (5) Angezeigter Name des Push Service Jobs  
Der Name identifiziert den Push Service Job und darf nur einmal vorkommen.
- (6) Bereich zur Eingabe der allgemeinen Daten:  
Bezeichnung (Name des Push Service Jobs)  
Dateiformat  
Protokolltype (FTP upload / HTTP POST)  
Intervall  
Aktivierungsstatus

- (7) Bereich zur Eingabe der Zieldaten:  
 Server Port  
 Upload Dateiname  
 Anmeldung (Benutzer / Passwort)
- (8) Bereich zur Eingabe der Proxy-Daten:  
 Server Port  
 Benutzer  
 Passwort
- (9) Schaltfläche Löschen  
 Klicken der Schaltfläche löscht den ausgewählten Push Service Job

Sind mehrere Push Service Jobs installiert, kann immer nur ein Job geöffnet und verändert werden:

**Push Service**

☐ ✓ ☐ ✕

+ Add

▶ SunSpec v1 Inverter FTP Intern	State: Connection is not possible
▶ SunSpec v1 Inverter HTTP	State: Successful 2 minutes ago
▶ SolarApi Inverter HTTP	State: Successful 2 minutes ago
▶ SolarApi SensorCard HTTP	State: Successful 2 minutes ago
▶ SolarApi StringControl HTTP	State: Successful 2 minutes ago
▶ IO States	State: Successful 2 minutes ago
▶ SolarApi Logdata HTTP	State: Successful 37 minutes ago
▶ SolarAPI Error HTTP	State: Successful 7 minutes ago
▶ New FTP Service 8	State: disabled
▼ <b>New FTP Service 9</b>	State: Connection is not possible

Name:

Data format:  by

Interval:  ☒ activated

---

Server.Port:

Upload file name:

☒ Login:

User:

Password:

---

☐ Proxy

---

Couldn't connect to server

Liste mit mehreren Push Service Jobs, ein Job ist geöffnet und kann verändert werden; bei einem anderen Job wird eine Fehlerbeschreibung angezeigt, wenn der Mauszeiger über den Ursachen-Text gebracht wird.

## Allgemeines

Die Kommunikation mit dem Fronius Push Service kann aktuell nur durch das Erstellen einer HTTP-Anfrage an eine bestimmte URL erfolgen.

### Push Service

+ Add

New FTP Service 0 State: —

Name:

Data format:  by

Interval:  ☒ activated

Server:Port:

Upload file name:

☒ Login:  
User:   
Password:

☒ Proxy:  
Server:Port:   
User:   
Password:

User Interface auf der Web-Seite des Fronius Datamanagers

## Protokolle

Aktuell werden die Protokolle HTTP-POST und FTP-Upload (passiv oder aktiv) unterstützt.

### Proxy

Die volle Proxy-Unterstützung (mit oder ohne Referenzen) wurde implementiert.  
Ausnahme: Verwendung von Zertifikaten

### Login-Authentifizierung

Unterstützt werden folgende Authentifizierungen:

- Basic
- Digest

Das Fronius Push Service überprüft automatisch, welche Authentifikation auszuwählen ist.

## Formate

Aktuell werden folgende Formate unterstützt:

- SunSpec Datalogger v1.0b - inverter float model
- SunSpec Datalogger v1.2 - meter model
- SolarAPI v1 - CurrentData - Inverter
- SolarAPI v1 - CurrentData - SensorCard
- SolarAPI v1 - CurrentData - StringControl
- SolarAPI v1 - CurrentData - Meter
- SolarAPI v1 - Logdata - Data
- SolarAPI v1 - Logdata - Errors and Events
- Datamanager IO States
- Demo Content



## SunSpec Data-logger v1.0b - inverter float model

XML basierendes Datenpaket  
SunSpec Modelle 1 und (111, 112 oder 113)  
weitere Informationen unter:  
<http://sunspec.org>

### Beispiel für exportierte Inhalte

```
<SunSpecData v="1">
  <d man="Fronius" mod="Fronius_Galvo_2.0-1" sn="24251001236580052" t="2014-07-30T13:53:51+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius Galvo 2.0-1</p>
      <p id="Vr">0.1.14.0</p>
      <p id="SN">24251001236580052</p> *
      <p id="DA">16</p>
    </m>
    <m id="111">
      <p id="W">2011.000000</p>
      <p id="Wh">42121.003906</p>
      <p id="St">4</p>
    </m>
  </d>
  <d man="Fronius" mod="Fronius_Symo_6.7-3-M" sn="24071001231460050" t="2014-07-30T13:53:51+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius Symo 6.7-3-M</p>
      <p id="Vr">0.2.4.1</p>
      <p id="SN">24071001231460050</p> *
      <p id="DA">36</p>
    </m>
    <m id="113">
      <p id="W">6519.000000</p>
      <p id="Wh">6440126.500000</p>
      <p id="St">4</p>
    </m>
  </d>
</SunSpecData>
```

\* Die Geräte-Seriennummern (SN) werden bei Fronius Symo, Primo und Galvo erst ab Seriennummer 2713xxxx angezeigt. Bei älteren Geräten wird der Produktmatrix-Code (PMC) des Recerbo angezeigt.

**SunSpec Data-  
logger v1.2 - me-  
ter model**

XML basierendes Datenpaket  
SunSpec Modelle 1 und 213  
weitere Informationen unter:  
<http://sunspec.org>

Beispiel für exportierte Inhalte

```
<SunSpecData v="1.2">
  <d man="Fronius" mod="Fronius_SmartMeter" sn="14300002" t="2014-07-30T14:16:32+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius SmartMeter</p>
      <p id="SN">14300002</p>
      <p id="DA">1</p>
    </m>
    <m id="213">
      <p id="W">-840.310000</p>
      <p id="Hz">50.000000</p>
      <p id="WphA">-265.760000</p>
      <p id="WphB">-297.450000</p>
      <p id="WphC">-277.100000</p>
      <p id="AphA">1.443000</p>
      <p id="AphB">1.524000</p>
      <p id="AphC">1.431000</p>
      <p id="PhVphA">229.700000</p>
      <p id="PhVphB">232.300000</p>
      <p id="PhVphC">232.300000</p>
    </m>
  </d>
  <d man="Fronius" mod="Fronius_SmartMeter" sn="14300001" t="2014-07-30T14:16:32+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius SmartMeter</p>
      <p id="SN">14300001</p>
      <p id="DA">2</p>
    </m>
    <m id="213">
      <p id="W">-238.380000</p>
      <p id="Hz">50.000000</p>
      <p id="WphA">-238.380000</p>
      <p id="WphB">-0.000000</p>
      <p id="WphC">-0.000000</p>
      <p id="AphA">1.047000</p>
      <p id="AphB">0.013000</p>
      <p id="AphC">0.023000</p>
      <p id="PhVphA">229.300000</p>
      <p id="PhVphB">106.200000</p>
      <p id="PhVphC">106.100000</p>
    </m>
  </d>
</SunSpecData>
```

