



# Developers' Guidelines Enhanced Messaging Service

Ericsson is the leading provider in the new telecoms world, with communications solutions that combine telecom and datacom technologies with freedom of mobility for the user. With more than 100,000 employees in 140 countries, Ericsson simplifies communications for its customers - network operators, service providers, enterprises and consumers - the world over.

First edition (May 2001)

Publication number: EN/LZT 108 4946 R1A

This document is published by **Ericsson Mobile Communications AB**, without any warranty.

Improvements and changes to this text necessitated by typographical errors, inaccuracies of current information or improvements to programs and/or equipment, may be made by **Ericsson Mobile Communications AB** at any time and without notice. Such changes will, however, be incorporated into new editions of this document. Any hard copies of this document or any locally stored copies are to be regarded as temporary reference copies only.

All rights reserved.

© Ericsson Mobile Communications AB, 2001

# Contents

<b>PREFACE</b> .....	<b>5</b>
Purpose Of This Document .....	5
<b>EMS OVERVIEW AND COMPONENTS</b> .....	<b>6</b>
<b>PHONE OVERVIEWS</b> .....	<b>8</b>
<b>SOUNDS</b> .....	<b>12</b>
Receive Sound to Phone (Mobile Terminated) .....	12
Send Sound From Phone (Mobile Originated) .....	12
Encoding Sound For Sending Over The Air .....	13
<b>MELODIES</b> .....	<b>14</b>
Receive Melody To Phone (Mobile Terminated) .....	14
Send Melody From Phone (Mobile Originated) .....	15
Encoding Melody For Sending Over The Air .....	16
<b>PICTURES</b> .....	<b>20</b>
Receive Picture to Phone (Mobile Terminated) .....	20
Send Picture From Phone (Mobile Originated) .....	21
Encoding Picture For Sending Over The Air .....	24
<b>ANIMATIONS</b> .....	<b>26</b>
Receive Animation to Phone (Mobile Terminated) .....	26
Send Animation From Phone (Mobile Originated) .....	27
Encoding Animation For Sending Over The Air .....	27
<b>FORMATTED TEXT</b> .....	<b>29</b>
Receive Formatted Text to Phone (Mobile Terminated) .....	29
Send Formatted Text From Phone (Mobile Originated) .....	29
Encoding Formatted Text For Sending Over The Air .....	29
<b>CONCATENATED MESSAGES</b> .....	<b>31</b>
Concatenation Format .....	31
<b>TERMINOLOGY AND ABBREVIATIONS</b> .....	<b>33</b>
<b>RELATED INFORMATION</b> .....	<b>34</b>



# PREFACE

## Purpose Of This Document

The Developers' Guidelines for EMS is designed to give the reader a deeper insight of how to design applications for Enhanced Messaging Service (EMS). Initiated by Ericsson, EMS is now incorporated by the 3rd Generation Partnership Project (3GPP) into the Short Messaging Service (SMS) standard. The SMS standard is supported by the major network operators and mobile phone manufacturers.

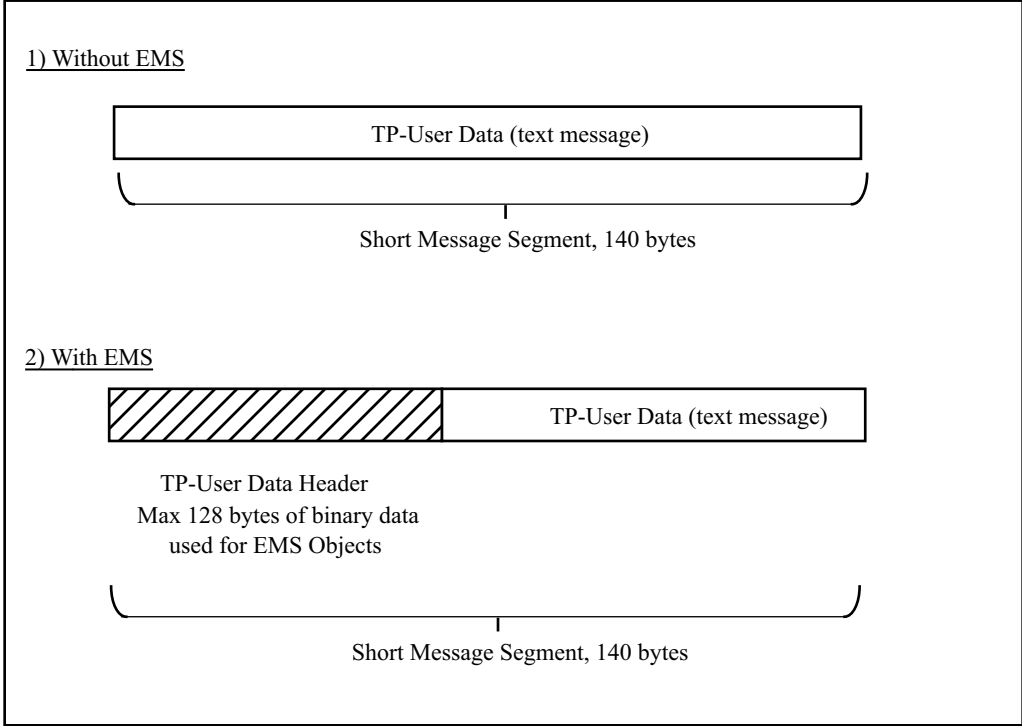
People who can benefit from this document include:

- Application providers
- Content providers
- Content aggregators
- Operators and service providers
- Software developers
- Business decision-makers

It is assumed that the reader has a basic understanding of SMS and EMS. An overview of the EMS standard and its possibilities can be found in the EMS White Paper, available on the Ericsson Mobility World. The site at <http://www.ericsson.com/mobilityworld> contains up-to-date information about technologies, products and tools.

# EMS OVERVIEW AND COMPONENTS

Enhanced Messaging Service is based on standard mechanisms in GSM SMS messaging. The first mechanism is called Transfer Protocol User Data Header (TP-UDH), which makes it possible to include binary data in a normal Short Message (SM) prior to the text message itself. The binary data is in the TP-UD field (message), which means that it consumes a part of the 140 bytes.



The diagram shows how the EMS objects consumes space in a Short Message. Note that in both standard SMS and EMS messages the text in the UD field may be represented in 7 bits (for normal European text) and hence requires bit filling or 16 bits (in the case of Chinese, Arabic etc.)

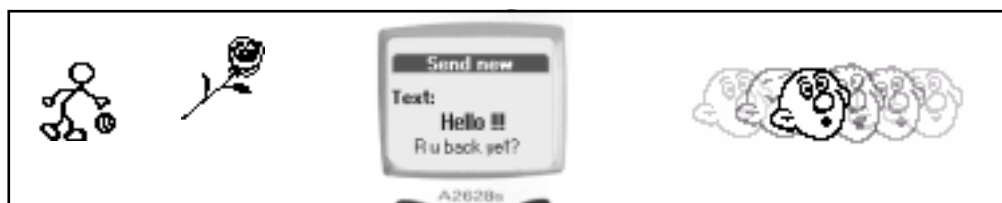
The components, or objects, that can be used in EMS are briefly described below. Each object is described in detail later in this document. Several objects (e.g. sounds, pictures, animations, melodies, formatting) can be inserted in one message.

