

D-Series Automation

Interactive Event Synchronisation Protocol

Version 1.0

7 February 2003

Introduction

D-Series Automation Systems support a number of interface methods and protocols to synchronise broadcast play out events with corresponding actions in Digital Broadcast, Conditional Access, Electronic Program Guide, Interactive Television and Uplink Delivery systems. These protocols are typically the intellectual property of the third party system provider.

This document describes an Interactive Event Synchronisation Protocol (IESP) that has been defined by Encoda Systems as a generic interface for use with EPG, Interactive TV, Browse Acquisition and similar third party applications. This interface exchanges XML formatted messages over Ethernet LANs using a simple TCP/IP sockets based protocol. The messages contain a configurable subset of data fields from events in the automation playout schedule.

Interface Platform

D-Series Automation Systems can use either a Conditional Access Interface (CAIF) platform or an Interactive Event Interface (IEIF) platform as a bridge between the internal automation system architecture and external systems. The IESP protocol described in this document is supported by both platforms. Use of an IEIF platform will be assumed for the purposes of this document.

An IEIF platform can act as an interactive event synchronisation client for up to two external systems. The number of active external systems is defined by an IEIF configuration parameter.

An IEIF platform can support a main and a backup LAN connection to each of the external systems. Backup LAN capabilities are enabled by IEIF configuration parameters.

An IEIF platform can support TCP socket connections to up to two port numbers on each of the external systems. The first port is used to send event synchronisation messages to the external system. The second port is optionally used to send look-ahead schedule information to the external system and to query is for item availability or event validity.

The IP addresses, subnet masks and private port numbers used to communicate with external systems are defined by IEIF configuration parameters and may be changed to suit the network architecture at the installed site.

Dual Redundant Platforms

Automation systems may be configured with either a single IEIF platform or with an optional dual redundant IEIF configuration consisting of two IEIF platforms.

In a dual redundant configuration one IEIF platform operates as the main interface to the external systems and the other IEIF is in a standby mode. The main interface actively

passes messages between automation systems and external systems. The standby IEIF only communicates with external systems to confirm that the LAN connections are working and the external systems are operational.

The main and standby IEIF platforms communicate with each other to confirm that they are operational. If the main platform fails the standby platform will change state and become the main. In addition, IEIF platforms support user interfaces that can force them to change state.

Common Functions

At system start up the IEIF will attempt to open a TCP/IP client connection to each external system over both the main and the optional backup LAN path. If a connection cannot be established the IEIF will attempt to open the connection at regular intervals. If the external system unexpectedly closes an open connection during normal operation the IEIF will attempt to reopen the connection at regular intervals. The connection retry interval for both conditions is defined by an IEIF configuration setting that has a default value of 30 seconds.

Once a connection has been established the IEIF will send a **Heartbeat** message to the first port number on each of the external systems in order to confirm that the external system application is operational. **Heartbeat** messages will be sent on both the main and optional backup LAN connection.

External systems should respond to **Heartbeat** messages by returning an **Acknowledge** message via the received LAN path within a time out period. The time out period is defined by an IEIF configuration setting with a default value of 10 seconds. If the external system detects an error in the received message it should return a **Not Acknowledged** message within the same time out period.

The IEIF will repeatedly send **Heartbeat** messages to each of the externals at regular intervals in order to confirm that it remains operational. The repeat interval for **Heartbeat** messages is defined by an IEIF configuration setting with a default value of 30 seconds. Note that the repeat interval may be temporarily extended if an event synchronisation message is pending.

Heartbeat Message.