## SolarAPI v1 - CurrentData - Inverter

JSON basierendes Datenpaket, ähnlich einer Web-Anfrage  
weitere Informationen unter:  
<http://www.fronius.com/de/solarenergie/produkte>  
Anlagen-Monitoring, Offene Schnittstellen, Fronius Solar API (JSON)

Beispiel für exportierte Inhalte

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T13:59:19+02:00",
    "RequestArguments" : {
      "Query" : "Inverter",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "PAC" : {
      "Unit" : "W",
      "Values" : {
        "16" : 2007,
        "36" : 6465,
        "66" : 1945,
        "78" : 3012,
        "83" : 2980,
        "84" : 19193,
        "85" : 18559,
        "87" : 8082,
        "97" : 2534
      }
    },
    "DAY_ENERGY" : {
      "Unit" : "Wh",
      "Values" : {
        "16" : 5027,
        "36" : 700,
        "66" : 8673,
        "78" : 13428,
        "83" : 13014,
        "84" : 21233,
        "85" : 86500,
        "87" : 35618,
        "97" : 10832
      }
    },
    "YEAR_ENERGY" : {
      "Unit" : "Wh",
      "Values" : {
        "16" : 41950,
        "36" : 5004877,

```

```
        "66" : 2088417,
        "78" : 2330199,
        "83" : 2255519,
        "84" : 15047895,
        "85" : 15048166,
        "87" : 6164320,
        "97" : 1879060
    },
    "TOTAL_ENERGY" : {
        "Unit" : "Wh",
        "Values" : {
            "16" : 42305,
            "36" : 6440730,
            "66" : 5817456,
            "78" : 9039449,
            "83" : 8762867,
            "84" : 58365852,
            "85" : 58408104,
            "87" : 23929897,
            "97" : 7285498
        }
    }
}
```

Beispiel für exportierte Inhalte

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:01:08+02:00",
    "RequestArguments" : {
      "DataCollection" : "NowSensorData",
      "DeviceClass" : "SensorCard",
      "DeviceId" : "All",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "0" : {
      "0" : {
        "Value" : 29,
        "Unit" : "\u00B0C"
      },
      "1" : {
        "Value" : 27,
        "Unit" : "\u00B0C"
      },
      "2" : {
        "Value" : 0,
        "Unit" : "W/m\u00B2"
      },
      "3" : {
        "Value" : 0,
        "Unit" : "km/h"
      },
      "4" : {
        "Value" : 0,
        "Unit" : "km/h"
      },
      "5" : {
        "Value" : 0,
        "Unit" : "A"
      }
    }
  }
}
```

**SolarAPI v1 -  
CurrentData -  
StringControl**

JSON basierendes Datenpaket, ähnlich einer Web-Anfrage  
weitere Informationen unter:  
<http://www.fronius.com/de/solarenergie/produkte>  
Anlagen-Monitoring, Offene Schnittstellen, Fronius Solar API (JSON)

Beispiel für exportierte Inhalte

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:02:11+02:00",
    "RequestArguments" : {
      "DataCollection" : "NowStringControlData",
      "DeviceClass" : "StringControl",
      "DeviceId" : "2",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "2" : {
      "1" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "2" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "3" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "4" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "5" : {
        "Value" : 0,
        "Unit" : "A"
      }
    }
  }
}
```

Beispiel für exportierte Inhalte

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:17:28+02:00",
    "RequestArguments" : {
      "Query" : "Meter",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "14300002" : {
      "TimeStamp" : 1406722647,
      "Enable" : 1,
      "Visible" : 1,
      "PowerReal_P_Sum" : -828.26,
      "Meter_Location_Current" : 0,
      "PowerReal_P_Phase_1" : -269.75,
      "PowerReal_P_Phase_2" : -274.79,
      "PowerReal_P_Phase_3" : -283.72,
      "PowerReactive_Q_Sum" : 474,
      "PowerReactive_Q_Phase_1" : 163.31,
      "PowerReactive_Q_Phase_2" : 154.32,
      "PowerReactive_Q_Phase_3" : 156.37,
      "Current_AC_Phase_1" : 1.457,
      "Current_AC_Phase_2" : 1.44,
      "Current_AC_Phase_3" : 1.45,
      "Voltage_AC_Phase_1" : 229.3,
      "Voltage_AC_Phase_2" : 231.9,
      "Voltage_AC_Phase_3" : 232,
      "Voltage_AC_PhaseToPhase_12" : 399.4,
      "Voltage_AC_PhaseToPhase_23" : 401.7,
      "Voltage_AC_PhaseToPhase_31" : 399.5,
      "Frequency_Phase_Average" : 50,
      "PowerApparent_S_Sum" : 954,
      "PowerFactor_Sum" : 0.86,
      "PowerFactor_Phase_1" : 0.85,
      "PowerFactor_Phase_2" : 0.87,
      "PowerFactor_Phase_3" : 0.87,
      "EnergyReal_WAC_Sum_Produced" : 23728,
      "EnergyReal_WAC_Sum_Consumed" : 1365,
      "EnergyReactive_VArAC_Sum_Produced" : 4020,
      "EnergyReactive_VArAC_Sum_Consumed" : 140770,
      "EnergyReal_WAC_Plus_Absolute" : 1365,
      "EnergyReal_WAC_Minus_Absolute" : 23728
    },
    "14300001" : {
      "TimeStamp" : 1406722646,
      "Enable" : 1,
```

```

"Visible" : 1,
"PowerReal_P_Sum" : -234.52,
"Meter_Location_Current" : 1,
"PowerReal_P_Phase_1" : -234.52,
"PowerReal_P_Phase_2" : 0,
"PowerReal_P_Phase_3" : 0,
"PowerReactive_Q_Sum" : -17.83,
"PowerReactive_Q_Phase_1" : -17.83,
"PowerReactive_Q_Phase_2" : 0,
"PowerReactive_Q_Phase_3" : 0,
"Current_AC_Phase_1" : 1.034,
"Current_AC_Phase_2" : 0.013,
"Current_AC_Phase_3" : 0.023,
"Voltage_AC_Phase_1" : 228.5,
"Voltage_AC_Phase_2" : 105.8,
"Voltage_AC_Phase_3" : 105.8,
"Voltage_AC_PhaseToPhase_12" : 295.9,
"Voltage_AC_PhaseToPhase_23" : 183.3,
"Voltage_AC_PhaseToPhase_31" : 295.9,
"Frequency_Phase_Average" : 50,
"PowerApparent_S_Sum" : 235,
"PowerFactor_Sum" : 0.99,
"PowerFactor_Phase_1" : 0.99,
"PowerFactor_Phase_2" : 1,
"PowerFactor_Phase_3" : 1,
"EnergyReal_WAC_Sum_Produced" : 0,
"EnergyReal_WAC_Sum_Consumed" : 7330,
"EnergyReactive_VArAC_Sum_Produced" : 5090,
"EnergyReactive_VArAC_Sum_Consumed" : 6110,
"EnergyReal_WAC_Plus_Absolute" : 0,
"EnergyReal_WAC_Minus_Absolute" : 7330