## Sounds And Melodies



EMS gives the user the ability to send and receive sounds. These can be pre-defined sounds, such as “Chime high” and “Notify”, or melodies (ring signals in the phone), downloaded from the Internet, received in SMS messages or composed by the user on the phone keypad or a PC.

## Pictures, Animations And Formatted Text



Phones supporting EMS include a set of pre-defined pictures for inserting in SMS messages. New pictures and animations are downloaded from the Internet or received in SMS messages. Pictures can be created and edited in the phone using a built-in Picture Editor. The users can format text in messages with different styles, sizes and alignment of text.

## Concatenated Messages

A part of the EMS standard is the support for concatenated messages, which means that the phone is able to automatically combine several messages both when creating and receiving EMS. This is useful to be able to build, and display, messages with rich content, since the amount of information in each SMS is limited by the SMS standards.

This mechanism permits longer messages than 140 bytes. In fact, the SMS standard allows up to 255 messages at 140 bytes each to be concatenated to one message up to about 38k bytes. However, current phones support concatenation of between 3 and 6 messages (product dependent).

## Identification Mechanism

Each object is identified by an Information Element (IE) in the TP-UDH field, which in turn is part of the TP-UD field. The IE contains a number of octets that identify the position and data for the EMS object(s) in the SM. The first octet identifies the absolute position of the object within and from the beginning of the SM data. In the case of pictures, animations, sounds and melodies, the second and following octets (up to n) are used for data or reference. In the case of text formatting, the second octet is used to indicate length of formatting and the third octet for the format value. (All these IEs are described in detail later in this document.)

EMS IEs of the same type may occur more than once in a single message or one segment of a concatenated SM.

# PHONE OVERVIEWS

These are the Ericsson phones that support Enhanced Messaging Service

---

Ericsson R520 (R2)	Characteristics
	<b>EMS related features:</b>
	<p>Sounds in messages: Yes Melodies in messages: Yes Melody composer: Yes <i>My Melodies</i>: 8 Pictures in messages: Yes Picture editor: Yes <i>My Pictures</i>: 30 Pre-defined pictures: 76 Pre-defined animations: Yes, 6 types User defined animations, receiving: Yes Formatted text in messages: No Concatenation: Yes, 6 messages</p>
	<b>Display size (W x H):</b> 101 x 67 pixels
	<b>Networks:</b> GSM 900, GSM 1800, GSM 1900
	<b>Features (general), examples:</b> Phone book in phone and on SIM (511 entries total), Bluetooth™ wireless technology support, Voice control, Speaker phone, Calendar, WAP Browser, Built-in E-mail client (POP/SMTP), Menu short-cuts, Sleeping display, Synchronization with PC, Infrared port, Built-in modem, Display light, Connected Line Identity Presentation (COLP), Games, Ring signal composition, SMS counter, SMS templates, and more
	<b>Features (operator-specific), examples:</b> GPRS, High Speed Data (HSCSD), WAP 1.2.1, WTLS class 3 (Encoding + Server Authentication + Client Certification), Circuit Switched Data (a.k.a GSM Data), SMS Cell Broadcast, SIM Application Toolkit, and more
	<b>Performance</b>
	Size: 130 x 50 x 16 mm
	Weight: 82 g without battery, 136 g with High Capacity Batt.
	Talk time: up to 25 hrs with High Capacity Battery
	Standby time: up to 735 hrs with High Capacity Battery

---



**EMS related features:**

Sounds in messages: No  
Melodies in messages: Yes  
Melody composer: Yes  
*My Melodies*: 8  
Pictures in messages: Yes  
Picture editor: Yes  
*My Pictures*: 30  
Pre-defined pictures: 69  
Pre-defined animations: No  
User defined animations, receiving: No  
Formatted text in messages: No  
Concatenation: Yes, 3 messages

**Display size (W x H):** 101 x 33 pixels

**Networks:** GSM 900, GSM 1800

**Features (general), examples:** Voice Answering, Voice Dialling, Mobile chat, Swatch® Internet time, SMS templates, Active flip, Games, Start-up/Shut-down shows, Phone book, Ring signal composition, SMS templates, and more

**Features (operator-specific), examples:** WAP 1.1, WTLS class 2 (Encoding + Server Authentication, Root Certificates pre-defined in phone), Circuit Switched Data (a.k.a GSM Data), SIM Application Toolkit, and more

**Performance**

Size: 101 x 54 x 28 mm

Weight: 128 g with kit battery

Talk time: up to 10 hrs with kit battery

Standby time: up to 200 hrs with kit battery

---

---

**Ericsson T29 (R2)****Characteristics**

---

**EMS related features:**

Sounds in messages: No  
Melodies in messages: Yes  
Melody composer: Yes  
*My Melodies*: 8  
Pictures in messages: Yes  
Picture editor: Yes  
*My Pictures*: 30  
Pre-defined pictures: 69  
Pre-defined animations: No  
User defined animations, receiving: No  
Formatted text in messages: No  
Concatenation: Yes, 3 messages

**Display size (W x H):** 101 x 33 pixels

**Networks:** GSM 900, GSM 1800

**Features (general), examples:** Voice Answering, Voice Dialling, Mobile chat, Swatch® Internet time, SMS templates, Active flip, Games, Start-up/Shut-down shows, Phone book, Ring signal composition, SMS templates, and more

**Features (operator-specific), examples:** WAP 1.1, WTLS class 2 (Encoding + Server Authentication, Root Certificates pre-defined in phone), Circuit Switched Data (a.k.a GSM Data), SIM Application Toolkit, and more

**Performance**

Size: 95 x 51 x 22 mm

Weight: 105 g with kit battery

Talk time: up to 12 hrs with High Capacity Battery

Standby time: up to 240 hrs with High Capacity Battery

---

**EMS related features:**

Sounds in messages: Yes  
Melodies in messages: Yes  
Melody composer: Yes  
*My Melodies*: 8  
Pictures in messages: Yes  
Picture editor: Yes  
*My Pictures*: 30  
Pre-defined pictures: 76  
Pre-defined animations: Yes, 6 types  
User defined animations, receiving: Yes  
Formatted text in messages: No  
Concatenation: Yes, 6 messages

**Display size (W x H):** 101 x 54 pixels

**Networks:** GSM 900, GSM 1800, GSM 1900

**Features (general), examples:** Phone book in phone and on SIM, Bluetooth™ wireless technology support, Voice control, Calendar, WAP Browser, Built-in E-mail client (POP/SMTP), Menu shortcuts, Synchronization with PC, Infrared port, Built-in modem, Display light, Connected Line Identity Presentation (COLP), Games, Ring signal composition, SMS counter, SMS templates, and more

**Features (operator-specific), examples:** GPRS, High Speed Data (HSCSD), WAP 1.2.1, WTLS class 3 (Encoding + Server Authentication + Client Certification), Circuit Switched Data (a.k.a GSM Data), SMS Cell Broadcast, SIM Application Toolkit, and more

**Performance**

Size: 96 x 50 x 18 mm

Weight: 86 g with kit battery (Ultra Slim Battery)

Talk time: up to 25 hrs with High Capacity Battery

Standby time: up to 735 hrs with High Capacity Battery

---

# SOUNDS

Support: R520 (R2), T39  
No support: T20e, T29 (R2)

The EMS message contains a reference to a pre-defined sound in the receiving phone. There are today 10 different sounds supported by the EMS standard, pre-defined in the phone. Since the sounds are specified in the EMS standard the actual message sent will only include a reference to the sound and not the entire sound itself. This makes the transmission very efficient.

