

FileAdapter per SafeMoney

Protocollo Scambio file per SafeMoney

Release 1.2.1 – 2021/01/25



| Numero Revisione | Descrizione | Data | Autore |
|------------------|--|------------|------------|
| 1.0.0 | Creazione documento | 2018/11/30 | D.Vezzuto |
| 1.1.0 | Aggiunta Gestione Configurazione, Withdrawal e Fast Withdrawal | 2019/01/08 | D.Vezzuto |
| 1.1.1 | Gestione Configurazione | 2019/08/02 | D.Vezzuto |
| 1.1.2 | Gestione File Paid | 2019/11/29 | D.Vezzuto |
| 1.2.0 | Gestione File Terminated.txt e EstNotDispensed | 2020/03/25 | D.Vezzuto |
| 1.2.1 | Modificati Withdrawal e Fast Withdrawal | 2021/01/25 | M.Paolillo |
| | | | |



Sommario

| | |
|--|----|
| FILEADAPTER: PROTOCOLLO SCAMBIO FILE PER DISPOSITIVI SAFEMONEY | 4 |
| DESCRIZIONE GENERALE | 4 |
| CONFIGURAZIONE ADAPTER | 4 |
| PROTOCOLLO..... | 4 |
| COMANDI OPERATIVI | 4 |
| COMANDI DI CONFIGURAZIONE..... | 5 |
| GESTIONE DEGLI ERRORI | 13 |
| DESCRIZIONE DELLA MACCHINA A STATI | 14 |
| APPENDICE: XML Schema Definition per Request e Response | 15 |



FILEADAPTER: PROTOCOLLO SCAMBIO FILE PER DISPOSITIVI SAFEMONEY

In questa guida viene descritto il protocollo di comunicazione *FileAdapter* che utilizza lo scambio file quale semplice modalità di comunicazione tra un software esterno e qualsiasi dispositivo della famiglia di casseti automatici SafeMoney.

DESCRIZIONE GENERALE

Al fine di rendere più rapido l'interfacciamento verso tutti i tipi di casseti SafeMoney, viene fornito un protocollo di comunicazione che utilizza i file quale modalità di comunicazione sia in termini operativi (pagamenti, prelievi...) sia in termine di configurazione.

Inoltre, a questo è possibile associare la visualizzazione di una UI costituita da un PANNELLO DI GESTIONE (si veda paragrafo 6.3), sviluppato per ambiente Windows (richiede .Net Framework 4), che rende visibile le operazioni che si stanno effettuando sui dispositivi SafeMoney.

In generale, con l'installazione del *FileAdapter* SafeMoney, infatti, viene creata automaticamente la cartella C:\Safemoney\ all'interno della quale avverrà la comunicazione attraverso i file.

CONFIGURAZIONE ADAPTER

È possibile configurare il componente software FileAdapter attraverso parametri passati a linea di comando. Tali parametri sono:

- **SHOW_UI**: indica se il Sistema deve mostrare o meno la form UI per la visualizzazione delle informazioni di pagamento e/o di impostazioni. 'YES' per visualizzare la UI.
- **ADVANCE_IP**: permette di impostare l'indirizzo IP e la porta del SafeMoney dotato di porta Ethernet (*SafeMoney Plus*, *SafeMoney Advance*, *SafeMoney Extreme...*). L'indirizzo deve essere nella seguente forma: 127.0.0.1:7409
- **PIN**: permette di specificare il Pin di accesso per la connessione remota al SafeMoney dotato di connessione Ethernet

PROTOCOLLO

Il protocollo prevede due tipologie di comandi: quelli operativi, rappresentati da file di tipo .txt, e quelli di configurazione, costituiti da file di tipo .xml.

COMANDI OPERATIVI

I comandi operativi sono tutti quei comandi che prevedono un flusso di banconote e/o monete con i casseti SafeMoney. Alla creazione di uno di questi file il *FileAdapter* risponderà con uno o più file che indicano le diverse fasi di lavorazione e/o le diverse informazioni gestite puntualmente dal dispositivo.