```

```

}

```

```

}

```

```

}

```



---

**SolarAPI v1 -  
CurrentData -  
Storages**

JSON basierendes Datenpaket, ähnlich einer Web-Anfrage  
weitere Informationen unter:  
<http://www.fronius.com/de/solarenergie/produkte>  
Anlagen-Monitoring, Offene Schnittstellen, Fronius Solar API (JSON)

Beispiel für exportierte Inhalte

```

{
  "Body" : {
    "0" : {
      "Controller" : {
        "Capacity_Maximum" : 3600,
        "Current_DC" : 1.4199999682605267,
        "DesignedCapacity" : 3600,
        "Details" : {
          "Manufacturer" : "Fronius_International",
          "Model" : "Fronius_Solar_Battery",
          "Serial" : "n.a.(err2)"
        },
        "Enable" : 1,
        "StateOfCharge_Relative" : 95,
        "Temperature_Cell" : 24.750004439055942,
        "TimeStamp" : 1484140041,
        "Voltage_DC" : 161.40000240504742,
        "Voltage_DC_Maximum_Cell" : 3.368000159971416,
        "Voltage_DC_Minimum_Cell" : 3.3610001596389338
      },
      "Modules" : [
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4099999684840441,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003146_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.550004450976871,
          "Temperature_Cell_Minimum" : 24.55000443607571,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.817002556170337,
          "Voltage_DC_Maximum_Cell" : 3.3660001598764211,
          "Voltage_DC_Minimum_Cell" : 3.3610001596389338
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003156_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.250004446506523,
          "Temperature_Cell_Minimum" : 24.450004434585594,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.832002556882799,
          "Voltage_DC_Maximum_Cell" : 3.3670001599239185,
          "Voltage_DC_Minimum_Cell" : 3.3620001596864313
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003174_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 96,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.650004437565826,
          "Temperature_Cell_Maximum" : 25.150004445016407,
          "Temperature_Cell_Minimum" : 24.450004434585594,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.821002556360327,
          "Voltage_DC_Maximum_Cell" : 3.368000159971416,
          "Voltage_DC_Minimum_Cell" : 3.3610001596389338
        }
      ]
    }
  }
}

```

```

{
  "Body" : {
    "0" : {
      "Controller" : {
        "Capacity_Maximum" : 3600,
        "Current_DC" : 1.4199999682605267,
        "DesignedCapacity" : 3600,
        "Details" : {
          "Manufacturer" : "Fronius_International",
          "Model" : "Fronius_Solar_Battery",
          "Serial" : "n.a.(err2)"
        },
        "Enable" : 1,
        "StateOfCharge_Relative" : 95,
        "Temperature_Cell" : 24.750004439055942,
        "TimeStamp" : 1484140041,
        "Voltage_DC" : 161.40000240504742,
        "Voltage_DC_Maximum_Cell" : 3.368000159971416,
        "Voltage_DC_Minimum_Cell" : 3.3610001596389338
      },
      "Modules" : [
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4099999684840441,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003146_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.550004450976871,
          "Temperature_Cell_Minimum" : 24.55000443607571,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.817002556170337,
          "Voltage_DC_Maximum_Cell" : 3.3660001598764211,
          "Voltage_DC_Minimum_Cell" : 3.3610001596389338
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003156_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.250004446506523,
          "Temperature_Cell_Minimum" : 24.450004434585594,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.832002556882799,
          "Voltage_DC_Maximum_Cell" : 3.3670001599239185,
          "Voltage_DC_Minimum_Cell" : 3.3620001596864313
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003174_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 96,
          "Status_BatteryCell" : 53
        }
      ]
    }
  }
}

```

**SolarAPI v1 -  
CurrentData -  
Powerflow**

JSON basierendes Datenpaket, ähnlich einer Web-Anfrage  
weitere Informationen unter:  
<http://www.fronius.com/de/solarenergie/produkte>  
Anlagen-Monitoring, Offene Schnittstellen, Fronius Solar API (JSON)

Beispiel für exportierte Inhalte

```
{
  "Body" : {
    "Inverters" : {
      "1" : {
        "Battery_Mode" : "normal",
        "DT" : 99,
        "E_Day" : 1223.9000244140625,
        "E_Total" : 662092.6875,
        "E_Year" : 504265.3125,
        "P" : -3,
        "SOC" : 95
      }
    },
    "Site" : {
      "BatteryStandby" : false,
      "E_Day" : 1223.9000000000001,
      "E_Total" : 662092.70000000007,
      "E_Year" : 504265.30000000005,
      "Meter_Location" : "grid",
      "Mode" : "bidirectional",
      "P_Akku" : -236.00999999999999,
      "P_Grid" : -4.2399999052286148,
      "P_Load" : 7.2399999052286148,
      "P_PV" : 287,
      "rel_Autonomy" : 100,
      "rel_SelfConsumption" : 100
    }
  },
  "Head" : {
    "RequestArguments" : null,
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    },
    "Timestamp" : "2017-01-11T14:07:43+01:00"
  }
}
```

## SolarAPI v1 - Logdata - Data

JSON basierendes Datenpaket, ähnlich einer Web-Anfrage  
Enthält alle gesammelten Daten für den aktuellen Tag.  
Wird dieses Format für weniger als tägliches Intervall verwendet, werden die Daten am externen Server verdoppelt. Wird das Intervall auf eine Woche eingestellt, werden Log-Daten-Anfragen für eine Woche erzeugt.  
Das kleinstmögliche Intervall ist eine Stunde!

weitere Informationen unter:

<http://www.fronius.com/de/solarenergie/produkte>

Anlagen-Monitoring, Offene Schnittstellen, Fronius Solar API (JSON)