## Receive Sound to Phone (Mobile Terminated)

When a message contains a sound, it will be indicated by a speaker symbol in the text.



The sound will be played when brought into focus. This enables multiple sounds to be included in a single message, played one at a time, and it prevents conflicts with other tasks in the phone, for example an ongoing phone call.

There can be multiple sounds in one message, and also in combination with pictures, melodies etc. If a message is concatenated in several segments, the phone automatically takes care of combining these into one message.

The actual sound produced by the phone is product and manufacturer-dependent. This means that each manufacturer can implement the sounds different from others.

Most phones without support for the EMS standard will simply ignore the sound reference when the message is received, and will just display the text in the message. This prevents the phone from displaying unreadable contents to the user.

## User Interaction After Receiving Sound

The user can play the sound(s).

*To play the sound*

1. Use the (up-)left and (down-)right buttons until the speaker symbol is highlighted in focus
2. The sound is played

## Send Sound From Phone (Mobile Originated)

When the sound (reference) is inserted into an outgoing message, the phone takes care of encoding it properly into the Short Message. If the option to send long messages is switched on in the phone, the phone also concatenates several messages, if necessary, see "Concatenated Messages" on page 31.

## User Interaction To Send Sound

*To insert a sound in an EMS message*

1. Create the message and type the text
2. At the desired position, press and hold \* or # or the Option key (depending on phone type) until the Options menu is displayed
3. Select Insert Object, Sound and choose from the list of available sounds
4. To insert more than one sound, or another EMS component, in the message, repeat from step 2



## Encoding Sound For Sending Over The Air

There are a number of fixed pre-defined sounds. Each sound no corresponds to a specific sound according to the table below. The presentations of these sounds are manufacturer specific.

Sound no	Description
0	Chimes high
1	Chimes low
2	Ding
3	TaDa
4	Notify
5	Drum
6	Claps
7	FanFare
8	Chord high
9	Chord low

The Information-Element-Data octet(s) placed in the TP-UDH, is coded as follows.

- Octet 1** Position indicating in the SM data the instant after which the sound should be played. It will be set to the number of characters from the beginning of the SM data after which the sound should be played.
- This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM
- Octet 2** Sound number. should be encoded as an integer value.

# MELODIES

Support: R520 (R2), T20e, T29 (R2), T39

No support: -

The melody is part of an EMS message. The melody data is transmitted according to a subset of the iMelody version 1.0 format, as specified by the Infrared Data Association. When the melody is sent from a phone, the phone takes care of encoding it into an object of the correct format. Melodies generated using other tools, for example by a service provider or contents creator, has to comply to this format.

## Receive Melody To Phone (Mobile Terminated)

When a message contains a melody, it will be indicated by a note symbol in the text.



The melody will be played when brought into focus. This prevents conflicts with other tasks in the phone, for example an ongoing phone call.

There can be multiple melodies in one message, and also in combination with pictures, sounds etc. If a message is concatenated in several segments, the phone automatically takes care of combining these into one message.

The actual tones in the melody played by the phone depends on the phone model. It is therefore important to verify the playback quality of the melody on the actual target devices. For information on support of tones in Ericsson phones, see "Encoding Melody For Sending Over The Air" on page 16.

Most phones without support for the EMS standard will simply ignore the melody information when the message is received, and will just display the text in the message. This prevents the phone from displaying unreadable contents to the user.

## User Interaction After Receiving Melody

The user can play the melody and also save the melody to a position in *My Melodies* for future use.

### *To play the melody*

1. Use the (up-)left and (down-)right buttons until the note symbol is highlighted in focus
2. The melody will be played

### *To save the melody to My Melodies*

1. Use the (up-)left and (down-)right buttons until the note symbol is highlighted in focus
2. Press Yes until the Options menu is displayed
3. Save the melody to a position in *My Melodies*

When the melody is saved in *My Melodies*, the user can use it as a ring tone, or edit it and also insert it in outgoing messages.

## Send Melody From Phone (Mobile Originated)

Melodies saved in *My Melodies* can be edited and inserted in EMS messages. To be able to use any of the pre-installed ring tones in EMS messages, the user first has to save the ring tone to a free position in *My Melodies*.

### Melody Composer

The built-in Melody composer allows the user to compose a new melody or edit any ring tone or melody in *My Melodies*.

Feature	Support in listed Ericsson phones	
Note range	Three octaves, from c to ++b	
Durations	c	150 ms
	c.	225 ms
	C	300 ms
	C.	450 ms
Melody length	up to 80 notes	
<i>My Melodies</i>	up to 8 different melodies	

For information on how to use the Melody composer, please refer to the User's Guide for the respective phone.

### User Interaction To Send Melody

When the melody is inserted into an outgoing message, the phone takes care of encoding it properly into the Short Message in the iMelody format. If the option to send long messages is switched on in the phone, the phone also concatenates several messages, if necessary, see "Concatenated Messages" on page 31.

However, if a melody exceeds 128 bytes (including header and footer) it will be truncated.

The user creates a message just like a normal SMS message and inserts the melody at the desired position.

#### *To insert a melody in an EMS message*

1. Create the message and type the text
2. At the desired position, press and hold \* or # or the Option key (depending on phone type) until the Options menu is displayed
3. Select Insert Object, Melody and choose from the list *My Melodies*
4. To insert more than one melody in the message, or another EMS component, repeat from step 2



## Encoding Melody For Sending Over The Air

A service provider or content creator can generate a melody to be sent to mobile phones in an EMS message. This makes it possible to provide customized ring tones and melodies, for example as a download service to users.

- The melody data, including header and footer, has to be encoded into the TP-UDH of the message. The phone will not recognise the melody data if it is transmitted in plain ASCII or any other format.
- The maximum melody data size is 128 bytes.
- Concatenation has to be used in order to include more than one melody in a message (each message segment can include only one melody).

The Information-Element-Data octet(s) placed in the TP-UDH, is coded as follows.

<b>Octet 1</b>	Position indicating in the SM data the instant the after which the sound should be inserted.
<b>Octet 2-n</b>	Protocol Data Unit. This octet(s) contains a User Defined Sound according to the iMelody format.

### The iMelody Format Support

The note range supported by the listed Ericsson phones is within the octaves 4, 5 and 6. However, it is possible that melodies received in EMS messages from other sources, such as another type of phone, can range out of the supported octaves. This is handled by the listed Ericsson phones according to the following scenarios:

#### 1) Receive an iMelody with notes out of range of the octaves 4, 5 and 6

a) If the iMelody fits within a three-octave “window”, it will be modified to fit within the supported range of notes. For example, iMelodies in the ranges 3, 4, 5 or 4, 5, 6 or 5, 6, 7 will all be modified and played using the octaves 4, 5 and 6. The modification is done in steps of a full octave.

b) If the iMelody does not fit within a three-octave “window”, but instead includes notes from four or five or more octaves, it will still be played.

First, the phone will parse the received iMelody to find the “best fit window” over a three octave range. For example, an iMelody where the majority of notes fall within the octaves 3, 4 and 5, but with a few notes in the octave 7, will be adjusted upwards and played in the range 4, 5, and 6.

Secondly, the phone will adjust those remaining notes out of range to a corresponding note in the octaves 4, 5 or 6. In the example above, the few notes from octave 7 in the received iMelody, will most likely be played in octave 6.

