Ondems SI-Manager – Release 1.3
Real-Time Management of DVB-SI Tables

1. Overview:

One of the digital television's many advantages is that additional information can be provided about the media, programs or events. This data is used by the receiver Set-Top Box (STB) to provide information to the viewers through the Electronic Program Guide (EPG). This additional data used by the Set-Top Boxes is transmitted in the DVB SI (Service Information) tables.
Digital television companies that broadcast and/or redistribute the MPEG-2 digital content are often confronted with the problem of managing DVB SI tables in order to maintain a DVB SI compliant data stream. These problems occur in digital broadcast in order to create the specific information associated with the digital content or in redistribution of the digital television when trying to multiplex/de-multiplex different transponders.
The *Ondems SI Manager* is a scalable and fully customizable solution designed to help professional broadcast companies as well as small digital television operators to create and manage the DVB SI content.

1.1. Product description:

Ondems SI-Manager is capable of generating EIT, TOT and TDT DVB-SI tables and capable of modifying the EIT tables. Its capabilities make it fit for several purposes:
1. Creation of EPG (Electronic Program Guide)
2. Multiplexing channels coming from different transponders into a single multiplex
3. Sending time and date (including timezone) information to Set-Top Boxes (STB) inside the network.

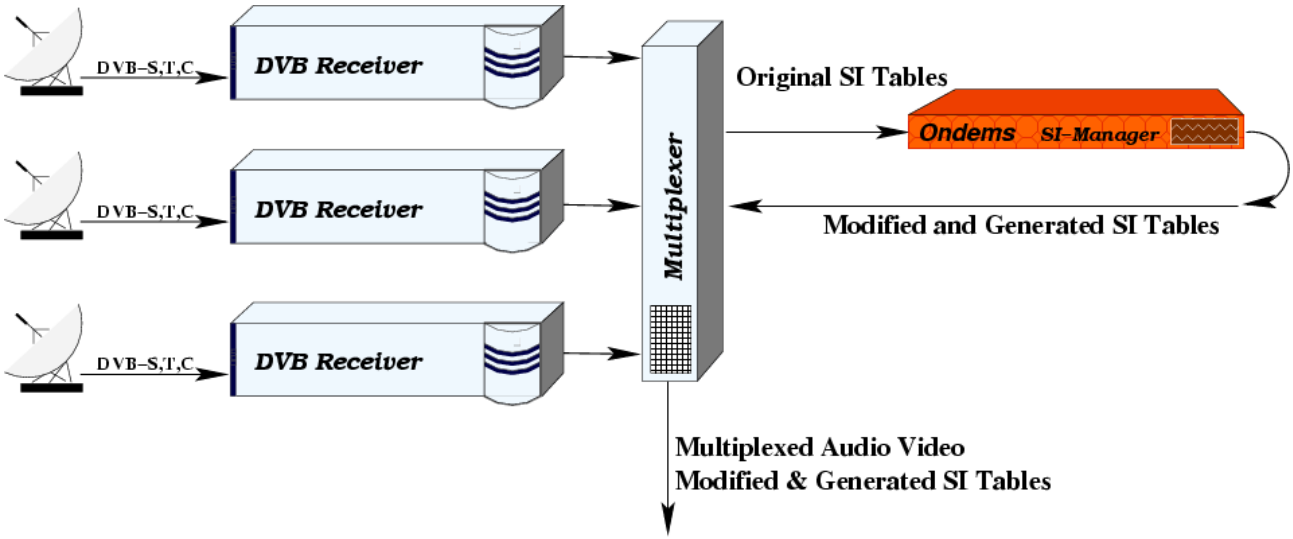

Figure 1: The use of Ondems SI-Manager in TV redistribution head-ends

The Ondems SI-Manager receives MPEG-2 TS packets containing SI tables. The received SI tables are processed according to the configuration and sent back into the system (multiplexer in Fig. 1).

1.2. Features (Technical specifications):

Some of the options and features presented are customizable depending on the client requirements. The features marked with (*) are currently under development. Please contact us for further details.

Ondems MediaGateway	
Target	Digital Television Systems: Cable Operators, Content providers
Media Formats	DVB compliant MPEG-2 Transport Stream: • Multi-Program Transport Stream (MPTS) • Single-Program Transport Stream (SPTS)

Input/Output	• DVB ASI (DEKTEC DTA-120, DTA-140)
	• ATM AAL0/AAL5 electrical/optical
	• IP Ethernet 10/100/1000 Base-TX (*)
	  • UDP (*)
	  • Multicast (*)
	• Up to 4 inputs/outputs

Output rate control	• For modified SI tables: same as input • For created EIT tables: preconfigured at 100 tables/second • For created TOT and TDT tables: preconfigured at 1 table/second
SI Table creation	• EIT • TOT • TDT