Beispiel für exportierte Inhalte

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:06:44+02:00",
    "RequestArguments" : {
      "Query" : "Inverter+SensorCard+Meter",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "inverter\97" : {
      "NodeType" : 193,
      "DeviceType" : 225,
      "Start" : "2014-07-30T00:00:00+02:00",
      "End" : "2014-07-30T23:59:59+02:00",
      "Data" : {
        "EnergyReal_WAC_Sum_Produced" : {
          "_comment" : "channelId=67830024",
          "Unit" : "Wh",
          "Values" : {
            "21300" : 0,
            "21600" : 2.1325,
            ...
            "50400" : 202.387222,
            "50700" : 201.4375
          }
        },
        "Voltage_DC_String_1" : {
          "_comment" : "channelId=66049",
          "Unit" : "V",
          "Values" : {
            "0" : 310,
            "300" : 307.4,
            ...
            "50400" : 344.8,
            "50700" : 352.4
          }
        },
        "Current_DC_String_1" : {
          "_comment" : "channelId=66050",
          "Unit" : "A",
          "Values" : {
            "21300" : 0,
            "21600" : 0.08,
```



## SolarAPI v1 - Logdata - Errors and Events

JSON basierendes Datenpaket, ähnlich einer Web-Anfrage  
Enthält alle gesammelten Fehler und Ereignisse für den aktuellen Tag.  
Wird dieses Format für weniger als tägliches Intervall verwendet, werden die Daten am externen Server verdoppelt. Wird das Intervall auf eine Woche eingestellt, werden Log-Daten-Anfragen für eine Woche erzeugt.  
Das kleinstmögliche Intervall sind 30 Minuten!

weitere Informationen unter:

<http://www.fronius.com/de/solarenergie/produkte>

Anlagen-Monitoring, Offene Schnittstellen, Fronius Solar API (JSON)

Beispiel für exportierte Inhalte

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:04:40+02:00",
    "RequestArguments" : {
      "Query" : "Errors+Events",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "inverter\16" : {
      "NodeType" : 112,
      "DeviceType" : 226,
      "Start" : "2014-07-30T00:00:00+02:00",
      "End" : "2014-07-30T23:59:59+02:00",
      "Data" : {
        "InverterErrors" : {
          "_comment" : "channelId=16646144",
          "Unit" : "?",
          "Values" : {
            "32040" : {
              "flags" : [
                "fatal",
                "official",
                "send"
              ],
              "#" : 509
            }
          }
        }
      }
    },
    "inverter\97" : {
      "NodeType" : 193,
      "DeviceType" : 225,
      "Start" : "2014-07-30T00:00:00+02:00",
      "End" : "2014-07-30T23:59:59+02:00",
      "Data" : {
        "InverterErrors" : {
          "_comment" : "channelId=16646144",
          "Unit" : "?",
          "Values" : {}
        }
      }
    }
  }
}
```

```

"broadcasts\" : {
  "Start" : "2014-07-30T00:00:00+02:00",
  "End" : "2014-07-30T23:59:59+02:00",
  "Data" : {
    "InverterEvents" : {
      "_comment" : "channelId=16711680",
      "Unit" : "?",
      "Values" : {
        "39720" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        },
        "47220" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        },
        "49980" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        },
        "50040" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        }
      }
    }
  }
}

```



## Datamanager IO States

JSON basierendes Datenpaket

Aktuell sind Richtung, Type und Funktionen programmiert. Weitere Änderungen können innerhalb folgender Grenzen erfolgen:

Type: Digital, Analog

Richtung: in, out

get (erhalten): true, false

set (einstellen): true, false

Funktion: Digital\_PowerControl\_Feedback\_Out\_1  
Digital\_PowerManagementRelay\_Out\_1  
Digital\_PowerControl\_In\_1 - 8.

(unzählige, nicht vorausberechenbare Anwendungsfälle)

Beispiel für exportierte Inhalte

```
{
  "pin_0" : {
    "function" : "Digital_PowerControl_Feedback_Out_1",
    "type" : "DIGITAL",
    "direction" : "out",
    "set" : false
  },
  "pin_1" : {
    "function" : "Digital_PowerManagementRelay_Out_1",
    "type" : "DIGITAL",
    "direction" : "out",
    "set" : false
  },
  "pin_2" : {
    "function" : "Digital_PowerControl_In_1",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  },
  "pin_3" : {
    "function" : "Digital_PowerControl_In_2",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  },
  "pin_4" : {
    "function" : "Digital_PowerControl_In_3",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  },
  "pin_5" : {
    "function" : "Digital_PowerControl_In_4",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  }
}
```

```
    },
    "pin_6" : {
        "function" : "Digital_PowerControl_In_5",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    },
    "pin_7" : {
        "function" : "Digital_PowerControl_In_6",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    },
    "pin_8" : {
        "function" : "Digital_PowerControl_In_7",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    },
    "pin_9" : {
        "function" : "Digital_PowerControl_In_8",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    }
}
```

**Demo Content**

Dummy-Inhalt, wird zur Überprüfung von Verbindungseinstellungen verwendet, ohne persönliche Inhalte wie Seriennummer oder Aktuelldaten zu übermitteln.

Beispiel für exportierte Inhalte

```
This is a demo file to check the service.  
File generated at 2014-07-30T14:07:02+02:00  
FRONIUS INTERNATIONAL 2014
```

---

## Intervalle

Bei werksseitiger Einstellung beginnt der Push Service um Mitternacht (00:00 h) mit dem Datenexport und wiederholt dies entsprechend den festgelegten Einstellungen.

Bei Anwender-spezifischen Konfigurationen führt der Push Service sofort alle installierten und aktivierten Jobs einmal aus, um Richtigkeit und Funktionalität überprüfen zu können. Danach arbeitet der Push Service wieder mit dem werksseitig eingestellten Export-Intervall.

Beispiel:

Falls ein Job auf ein Ein-Stunden-Intervall eingestellt ist, wird dieser alle 60 Minuten ausgeführt, beginnend um Mitternacht.

Die Anwendung einer neuen Konfiguration um beispielsweise 06:45 h lässt das Push Service um 06:45 h einen Test durchführen. Danach arbeitet der Push Service wieder mit dem ursprünglichen Export-Intervall, folglich wird der nächste Job um 07:00 h ausgeführt.

### Intervalle

- 10 Sekunden
- 1 Minute
- 5 Minuten
- 10 Minuten
- 15 Minuten
- 30 Minuten
- 1 Stunde
- 2 Stunden
- 4 Stunden
- 12 Stunden
- 1 Tag (jeden Tag um 23:50 h / 11:50 pm)
- 1 Woche (jeden Sonntag um 23:50 h / 11:50 pm)

**WICHTIG!** Für die letzten zwei Optionen muss der Nachtmodus des Wechselrichters aktiviert sein, um eine Datenübertragung zu gewährleisten.

# Contents

- Operating Instructions..... 30
  - General..... 30
  - Overview ..... 31
- Details..... 33
  - General..... 33
  - Protocols..... 33
  - Formats..... 33
  - SunSpec Datalogger v1.0b - inverter float model..... 35
  - SunSpec Datalogger v1.2 - meter model..... 36
  - SolarAPI v1 - CurrentData - Inverter ..... 37
  - SolarAPI v1 - CurrentData - SensorCard ..... 39
  - SolarAPI v1 - CurrentData - StringControl ..... 40
  - SolarAPI v1 - CurrentData - Meter ..... 41
  - SolarAPI v1 - CurrentData - Storages ..... 43
  - SolarAPI v1 - CurrentData - Powerflow ..... 46
  - SolarAPI v1 - Logdata - Data ..... 47
  - SolarAPI v1 - Logdata - Errors and Events..... 49
  - Datamanager IO States ..... 51
  - Demo Content ..... 53
  - Intervals..... 54

# Operating Instructions

---

## General

Using the push service, current and log data of the inverter can be exported to an external server in different formats or using different protocols. The prerequisite for this is a Fronius Datamanager or Fronius Datamanager 2.0.

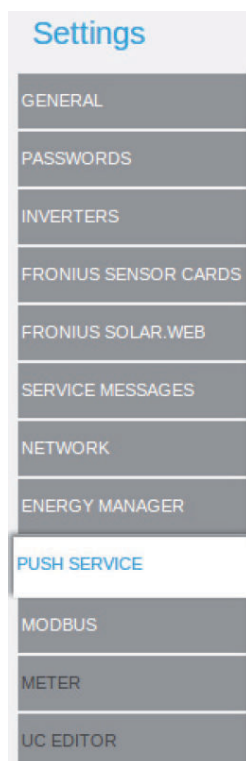
Formats define how the information is sent.