The format of the **Heartbeat** message is:

```
<iesp>  
  <MesgNum>12345</MesgNum>  
  <Heartbeat>  
    <Status>Main</Status>  
  </Heartbeat>  
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Status tag contains an ASCII string indicating the main or standby state of the IEIF platform. Valid status string values are:

Status String	Meaning
Main	This IEIF platform is currently acting as the Main interface.
Standby	This IEIF platform is currently acting the Standby interface.
ToMain	This IEIF platform is transitioning from Standby to Main.
ToStandby	This IEIF platform is transitioning from Main to Standby.
?????	This IEIF platform is in an erroneous or indeterminate state.

If the automation system is only configured with a single IEIF platform the status string will always contain “Main” unless an error has occurred.

Acknowledge Message.

The format of the **Acknowledge** message is:

```
<iesp>  
  <MesgNum>12345</MesgNum>  
  <Ack>  
    <Status>Main</Status>  
  </Ack>  
</iesp>
```

where;

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Status tag contains an ASCII string indicating the main or standby state of the IEIF platform. Valid status string values are:

Status String	Meaning
Main	This is currently the Main external system platform.
Standby	This is currently the Standby external system platform.
ToMain	This external system platform is transitioning from Standby to Main.
ToStandby	This external system platform is transitioning from Main to Standby.
????	This external system platform is in an erroneous or indeterminate state.

If the external system does not support a main and standby system configuration it should always return “Main”.

Not Acknowledged Message.

The format of the **Not Acknowledged** message is:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <Nak>
    <Status>Main</Status>
    <Error>ERROR: nnnn aaaaaaaaaaaaaaaaaa</Error>
  </Nak>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Status tag contains an ASCII string indicating the main or standby state of the IEIF platform. Valid status string values are:

Status String	Meaning
Main	This is currently the Main external system platform.
Standby	This is currently the Standby external system platform.
ToMain	This external system platform is transitioning from Standby to Main.
ToStandby	This external system platform is transitioning from Main to Standby.
?????	This external system platform is in an erroneous or indeterminate state.

If the external system does not support a main and standby system configuration it should always return “Main”.

The Error tag contains an optional external system defined error message string of up to 128 printable ASCII characters, excluding XML tag delimiters. This error message string is used for diagnostic logging purposes only. The tag may be blank if the external system does not support error messages.

Event Trigger Synchronisation Mode

If an automation system is configured to synchronise external system in Event Trigger mode it will use the IEIF to send messages to the external system either for every broadcast event or only for broadcast events that containing a non-blank value in a specific event field.

An automation system configuration parameter determines if messages are sent for every broadcast event or only for events containing a non-blank value. A further automation system configuration parameter is used to define which event field the automation system should check for non-blank values.

Event Trigger mode uses a configurable sequence of Cue, Start and Stop or Pause control messages to synchronise the external system. An automation system configuration parameter can be used to defines which control sequence to use in order to match the control capabilities or requirements of the external systems. For example:

- a. Only send Cue messages.
- b. Only send Start messages.
- c. Send Start and Stop messages.
- d. Send Cue and Start messages.
- e. Send Cue, Start and Stop messages.

Stop messages may optionally be replaced by Pause messages on an event by event basis.

Event Trigger mode can also use Clear messages to notify the external system that a cued or started event has been deleted before it went on-air. Edit messages may also be sent to notify the external system that data about the current broadcast event has changed.

External systems should use the type and content of each Event Trigger synchronisation message to determine if they need to react to the message by performing some internal function.

The purpose, format and specification of each Event Trigger message is described below.

Event Trigger Cue Message.

Event Trigger **Cue** messages may be sent to the external system to prepare it to react to the start of a new broadcast event

When these messages are enabled the IEIF will send an Event Trigger **Cue** to the external system M:SS minutes and seconds before the approximate start time of the new broadcast event. This offset time is defined by an automation system configuration parameter with a maximum value of 2:00 minutes and seconds.

Note that if a series of short broadcast events are scheduled to play to air each one will be individually cued M:SS minutes and seconds before its approximate start time, i.e. more than one event may be in the **Cued** state at any one time.

External systems should respond to Event Trigger **Cue** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected.

The format of the Event Trigger **Cue** message is:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <EvCue>
    <Channel>

    </Channel>
    <Event>

    </Event>
  </EvCue>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag element contains a user configurable structure identifying the automation bus that is about to broadcast the event.