In particolare, di seguito sono elencati i file che costituiscono un *input* per il sistema:

- **tobepaid.txt**, il contenuto del file indica la cifra che si vuole pagare in centesimi con un numero di 9 cifre (ad esempio, se la cifra è di 5 Euro, il file dovrà contenere la stringa 000000500). Il file può contenere anche un importo negativo (ad es. -000000500), in questo caso la cassa erogherà direttamente l'importo (da utilizzare nel caso di prelievo con cui, ad esempio, pagare fornitori, vincite, ecc.).
- **void.txt**, il file, senza alcun contenuto, indica al sistema di interrompere l'operazione di pagamento precedentemente avviata
- **getLevels.txt**, il file, senza alcun contenuto, richiede al sistema di indicare i valori attuali dei livelli di monete e banconote

Viceversa, il sistema mostrerà i seguenti file di *output* che rappresentano lo stato del sistema sia in termini di risposta ai comandi inseriti sia in termini di funzionamento del sistema stesso

- **amount.txt**, indica l'ammontare puntuale di monete e/o banconote, formattato a 9 cifre, inserito a fronte del pagamento richiesto
- **paid.txt**, generato quando l'ammontare inserito è uguale o maggiore di quanto richiesto, contiene l'importo totale versato espresso con numero formattato a 9 cifre.
- **dispensed.txt**, generato a fine transazione, contiene il totale erogato dal cassetto automatico espresso con numero formattato a 9 cifre.
- **notdispensed.txt**, generato a fine transazione, contiene il totale non erogato dal cassetto automatico espresso con numero formattato a 9 cifre.
- **estnotdispensed.txt**, generato non appena il sistema calcola la stima del resto che andrà a dispensare, contiene il totale previsto del resto non erogato dal cassetto automatico espresso con numero formattato a 9 cifre.
- **terminated.txt**, indica che la transazione attualmente in corso è terminata
- **status.txt**, generato all'avvio del *FileAdapter*, contiene lo stato puntuale della macchina nella forma: *CODE(NNNN): STATUS [DESCRIPTION]* (ad es. 0001: CONNECTED), in cui la DESCRIPTION è opzionale e, se presente, contiene informazioni aggiuntive utili soprattutto in caso di errore
- **connected.txt**, se presente indica che il cassetto è connesso e correttamente funzionante
- **disconnected.txt**, se presente indica che il cassetto è disconnesso o non correttamente funzionante
- **loading.txt**, indica l'ammontare puntuale di monete e/o banconote durante la fase di caricamento cassetto

COMANDI DI CONFIGURAZIONE

I comandi di configurazione sono tutti quei comandi che prevedono richieste di visualizzazione e/o impostazione dei livelli del SafeMoney. Alla creazione del comando, il *FileAdapter* risponderà con un singolo file di risposta contenente le informazioni richieste e/o la conferma dell'avvenuta operazione.

In particolare, il file di richiesta è di tipo xml e denominato sempre come *CMD_REQUEST.xml* e verrà posto nella cartella "C:\Safemoney\", a questo il sistema risponderà elaborando la richiesta e riponendo nella medesima cartella il file *CMD_RESPONSE.xml* con il contenuto richiesto.

La struttura di una richiesta ha sempre la seguente forma:

```
<?xml version="1.0" encoding="utf-8"?>
<request type="coinCfg" id="123456abc">
  <item>
    <param value="denomination">20</param>
    <param value="quantity">-1</param>
    <param value="currency">EUR</param>
  </item>
  ...
</request>
```

dove l'attributo *type* specifica la vera e propria richiesta, *id* permette di assegnare un identificativo alla richiesta (opzionale) mentre all'interno del tag *<item>* sono contenuti i diversi parametri utili ai fini dell'evasione della richiesta stessa.