SI Table modification	• EIT:
	  • TSID (transport_stream_id)
	  • ONID (original_network_id)
	  • SID (service_id)
	  • Table ID (table_id)

Supported Linux Distributions (tested)	• Mandrake 9.x, 8.2 (off the shelf) • SuSE 9.0, 8.x (off the shelf) • RedHat 9.0 (off the shelf) • Debian Woody (requires updating some libraries) • Other distributions should work as well. Please contact Ondems for details and support

1.3. Added features and major modifications

1.3.1. From version 1.0 to version 1.1:

- SID modification for EIT tables
- Creation of EIT tables

From version 1.1 to version 1.2:

- Creation of TOT and TDT tables

1.3.2. From version 1.2 to version 1.3:

- Added support for ASI Input/Output

2. Installation

Ondems SI-Manager can be delivered in 3 ways:
1. A fully installed and preconfigured box (recommended option).
2. Installed and preconfigured on customer's hardware (PC running Linux)
3. As a software package (RPM).

After the installation, the SI-Manager is configured as a service in the Linux OS. The service is automatically started in the run-levels 3 or 5 and stopped in run-levels 0, 1, 2 and 6. However, the service might be disabled if the appropriate configuration file does not exist. The product has been successfully tested with various versions of the following Linux distributions: Mandrake (9.1, 9.2), SuSE (9.0), RedHat (9.0) and Debian Woody.

2.1. Prerequisites

The host PC that is running the SI-Manager should feature a processor faster than 1 GHz and at least 128 RAM. A recent Linux version has to be supported and installed.
In addition to this, the drivers for the appropriate I/O interfaces have to be installed. The drivers for the supported ATM cards are part of the Linux kernel and they are automatically loaded by the SI-Manager at startup. The drivers for the DEKTEC ASI cards (Dta1xx) are not part of the Linux kernel, and they have to be installed separately.

2.2. Installing the SI-Manager

1. Install the rpm package:

   ```
   rpm -i si-manager-1.2-1.i586.rpm
   ```

2. Modify the configuration file:

   ```
   /etc/si-manager/si-manager.config
   ```

3. Restart the SI-Manager service:

   ```
   /etc/init.d/si-manager restart
   ```

4. Check the log file:

   ```
   tail -f /var/log/si-manager.log
   ```

2.3. Quick Testing Guide:

1. Connect the input to the SI-Manager
2. Check the log file:

   ```
   tail -f /var/log/si-manager.log
   ```

3. If the log file does not specify any errors, check the output streams

2.4. Files Location:

SI-Manager files are located in:

```
/usr/local/Ondems/si-manager
```

SI-Manager binary:

```
/usr/local/Ondems/si-manager/si-manager
```

SI-Manager service script:

```
/etc/init.d/si-manager
```

Configuration file (the service is disabled if this file does not exist):

```
/etc/si-manager/si-manager.config
```

Examples of configuration files are located in:

```
/usr/local/Ondems/si-manager/config/*
```

Installed man pages (in /usr/local/man):

```
si-manager (1)
si-manager.config (5)
si-manager.event (5)
```

When running, Ondems SI-Manager will also create the following log files:

```
/var/log/si-manager.log
/var/log/si-manager.err
```

2.5. Starting, stopping and verifying the SI-Manager:

Starting the SI-Manager:

```
/etc/init.d/si-manager start
```

Stopping the SI-Manager:

```
/etc/init.d/si-manager stop
```

Verify if the SI-Manager is running:

```
/etc/init.d/si-manager status
```

Note: Please always use the above commands (scripts) to start and stop the SI-Manager. In addition to starting/stopping the SI-Manager, the above commands (scripts) also load/unload the ASI drivers.

Starting the SI-Manager executable (si-manager) alone may lead to failure because the ASI driver is not loaded by the Linux kernel.

3. Configuration

Ondems SI-Manager supports an xml-based configuration file. In addition to this file, additional xml-based files specify EPG/EIT creation data (if needed). The xml-based configuration files can be generated using an external application (GUI). The biggest advantage of this approach is that the process of generating the configuration does not interfere in any way with the operation of the SI-Manager, ensuring to it a greater reliability. When the new configuration is ready, it can replace the old configuration.

3.1. Description of the configuration file format

The configuration file is located in:

```
/etc/si-manager/si-manager.config
```

and has xml format. The configuration file is composed of 4 parts:

1. The xml header
2. The time specification part, used for generating TOT and TDT tables
3. The EIT modification specification part, used for modifying EIT tables
4. The EIT generation specification part, used for generating EIT tables

The next subsections describe each of these parts. In the provided examples, the fields highlighted with bold characters are example values that may be changed in real configurations.