The Event tag contains a user configurable structure that is populated with the content of specific data fields from the broadcast event.

Event Trigger Start Message.

Event Trigger **Start** messages may be sent to the external system to indicate that a new broadcast event has started or is about to start.

When these messages are enabled the IEIF will send an Event Trigger **Start** message to the external system SS:FF seconds and frames before the start of a new broadcast event. This offset time is defined by an automation system configuration parameter with a maximum value 10:00 seconds. An offset time of zero (00:00) is valid.

Note that Event Trigger **Start** messages will be sent immediately if a manual or external action with no pre-roll period is used to start the new broadcast event.

External systems should respond to Event Trigger **Start** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Event Trigger **Start** messages.

The format of an Event Trigger **Start** message is:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <EvStart>
    <Channel>

    </Channel>
    <Event>

    </Event>
  </EvStart>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag contains a user configurable structure identifying the automation bus that is about to broadcast the event.

The Event tag contains a user configurable structure that is populated with the content of specific data fields from the broadcast event.

Event Trigger Stop Message.

Event Trigger **Stop** messages may be sent to the external system to indicate that the current broadcast event has ended or is about to end.

When these messages are enabled the IEIF will send an Event Trigger **Stop** message to the external system SS:FF seconds and frames before the end of the current broadcast event. This offset time is defined by an automation system configuration parameter with a maximum value 10:00 seconds. An offset time of zero (00:00) is valid.

Note that an Event Trigger **Stop** message will be sent immediately if a manual or external action with no pre-roll period is used to start the next broadcast event.

External systems should respond to Event Trigger **Stop** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Event Trigger **Stop** messages.

The format of an Event Trigger **Stop** message is:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <EvStop>
    <Channel>

    </Channel>
    <Event>

    </Event>
  </EvStop>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag contains a user configurable structure identifying the automation bus that is about to broadcast the event.

The Event tag contains a user configurable structure that is populated with the content of specific data fields from the broadcast event.

Event Trigger Pause Message.

Event Trigger **P**ause messages may be sent to the external system to indicate that the current broadcast event has stopped but may resume again at a later time, e.g. after a commercial break.

These messages may be sent in place of an Event Trigger **S**top message when enabled by the contents of a control field in the current broadcast event.

When these messages are enabled the IEIF will send an Event Trigger **P**ause message to the external system SS:FF seconds and frames before the end of the current broadcast event. This offset time is the same as that defined for Event Trigger Stop messages.

Note that an Event Trigger **P**ause message will be sent immediately if a manual or external action with no pre-roll period is used to start the next broadcast event.

External systems should respond to Event Trigger **P**ause messages by returning an **A**cknowledge message, or a **N**ot **A**cknowledged message if an error is detected or the external system does not support Event Trigger **P**ause messages.

The format of an Event Trigger **P**ause message is:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <EvPause>
    <Channel>

    </Channel>
    <Event>

    </Event>
  </EvPause>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag contains a user configurable structure identifying the automation bus that is about to broadcast the event.

The Event tag contains a user configurable structure that is populated with the content of specific data fields from the broadcast event.

Paused events may be resumed at some time in the future by sending Event Trigger **Cue** and/or **Start** messages for an event with the same primary synchronisation attributes. Note that if the future event is deleted from the broadcast schedule the automation system will not take deliberate action to notify the external system via the IEIF. External systems should remove events from the paused state if they have not been restarted after a significant time out period (e.g. 10 to 20 minutes) or should use IEIF item availability requests or look ahead schedule download information to determine if the event is still present in the broadcast schedule.

Event Trigger Clear Message.

Event Trigger **Clear** messages may be sent by the IEIF to notify the external system that an event identified in a **Cue** or **Start** message has been deleted from the automation payout schedule before the broadcast event started.

These messages are enabled if the **Cue** or **Start** messages are enabled.

External systems should respond to Event Trigger **Clear** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Event Trigger **Clear** messages.

The format of an Event Trigger **Clear** message is:

```
<iesp>  
  <MesgNum>12345</MesgNum>  
  <EvClear>  
    <Channel>  
  
    </Channel>  
    <Event>  
  
    </Event>  
  </EvClear>  
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag contains a user configurable structure identifying the automation bus that is about to broadcast the event.