#### 2) Storing of an EMS message with a melody

When a message containing a melody is stored, it will contain the original iMelody data received.

#### 3) Saving a melody

When a received melody is saved it will be stored in its modified format. For example, a received iMelody in the octaves 5, 6, 7 and 8 will be modified and stored in the range 4, 5 and 6 as explained in (1a) and (1b).

#### 4) Sending an iMelody

a) When a melody is created in the Melody Composer, it is translated to iMelody format in the octaves 4, 5 and 6.

b) When a melody is received, then saved and then included in an outgoing message, it will be sent in iMelody format in the octaves 4, 5 and 6. The modification rules according to (1a) and (1b) are applied.



c) When forwarding or replying to a message containing a melody, the melody will be sent in iMelody format in the octaves 4, 5 and 6. The modification rules according to (1a) and (1b) are applied.

This table shows how the iMelody v1.0 format is supported by Ericsson phones:

iMelody format	Support in listed Ericsson phones		
	iMelody notation	Melody Composer notation	Comment
Durations	3	c	
	2	c.	
	1	C	
	0	C.	
Flat note	Yes, &	(b)	
Sharp note	Yes, #	#	
Rest	Yes, r3 (r2, r1, r0)	p (p. , P , P. )	
Beat, Style	No		
Volume, Volume modifiers	No, (V), (V+, V-)		
Duration Specifier	No, (. : ;)		
Led, Vibe, Backlight, Repeat	No		

The following table details, for each of the supported octaves, how the notes are represented in the Melody Composer format, also known as the Extended EMelody format (upper rows) and iMelody format (lower rows):

**Octave 4 (+c is 1047Hz approx., +b is 1991Hz approx.)**

	+c	+#c	+(b)d	+d	+#d	+(b)e	+e	+f	+#f	+(b)g	+g	+#g	+(b)a	+a	+#a	+(b)b	+b	+b.	+B	+B.
*4	c3	#c3	&d3	d3	#d3	&e3	e3	f3	#f3	&g3	g3	#g3	&a3	a3	#a3	&b3	b3	b2	b1	b0

**Octave 5 (++c is 2072Hz approx., ++b is 3906Hz approx.)**

	++c	++#c	++(b)d	++d	++#d	++(b)e	++e	++f	++#f	++(b)g	++g	++#g	++(b)a	++a	++#a	++(b)b	++b	++b.	++B	++B.
*5	c3	#c3	&d3	d3	#d3	&e3	e3	f3	#f3	&g3	g3	#g3	&a3	a3	#a3	&b3	b3	b2	b1	b0

**Octave 6 (+++c is 4144Hz approx., +++b is 7812Hz approx.)**

	+++c	+++#c	+++(b)d	+++d	+++#d	+++(b)e	+++e	+++f	+++#f	+++(b)g	+++g	+++#g	+++(b)a	+++a	+++#a	+++(b)b	+++b	+++b.	+++B	+++B.
*6	c3	#c3	&d3	d3	#d3	&e3	e3	f3	#f3	&g3	g3	#g3	&a3	a3	#a3	&b3	b3	b2	b1	b0

## The iMelody Object

The complete iMelody object consists of a header, the iMelody data and a footer.

However, Ericsson phones are able to receive and recognise incomplete iMelody objects also, according to this:

**a) Full header and footer** BEGIN:IMELODY  
 VERSION:1.0  
 FORMAT:CLASS1.0  
 < iMelody >  
 END:IMELODY

**b) Partial header, footer** BEGIN:IMELODY  
 < iMelody >  
 END:IMELODY

**c) No header and footer** < iMelody >

Only (a) is according to the standard and hence only (a) is guaranteed to work with other manufacturers' phones. When the melody is sent from an Ericsson phone, it is always sent with header and footer according to a) above, regardless of how the melody was received to the phone.

Without any detailed explanations, this is an example of how the data for a Class1.0 iMelody object is composed, taken from Beethoven's 9th Symphony. In addition to the required Header, iMelody and Footer information shown in this example, an iMelody object can include optional Header information and iMelody modifiers, such as name of melody, name of composer, beat, volume etc.

However, it is not recommended to include the optional information, since it will be ignored by most phones and it consumes valuable space in the Short Message.

<b>iMelody example data</b>	<b>Description</b>
BEGIN:IMELODY VERSION:1.0 FORMAT:CLASS1.0	Header information
MELODY: *3f3f3f3#c1#d3#d3#d3c1r3f3f3f3#c3#f3#f3#f3f3*4#c3#c3 #c3*3#a1	Required iMelody
END:IMELODY	Footer information

The same example in other iMelody representations

---

**iMelody representation of “Beethoven’s 9th” sample**

---

Hex Data representation	0x42, 0x45, 0x47, 0x49, 0x4E, 0x3A, 0x49, 0x4D, 0x45, 0x4C, 0x4F, 0x44, 0x59, 0x0D, 0x0A, 0x56, 0x45, 0x52, 0x53, 0x49, 0x4F, 0x4E, 0x3A, 0x31, 0x2E, 0x30, 0x0D, 0x0A, 0x46, 0x4F, 0x52, 0x4D, 0x41, 0x54, 0x3A, 0x43, 0x4C, 0x41, 0x53, 0x53, 0x31, 0x2E, 0x30, 0x0D, 0x0A, 0x4D, 0x45, 0x4C, 0x4F, 0x44, 0x59, 0x3A, 0x2A, 0x33, 0x66, 0x33, 0x66, 0x33, 0x66, 0x33, 0x23, 0x63, 0x31, 0x23, 0x64, 0x33, 0x23, 0x64, 0x33, 0x23, 0x64, 0x33, 0x63, 0x31, 0x72, 0x33, 0x66, 0x33, 0x66, 0x33, 0x66, 0x33, 0x23, 0x63, 0x33, 0x23, 0x66, 0x33, 0x23, 0x66, 0x33, 0x23, 0x66, 0x33, 0x23, 0x66, 0x33, 0x2A, 0x34, 0x23, 0x63, 0x33, 0x23, 0x63, 0x33, 0x23, 0x63, 0x33, 0x2A, 0x33, 0x23, 0x61, 0x31, 0x0D, 0x0A, 0x45, 0x4E, 0x44, 0x3A, 0x49, 0x4D, 0x45, 0x4C, 0x4F, 0x44, 0x59, 0x0D, 0x0A,
User Data representation	0x96, #UDL (7 bit alphabet, which means the value is filled to an octet) 0x82, #UDHL 0x0C, #IEI 0x80, #IEIL 0x00, #POS 0x42, 0x45, 0x47, 0x49, 0x4E, 0x3A, 0x49, 0x4D, 0x45, 0x4C, 0x4F, 0x44, 0x59, 0x0D, 0x0A, 0x56, 0x45, 0x52, 0x53, 0x49, 0x4F, 0x4E, 0x3A, 0x31, 0x2E, 0x30, 0x0D, 0x0A, 0x46, 0x4F, 0x52, 0x4D, 0x41, 0x54, 0x3A, 0x43, 0x4C, 0x41, 0x53, 0x53, 0x31, 0x2E, 0x30, 0x0D, 0x0A, 0x4D, 0x45, 0x4C, 0x4F, 0x44, 0x59, 0x3A, 0x2A, 0x33, 0x66, 0x33, 0x66, 0x33, 0x66, 0x33, 0x23, 0x63, 0x31, 0x23, 0x64, 0x33, 0x23, 0x64, 0x33, 0x23, 0x64, 0x33, 0x63, 0x31, 0x72, 0x33, 0x66, 0x33, 0x66, 0x33, 0x66, 0x33, 0x23, 0x63, 0x33, 0x23, 0x66, 0x33, 0x23, 0x66, 0x33, 0x23, 0x66, 0x33, 0x2A, 0x34, 0x23, 0x63, 0x33, 0x23, 0x63, 0x33, 0x23, 0x63, 0x33, 0x2A, 0x33, 0x23, 0x61, 0x31, 0x0D, 0x0A, 0x45, 0x4E, 0x44, 0x3A, 0x49, 0x4D, 0x45, 0x4C, 0x4F, 0x44, 0x59, 0x0D, 0x0A,
Packed representation	424547494E3A494D454C4F44590D0A56455253494F4E3A312E300D0A464F524D41543A434C415353312E300D0A4D454C4F44593A2A33663366336633236332366332366332366332366332A342363332363332363332A332361310D0A454E443A494D454C4F44590D0A