3.1.1. Supported Input/Output URLs

The SI-Manager can receive information from several input types, process it and send it to several output types. Currently, the supported inputs are ATM and local file, and the supported output is ATM. In the future, support for ASI and IP will be added. Each input or output is specified in the configuration file by an URL.
Currently, the SI-Manager supports several types of URLs:

1. File URL:
   `url="file:///absolute/path/to/file"`
   Example:
   `url="file:///etc/si-manager/EPG/today.config"`
2. ATM AAL5 Interface VPI/VCI URL:
   `url="atm://interface.vpi.vci"`
   Example:
   `url="atm://0.0.100"`

In addition to the above URLs, two other URLs will be supported in the near future:

3. ASI card/channel URL:
   `url="asi://card_type.card_no.channel_no"`
   Currently the following DVB-ASI cards are supported:
   Dektec DTA-100 OUT (card_type=100)
   Dektec DTA-120 IN (card_type=120)
   Dektec DTA-140 IN/OUT (card_type=140)
   Example:
   `url="asi://120.0.0"`
4. IP Multicast URL:
   `url="mcast://ip_address:port"`
   Example:
   `url="mcast://239.252.1.1:5555"`

3.1.2. The xml header.

This part has two lines. The first line describes the character encoding. The second line specifies the lowest version of the SI-Manager software that completely understands this configuration file. Below is an example of such header:

```
<?xml version="1.1" encoding="UTF-8"?>
<simanager name="SI-Manager" version="1.1">
```

The above header specifies the lowest SI-Manager version for this configuration file as being 1.1. This means that the configuration file does not contain configuration for features that were added in versions greater than 1.1. For example, the configuration file should not contain any time-specification part (used for generating TOT and TDT tables), since this generation was added in the version 1.2.

3.1.3. The time specification part

This part defines an arbitrary number of TOT tables. Each table has a name, which is used for any reference to that table. Currently, this name is used for specifying which TOT data should be used when sending TOT tables on specific outputs. Each TOT may have several timezones. Each timezone has:

- A 3 letters country code. This code identifies the country of the timezone,
- A country region ID. This is used for countries that span across several timezones (example: Russia, US, Canada). For countries with only one timezone, this code is 0.
- Offset polarity. This can be 0 or 1. The 0 offset is used for timezones east of UTC (Universal Time, also known as GMT, or Greenwich Mean Time), where a number of hours is added to the GMT time. The offset 1 is used for timezones west of UTC, where a number of hours is subtracted from the GMT time.
- Time offset. The time offset specifies the region's offset from UTC (GMT).
- Date and time of change to summer or winter time.
- The new time offset after the change to summer or winter time.

Here it is an example of a time specification part that describes the winter time TOT for two countries, Finland and Sweden. The date for the change to summer time is the actual date this event happen in 2004:

```
<timespec>
  <tot name="FinSwe_Winter">
    <region three_letter_country_code="fin" country_region_ID_hex="0" time_offset_polarity="0">
      <time_offset  hrs="02"  min="00"/>
      <time_of_change  day="28"  month="03"  year="2004"  hrs="02"  min="00"  sec="00"/>
      <next_time_offset  hrs="03"  min="00"/>
    </region>
    <region three_letter_country_code="swe" country_region_ID_hex="0" time_offset_polarity="0">
      <time_offset  hrs="01"  min="00"/>
      <time_of_change  day="28"  month="03"  year="2004"  hrs="02"  min="00"  sec="00"/>
      <next_time_offset  hrs="02"  min="00"/>
    </region>
  </tot>
</timespec>
```

In the above example, the name of the specified TOT data is FinSwe_Winter. The example specifies that the local time in Finland during the winter is GMT+2, and after the change to summer time, which happens on 28th of March at 2 AM, the local time will be GMT+3. For Sweden, the example specifies that the winter local time is GMT+1, and after the change to summer time, the local time is GMT+2.
The GMT date and time necessary for the creation of TOT and TDT tables is taken from the local machine clock. The TOT and TDT tables are broadcasted once every second on the specified outputs (see next sections).

3.1.4. The EIT modification specification part

This part defines how to modify EIT tables coming from different inputs and to what outputs they should be sent after the modification. Currently, the SI-Manager supports the modification of:

- TSID (Transport Stream ID)
- ONID (Original Network ID)
- SID (Service ID)
- table_id
- version_number (modified automatically)

inside the EIT tables.
Below it is a simple example of how to modify the EIT tables that correspond to two TV channels:

```
<input  url="atm://0.0.100"  type="transport-stream">
  <service  pid="0x12"  service_id="33">
    <change  service_id="34"  transport_stream_id="1000"  original_network_id="2000"/>
    <output  url="atm://0.0.200"  type="transport-stream"  mode="cbr/128kbps" network="actual"
             tot="present" tdt="present" tot_name="FinSwe_Winter"/>
  </service>
  <service  pid="0x12"  service_id="65">
    <change  transport_stream_id="1000"  original_network_id="2000"/>
    <output  url="atm://0.0.200"  type="transport-stream"  mode="cbr/128kbps" network="other"
             tot="present" tdt="present" tot_name="FinSwe_Winter"/>
  </service>
</input>
```