Protocols define the type of data transfer.

There are no pre-installed jobs.

Up to 10 jobs can be installed using this function. Each job may use different protocols, formats or external servers.

The Push Service function can be found on the Fronius Datamanager website in the Settings menu:



## Push Service

☒ (1)    ☐ (2)

(3)

▼ **New FTP Service 0** (5)
(4) State: ---

Name:

Data format:  by  (6)

Interval:  activated ☒

---

Server:Port:  (7)

Upload file name:

☐ Login:

---

☒ Proxy

Server:Port:  (8)

User:

Password:

---

(9)

- (1) "Apply/Save" button  
Saves the configuration and starts an immediate test run of all the installed and enabled services
- (2) "Cancel/Discard entry" button  
Discards any changes and reloads the original configuration
- (3) "Add" button  
Click this button to add a new push service job. The new job is saved by clicking the "Apply/Save" button (1).
- (4) State  
Displays the current status of the respective push service job.  
In the event of an error, detailed information can be displayed by hovering the cursor above the cause text. The detailed information is only displayed in English.
- (5) Displayed name of the push service job  
The name identifies the push service job and must be unique.
- (6) Area for entering general data:  
Name (name of the push service job)  
Data format  
Protocol type (FTP upload/HTTP POST)  
Interval  
Activation status

- (7) Area for entering destination data:  
 Server port  
 Upload file name  
 Login (user/password)
- (8) Area for entering proxy data:  
 Server port  
 User  
 Password
- (9) "Delete" button  
 Click this button to delete the selected push service job

Only one job can be opened and edited at any given time, regardless of how many push service jobs have been installed:

**Push Service**

+ Add

▶ SunSpec v1 Inverter FTP Intern	State: Connection is not possible
▶ SunSpec v1 Inverter HTTP	State: Successful 2 minutes ago
▶ SolarApi Inverter HTTP	State: Successful 2 minutes ago
▶ SolarApi SensorCard HTTP	State: Successful 2 minutes ago
▶ SolarApi StringControl HTTP	State: Successful 2 minutes ago
▶ IO States	State: Successful 2 minutes ago
▶ SolarApi Logdata HTTP	State: Successful 37 minutes ago
▶ SolarAPI Error HTTP	State: Successful 7 minutes ago
▶ New FTP Service 8	State: disabled
▼ <b>New FTP Service 9</b>	State: Connection is not possible

Name:

Data format:  by

Interval:  ☒ activated

---

Server:Port:

Upload file name:

☒ Login:

User:

Password:

---

☐ Proxy

---

List of multiple push service jobs. One job has been opened and can be edited; for all other jobs, an error message will be displayed if the cursor is hovered over the cause text.



# Details

## General

At present the only way to communicate with the Fronius push service is to make an HTTP request to a specific URL.

### Push Service

☐ ☐

+ Add

New FTP Service 0

State: --

Name:

New FTP Service 0

Data format:

SunSpec Datalogger v1.0b - Inverter float model

by

FTP upload

Interval:

10 sec

activated

Server:Port:

MyServer:21

Upload file name:

/anypath/anyfile{DATE}{TIME}.any

Login:

User:

Password:

Proxy:

Server:Port:

http://anyserver:8080

User:

anyuser

Password:

\*\*\*\*\*

Delete

User interface on the Fronius Datamanager web page

## Protocols

The HTTP POST and FTP upload (passive or active) protocols are currently supported.

### Proxy

Full proxy support (with or without references) has been implemented.  
Exception: use of certificates

### Login authentication

The following authentications are supported:

- Basic
- Digest

The Fronius push service automatically checks which authentication should be selected.

## Formats

The following formats are currently supported:

- SunSpec Datalogger v1.0b - inverter float model
- SunSpec Datalogger v1.2 - meter model
- SolarAPI v1 - CurrentData - Inverter
- SolarAPI v1 - CurrentData - SensorCard
- SolarAPI v1 - CurrentData - StringControl
- SolarAPI v1 - CurrentData - Meter
- SolarAPI v1 - Logdata - Data
- SolarAPI v1 - Logdata - Errors and Events
- Datamanager IO States
- Demo Content

**SunSpec Data-logger v1.0b - inverter float model**

XML-based data package  
SunSpec Models 1 and (111, 112 or 113)  
For more information see:  
<http://sunspec.org>

Example of pushed content

```
<SunSpecData v="1">
  <d man="Fronius" mod="Fronius_Galvo_2.0-1" sn="24251001236580052" t="2014-07-30T13:53:51+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius Galvo 2.0-1</p>
      <p id="Vr">0.1.14.0</p>
      <p id="SN">24251001236580052</p> *
      <p id="DA">16</p>
    </m>
    <m id="111">
      <p id="W">2011.000000</p>
      <p id="Wh">42121.003906</p>
      <p id="St">4</p>
    </m>
  </d>
  <d man="Fronius" mod="Fronius_Symo_6.7-3-M" sn="24071001231460050" t="2014-07-30T13:53:51+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius Symo 6.7-3-M</p>
      <p id="Vr">0.2.4.1</p>
      <p id="SN">24071001231460050</p> *
      <p id="DA">36</p>
    </m>
    <m id="113">
      <p id="W">6519.000000</p>
      <p id="Wh">6440126.500000</p>
      <p id="St">4</p>
    </m>
  </d>
</SunSpecData>
```

\* The device serial numbers (SN) are only displayed for Fronius Symo, Primo and Galvo from serial number 2713xxxx. For older devices, the product matrix code (PMC) of the Recerbo is displayed.

**SunSpec Data-  
logger v1.2 - me-  
ter model**

XML-based data package  
SunSpec Models 1 and 213  
For more information see:  
<http://sunspec.org>

Example of pushed content

```
<SunSpecData v="1.2">
  <d man="Fronius" mod="Fronius_SmartMeter" sn="14300002" t="2014-07-30T14:16:32+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius SmartMeter</p>
      <p id="SN">14300002</p>
      <p id="DA">1</p>
    </m>
    <m id="213">
      <p id="W">-840.310000</p>
      <p id="Hz">50.000000</p>
      <p id="WphA">-265.760000</p>
      <p id="WphB">-297.450000</p>
      <p id="WphC">-277.100000</p>
      <p id="AphA">1.443000</p>
      <p id="AphB">1.524000</p>
      <p id="AphC">1.431000</p>
      <p id="PhVphA">229.700000</p>
      <p id="PhVphB">232.300000</p>
      <p id="PhVphC">232.300000</p>
    </m>
  </d>
  <d man="Fronius" mod="Fronius_SmartMeter" sn="14300001" t="2014-07-30T14:16:32+02:00">
    <m id="1">
      <p id="Mn">Fronius</p>
      <p id="Md">Fronius SmartMeter</p>
      <p id="SN">14300001</p>
      <p id="DA">2</p>
    </m>
    <m id="213">
      <p id="W">-238.380000</p>
      <p id="Hz">50.000000</p>
      <p id="WphA">-238.380000</p>
      <p id="WphB">-0.000000</p>
      <p id="WphC">-0.000000</p>
      <p id="AphA">1.047000</p>
      <p id="AphB">0.013000</p>
      <p id="AphC">0.023000</p>
      <p id="PhVphA">229.300000</p>
      <p id="PhVphB">106.200000</p>
      <p id="PhVphC">106.100000</p>
    </m>
  </d>
</SunSpecData>
```