For reference, this is the same example in the built-in Melody Composer in the phone

---

**“Beethoven’s 9th” sample in Melody Composer (eMelody format)**

---

fff#C#d#d#dCpfff#c#f#f#ff+#c+#c+#c#A fffc#f#f#ff+#d+#d+#d

---

# PICTURES

Support: R520 (R2), T20e, T29 (R2), T39

No support: -

## Receive Picture to Phone (Mobile Terminated)

The inserted pictures are shown in the message.



How the picture is displayed on the phone obviously depends on the size of the display. It is therefore important to verify the visual appearance of the picture on the actual target devices. If a picture is wider than the display, it will be truncated. If it is higher than the display, the user can scroll up and down to view it.

---

### Variable picture formats

Width:	In multiples of 8 pixels up to the screen width (i.e. 8, 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96)  The max screen width currently is 101, so 96 is the nearest to this, any larger and the width will be truncated.
Height:	From 1 to 1024 pixels. <b>Note:</b> The total product of Width and Height must never exceed 1024 pixels.

---

### Fixed picture formats

Small – 16 x 16 pixels (32 bytes)  
Large – 32 x 32 pixels (128 bytes)

Most phones without support for the EMS standard will simply ignore the picture information when the message is received, and will just display the text in the message. This prevents the phone from displaying unreadable contents to the user.

## User Interaction After Receiving Picture

The user can save the picture to a position in *My Pictures* for future use.

### *To save the picture to My Pictures*

1. Use the (up-)left and (down-)right buttons until the picture is highlighted in focus
2. Press Yes until the Options menu is displayed
3. Save the picture to a position in *My Pictures*

When the picture is saved in *My Pictures*, the user can edit it and also insert it in outgoing messages.

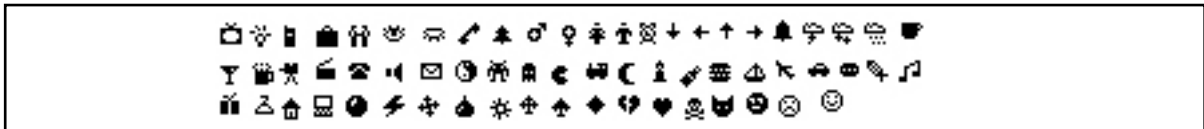
## Send Picture From Phone (Mobile Originated)

The user can insert pictures in messages. This can be pre-installed pictures in the phone, or pictures that the user has saved in *My Pictures*.

The pre-installed pictures are grouped in categories to make it easy to find them:

- Entertainment
- Emotions
- Personal
- Weather
- Sport
- Travel
- Objects 1
- Objects 2

These are examples of pictures in the format 8 x 8 pixels that can be pre-installed in phones:



These are examples of pictures in the format 16 x 16 pixels that can be pre-installed in phones:



These are examples of pictures in the format 32 x 32 pixels that can be pre-installed in phones:

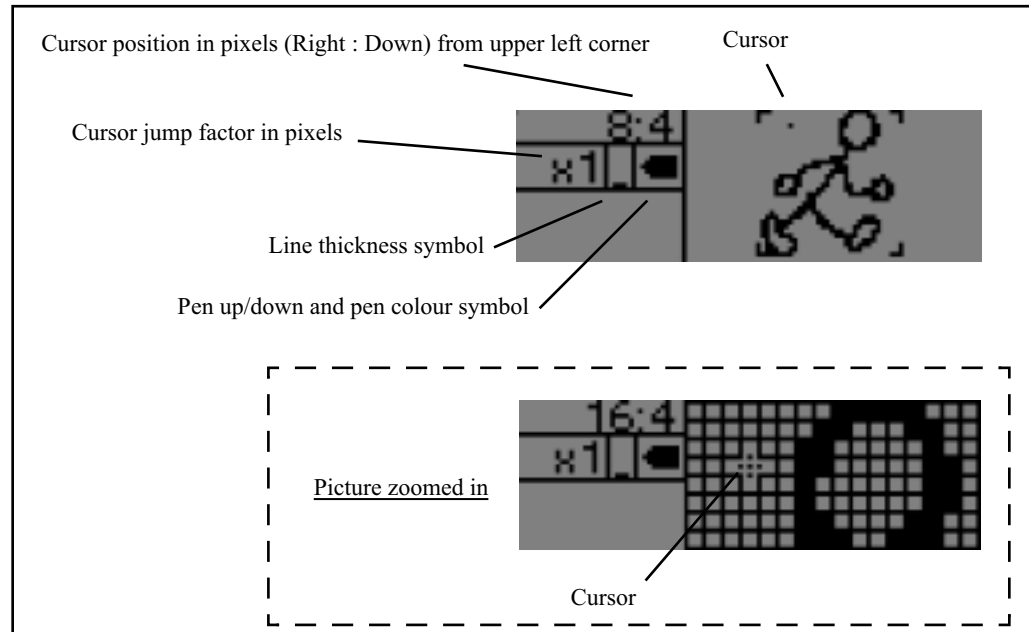


## Picture Editor

The built-in Picture editor allows the user to create a new picture or edit any picture in *My Pictures*.

Feature	Support and use in listed Ericsson phones	
Edit picture saved in <i>My Pictures</i>	Small, Large	Width x Height: see table “Fixed picture formats” on page 20
	Variable	Width x Height: see table “Variable picture formats” on page 20
Create new picture in <i>My Pictures</i>	User defined size	Width: 8, 16 or 32 pixels, Height: 8, 16 or 32 pixels,
Pen tool control, using the keypad on the phone	Keys 1, 2, 3,	Move pen Up (left, center, right)
	Keys 4, 6	Move pen Left, Right
	Keys 7, 8, 9	Move pen Down (left, center, right)
	Key 5	Lift pen up, or put pen down. Long press: zoom in, zoom out
	Key 0	Toggle line thickness 1, 3, 5 etc. pixels
	Key #	Toggle jump factor 1, 3, 5 etc. pixels
	Key *	Toggle black or white colour

For information on how to use the Picture editor, please refer to the User's Guide for the respective phone.