The input tag specifies the URL of the input and the type of the stream. Each input supports several services (TV channels). A service is identified by its service id and by the PID of the EIT SI table – normally 18 (0x12). For each service, the new values for TSID, ONID and SID can be specified. The modified EIT tables are forwarded to one (or

several) outputs, specified URLs. The URL tag may specify also the allocated bandwidth for the output and if TOT and TDT tables should be present on that output.

A more complex example is presented below:

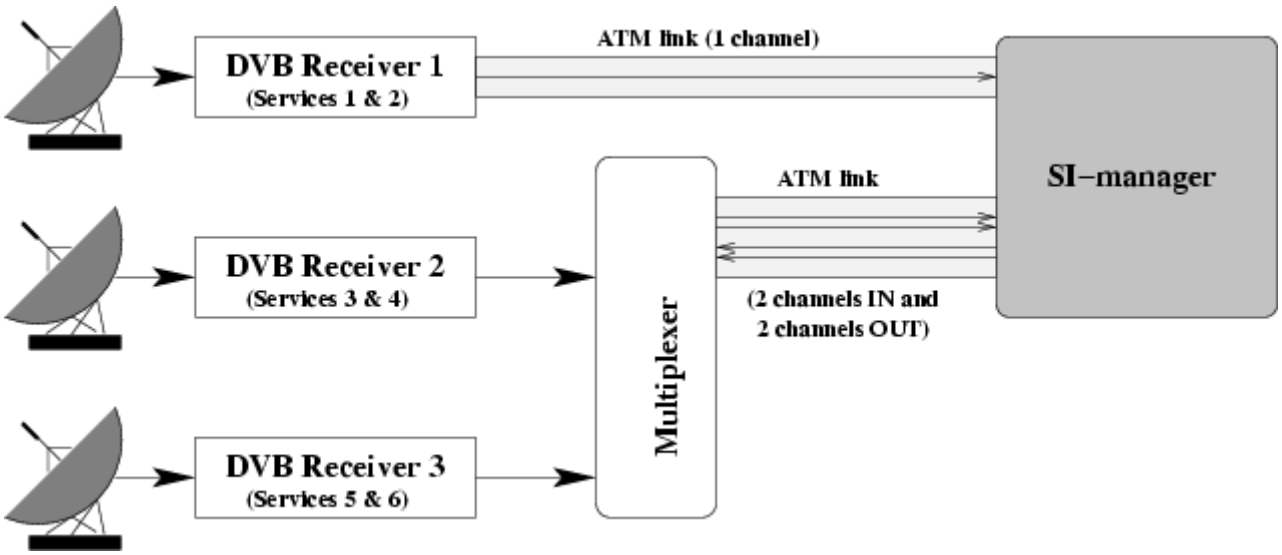

Figure 2: Example of DVB-C Channel creation out of Services coming from multiple Transponders

The system receives 3 Transponders from satellite, and each Transponder has two Services (TV channels). We want to create two DVB-C Channels as follows: Channel1 should contain Services 1, 2 and 3, and Channel 2 should contain Services 4, 5 and 6. Both two channels should have correct EIT tables and each of them should also have the EIT tables of the other Channel, so that viewers can display a complete schedule on their STB. The desired configuration is presented if Fig. 3 below.