The Event tag contains a user configurable structure that is populated with the content of specific data fields from the broadcast event.

Event Trigger Edit Message.

Event Trigger **Edit** messages may be sent to notify the external system that data about the current broadcast event has changed.

External systems should respond to Event Trigger **Edit** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Event Trigger **Edit** messages.

The format of an Event Trigger **Edit** message is:

```
<iesp>
  <MsgNum>12345</MsgNum>
  <EvEdit>
    <Channel>

    </Channel>
    <Event>

    </Event>
  </EvEdit>
</iesp>
```

where:

The **MsgNum** tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The **Channel** tag contains a user configurable structure identifying the automation bus that is broadcasting the event.

The **Event** tag contains a user configurable structure that is populated with the content of specific data fields from the current broadcast event.

Event Trigger Look Ahead Schedule.

The automation system can optionally use the IEIF to transfer an Event Trigger look ahead schedule to the external system and to as it to validate the contents of each event.

When configured to support Event Trigger look ahead schedule transfers the automation system will asynchronously cycle through each IEIF enabled channel and identify up to 20 future Event Trigger synchronisation enabled broadcast events that fall within an on-air look ahead window.

As each broadcast event is identified the automation system will use the IEIF to send an Event Trigger **List** message to a second port on the external system. This message will contain the same information about the future broadcast event as other Event Trigger messages. The external system may use a series of these message to build its own an internal broadcast event schedule.

Note that in practise the on-air look ahead window may limit the actual number of events transferred to significantly less then 20 depending upon the style and contents the broadcast schedule on each channel.

The automation system can optionally use the look ahead schedule transfer process to ask the external system to confirm if it contains or can validate key elements of each future broadcast event. For example, if an EPG system's database contains a matching event record or if an Interactive TV data carousel can access the specified interactive application control file. When this option is enabled an **AvailReq** flag will be set in each Event Trigger **List** message.

External systems should respond to Event Trigger **List** message by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Event Trigger **List** messages.

If the an **AvailReq** flag is set in an Event Trigger **List** message the external system should examine the contents of the message and asynchronously reply by returning an **EvAvailReply** message indicating if the contents of the broadcast event is valid. External systems should use their own internal selection criteria to determine if an event is valid.

The IEIF will respond to an **EvAvailReply** message by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected.

The format of an Event Trigger **List** message from the IEIF is:

```
<iesp>
  <MsgNum>nnnnn</MsgNum>
  <EvList>
    <Channel>

    </Channel>
    <Event>

    </Event>
    <Handle>hhhhhhh</Handle>
    <EntryNum>01</EntryNum>
    <Last>N</Last>
    <AvailReq>Y</AvailReq>
  </EvList>
</iesp>
```

where:

The **MsgNum** tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The **Channel** tag contains a user configurable structure identifying the automation bus that is schedule to broadcast the event.

The **Event** tags contain a user configurable structure that is populated with the content of specific data fields from the associated broadcast event.

The **Handle** tag contains a unique four byte automation event handle encoded as eight ASCII hexadecimal digits.

The **EntryNum** tag contains a two numeric ASCII character look ahead schedule entry number in the range “00” to “30”, where “00” is the on-air event, “01 is the first future event and so on.

The **Last** tag contains the single character “Y” if this is the last entry in the look ahead schedule, or “N” if it is not the last entry.

The **AvailReq** tag contains the single character “Y” if the external system should confirm the availability of this event.

