

INTERNATIONAL STANDARD

ISO/IEC
9636-5

First edition
1991-12-15

**Information technology — Computer graphics —
Interfacing techniques for dialogues with
graphical devices (CGI) — Functional
specification —**

Part 5:
Input and echoing

*Technologies de l'information — Infographie — Interfaces pour
l'infographie — Spécifications fonctionnelles —*

Partie 5: Entrée et résonance



Reference number
ISO/IEC 9636-5:1991(E)

Contents

	Page
Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	2
3 Concepts	3
3.1 Introduction	3
3.2 Basic input model	3
3.3 Logical input devices	3
3.4 Measures	4
3.4.1 Measures of input class LOCATOR	4
3.4.2 Measures of input class STROKE	4
3.4.3 Measures of input class VALUATOR	5
3.4.4 Measures of input class CHOICE	5
3.4.5 Measures of input class PICK	5
3.4.6 Measures of input class STRING	5
3.4.7 Measures of input class RASTER	5
3.4.8 Measures of input class GENERAL	6
3.5 Coordinate systems for VDC measures	6
3.6 Triggers	8
3.6.1 Trigger association	8
3.6.2 Timeouts	8
3.6.3 The break action	8
3.7 Input methods and state model	9
3.7.1 Logical input device model	9
3.7.2 Request input	10
3.7.3 Sample input	10
3.7.4 Remote echoing	10
3.7.5 Event input	13
3.8 Prompting, echoing, and acknowledgement	15
3.8.1 Prompting	15
3.8.2 Echoing	16
3.8.3 Acknowledgement	18
3.9 Portioning of returned input data	19
3.10 State restrictions	19
3.11 Inquiry	20
4 Interactions with other parts of ISO/IEC 9636	21
4.1 Interactions with more than one part of ISO/IEC 9636	21
4.2 Interactions with ISO/IEC 9636-1 (Overview)	21
4.3 Interactions with ISO/IEC 9636-2 (Control)	21
4.4 Interactions with ISO/IEC 9636-3 (Output)	21
4.5 Interactions with ISO/IEC 9636-4 (Segments)	21
4.6 Interactions with ISO/IEC 9636-6 (Raster)	22
5 Abstract specification of functions	23
5.1 Introduction	23
5.1.1 Data types employed	23
5.1.2 Validity of returned information	23
5.2 Input control functions	23
5.2.1 INITIALIZE LOGICAL INPUT DEVICE	23
5.2.2 RELEASE LOGICAL INPUT DEVICE	24

5.2.3	ECHO CONTROLS	24
5.2.4	PUT CURRENT <input class> MEASURE	25
5.2.5	ECHO DATA	26
5.2.6	<input class> DEVICE DATA	27
5.2.7	ASSOCIATE TRIGGERS	30
5.2.8	GET ADDITIONAL STROKE DATA	30
5.2.9	GET ADDITIONAL PICK DATA	31
5.2.10	GET ADDITIONAL STRING DATA	31
5.2.11	GET ADDITIONAL RASTER DATA	32
5.3	Request and sample functions	32
5.3.1	REQUEST <input class>	32
5.3.2	SAMPLING STATE	34
5.3.3	SAMPLE <input class>	34
5.4	Echo request input functions	35
5.4.1	INITIALIZE ECHO REQUEST	35
5.4.2	ECHO REQUEST <input class>	36
5.5	Event input functions	38
5.5.1	INITIALIZE EVENT QUEUE	38
5.5.2	RELEASE EVENT QUEUE	38
5.5.3	ENABLE EVENTS	39
5.5.4	DISABLE EVENTS	39
5.5.5	EVENT QUEUE BLOCK CONTROL	40
5.5.6	FLUSH EVENTS	40
5.5.7	FLUSH DEVICE EVENTS	40
5.5.8	AWAIT EVENT	41
5.5.9	DEQUEUE <input class> EVENT	42
5.5.10	EVENT QUEUE TRANSFER	43
5.6	Echo output functions	43
5.6.1	INITIALIZE ECHO OUTPUT	43
5.6.2	RELEASE ECHO OUTPUT	44
5.6.3	ECHO OUTPUT CONTROLS	44
5.6.4	PERFORM ACKNOWLEDGEMENT	45
5.6.5	UPDATE <input class> ECHO OUTPUT	45
5.6.6	ECHO OUTPUT DATA	46
6	Input and echoing inquiry functions	48
6.1	Introduction	48
6.1.1	Data types employed	48
6.1.2	Validity of returned information	48
6.2	Input description table	48
6.2.1	INQUIRE INPUT CAPABILITY	48
6.2.2	INQUIRE LIST OF AVAILABLE INPUT DEVICES	48
6.3	Class-independent logical input device description table	49
6.3.1	INQUIRE COMMON INPUT DEVICE PROPERTIES	49
6.3.2	INQUIRE LIST OF SUPPORTED ECHO TYPES	49
6.3.3	INQUIRE LIST OF SUPPORTED PROMPT TYPES	49
6.3.4	INQUIRE LIST OF SUPPORTED ACKNOWLEDGEMENT TYPES	50
6.3.5	INQUIRE LIST OF ASSOCIABLE TRIGGERS	50
6.4	Class-specific logical input device description table	50
6.4.1	INQUIRE LOCATOR CAPABILITIES	50
6.4.2	INQUIRE STROKE CAPABILITIES	51
6.4.3	INQUIRE CHOICE CAPABILITIES	51
6.4.4	INQUIRE PICK CAPABILITIES	51
6.4.5	INQUIRE STRING CAPABILITIES	51
6.4.6	INQUIRE LIST OF AVAILABLE INPUT CHARACTER SETS	52
6.4.7	INQUIRE RASTER INPUT CAPABILITIES	52
6.4.8	INQUIRE LIST OF PERMITTED RASTER SPOT CENTRE SEPARATIONS	52
6.4.9	INQUIRE GENERAL CAPABILITIES	52
6.4.10	INQUIRE LIST OF SUPPORTED GENERAL MEASURE FORMATS	53
6.5	Class-independent logical input device state list	53
6.5.1	INQUIRE COMMON LOGICAL INPUT DEVICE STATE	53
6.5.2	INQUIRE LIST OF ASSOCIATED TRIGGERS	53