*The visual appearance of the picture editor in a small display is shown in this illustration.*

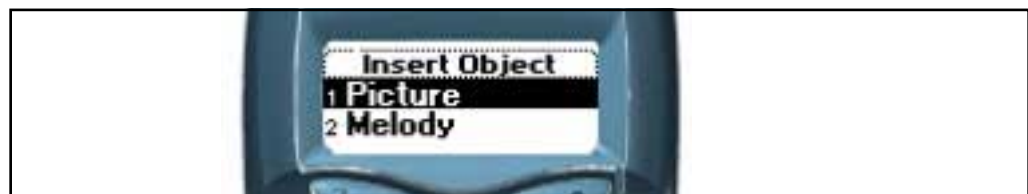
## User Interaction To Send Picture

When the picture is inserted into an outgoing message, the phone takes care of encoding it properly into the Short Message. If the option to send long messages is switched on in the phone, the phone also concatenates several messages, if necessary, see "Concatenated Messages" on page 31.

The user creates a message just like a normal SMS message and inserts the picture at the desired position.

### *To insert a picture in an EMS message*

1. Create the message and type the text
2. At the desired position, press and hold \* or # or the Option key (depending on phone type) until the Options menu is displayed
3. Select Insert Object, Picture and choose from the pre-defined pictures or from the list *My Pictures*
4. To insert more than one picture in the message, or another EMS component, repeat from step 2



# Encoding Picture For Sending Over The Air

## Large Picture

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows:

- |                  |   |
|------------------|---|
| <b>Octet 1</b>   | Position indicating in the SM data the instant the picture should be displayed. Set to the number of characters from the beginning of the SM data after which the picture should be displayed. This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM |
| <b>Octet 2-n</b> | Protocol Data Unit as described in “Picture Format” on page 25. This octet(s) contains a Large Picture.   |

## Small Picture

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows:

- |                  |  |
|------------------|--|
| <b>Octet 1</b>   | Position indicating in the SM data the instant the picture should be displayed in the SM data. Set to the number of characters from the beginning of the SM data after which the picture should be displayed. This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM |
| <b>Octet 2-n</b> | Protocol Data Unit as described in “Picture Format” on page 25. This octet(s) contains a Small Picture.  |

## Variable Picture

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows:

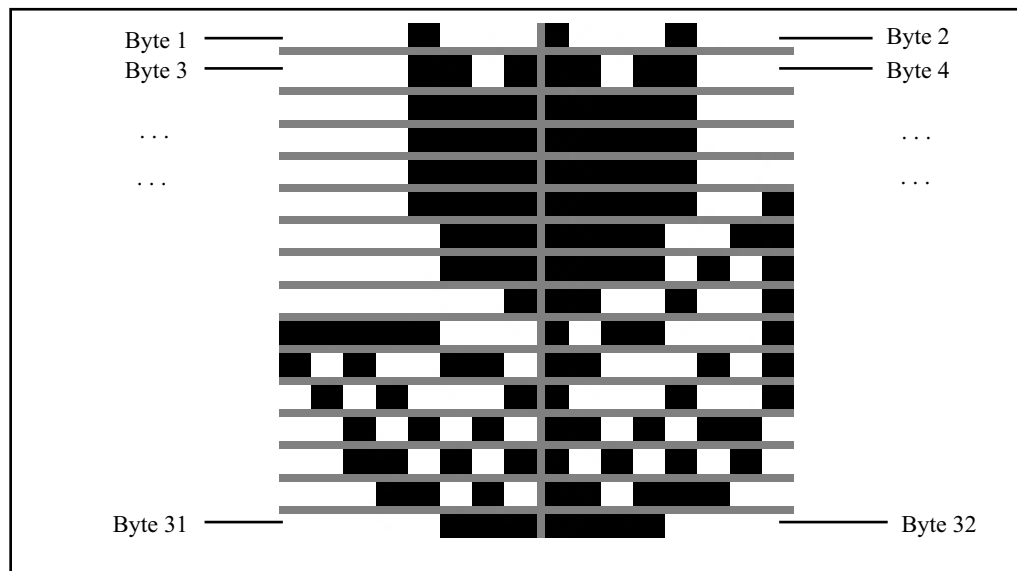
- |                  |  |
|------------------|--|
| <b>Octet 1</b>   | Position indicating in the SM data the instant the picture should be displayed in the SM data. Set to the number of characters from the beginning of the SM data after which the picture should be displayed. This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM |
| <b>Octet 2</b>   | Horizontal dimension of the picture. This octet contains the horizontal number of 8 pixels, which means this value is multiplied by 8 to get the total number of horizontal pixels.  |
| <b>Octet 3</b>   | Vertical dimension of the picture. This octet contains the vertical number of pixels.  |
| <b>Octet 4-n</b> | Protocol Data Unit as described in “Picture Format” on page 25. This octet(s) contains a Variable Picture line by line from top left to bottom right.  |

The values of the horizontal and vertical dimensions must be chosen properly by the sending entity. If the calculated size of this IE exceeds the limits of a single SM or segment it is discarded by the receiving entity.



## Picture Format

Pictures are coded from upper left to lower right and in each byte the most significant bit represent the pixel at the left. The pictures are plain black and white, no colours or grey scales are supported. The bitvalue "0" represents a white pixel and the bitvalue "1" represents a black pixel.



*This illustration shows the format of a Small picture (16 x 16), where:  
Byte 1 = 0A, Byte 2= AA ... Byte 31 = 07, Byte 32 = F0*

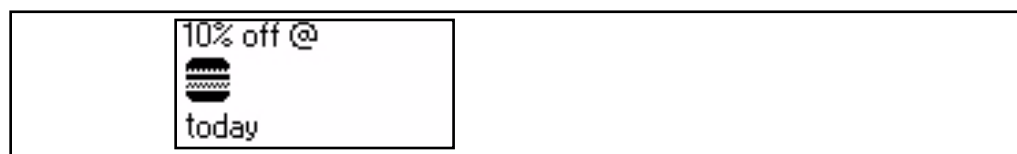
## Picture Example

Example of a message including one small picture is coded as follows:

:

TP UDHI	User Data Header	User Data
1	UDHL=24 IEI=11 IEIDL=22 IED1=11 <Protocol Data Unit for the Small picture, 32 bytes>	10% off @ [<CR><LF><CR> ><LF>] today

Should be displayed as



# ANIMATIONS

Support: R520 (R2), T39  
No support: T20e, T29 (R2)

- Pre-defined** There are a number of pre-defined animations that can be received and sent. The EMS message contains only a reference to the animation. Therefore the pre-defined animation is only limited by the performance and memory capacity of the phone.
- User Defined** The user defined animations can only be received to the phone. Animations are coded as 4 sequential pictures, with the first picture sent first.

## Receive Animation to Phone (Mobile Terminated)

When a message contains an animation, it will be indicated in the text by the first frame of the animation. The animation will be played when brought into focus. This prevents conflicts with other tasks in the phone, for example an ongoing phone call.

There can be multiple animations in one message, and also in combination with pictures, sounds etc. If a message is concatenated in several segments, the phone automatically takes care of combining these into one message.

Animation type	Size
Pre-defined animation	A sequence of pictures of suitable size, pre-installed in the phone. Only a reference is transmitted. In listed Ericsson phones, a sequence is 6 pictures.
User defined, Small	4 pictures of 8 x 8 pixels (4 times 8 bytes transmitted)
User defined, Large	4 pictures of 16 x 16 pixels (4 times 32 bytes transmitted)

The actual frame rate played by the phone depends on the phone model. It is therefore important to verify the playback quality of the animation on the actual target devices.