La struttura della risposta ha sempre la seguente forma:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="getRoute" request_id="1234" timestamp="20181128_163338" code="ok">
  <param value="1" type="coin" reference="RE" min="0" max="0" />
  <param value="2" type="coin" reference="RE" min="0" max="0" />
  ...
</response>
```

dove l'attributo *type* indica il tipo di richiesta servita, *id* l'eventuale identificativo associato alla richiesta (opzionale), mentre all'interno dei tag *param* sono contenuti i diversi valori di risposta della richiesta stessa.

Di seguito, viene riportato l'elenco completo delle tipologie di CMD_REQUEST accettate dal sistema:

- **coinMaxLevel:** permette la configurazione del livello massimo di monete nel cassetto, oltre tale valore saranno reindirizzate direttamente nella cashbox

```
<?xml version="1.0" encoding="utf-8"?>
<request type="coinMaxLevel">
  <item>
    <param value="denomination">20</param>
    <param value="quantity">150</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">50</param>
    <param value="quantity">180</param>
    <param value="currency">EUR</param>
  </item>
  ...
</request>
```

- **coinMinLevel:** permette la configurazione del livello minimo di monete nel cassetto, se effettuato un comando di *empty partial* tale livello sarà mantenuto nel riciclatore

```
<?xml version="1.0" encoding="utf-8"?>
<request type="coinMinLevel">
  <item>
    <param value="denomination">20</param>
    <param value="quantity">50</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">50</param>
    <param value="quantity">80</param>
    <param value="currency">EUR</param>
  </item>
  ...
</request>
```

- **coinRoute:** permette la configurazione dell'instradamento delle monete nel cassetto, i valori ammissibili sono RE: *Recycling*, CB: *Cashbox*, DS: *Disabled*

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<request type="coinRoute">
  <item>
    <param value="denomination">1</param>
    <param value="route">DS</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">20</param>
    <param value="route">RE</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">50</param>
    <param value="route">CB</param>
    <param value="currency">EUR</param>
  </item>...
</request>

```

- **coinCfg:** al fine di ottenere una rapida configurazione della *route* delle monete o del livello massimo viene fornita una richiesta specifica che indichi al sistema, per ciascun taglio, se si vuole abilitare/disabilitare o fissarne il valore massimo. Questo avviene passando nel tag *param* con value *quantity* i seguenti valori: -1 per taglio disabilitato, 0 per taglio da mettere direttamente in *Cashbox*, oppure un *Numero*>0 per taglio in *Recycling* e con valore massimo pari a *Numero*.

```

<?xml version="1.0" encoding="utf-8"?>
<request type="coinCfg">
  <item>
    <param value="denomination">20</param>
    <param value="quantity">-1</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">50</param>
    <param value="quantity">0</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">100</param>
    <param value="quantity">2</param>
    <param value="currency">EUR</param>
  </item>
  ...
</request>

```

- **noteMaxLevel:** permette la configurazione del livello massimo di banconote nel cassetto, oltre tale valore saranno reindirizzate direttamente nella cashbox

```

<?xml version="1.0" encoding="utf-8"?>
<request type="noteMaxLevel">
  <item>

```

```

        <param value="denomination">1000</param>
        <param value="quantity">3</param>
        <param value="currency">EUR</param>
    </item>
    ...
</request>

```

- **noteMinLevel:** permette la configurazione del livello minimo di banconote nel cassetto, se effettuato un comando di *empty partial* tale livello sarà mantenuto nel riciclatore

```

<?xml version="1.0" encoding="utf-8"?>
<request type="noteMinLevel">
    <item>
        <param value="denomination">500</param>
        <param value="quantity">1</param>
        <param value="currency">EUR</param>
    </item>
    <item>
        <param value="denomination">1000</param>
        <param value="quantity">1</param>
        <param value="currency">EUR</param>
    </item>
    ...
</request>

```

- **noteRoute:** permette la configurazione dell'instradamento delle banconote nel cassetto, i valori ammissibili sono RE: *Recycling*, CB: *Cashbox*, DS: *Disabled*

```

<?xml version="1.0" encoding="utf-8"?>
<request type="noteRoute">
    <item>
        <param value="denomination">500</param>
        <param value="route">DS</param>
        <param value="currency">EUR</param>
    </item>
    <item>
        <param value="denomination">1000</param>
        <param value="route">RE</param>
        <param value="currency">EUR</param>
    </item>
    <item>
        <param value="denomination">20000</param>
        <param value="route">CB</param>
        <param value="currency">EUR</param>
    </item>
    ...
</request>