**SolarAPI v1 -  
CurrentData -  
Inverter**

JSON-based data package, similar to a web request  
For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T13:59:19+02:00",
    "RequestArguments" : {
      "Query" : "Inverter",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "PAC" : {
      "Unit" : "W",
      "Values" : {
        "16" : 2007,
        "36" : 6465,
        "66" : 1945,
        "78" : 3012,
        "83" : 2980,
        "84" : 19193,
        "85" : 18559,
        "87" : 8082,
        "97" : 2534
      }
    },
    "DAY_ENERGY" : {
      "Unit" : "Wh",
      "Values" : {
        "16" : 5027,
        "36" : 700,
        "66" : 8673,
        "78" : 13428,
        "83" : 13014,
        "84" : 21233,
        "85" : 86500,
        "87" : 35618,
        "97" : 10832
      }
    },
    "YEAR_ENERGY" : {
      "Unit" : "Wh",
      "Values" : {
        "16" : 41950,
        "36" : 5004877,

```

```

        "66" : 2088417,
        "78" : 2330199,
        "83" : 2255519,
        "84" : 15047895,
        "85" : 15048166,
        "87" : 6164320,
        "97" : 1879060
    },
    "TOTAL_ENERGY" : {
        "Unit" : "Wh",
        "Values" : {
            "16" : 42305,
            "36" : 6440730,
            "66" : 5817456,
            "78" : 9039449,
            "83" : 8762867,
            "84" : 58365852,
            "85" : 58408104,
            "87" : 23929897,
            "97" : 7285498
        }
    }
}

```

**SolarAPI v1 -  
CurrentData -  
SensorCard**

JSON-based data package, similar to a web request  
For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

NE

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:01:08+02:00",
    "RequestArguments" : {
      "DataCollection" : "NowSensorData",
      "DeviceClass" : "SensorCard",
      "DeviceId" : "All",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "0" : {
      "0" : {
        "Value" : 29,
        "Unit" : "\u00B0C"
      },
      "1" : {
        "Value" : 27,
        "Unit" : "\u00B0C"
      },
      "2" : {
        "Value" : 0,
        "Unit" : "W/m\u00B2"
      },
      "3" : {
        "Value" : 0,
        "Unit" : "km/h"
      },
      "4" : {
        "Value" : 0,
        "Unit" : "km/h"
      },
      "5" : {
        "Value" : 0,
        "Unit" : "A"
      }
    }
  }
}
```

---

**SolarAPI v1 -  
CurrentData -  
StringControl**

JSON-based data package, similar to a web request  
For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:02:11+02:00",
    "RequestArguments" : {
      "DataCollection" : "NowStringControlData",
      "DeviceClass" : "StringControl",
      "DeviceId" : "2",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "2" : {
      "1" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "2" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "3" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "4" : {
        "Value" : 0,
        "Unit" : "A"
      },
      "5" : {
        "Value" : 0,
        "Unit" : "A"
      }
    }
  }
}
```



**SolarAPI v1 -  
CurrentData -  
Meter**

JSON-based data package, similar to a web request  
For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:17:28+02:00",
    "RequestArguments" : {
      "Query" : "Meter",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "14300002" : {
      "TimeStamp" : 1406722647,
      "Enable" : 1,
      "Visible" : 1,
      "PowerReal_P_Sum" : -828.26,
      "Meter_Location_Current" : 0,
      "PowerReal_P_Phase_1" : -269.75,
      "PowerReal_P_Phase_2" : -274.79,
      "PowerReal_P_Phase_3" : -283.72,
      "PowerReactive_Q_Sum" : 474,
      "PowerReactive_Q_Phase_1" : 163.31,
      "PowerReactive_Q_Phase_2" : 154.32,
      "PowerReactive_Q_Phase_3" : 156.37,
      "Current_AC_Phase_1" : 1.457,
      "Current_AC_Phase_2" : 1.44,
      "Current_AC_Phase_3" : 1.45,
      "Voltage_AC_Phase_1" : 229.3,
      "Voltage_AC_Phase_2" : 231.9,
      "Voltage_AC_Phase_3" : 232,
      "Voltage_AC_PhaseToPhase_12" : 399.4,
      "Voltage_AC_PhaseToPhase_23" : 401.7,
      "Voltage_AC_PhaseToPhase_31" : 399.5,
      "Frequency_Phase_Average" : 50,
      "PowerApparent_S_Sum" : 954,
      "PowerFactor_Sum" : 0.86,
      "PowerFactor_Phase_1" : 0.85,
      "PowerFactor_Phase_2" : 0.87,
      "PowerFactor_Phase_3" : 0.87,
      "EnergyReal_WAC_Sum_Produced" : 23728,
      "EnergyReal_WAC_Sum_Consumed" : 1365,
      "EnergyReactive_VArAC_Sum_Produced" : 4020,
      "EnergyReactive_VArAC_Sum_Consumed" : 140770,
      "EnergyReal_WAC_Plus_Absolute" : 1365,
      "EnergyReal_WAC_Minus_Absolute" : 23728
    },
    "14300001" : {
      "TimeStamp" : 1406722646,
      "Enable" : 1,
```

```

"Visible" : 1,
"PowerReal_P_Sum" : -234.52,
"Meter_Location_Current" : 1,
"PowerReal_P_Phase_1" : -234.52,
"PowerReal_P_Phase_2" : 0,
"PowerReal_P_Phase_3" : 0,
"PowerReactive_Q_Sum" : -17.83,
"PowerReactive_Q_Phase_1" : -17.83,
"PowerReactive_Q_Phase_2" : 0,
"PowerReactive_Q_Phase_3" : 0,
"Current_AC_Phase_1" : 1.034,
"Current_AC_Phase_2" : 0.013,
"Current_AC_Phase_3" : 0.023,
"Voltage_AC_Phase_1" : 228.5,
"Voltage_AC_Phase_2" : 105.8,
"Voltage_AC_Phase_3" : 105.8,
"Voltage_AC_PhaseToPhase_12" : 295.9,
"Voltage_AC_PhaseToPhase_23" : 183.3,
"Voltage_AC_PhaseToPhase_31" : 295.9,
"Frequency_Phase_Average" : 50,
"PowerApparent_S_Sum" : 235,
"PowerFactor_Sum" : 0.99,
"PowerFactor_Phase_1" : 0.99,
"PowerFactor_Phase_2" : 1,
"PowerFactor_Phase_3" : 1,
"EnergyReal_WAC_Sum_Produced" : 0,
"EnergyReal_WAC_Sum_Consumed" : 7330,
"EnergyReactive_VArAC_Sum_Produced" : 5090,
"EnergyReactive_VArAC_Sum_Consumed" : 6110,
"EnergyReal_WAC_Plus_Absolute" : 0,
"EnergyReal_WAC_Minus_Absolute" : 7330