Most phones without support for the EMS standard will simply ignore the animation information when the message is received, and will just display the text in the message. This prevents the phone from displaying unreadable contents to the user.

## User Interaction After Receiving Animation

The user can play the animation.

### *To play the animation*

1. Use the (up-)left and (down-)right buttons until the animation frame is highlighted in focus
2. The animation will be played

## Send Animation From Phone (Mobile Originated)

The user can insert a pre-defined animation into a message. When the animation (reference) is inserted into an outgoing message, the phone takes care of encoding it properly into the Short Message. If the option to send long messages is switched on in the phone, the phone also concatenates several messages, if necessary, see “Concatenated Messages” on page 31.

### User Interaction To Send Animation

*To insert an animation in an EMS message*

1. Create the message and type the text
2. At the desired position, press and hold \* or # or the Option key (depending on phone type) until the Options menu is displayed
3. Select Insert Object, Animation and choose from the list of available animations
4. To insert more than one animation, or another EMS component, in the message, repeat from step 2



## Encoding Animation For Sending Over The Air

### Pre-defined Animation

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows.

- |                |   |
|----------------|---|
| <b>Octet 1</b> | Position indicating in the SM data the instant the animation should be displayed. Set to the number of characters from the beginning of the SM data after which the animation should be displayed. This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM |
| <b>Octet 2</b> | Animation number. should be encoded as an integer value, see “Animation Format” on page 28.   |

### Large User Defined Animation

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows:

- |                  |  |
|------------------|--|
| <b>Octet 1</b>   | Position indicating the instant the animation should be displayed. Set to the number of characters from the beginning of the SM data after which the animation should be displayed. This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM |
| <b>Octet 2-n</b> | Protocol Data Unit as described in “Animation Format” on page 28. This octet(s) contains a Large Animation.  |




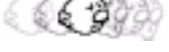


## Small User Defined Animation

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows:

<b>Octet 1</b>	Position indicating the instant the animation should be displayed. Set to the number of characters from the beginning of the SM data after which the animation should be displayed. This octet is coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM
<b>Octet 2-n</b>	Protocol Data Unit as described in “Animation Format” on page 28. This octet(s) contains a Small Animation.

## Animation Format

**Pre-defined** There are a number of pre-defined animations. Each animation number corresponds to a specific animation according to the table below.

Animation no	Description	Animation
0	I am ironic, flirty	
1	I am glad	
2	I am sceptic	
3	I am sad	
4	WOW!	
5	I am crying	

**User Defined** Animations are coded as 4 sequential pictures, with the first picture sent first. The picture data of each picture should be coded as described in “Picture Format” on page 25.

Each animation requires a concatenated message of at least four segments.

Animation type	Size
Small animation	4 pictures of 8 x 8 pixels (4 times 8 bytes)
Large animation	4 pictures of 16 x 16 pixels (4 times 32 bytes)

# FORMATTED TEXT

Support: -

No support: R520 (R2), T20e, T29 (R2), T39

## Receive Formatted Text to Phone (Mobile Terminated)

The formatted text is shown in the message.



The EMS text formatting modifiers are Bold, Italic, Strikethrough, Underline, Small, Normal, Large, in combination with alignment Right, Center, Left. Note that some phones may not support all listed formatting modifiers.

Most phones without support for the EMS standard will simply ignore the formatting information when the message is received, and will just display the text in normal style. This prevents the phone from displaying unreadable contents to the user.

## Send Formatted Text From Phone (Mobile Originated)

The user can format the text in messages sent from the phone. Note that some phones may not support all listed formatting modifiers.

## Encoding Formatted Text For Sending Over The Air

The Information-Element-Data octet(s) placed in the TP-UDH is coded as follows.

- |                |  |
|----------------|--|
| <b>Octet 1</b> | Start position of the text formatting. Set to the number of characters from the beginning of the SM data after which the formatting applies.<br><br>This octet should be coded as an integer value in the range 0 (beginning of the SM data) to the maximum number of characters included in the SM data of one single SM or one segment of a concatenated SM. |
| <b>Octet 2</b> | Text formatting length. Gives the number of formatted characters. This octet should be coded as an integer value in the range 1 to the maximum number of characters for which the formatting applies in one single SM or one segment of a concatenated SM.   |
| <b>Octet 3</b> | Formatting mode value coded as following:<br><br>Octet 3: Bit 7 Bit 6 Bit 5 Bit 4 Bit 3 Bit 2 Bit 1 Bit 0<br>The description of each bit is found in “Text Format” on page 30  |

## Text Format

Note that to some languages, some of these formatting modifiers are irrelevant.

Bit 1	Bit 0	Alignment *)
0	0	Left
0	1	Center
1	0	Right
1	1	Language dependent ( <b>default</b> )

\*) Note: in case formatting text is inserted on the same line as previous non formatting text or with a different mode value, the alignment value must be set to the same value as the previous formatted pre-defined object.

Bit 3	Bit 2	Font size
0	0	Normal ( <b>default</b> )
0	1	Large
1	0	Small
1	1	<i>Reserved</i>

Bit 4	Style Bold
1	Bold on
0	Bold off ( <b>default</b> )

Bit 5	Style Italic
1	Italic on
0	Italic off ( <b>default</b> )

Bit 6	Style Underlined
1	Underlined on
0	Underlined off ( <b>default</b> )

Bit 7	Style Strikethrough
1	Strikethrough on
0	Strikethrough off ( <b>default</b> )

# CONCATENATED MESSAGES

Support: R520 (R2), T20e, T29 (R2), T39

No support: -

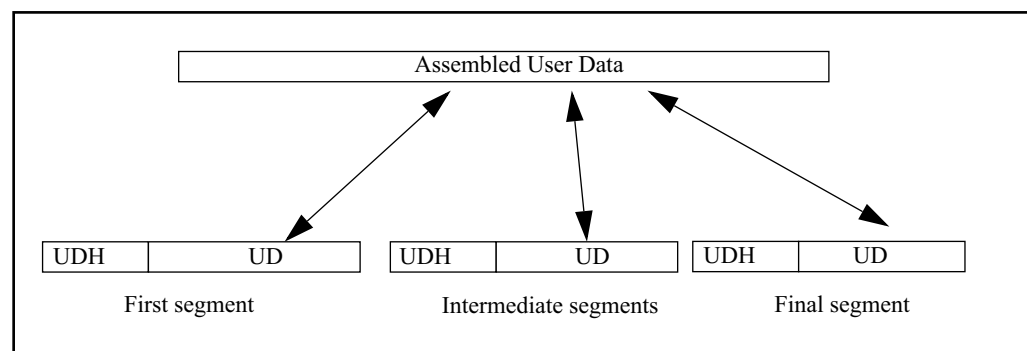
Concatenated messages are required in most cases when using several types of EMS elements, since it is only possible to send one large picture/large animation/melody in one single SM. After including either of these elements, there are very few characters left to the text part:

- For 8 bit messages, there are only 3 (or 9 if no concatenation is used) characters left
- For 16 bit messages, there are only 2 (or 8 if no concatenation is used) characters left

(Both 8 bit and 16 bit message reference is supported, though 8 bit is recommended, since some networks do not properly support 16 bit message reference.)

The concatenation mechanism permits longer messages than 140 bytes. In fact, the SMS standard allows up to 255 messages at 140 bytes each to be concatenated to one message up to about 38k bytes. However, current phones support concatenation of between 3 and 6 messages (product dependent). Larger concatenated messages are treated as separate messages.