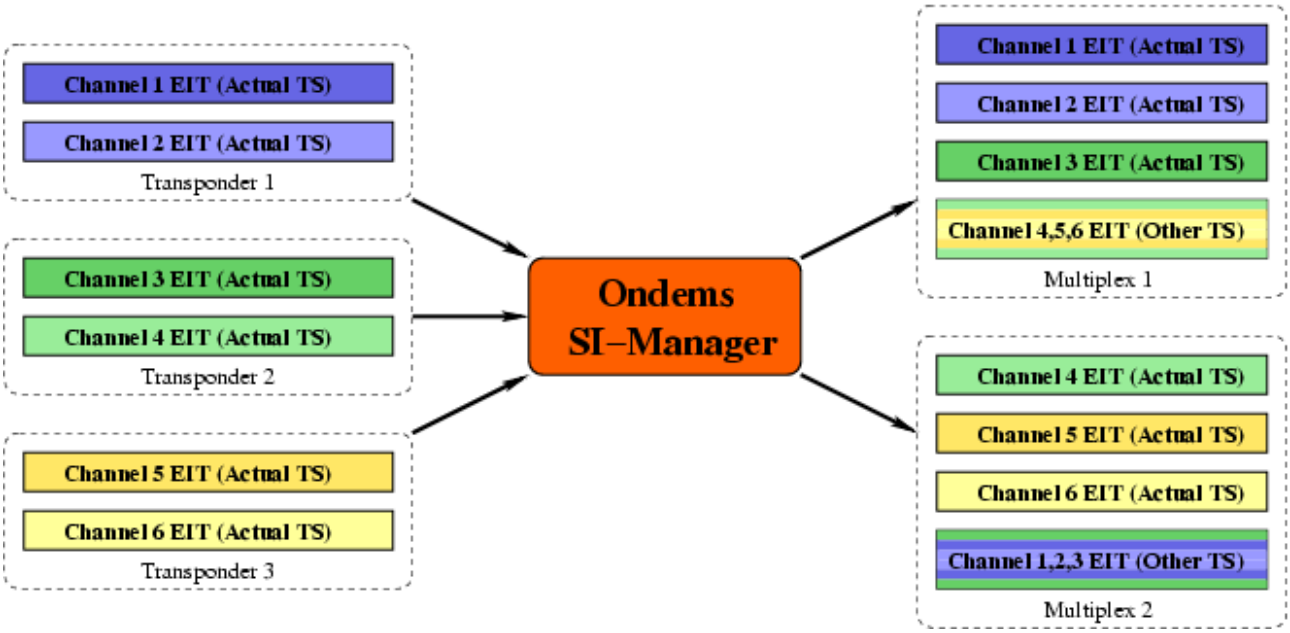

Figure 3: Example showing the remultiplexing of 6 TV Channels from 3 Satellite Transponders into 2 DVB-C Channels

Services 1 and 2 come from the DVB Receiver 1 to the SI-Manager on an ATM link with one channel. Services 3, 4, 5 and 6 come from the Multiplexer to the SI-Manager on the second ATM link. Each of the two Transponders coming from the Multiplexer uses a separate ATM channel.
After the EIT processing, the SI-Manager should send back the EIT tables to the Multiplexer using the second ATM link. The EIT tables for each DVB-C Channel should be sent on a separate ATM channel.
This system requires the following section for EIT modification:

```
<input  url="atm://0.0.101"  type="transport-stream">
  <service  pid="0x12"  service_id="1">
    <change  transport_stream_id="1"  original_network_id="100"/>
    <output  url="atm://1.0.201"  type="transport-stream"  mode="cbr/128kbps" network="actual"/>
    <output  url="atm://1.0.202"  type="transport-stream"  mode="cbr/128kbps" network="other"/>
  </service>
  <service  pid="0x12"  service_id="2">
    <change  transport_stream_id="1"  original_network_id="100"/>
    <output  url="atm://1.0.201"  type="transport-stream"  mode="cbr/128kbps" network="actual"/>
    <output  url="atm://1.0.202"  type="transport-stream"  mode="cbr/128kbps" network="other"/>
  </service>
</input>
<input  url="atm://1.0.110"  type="transport-stream">
  <service  pid="0x12"  service_id="3">
    <change  transport_stream_id="1"  original_network_id="100"/>
    <output  url="atm://1.0.201"  type="transport-stream"  mode="cbr/128kbps" network="actual"/>
    <output  url="atm://1.0.202"  type="transport-stream"  mode="cbr/128kbps" network="other"/>
  </service>
  <service  pid="0x12"  service_id="4">
    <change  transport_stream_id="2"  original_network_id="100"/>
    <output  url="atm://1.0.201"  type="transport-stream"  mode="cbr/128kbps" network="other"/>
    <output  url="atm://1.0.202"  type="transport-stream"  mode="cbr/128kbps" network="actual"/>
  </service>
</input>
<input  url="atm://1.0.111"  type="transport-stream">
  <service  pid="0x12"  service_id="5">
    <change  transport_stream_id="2"  original_network_id="100"/>
    <output  url="atm://1.0.201"  type="transport-stream"  mode="cbr/128kbps" network="other"/>
    <output  url="atm://1.0.202"  type="transport-stream"  mode="cbr/128kbps" network="actual"/>
  </service>
  <service  pid="0x12"  service_id="6">
    <change  transport_stream_id="2"  original_network_id="100"/>
    <output  url="atm://1.0.201"  type="transport-stream"  mode="cbr/128kbps" network="other"/>
    <output  url="atm://1.0.202"  type="transport-stream"  mode="cbr/128kbps" network="actual"/>
  </service>
</input>
```