```

```

}

```

```

}

```

```

}

```

---

**SolarAPI v1 -  
CurrentData -  
Storages**

JSON-based data package, similar to a web request  
For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```

{
  "Body" : {
    "0" : {
      "Controller" : {
        "Capacity_Maximum" : 3600,
        "Current_DC" : 1.4199999682605267,
        "DesignedCapacity" : 3600,
        "Details" : {
          "Manufacturer" : "Fronius_International",
          "Model" : "Fronius_Solar_Battery",
          "Serial" : "n.a.(err2)"
        },
        "Enable" : 1,
        "StateOfCharge_Relative" : 95,
        "Temperature_Cell" : 24.750004439055942,
        "TimeStamp" : 1484140041,
        "Voltage_DC" : 161.40000240504742,
        "Voltage_DC_Maximum_Cell" : 3.368000159971416,
        "Voltage_DC_Minimum_Cell" : 3.3610001596389338
      },
      "Modules" : [
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4099999684840441,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003146_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.550004450976871,
          "Temperature_Cell_Minimum" : 24.55000443607571,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.817002556170337,
          "Voltage_DC_Maximum_Cell" : 3.3660001598764211,
          "Voltage_DC_Minimum_Cell" : 3.3610001596389338
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003156_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.250004446506523,
          "Temperature_Cell_Minimum" : 24.450004434585594,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.832002556882799,
          "Voltage_DC_Maximum_Cell" : 3.3670001599239185,
          "Voltage_DC_Minimum_Cell" : 3.3620001596864313
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003174_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 96,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.650004437565826,
          "Temperature_Cell_Maximum" : 25.150004445016407,
          "Temperature_Cell_Minimum" : 24.450004434585594,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.821002556360327,
          "Voltage_DC_Maximum_Cell" : 3.368000159971416,
          "Voltage_DC_Minimum_Cell" : 3.3610001596389338
        }
      ]
    }
  }
}

```

```

{
  "Body" : {
    "0" : {
      "Controller" : {
        "Capacity_Maximum" : 3600,
        "Current_DC" : 1.4199999682605267,
        "DesignedCapacity" : 3600,
        "Details" : {
          "Manufacturer" : "Fronius_International",
          "Model" : "Fronius_Solar_Battery",
          "Serial" : "n.a.(err2)"
        },
        "Enable" : 1,
        "StateOfCharge_Relative" : 95,
        "Temperature_Cell" : 24.750004439055942,
        "TimeStamp" : 1484140041,
        "Voltage_DC" : 161.40000240504742,
        "Voltage_DC_Maximum_Cell" : 3.368000159971416,
        "Voltage_DC_Minimum_Cell" : 3.3610001596389338
      },
      "Modules" : [
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4099999684840441,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003146_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.550004450976871,
          "Temperature_Cell_Minimum" : 24.55000443607571,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.817002556170337,
          "Voltage_DC_Maximum_Cell" : 3.3660001598764211,
          "Voltage_DC_Minimum_Cell" : 3.3610001596389338
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003156_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 95,
          "Status_BatteryCell" : 53,
          "Temperature_Cell" : 24.750004439055942,
          "Temperature_Cell_Maximum" : 25.250004446506523,
          "Temperature_Cell_Minimum" : 24.450004434585594,
          "TimeStamp" : 1484140041,
          "Voltage_DC" : 53.832002556882799,
          "Voltage_DC_Maximum_Cell" : 3.3670001599239185,
          "Voltage_DC_Minimum_Cell" : 3.3620001596864313
        },
        {
          "Capacity_Maximum" : 1200,
          "Current_DC" : 1.4199999682605267,
          "CycleCount_BatteryCell" : 11,
          "Details" : {
            "Manufacturer" : "Sony",
            "Model" : "P49929880N",
            "Serial" : "S012003174_"
          },
          "Enable" : 1,
          "StateOfCharge_Relative" : 96,
          "Status_BatteryCell" : 53
        }
      ]
    }
  }
}

```

**SolarAPI v1 -  
CurrentData -  
Powerflow**

JSON-based data package, similar to a web request  
For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```
{
  "Body" : {
    "Inverters" : {
      "1" : {
        "Battery_Mode" : "normal",
        "DT" : 99,
        "E_Day" : 1223.9000244140625,
        "E_Total" : 662092.6875,
        "E_Year" : 504265.3125,
        "P" : -3,
        "SOC" : 95
      }
    },
    "Site" : {
      "BatteryStandby" : false,
      "E_Day" : 1223.9000000000001,
      "E_Total" : 662092.70000000007,
      "E_Year" : 504265.30000000005,
      "Meter_Location" : "grid",
      "Mode" : "bidirectional",
      "P_Akku" : -236.00999999999999,
      "P_Grid" : -4.2399999052286148,
      "P_Load" : 7.2399999052286148,
      "P_PV" : 287,
      "rel_Autonomy" : 100,
      "rel_SelfConsumption" : 100
    }
  },
  "Head" : {
    "RequestArguments" : null,
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    },
    "Timestamp" : "2017-01-11T14:07:43+01:00"
  }
}
```

## SolarAPI v1 - Logdata - Data

JSON-based data package, similar to a web request

Contains all the collated data for the current day.

Data will be duplicated on the external server if this format is used for an interval more frequent than daily. If the interval is set to one week, log data requests for one week are generated.

The smallest possible interval is one hour.

For more information go to:

<http://www.fronius.com/en/photovoltaics/products>

System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:06:44+02:00",
    "RequestArguments" : {
      "Query" : "Inverter+SensorCard+Meter",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "inverter\97" : {
      "NodeType" : 193,
      "DeviceType" : 225,
      "Start" : "2014-07-30T00:00:00+02:00",
      "End" : "2014-07-30T23:59:59+02:00",
      "Data" : {
        "EnergyReal_WAC_Sum_Produced" : {
          "_comment" : "channelId=67830024",
          "Unit" : "Wh",
          "Values" : {
            "21300" : 0,
            "21600" : 2.1325,
            "...",
            "50400" : 202.387222,
            "50700" : 201.4375
          }
        },
        "Voltage_DC_String_1" : {
          "_comment" : "channelId=66049",
          "Unit" : "V",
          "Values" : {
            "0" : 310,
            "300" : 307.4,
            "...",
            "50400" : 344.8,
            "50700" : 352.4
          }
        },
        "Current_DC_String_1" : {
          "_comment" : "channelId=66050",
          "Unit" : "A",
          "Values" : {
            "21300" : 0,
            "21600" : 0.08,
```

```

        "21900" : 0.08,
        "22200" : 0.09,
        ...
        "49800" : 8.82,
        "50100" : 7.14,
        "50400" : 7.28,
        "50700" : 7.55
    }
}