For example, on a phone that supports a concatenation of 6, if a message was received that comprised of 10 concatenations then the user would see this as two separate message; one with 6 SM segments, the other with 4 segments.



*The illustration shows how the message is built up by three segments.*

When sending and receiving from and to the phone, the phone automatically takes care of this process.

## Concatenation Format

Concatenation works similar to EMS features whereby each SM segment in a concatenated message includes a concatenation IE in the UDH:

- |                |  |
|----------------|--|
| <b>Octet 1</b> | Provides a message reference (8 bit or 16 bit) so that different concatenated messages can be distinguished. |
| <b>Octet 2</b> | Gives the number of SM segments in the concatenated message.   |
| <b>Octet 3</b> | Gives the SM segment sequence number within the concatenated message.  |

If one or more objects are embedded in one segment of a concatenated message, the IE octet indicating its/their position within the SM data cannot be set to a value that would refer to a position in the next segment(s). For example, if a formatted text that could not be conveyed in one segment it has to be split in several segments. In that case, the IE relating to the formatting has to be repeated in all the segments in which it applies. The following example shows a concatenated EMS message.

## EMS Concatenation Example

Example of a message including 1 Large Picture, 3 Small animations and 2 User defined Melodies together with some text.

The EMS message: <Large Picture1> <User Defined Melody 1> **Hello All, This is a real Enhanced Message** <Small Animation 1> . **I can send** <Small Animation 2> **and receive** <Small Animation 3> **really advanced EMS messages** <User Defined Melody2>

This EMS message has to use concatenated messages. Note that the receiving phone will have to be capable of receiving a concatenated message made up of 5 segments, otherwise the message will be truncated or partly ignored.

The message will typically contain the following data:

SM	User Data Header	User Data
1	IEI=10 (Large Picture) IED1=00 (beginning of the SM) <Large Picture 1 (128 bytes)>	[<CR><LF>] space
2	IEI=0C (User Defined Sound) IED1=00 (beginning of the SM) <User Melody 1 (128 bytes max)>	Hel (only 3 characters left for text in this SM)
3	IEI=0F (Small Animation) IED1=27 (39th position) <Small Animation 1 (32 bytes)> IEI=0F (Small Animation) IED1=32 (50th position) <Small Animation 2 (32 bytes)>	lo All, This is a real Enhanced Message. I can send and receive
4	IEI=0F (Small Animation) IED1=00 (beginning of the SM) <Small Animation 3 (32 bytes)>	really advanced EMS messages.
5	IEI=0C (User Defined Sound) IED1=00 (beginning of the SM) <User Melody 2 (128 bytes max)>	[<CR><LF>]



# TERMINOLOGY AND ABBREVIATIONS

3GPP	3rd Generation Partnership Project.
CR	Carriage Return - a carriage return is a special code that moves the cursor to the beginning of the current line.
EMS	Enhanced Messaging Service, an enhancement of the SMS standard. This makes it possible to include pictures, melodies, sounds and animations in messages, and also to receive and edit new pictures and melodies on the phone.
ETSI	European Telecommunications Standards Institute.
GSM	Global System for Mobile Communications. GSM is the world's most widely-used digital mobile phone system, now operating in over 100 countries around the world, particularly in Europe and Asia-Pacific.
Hexadecimal	Refers to the base-16 number system, which consists of 16 unique symbols: the numbers 0 to 9 and the letters A to F. For example, the decimal number 15 is represented as F in the hexadecimal numbering system. The hexadecimal system is useful because it can represent every byte (8 bits) as two consecutive hexadecimal digits.
IE	Information Element
IED(n)	Information Element Data (octet number)
iMelody	The iMelody format, specified by IrDA, is a minimal set of tones that can be used in EMS applications to transfer melodies, such as ring tones, between devices.
IrDA	Infrared Data Association.
LF	Line Feed
Octet	An octet is 8 bits. It is equivalent to a byte, as long as the byte is also 8 bits.
Service Provider	A company that provides services, for example subscriptions to mobile phone users.
SM	Short Message.
SMS	Short Message Service. Allows messages of up to 160 characters to be sent and received via the network operator's message centre to your mobile phone. Messages are stored if the phone is off or out of reach ensuring that they reach you. To use this service, it must be supported by your network.
TP	Transfer Protocol.
UDH	User Data Header.
WAP	Wireless Application Protocol. Handheld devices, low bandwidth, binary coded, a deck/card metaphor to specify a service. A card is typically a unit of interaction with the user, that is, either presentation of information or request for information from the user. A collection of cards is called a deck, which usually constitutes a service.

# RELATED INFORMATION

## Documents

- EMS specifications are found in “Technical Specification 3GPP; Technical realization of the Short Message Service (SMS)” ((3G TS 23.040 V4.0.0 (2000-07))
- iMelody standard issued by Infrared Data Association.
- EMS White Paper
- Ericsson R520 (R2) White Paper
- Ericsson T20e White Paper
- Ericsson T29 (R2) White Paper
- Ericsson T39 White Paper

## Links

- <http://mobileinternet.ericsson.com> - a site for the mobile phone user
- <http://www.ericsson.com/mobilityworld> - information, tools, white papers and software updates on Ericsson products and technologies; check frequently!
- <http://www.gprsworld.com> - home of the Mobile Applications Initiative created by Ericsson
- <http://www.3gpp.org> - home of the 3rd Generation Partnership Project
- <http://www.irda.com> - home of the Infrared Data Association
- <http://www.etsi.org> - home of the European Telecommunications Standards Institute

## Trademarks And Acknowledgements

The IrDA Feature Trademark is owned by the Infrared Data Association and is used under licence there from.





# Index

## A

Abbreviations 33  
Acknowledgement 34  
Animation format 28

## C

Concatenated messages 7  
Concatenation example 32

## D

Documents 34  
Duration 15

## E

EMelody format 17  
EMS 5  
EMS components 6  
EMS concatenation example 32  
EMS overview 6  
Enhanced Messaging Service 5  
Extended EMelody format 17

## I

Identification mechanism for EMS 7

## L

Large picture specification 24  
Large user defined animation 27  
Linked messages 7  
Links 34  
Long message 7

## M

Melody composer 15  
Melody length 15

## N

Note range 15

## P

Phones that support EMS 8  
Picture editor 22  
Picture editor display 23  
Picture example 25  
Picture format 25  
Picture formats 20  
Picture height 20  
Picture width 20  
Pictures, animations and formatted text 7  
Pre-defined animation 27  
Pre-defined animation numbers 28  
Pre-defined sound numbers 13

Preface 5  
Purpose of this document 5

## **R**

R520 (R2) 8  
Related information 34

## **S**

Send sound from phone 12  
Short message, diagram 6  
Small picture specification 24  
Small user defined animation 28  
Sounds and melodies 6

## **T**

T20e 9  
T29 (R2) 10  
T39 11  
Terminology and abbreviations 33  
Text format specification 30  
The iMelody format support 16  
The iMelody object 18  
Trademarks and acknowledgements 34

## **U**

User interaction after receiving animation 26  
User interaction after receiving melody 14  
User interaction after receiving picture 20  
User interaction after receiving sound 12  
User interaction to send animation 27  
User interaction to send melody 15  
User interaction to send picture 23  
User interaction to send sound 13

## **V**

Variable picture specification 24