```

- **noteCfg:** al fine di ottenere una rapida configurazione della *route* delle banconote o del livello massimo viene fornita una richiesta specifica che indichi al sistema, per ciascun taglio, se si vuole abilitare/disabilitare o fissarne il valore massimo. Questo avviene passando nel tag *param* con value *quantity* i seguenti valori: -1 per taglio disabilitato, 0 per taglio da

mettere direttamente in *Cashbox*, oppure un *Numero*>0 per taglio in *Recycling* e con valore massimo pari a *Numero*.

```
<?xml version="1.0" encoding="utf-8"?>
<request type="noteCfg">
  <item>
    <param value="denomination">500</param>
    <param value="quantity">-1</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">1000</param>
    <param value="quantity">0</param>
    <param value="currency">EUR</param>
  </item>
  <item>
    <param value="denomination">2000</param>
    <param value="quantity">2</param>
    <param value="currency">EUR</param>
  </item>
  ...
</request>
```

- **getRoute:** permette di ottenere la configurazione della *route* di tutte i tagli gestiti dal SafeMoney

```
<?xml version="1.0" encoding="utf-8"?>
<request type="getRoute"/>
```

La risposta sarà del tipo:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="getRoute" request_id="1234" timestamp="20181128_163338" code="ok">
  <param value="1" type="coin" reference="RE" min="0" max="0" />
  <param value="2" type="coin" reference="RE" min="0" max="0" />
  <param value="5" type="coin" reference="RE" min="0" max="0" />
  ...
  <param value="50000" type="note" reference="CB" min="0" max="0" />
</response>
```

- **getLevels:** permette di ottenere l'inventario macchina nel momento della richiesta sia in cashbox che in recycling:

```
<?xml version="1.0" encoding="utf-8"?>
<request type="getLevels"/>
```

La risposta sarà del tipo:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="GETLEVELS" timestamp="20181129_111316" code="ok">
  <param value="1" type="coin" reference="CB">12</param>
  <param value="2" type="coin" reference="CB">23</param>
  <param value="5" type="coin" reference="CB">54</param>
  ...
</response>
```

```

<param value="1" type="coin" reference="RE">43</param>
<param value="2" type="coin" reference="RE">23</param>
<param value="5" type="coin" reference="RE">12</param>
...
<param value="500" type="note" reference="CB">10</param>
<param value="1000" type="note" reference="CB">35</param>
...
<param value="500" type="note" reference="RE">24</param>
<param value="1000" type="note" reference="RE">21</param>
...
</response>

```

- **getLevels.txt:** per compatibilità con i precedenti driver SafeMoney, è stata mantenuta la modalità di richiesta inventario attraverso il file *getLevels.txt*, a questo il sistema risponde con il file *levels.xml* contenente le medesime informazioni ottenute a partire dal *CMD_REQUEST.xml* di tipo *getLevels* ma organizzato secondo il seguente schema:

```

<?xml version="1.0" encoding="utf-8"?>
<levels timestamp="20181130_104951" code="ok">
  <cashbox>
    <channel1 value="1">23</channel1>
    <channel2 value="2">43</channel2>
    ...
  </cashbox>
  <coins>
    <channel1 value="1">65</channel1>
    <channel2 value="2">43</channel2>
    ...
  </coins>
  <stacker>
    <channel1 value="500">3</channel1>
    <channel2 value="1000">6</channel2>
    ...
  </stacker>
  <notes>
    <channel1 value="500">12</channel1>
    <channel2 value="1000">33</channel2>
    ...
  </notes>
</levels>

```

- **loadStart:** permette di impostare il SafeMoney in modalità di caricamento

```

<?xml version="1.0" encoding="utf-8"?>
<request type="loadstart" id="123" />