The format of an **EvAvailReply** message from the external system is:

```
<iesp>
  <MesgNum>56789</MesgNum>
  <EvAvailRep>
    <Channel>

    </Channel>
    <Handle>hhhhhhhh</Handle>
    <EntryNum>01</EntryNum>
    <Confirm>Y</Confirm>
  </EvAvailRep>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”. This message number is generated by the external system.

The Channel tag contains a user configurable structure identifying the automation bus that is schedule to broadcast the event.

The Handle tag contains the four byte automation event handle from the original EvList message.

The EntryNum tag contains the entry number from the original EvList message.

The Confirm tag contains the single character “Y” if the EvList event entry is valid and “N” if it is invalid.

Now/Next Synchronisation Mode

If an automation system is configured to synchronise external systems in Now/Next mode it will use the IEIF to send messages to the external system at the start of each new broadcast event on an IEIF enabled channel that contains a non-blank value in a specific event field. These broadcast events are usually the first event in a published program. An automation system configuration parameter is used to define which event field the automation system should check for non-blank values.

If an automation system is configured to synchronise external systems in Now/Next mode it will cycle through the schedule on each IEIF enabled channel and attempt to locate each new Now/Next enabled broadcast event, the first future Now/Next enabled broadcast event and an optional second future event that falls within an on-air look ahead window.

In practice the on-air look ahead window may limit detection of a first future Now/Next enabled broadcast event or optional second future depending upon the style and contents the broadcast schedule on each channel.

Now/Next mode uses Start messages to synchronise the external system. These messages contain information about the new broadcast event, the first future Now/Next enabled event and an optional second future event.

Now/Next mode can also use Edit messages to notify the external system that information about the first future Now/Next event or optional second future event has changed.

Note that on-air broadcast events cannot be stopped, paused or edited. Now/Next mode assumes that the external system's on-air broadcast events will always end or be edited or replaced by starting a new event.

The purpose, format and specification of each Now/Next message is described below.

Now/Next Start Message.

When Now/Next mode is enabled Now/Next **Start** messages will be sent to the external system to indicate that a new broadcast event with a non-blank event field has started.

External systems should respond to Now/Next **Start** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Now/Next **Start** messages.

The format of a Now/Next **Start** message is:

```
<iesp>
  <MsgNum>123435</MsgNum>
  <NnStart>
    <Channel>

    </Channel>
    <Now>
      <Event>

      </Event>
      <Handle>hhhhhhhh</Handle>
    </Now>
    <Next>
      <Event>

      </Event>
      <Handle>hhhhhhhh</Handle>
    </Next>
    <Next+1>
      <Event>

      </Event>
      <Handle>hhhhhhhh</Handle>
    </Next+1>
  </NnStart>
</iesp>
```

where:

The **MsgNum** tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The **Channel** tag contains a user configurable structure identifying the automation bus that is broadcasting the event.

The **Event** tags contain a user configurable structure that is populated with the content of specific data fields from the associated broadcast event.

The **Handle** tag contains a unique four byte automation event handle encoded as eight ASCII hexadecimal digits.

The Now tag contains an Event tag populated with data from the new broadcast event .

The Next tag contains an Event tag populated with data from the first future broadcast event that contains a non-blank event field.

The Next+1 tag contains an Event tag populated with data from the second future broadcast event that contains a non-blank event field.

Note that event data in the Next or Next+1 tag will be blank if the automation system cannot find a future broadcast event with a non-blank event field within a look-ahead window defined by an automation system configuration parameter.

Now/Next Edit Message.

Now/Next **Edit** messages may be sent to the external system to indicate that the “Next” or “Next+1” broadcast event has changed. These messages are typically sent when an automation system operator adds, deletes or replaces a future broadcast event. These messages may also be sent if an operator edits a future broadcast event, e.g. by changing its published start time.

External systems should respond to Now/Next **Edit** messages by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Now/Next **Edit** messages.

The format of a Now/Next **Edit** message is:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <NnEdit>
    <Channel>

    </Channel>
    <Next>
      <Event>

      </Event>
      <Handle>hhhhhhhh</Handle>
    </Next>
    <Next+1>
      <Event>

      </Event>
      <Handle>hhhhhhhh</Handle>
    </Next+1>
  </NnEdit>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag contains a user configurable structure identifying the automation bus that is about to broadcast the event.

The Event tags contain a user configurable structure that is populated with the content of specific data fields from the associated broadcast event.

The Handle tag contains a unique four byte automation event handle encoded as eight ASCII hexadecimal digits.

The Next tag contains an Event tag populated with data from the first future broadcast event that contains a non-blank event field.

The Next+1 tag contains an Event tag populated with data from the second future broadcast event that contains a non-blank event field.

Note that event data in the Next or Next+1 tag will be blank if the automation system cannot find a future broadcast event with a non-blank event field within a look-ahead window defined by an automation system configuration parameter.

Now/Next Look Ahead Schedule.

The automation system can optionally use the IEIF to transfer a Now/Next look ahead schedule to the external system.

When configured to support Now/Next look ahead schedule transfers the automation system will asynchronously cycle slowly through each IEIF enabled channel and identify up to 50 future Now/Next synchronisation enabled broadcast event.

As each broadcast event is identified the automation system will use the IEIF to send a Now/Next **List** message to a second port on the external system. This message will contain the same information about the future broadcast event as other Now/Next messages. The external system may use a series of these message to build its own an internal broadcast event schedule.

External systems should respond to Now/Next **List** message by returning an **Acknowledge** message, or a **Not Acknowledged** message if an error is detected or the external system does not support Now/Next **List** messages.

The format of a Now/Next **List** message from the IEIF is:

```
<iesp>
  <MesgNum>nnnnn</MesgNum>
  <NnList>
    <Channel>

    </Channel>
    <Event>

    </Event>
    <Handle>hhhhhhh</Handle>
    <EntryNum>01</EntryNum>
    <Last>N</Last>
  </NnList>
</iesp>
```

where:

The MesgNum tag contains a five numeric ASCII character message number in the range “00000” to “65535”.

The Channel tag contains a user configurable structure identifying the automation bus that is schedule to broadcast the event.

The Event tags contain a user configurable structure that is populated with the content of specific data fields from the associated broadcast event.

The Handle tag contains a unique four byte automation event handle encoded as eight ASCII hexadecimal digits.

The EntryNum tag contains a two numeric ASCII character look ahead schedule entry number in the range “01” to “50”, where “01” is the first future event and so on.

The Last tag contains the single character “Y” if this is the last entry in the look ahead schedule, or “N” if it is not the last entry.

Channel structure

An automation system configuration table is used to define the selection of automation bus configuration fields to include in a Channel structure. This table also allows the XML tag name to be defined in order to match a field name or schema supported by the external system.

The following table lists the valid automation system fields, their default XML tag name and a description of their format, size and content.

Automation field	XML Tag	Description
bus (name)	Name	An up to 8 ASCII character field containing an automation channel identification mnemonic.
bus description	Desc	An up to 20 ASCII character field containing the displayed automation channel description.
Timezone	Timezone	An up to 4 ASCII character field containing the channel's timezone mnemonic.
Network Name	Network	An up to 32 ASCII character field, typically containing the channel's network name.
Station ID	StationID	An up to 6 ASCII character field, typically containing the channel's station call sign or ID.
Transmission Signal ID	TSID	A 4 ASCII character field, typically containing the channel's transmission signal ID in hexadecimal digits.

An example of a populated Channel structure containing all five fields for the fictional station KXYZ-TV would be as follows:

```
<iesp>
  <MesgNum>12345</MesgNum>
  <
    >
    <Channel>
      <Name>KXYZ-TV</Name>
      <Desc>KXYZ-TV CH29 Houston</Desc>
      <Timezone>EST</Timezone>
      <Network>DuMont Television Network</Network>
      <StationID>KXYZ</StationID>
      <TSID>FFFF</TSID>
    </Channel>
  </
  >
```

In practise a smaller subset of fields is likely to be used as a single channel identifier, such as "Name", may be all that is required by the external system.

Event Structure

An automation system configuration table is used to define a sequence of up to 16 event fields to include in the Event structure. This configuration table also allows the XML tag name of each field to be defined.

The following tables list the subset of automation system event fields available for use in an Event structure. The tables include each field's default XML tag name together with a description of its format, size and content. The Event structure can include any of the automation event fields listed in the tables.

Typical Event Trigger mode fields:

Automation event field	XML Tag	Format	Description
ev_num	EventNum		An up to 20 ASCII character code identifying the broadcast play out event.
utc_date	Date	dd/mm/yy	The scheduled start date of the broadcast event in UTC.
time	Time	hh:mm:ss:ff	The scheduled start time of the broadcast event in UTC.
duration	Duration	h:mm:ss:ff	The scheduled duration of the broadcast event.
start_type	StartType	“- “ or “TT”, “MAN”, “EXT”, “NET” or blank	A 3 ASCII character mnemonic indicating if the event is scheduled to start at the scheduled time, by a manual cue, by a cue from an external source or a channel associated cue. Blank if it starts when the previous event ends.
end_type	EndType	“MAN”, “EXT”, “NET” or blank	A 3 ASCII character mnemonic indicating if the event is scheduled to end by a manual operator cue, a cue from an external source or a channel associated cue. Blank if it ends when the duration expires.
video_item	VidID		Up to 32 ASCII characters containing the Material ID of the video item that is scheduled to play to air
type_material	Seg	“Mnn,” “S ”, “C ”, “T ”, “I ”, “K ”, “X ” or blank	A 3 ASCII character type of material mnemonic. “Mnn” indicates a multi segment program, where “nn” is a two digit segment number in the range 01 to 99.
bus_item or: bus_item2 bus_item3 bus_item4	ID		Up to 32 ASCII characters containing the ID or file name of the item to be triggered. The automation system may be configured to automatically populate this field with the Material ID of the video item that is scheduled to play to air.
comment1	Title		Up to 66 characters containing a multilingual event title or description, encoded in UTF-8.
agency_code	AgencyCode		Up to 32 ASCII characters containing the ISCI, V-ISAN, Ad-ID or similar unique material identifier code of the item to be triggered.
ca_code	Code		An up to 20 ASCII character keyword or event identification code to be passed to the external system.
ca_iactive	Interactive	“I” or blank.	A single ASCII character “I”, optionally used to indicate that the broadcast event has an associated Interactive TV event.
ca1_data	Data		Up to 256 bytes of binary data encoded as up to 512 ASCII hexadecimal digits.

Typical Now/Next mode fields:

Automation event field	XML Tag	Format	Description
si_event_id	PgmID		An up to 16 ASCII character ID used to identify events in the same published program.
text_out1	PgmTitle		Up to 40 characters containing multilingual text encoded in UTF-8.
text_out2	PgmDesc		Up to 40 characters containing multilingual text encoded in UTF-8.
vps_date	PgmDate	dd.mm	The program's published start date.
vps_time	PgmTime	hh:mm	The program's published start time.
program_rating	PgmRating		An up to 9 ASCII character Program Rating mnemonic.
rating_fv	PRfantasy	"F" or blank.	A single ASCII character Program Rating flag – Fantasy Violence
rating_v	PRviolence	"V" or blank.	A single ASCII character Program Rating flag – Violence
rating_s	PRsexual	"S" or blank.	A single ASCII character Program Rating flag – Sexual situation
rating_l	PRlang	"L" or blank.	A single ASCII character Program Rating flag – Coarse language
rating_d	PRdialogue	"D" or blank.	A single ASCII character Program Rating flag – Suggestive dialogue

Revision History.**Revision 1.0 (7 February 2003)**

- First Draft. Replaces draft XML event synchronisation document.