From the above configuration example, we can distinguish the 3 input URLs. For the ATM card “0” we are using the channel 101, and for the ATM card 1 we are using the channels 110 and 111. Each of the mentioned ATM channels receives EIT tables corresponding to their two services. The configuration specifies that the ONID for all Services should be changed to 100. In addition to this, the TSID for Services 1, 2 and 3 should be changed to 1 (they will be broadcasted as DVB-C Channel1) and the TSID for Services 4, 5 and 6 should be changed to 2 (they will be broadcasted as DVB-C Channel2). Each modified EIT is sent on both output ATM channels (201 and 202 of card “1”), because each DVB-C Channel should have both its EPG information and also the other Channel's EPG, so that viewers can display a complete schedule on their STB. EIT Tables from Services 1, 2 and 3, sent to DVB-C Channel 1 have their “table_id “ set to “actual”. The same services have the “table_id” set to “other” when they are sent to DVB-C Channel 2. In a similar way, EIT Tables from Services 4, 5 and 6, sent to DVB-C Channel 2 have their “table_id “ set to “actual”. These services have the “table_id” set to “other” when they are sent to DVB-C Channel 1.

3.1.5. The EIT generation specification part

This part defines how to generate EIT tables. The EPG information resides in separate files. The main configuration file specifies these EPG information files as input URL. One such EPG information file contains EPG information for several transponders/multiplexes. The SI-Manager will create EIT tables for all transponders/multiplexes present in the file. These EIT tables are then all sent on each transponder/multiplex. On each such transponders, the SI-Manager will create proper “Actual transport stream” and “Other transport stream” EIT tables.
This concept is illustrated in the figure below:

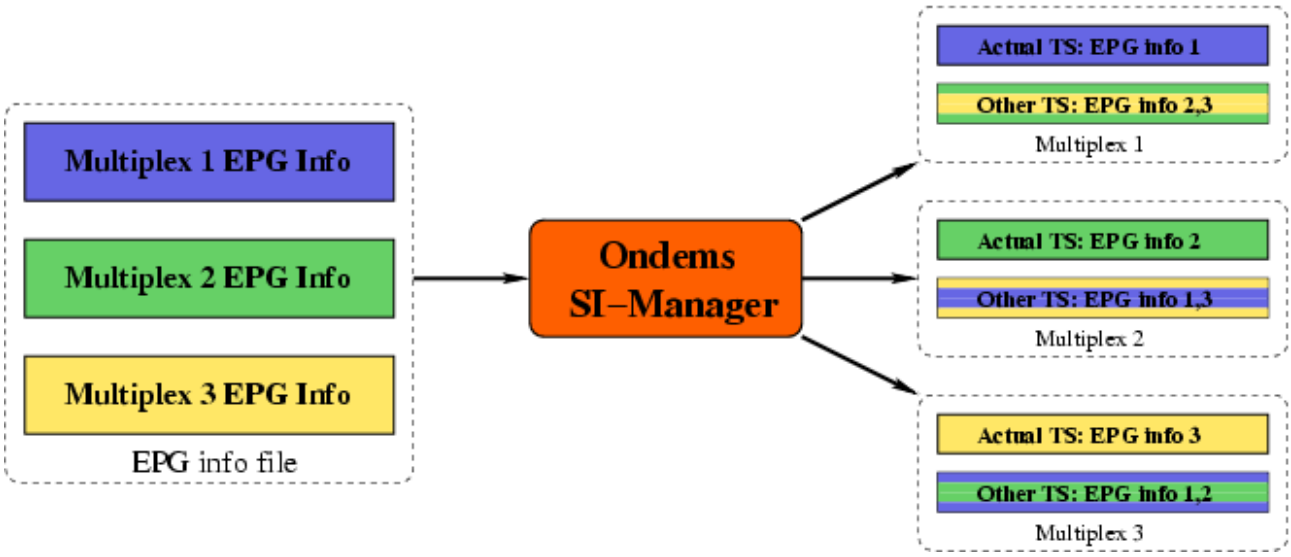

Figure 4: EIT creation for several Transponders from the same info file

In the above configuration, each Transponder has its own TSID (suppose that for this example, they are 1, 2 and respectively 3). On Output 1 all EIT tables generated from the Transponder 1 EPG info should have their "table id" set to "actual TS", while all other EIT tables (generated from Transponders 2 and 3 EPG info) should have their "table id" set to "other TS". Similarly, for Output 2, all EIT tables generated from Transponder 2 EPG info should have their "table id" set to "actual TS", while all other EIT tables (generated from Transponders 1 and 3 EPG info) should have their "table id" set to "other TS".
The following configuration section corresponds to the above example:

```
<input  url="file:////etc/si-manager/ondems.event"  type="event-file">
  <service  pid="0x12"  transport_stream_id="1"  original_network_id="100">
    <output  url="atm://0.0.101"  type="transport-stream"  mode="cbr/128kbps"/>
  </service>
  <service  pid="0x12"  transport_stream_id="2"  original_network_id="100">
    <output  url="atm://0.0.102"  type="transport-stream"  mode="cbr/128kbps"/>
  </service>
  <service  pid="0x12"  transport_stream_id="3"  original_network_id="100">
    <output  url="atm://0.0.103"  type="transport-stream"  mode="cbr/128kbps"/>
  </service>
</input>
```

Each service tag identifies one Transponder/Multiplex from the EPG information file, by specifying its TSID and ONID. The specified Transponder/Multiplex identifies the "actual TS". The rest of the Transponders/Multiplexes will be considered the "other TS".