```

La risposta sarà del tipo:

```

<?xml version="1.0" encoding="utf-8"?>
<response type="loadStart" request_id="123" timestamp="20181128_163338" code="ok"/>

```

Durante questa fase è possibile conoscere l'importo caricato leggendo il contenuto del file **loading.txt**

- **loadEnd**: permette di terminare la fase di caricamento del SafeMoney

```
<?xml version="1.0" encoding="utf-8"?>
<request type="loadEnd" id="123" />
```

La risposta sarà del tipo:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="loadEnd" request_id="123" value="1200" timestamp="20210122_171828"
code="ok">
  <param value="1" type="coin">0</param>
  <param value="2" type="coin">0</param>
  <param value="5" type="coin">0</param>
  <param value="10" type="coin">0</param>
  <param value="20" type="coin">0</param>
  <param value="50" type="coin">0</param>
  <param value="100" type="coin">6</param>
  <param value="200" type="coin">3</param>
  <param value="500" type="note">0</param>
  <param value="1000" type="note">0</param>
  <param value="2000" type="note">0</param>
  <param value="5000" type="note">0</param>
  <param value="10000" type="note">0</param>
  <param value="20000" type="note">0</param>
  <param value="50000" type="note">0</param>
</response>
```

- **emptyPartial**: permette di effettuare lo svuotamento a livello Min (livello *fondocassa*) il SafeMoney nella parte di device monete e/o nella parte di device banconote semplicemente indicando lo specifico *deviceType*

```
<?xml version="1.0" encoding="utf-8"?>
<request type="emptyPartial">
  <item>
    <param value="deviceType">COIN</param>
  </item>
  <item>
    <param value="deviceType">NOTE</param>
  </item>
</request>
```

La risposta sarà del tipo:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="emptyPartial" timestamp="20181128_163338" code="ok"/>
```

- **emptyTotal**: permette di effettuare lo svuotamento totale il SafeMoney nella parte di device monete e/o nella parte di device banconote semplicemente indicando lo specifico *deviceType*

```
<?xml version="1.0" encoding="utf-8"?>
<request type="emptyTotal">
```

```

<item>
    <param value="deviceType">COIN</param>
</item>
<item>
    <param value="deviceType">NOTE</param>
</item>
</response>

```

La risposta sarà del tipo:

```

<?xml version="1.0" encoding="utf-8"?>
<response type="EmptyTotal" timestamp="20181128_163338" code="ok"/>

```

- **fastWithdrawal:** permette di effettuare un prelievo dal SafeMoney di un richiesto valore indipendentemente dallo specifico *deviceType*

```

<?xml version="1.0" encoding="utf-8"?>
<request type="fastWithdrawal" id="123">
    <item>
        <param value="toBePaid">1075</param>
        <param value="description">Cassa 1 – TEST WITHDRAWAL</param>
    </item>
</request>

```

La risposta sarà del tipo:

```

<?xml version="1.0" encoding="utf-8"?>
<response type="fastWithdrawal" request_id="123" value="1075"
timestamp="20210122_162327" code="ok" />

```

- **withdrawal:** permette di effettuare un prelievo dal SafeMoney nella parte di device monete e/o nella parte di device banconote semplicemente indicando lo specifico *deviceType*

```

<?xml version="1.0" encoding="utf-8"?>
<request type="withdrawal">
    <item>
        <param value="denomination">500</param>
        <param value="quantity">2</param>
        <param value="deviceType">NOTE</param>
    </item>
    <item>
        <param value="denomination">100</param>
        <param value="quantity">1</param>
        <param value="deviceType">COIN</param>
    </item>
</request>

```

La risposta sarà del tipo:

```

<response type="withdrawal" value="600" request_id="123"
timestamp="20210122_161526" code="ok">
    <param value="500" type="NOTE" dispensed="1" notdispensed="1" requested="2" />
    <param value="100" type="COIN" dispensed="1" notdispensed="0" requested="1" />
</response>