    },
    "meter:S00112" : {
        "Start" : "2014-07-30T00:00:00+02:00",
        "End" : "2014-07-30T23:59:59+02:00",
        "Data" : {
            "Meter_Location_Current" : {
                "_comment" : "channelId=117050390",
                "Unit" : "1",
                "Values" : {
                    "39720" : 0,
                    ...
                    "50700" : 0
                }
            },
            "EnergyReal_WAC_Minus_Relative" : {
                "_comment" : "channelId=184615176",
                "Unit" : "Wh",
                "Values" : {
                    "39720" : 1,
                    "40020" : 123,
                    ...
                    "49800" : 166,
                    "50100" : 55,
                    "50400" : 166,
                    "50700" : 164
                }
            }
        }
    },
    "datamanager:\\dc\\f000a5b5\\" : {
        "Start" : "2014-07-30T00:00:00+02:00",
        "End" : "2014-07-30T23:59:59+02:00",
        "Data" : {
            "Digital_PowerManagementRelay_Out_1" : {
                "_comment" : "channelId=123407124",
                "Unit" : "1",
                "Values" : {
                    "47215" : 0,
                    "49998" : 0
                }
            }
        }
    }
}

```



## SolarAPI v1 - Logdata - Errors and Events

JSON-based data package, similar to a web request  
Contains all the collated errors and events for the current day.  
Data will be duplicated on the external server if this format is used for an interval more frequent than daily. If the interval is set to one week, log data requests for one week are generated.  
The smallest possible interval is 30 minutes.

For more information go to:  
<http://www.fronius.com/en/photovoltaics/products>  
System Monitoring, Open Interfaces, Fronius Solar API (JSON)

Example of pushed content

```
{
  "Head" : {
    "Timestamp" : "2014-07-30T14:04:40+02:00",
    "RequestArguments" : {
      "Query" : "Errors+Events",
      "Scope" : "System"
    },
    "Status" : {
      "Code" : 0,
      "Reason" : "",
      "UserMessage" : ""
    }
  },
  "Body" : {
    "inverter\16" : {
      "NodeType" : 112,
      "DeviceType" : 226,
      "Start" : "2014-07-30T00:00:00+02:00",
      "End" : "2014-07-30T23:59:59+02:00",
      "Data" : {
        "InverterErrors" : {
          "_comment" : "channelId=16646144",
          "Unit" : "?",
          "Values" : {
            "32040" : {
              "flags" : [
                "fatal",
                "official",
                "send"
              ],
              "#" : 509
            }
          }
        }
      }
    },
    "inverter\97" : {
      "NodeType" : 193,
      "DeviceType" : 225,
      "Start" : "2014-07-30T00:00:00+02:00",
      "End" : "2014-07-30T23:59:59+02:00",
      "Data" : {
        "InverterErrors" : {
          "_comment" : "channelId=16646144",
          "Unit" : "?",
          "Values" : {}
        }
      }
    }
  }
}
```

```

"broadcasts\" : {
  "Start" : "2014-07-30T00:00:00+02:00",
  "End" : "2014-07-30T23:59:59+02:00",
  "Data" : {
    "InverterEvents" : {
      "_comment" : "channelId=16711680",
      "Unit" : "?",
      "Values" : {
        "39720" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        },
        "47220" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        },
        "49980" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        },
        "50040" : {
          "flags" : [
            "send"
          ],
          "#" : 3,
          "desc" : "Power_limitation_100%",
          "attr" : {
            "Power" : "100_[%]",
            "Radiant" : "255_[1]",
            "affect" : "p"
          }
        }
      }
    }
  }
}

```

## Datamanager IO States

JSON-based data package

The direction, type and functions are currently programmed. Other changes can be made within the following limits:

Type:	Digital, Analog
Direction:	in, out
get:	true, false
set:	true, false
Function:	Digital_PowerControl_Feedback_Out_1 Digital_PowerManagementRelay_Out_1 Digital_PowerControl_In_1 - 8.

(countless, unpredictable use cases)

Example of pushed content

```
{
  "pin_0" : {
    "function" : "Digital_PowerControl_Feedback_Out_1",
    "type" : "DIGITAL",
    "direction" : "out",
    "set" : false
  },
  "pin_1" : {
    "function" : "Digital_PowerManagementRelay_Out_1",
    "type" : "DIGITAL",
    "direction" : "out",
    "set" : false
  },
  "pin_2" : {
    "function" : "Digital_PowerControl_In_1",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  },
  "pin_3" : {
    "function" : "Digital_PowerControl_In_2",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  },
  "pin_4" : {
    "function" : "Digital_PowerControl_In_3",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  },
  "pin_5" : {
    "function" : "Digital_PowerControl_In_4",
    "type" : "DIGITAL",
    "direction" : "in",
    "get" : false
  }
}
```

```
    },
    "pin_6" : {
        "function" : "Digital_PowerControl_In_5",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    },
    "pin_7" : {
        "function" : "Digital_PowerControl_In_6",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    },
    "pin_8" : {
        "function" : "Digital_PowerControl_In_7",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    },
    "pin_9" : {
        "function" : "Digital_PowerControl_In_8",
        "type" : "DIGITAL",
        "direction" : "in",
        "get" : false
    }
}
```

---

**Demo Content**

Dummy content is used to check the connection settings without having to send private data such as serial numbers or current data.

Example of pushed content

```
This is a demo file to check the service.  
File generated at 2014-07-30T14:07:02+02:00  
FRONIUS INTERNATIONAL 2014
```

EN

---

## Intervals

According to the default configuration, the push service will start the data export at midnight (00:00 am) and repeat it as per the defined settings. Where user-specific configurations are applied, the push service immediately executes all installed and enabled jobs in order to check accuracy and functionality. The push service will then revert to operating at the default export interval.

Example:

If a job is set to a one hour interval, then this will be performed every 60 minutes starting at midnight.

If a new configuration is applied at 6:45 am, the push service will carry out a test run at 6:45 am. From this point on the push service will return to its original export interval; the next job will thus be carried out at 7:00 am.

### Intervals

- 10 seconds
- 1 minute
- 5 minutes
- 10 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2 hours
- 4 hours
- 12 hours
- 1 day (every day at 23:50 / 11:50 pm)
- 1 week (every Sunday at 23:50 / 11:50 pm)

**IMPORTANT!** Night mode must be enabled on the inverter to ensure a reliable data transfer for the last two options.





[fronius.com/en/solar-energy/installers-partners/products-solutions/monitoring-digital-tools](https://fronius.com/en/solar-energy/installers-partners/products-solutions/monitoring-digital-tools)

MONITORING &  
DIGITAL TOOLS

**Fronius International GmbH**

Froniusstraße 1  
4643 Pettenbach  
Austria  
[contact@fronius.com](mailto:contact@fronius.com)  
[www.fronius.com](http://www.fronius.com)

At [www.fronius.com/contact](http://www.fronius.com/contact) you will find the contact details of all Fronius subsidiaries and Sales & Service Partners.