The EPG information files have the following format:

```
<network name="name">
  <service  id="id"  transport_stream_id="id"  original_network_id="id"  valid="now">
    <event  id="id"  start_time="dd/mm/yy  hh:mm:ss"  duration="hh:mm:ss" freeCA="0/1">
      <content  dvb="id"  user="id"/>
      ...
      <component stream_content="id"  type="id"  tag="id"  language="code"
                    text_encoding="encoding"  text="text"/>
      ...
      <language  code="code">
        <short_event  name_encoding="encdng"  name="text"  text_encoding="encoding"  text="text"/>
        <extended_event  text_encoding="encoding"  text="text"/>
      </language>
      ...
      <parental_rating>
        <country code="code" rating="id"/>
      </parental_rating>
      ...
    </event>
    ...
```

```
    </service>
    ...
</network>
```

The above tags and fields have the following meaning:

- network: it is used to bind together several services. The network name is currently not used internally by the SI-Manager. However, it is recommended that services are grouped together .
- service: a service specifies one channel. The channel is identified by its Service ID, TSID and ONID.
- event: inside each channel, several events are defined. An event is a TV channel programing unit, like a movie, show or news. Each event is identified by an ID (it has to be unique inside the service), a starting time and day, a duration and a flag specifying if the event is broadcasted in clear or encrypted. Assigning IDs to events has to be done carefully, so that inserting a new event inside an already broadcasted schedule should not modify the event IDs of any other events. The reason for this is to maintain consistency for viewers that have scheduled the recording of an event based on its ID. Changing the ID of that event would make the viewer's STB to record another event, the one that has taken the ID of the initially scheduled event.
- content: this tag describes the event type. The dvb field has 8 bits. The first 4 bits encode the main event type (e.g. movie, news, show, sports, etc). The next 4 bits encode details about the main event type (e.g. for a movie: thriller, adventure, comedy, etc). The encoding of event types and details is described in the DVB document "Specification for Service Information (SI) in DVB systems", Table 18 (pg. 40-45). The user field also has 8 bits and it is specified by the broadcaster.
  There may be several content tags associated with an event, because the event may be described using more than one content descriptors. For example, a movie can be a science-fiction comedy, and thus requiring two content descriptors:
  - first to specify a science fiction movie (<content dvb="**0x0103**" user="**0**"/>) and
  - the second to specify a comedy (<content dvb="**0x0104**" user="**0**"/>)
- component: this tag describes each stream composing the event. The stream_content is a 4 bit field that encodes the main stream type (video, audio, data). The type field encodes additional information about the stream (e.g. aspect ratio and frame-rate for video). The encoding of event types and details is described in the DVB document "Specification for Service Information (SI) in DVB systems", Table 16 (pg. 38-39).
  The tag field has 8 bits and has the same value as the component_tag field in the stream identifier descriptor (if present in PMT) for the component stream.
  The language field specifies the language of the component (in case of audio or data). The language is specified as 3-letters language code.
  The text field provides a text description of the component. The character set that is used is specified in the text_encoding field.
- language: this tag is mainly a wrapper for two important tags inside it: the short_event tag and the extended_event tag. The language tag also specifies the language for these two tags (as 3-letters language code).
- short_event: this tag provides the name of the event (inside the name field) and a short description of the event (inside the text field). The name_encoding and text_encoding fields specify the character set used in the name and respectively text fields.
- extended_event: this tag provides an extended description of the event, inside the text field. The character set that is used is specified in the text_encoding field.
- parental_rating: this tag is a wrapper for several tags specifying the actual parental rating for different countries:
- country: this tag specifies the parental rating (inside the rating field) for a specified country (inside the code field, as 3-letters code). The possible parental ratings IDs are specified in the DVB document "Specification for Service Information (SI) in DVB systems", Table 55 (pg. 65).