```

- **settings:** permette di effettuare la configurazione del FileAdapter e di visualizzare o meno la UI associata all'applicativo stesso

```
<?xml version="1.0" encoding="utf-8"?>
<request type="settings">
  <item>
    <param value="show_ui">true</param>
  </item>
</request>
```

La risposta sarà del tipo:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="settings" timestamp="2018012_142112" code="ok"/>
```

GESTIONE DEGLI ERRORI

La gestione degli errori viene effettuata sia nel caso dei comandi operativi che di configurazione in modo da avere sempre sotto controllo qualsiasi tipologia di segnalazione avvenga dal dispositivo SafeMoney.

Come già anticipato, il file di *status.txt* contiene puntualmente lo stato del dispositivo in termini di codice e descrizione e, nel caso di segnalazioni da parte del SafeMoney, anche la descrizione della segnalazione stessa. Ad esempio, nel caso di segnalazione di tipo “cassetto aperto”, il file in questione conterrà la dicitura “0001: IN CONNESSIONE - UNITA' BANCONOTE: cassetto mancante”. Viceversa, per ogni richiesta inviata al sistema attraverso il file CMD_REQUEST.xml nella risposta ricevuta è possibile che sia presente un tag *error* nel cui attributo *type* è presente il messaggio inviato dal sistema; per ulteriori approfondimenti, al suo interno è presente il tag *CDATA* con l'eventuale *stackTrace* che lo ha generato.

La risposta sarà del tipo:

```
<?xml version="1.0" encoding="utf-8"?>
<response type="GETLEVELS" timestamp="20181129_122308" code="nok">
  <error type="Unable to process any command during offline
status"><![CDATA[System.ArgumentException: Unable to process any command during offline
status in SM.dotnet.driver.adapters.FileAdapter.manageCommand(CommandRequest cmd, String
cmdAsString) in
E:\ditron\drivers\DotNetDriverSafeMoneyAdapter\head\src\DotNetDriverSafeMoneyAdapter\adapters\
FileAdapter.cs:riga 440]]></error>
  <request><![CDATA[<?xml version="1.0" encoding="utf-8"?>
<request type="GETLEVELS">
</request>]]></request>
</response>
```

DESCRIZIONE DELLA MACCHINA A STATI

Al fine di ottenere una schematizzazione del comportamento del sistema *FileAdapter* assunto durante l'intero ciclo di vita, di seguito viene riportato uno state diagram che sintetizza il modello di stato del *FileAdapter* durante l'invio e la ricezione sia dei comandi operativi sia di quelli di configurazione.

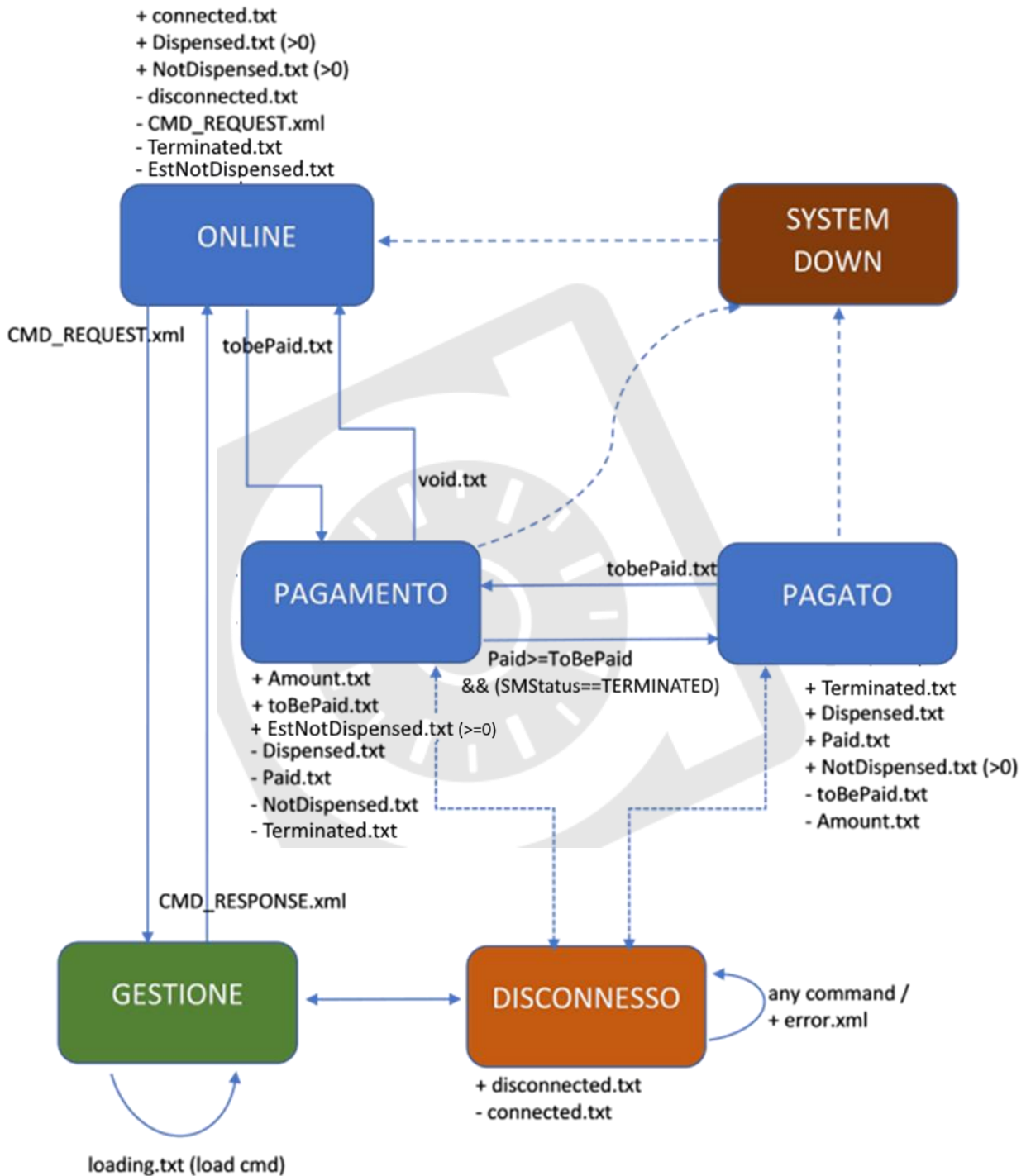


Figura 1 - FileAdapter: State Diagram

In blu sono riportati gli stati relativi ai comandi operativi, in verde lo stato assunto durante l'invio di comandi di configurazione mentre in arancione gli stati di Disconnessione del *FileAdapter* dal SafeMoney e di "System Down", ovvero di non funzionamento del *FileAdapter* stesso.

APPENDICE: XML Schema Definition per Request e Response

Di seguito l' XSD per il file di richiesta CMD_REQUEST.xml:

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="request">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="item" maxOccurs="unbounded" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="param" maxOccurs="unbounded" minOccurs="0">
                <xs:complexType>
                  <xs:simpleContent>
                    <xs:extension base="xs:string">
                      <xs:attribute type="xs:string" name="value"
use="optional"/>
                    </xs:extension>
                  </xs:simpleContent>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute type="xs:string" name="type"/>
      <xs:attribute type="xs:string" name="id"/>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Di seguito l' XSD per il file di risposta CMD_RESPONSE.xml:

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="response">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="param" maxOccurs="unbounded" minOccurs="0">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:byte">
                <xs:attribute type="xs:byte" name="value" use="optional"/>
                <xs:attribute type="xs:string" name="type" use="optional"/>
                <xs:attribute type="xs:string" name="reference" use="optional"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
</xs:element>
<xs:element name="error">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute type="xs:string" name="type"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element type="xs:string" name="request"/>
</xs:sequence>
<xs:attribute type="xs:string" name="type"/>
<xs:attribute type="xs:string" name="timestamp"/>
<xs:attribute type="xs:string" name="code"/>
</xs:complexType>
</xs:element>
</xs:schema>
```