Values for the character set encoding field:	
default	Default Latin character set
Cyrillic	ISO8859-5 Cyrillic character set
Arabic	ISO8859-6 Arabic character set
Greek	ISO8859-7 Greek character set
Hebrew	ISO8859-8 Hebrew character set
Nordic	ISO8859-9 French/German/Nordic character set
Korean	KSC5601 character set
ISO8859-n	ISO8859-n character set
multilingual	ISO10646 multilingual character set (Unicode)

Below is an example of a configuration specification for the “Harry Potter and the Philosopher's Stone” movie, scheduled to start on a channel at 20:00 on 17th of April, 2004. The movie is available in two languages, English and French.

```
<network name="net1">
  <service  id="1"  transport_stream_id="1"  original_network_id="100"  valid="now">
    <event  id="123"  start_time="17/04/04  20:00:00"  duration="2:32:00" freeCA="0">
      <content  dvb="0x0102"  user="0"/>
      <content  dvb="0x0103"  user="0"/>
      <component stream_content="0x01"  type="0x02"  tag="1"/>
                      text_encoding="default"  text="Video, 16:9, PAL "/>
      <component stream_content="0x02"  type="0x05"  tag="2"  language="eng"
                      text_encoding="default"  text="English audio track, surround"/>
      <component stream_content="0x02"  type="0x05"  tag="3"  language="fre"
                      text_encoding="default"  text="French audio track, surround"/>
      <component stream_content="0x03"  type="0x12"  tag="4"  language="fin"
                      text_encoding="default"  text="Finnish subtitles for 16:9"/>
      <language  code="eng">
        <short_event  name_encoding="default"  name="Harry Potter and the Philosopher's Stone"
                     text_encoding="default"  text="Great Movie, worth seeing"/>
        <extended_event  text_encoding="default"  text="Rescued from the outrageous neglect of his
                                                        aunt and uncle, a young boy with a great
                                                        destiny proves his worth while attending
                                                        Hogwarts School of Witchcraft and Wizardry."/>
      </language>
      <parental_rating>
        <country code="gbr" rating="0x8"/>
        <country code="fra" rating="0x8"/>
        <country code="fin" rating="0x8"/>
      </parental_rating>
    </event>
  </service>
</network>
```

The information from the above configuration example was taken from the Internet Movie Database (IMDB), at:
http://www.imdb.com/title/tt0241527/

4. Troubleshooting

Before reporting any problem, please verify the log file produced by the SI-Manager:

```
tail -f /var/log/si-manager.log
```

When reporting a problem, please also send the configuration file (`/etc/si-manager/si-manager.config`) and the log files (`/var/log/si-manager.log` and `/var/log/si-manager.err`)

5. Background Information

5.1. MPEG-2 Transport Stream Format

MPEG-2 combines one or more Packetized Elementary Streams which have a common time-base into a single stream. There are two formats for MPEG-2 Streams:

- MPEG-2 Program Stream is using packets of variable and relatively great lengths being suitable in error-free environment, such as non real-time software processing
- MPEG-2 Transport Stream is using 188 bytes fixed packet length being suitable for environments where errors are likely, such as storage or transmission in lossy or noisy media and real-time environment

5.2. Transport Stream Packets

MPEG-2 Transport Stream packets are 188 bytes length. Each packet has a header with an optional Adaptation Field structure and the payload. The important fields in the header are:

- *Synchronization byte* with the value of 0x47 in hexadecimal (71 in decimal)
- *PID* which select the stream component
- *Continuity Counter* incremented by one for each packet with the current PID
- *Program Clock Reference* is inserted in the Adaptation Field structure of the packet to provide a accurate 27Mhz clock for the stream

5.3. Transport Stream Structure

The MPEG-2 Transport Stream is composed by one or more Program Elementary Streams (PES), each of these streams is encapsulated in Transport Stream packets with different PID values. Apart from Program Elementary Streams in the Transport Stream there are Service Information tables encapsulated with the according PID values.
Some of the PID values are reserved for special Service Information tables. Value 0 is reserved for Program Association Table, 1 is reserved for Conditional Access Table, values between 2 and 31 are reserved for DVB Service Information Tables and the value of 8191 is reserved for Null packets. The Null packets are inserted in the stream to shape it to a constant bitrate form. These Null packets are discarded by the receivers.

5.4. Service Information Tables

Program Association Table (PAT) maintains the list of the programs (for each program the information that is stored is the PID value of the Program Map Table). Program Map Table (PMT) maintains the list of the PID values of the streams within the program (video, audio, subtitle). Each program is driven by its own PCR clock. The PCR PID is also provided in the PMT.
Apart from MPEG-2 Service Information tables, the DVB Service Information tables provide an extended range of services on top of the standard Service Information tables:

- Event Information Tables (EIT) provide Electronic Program Guide (EPG) for present / following / schedule events
- Time and Date Tables (TDT) provide universal time information for the programs
- Time Offset Tables (TOT) provide local and universal time information for the programs
- Network Information Tables (NIT) provide network characteristics and frequencies
- Service Description Tables (SDT) provide program name definition

All of these DVB Service Information tables are used by the Set-Top Box receivers to provide descriptions of the programs and events on the digital television networks. This data may also be provided for other transport streams.