6.5.3	INQUIRE ECHO DATA RECORD	54
6.5.4	INQUIRE INPUT DEVICE DATA RECORD	54
6.6	Class-specific logical input device state list	54
6.6.1	INQUIRE LOCATOR STATE	54
6.6.2	INQUIRE STROKE STATE	55
6.6.3	INQUIRE VALUATOR STATE	55
6.6.4	INQUIRE CHOICE STATE	56
6.6.5	INQUIRE PICK STATE	56
6.6.6	INQUIRE STRING STATE	56
6.6.7	INQUIRE RASTER INPUT STATE	57
6.6.8	INQUIRE GENERAL STATE	57
6.7	Event input state list	57
6.7.1	INQUIRE EVENT INPUT STATE	57
6.8	Echo output description table	58
6.8.1	INQUIRE ECHO OUTPUT CAPABILITIES	58
6.8.2	INQUIRE LIST OF ECHO OUTPUT ECHO TYPES	58
6.8.3	INQUIRE LIST OF ECHO OUTPUT PROMPT TYPES	58
6.8.4	INQUIRE LIST OF ECHO OUTPUT ACKNOWLEDGEMENT TYPES	59
6.8.5	INQUIRE LIST OF SUPPORTED GENERAL FORMAT IDENTIFIERS	59
6.9	Echo entity state list	59
6.9.1	INQUIRE LIST OF CURRENTLY EXISTING ECHO ENTITIES	59
6.10	Individual echo entity state list	60
6.10.1	INQUIRE ECHO ENTITY STATE	60
6.10.2	INQUIRE ECHO OUTPUT DATA RECORD	60
7	CGI description tables and state lists	61
7.1	Description tables	61
7.1.1	Input capability	61
7.1.2	Class-independent logical input capability	61
7.1.3	Echo output capability	63
7.2	State lists	64
7.2.1	Input state	64
7.2.2	Class-independent logical input device state	64
7.2.3	Class-specific logical input device state	65
7.2.4	Events	67
7.2.5	Echo entity state	68
A	Formal grammar of the functional specification	69
B	Input errors	92
C	Guidelines for CGI implementors	94

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 9636-5 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

ISO/IEC 9636 consists of the following parts, under the general title *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification*:

- Part 1: Overview, profiles, and conformance
- Part 2: Control
- Part 3: Output
- Part 4: Segments
- Part 5: Input and echoing
- Part 6: Raster

iTeh STANDARD PREVIEW
(standards.iteh.ai)

Annexes A and B form an integral part of this part of ISO/IEC 9636. Annex C is for information only.

ISO/IEC 9636-5:1991
<https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991>

Introduction

This part of ISO/IEC 9636 describes the interface between a device-independent part of a graphics system and INPUT or OUTPUT Virtual Devices from which graphical and non-graphical inputs can be obtained.

CGI input functions control and perform different methods of input from a Virtual Device, allowing the return of an input value according to user requirement.

CGI input functions return a variety of data types which correspond to output data commonly used in graphics systems.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 9636-5:1991

<https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991>

Information technology – Computer graphics – Interfacing techniques for dialogues with graphical devices (CGI) – Functional specification –

Part 5: Input and echoing

1 Scope

This part of ISO/IEC 9636 defines those functions of the Computer Graphics Interface concerned with obtaining graphical and non-graphical input from a Virtual Device of device class INPUT or OUTIN. This part of ISO/IEC 9636 also defines functions to support echoing of input operations on separate Virtual Devices.

This part of ISO/IEC 9636 is part 5 of ISO/IEC 9636, and should be read in conjunction with ISO/IEC 9636-1, ISO/IEC 9636-2, and ISO/IEC 9636-4. The relationship of this part of ISO/IEC 9636 to the other parts of of ISO/IEC 9636 is described in ISO/IEC 9636-1 and in clause 4.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 9636-5:1991](https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991)

<https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991>

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9636. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9636 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 9636-1 : 1991 *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 1: Overview, profiles, and conformance.*

ISO/IEC 9636-2 : 1991 *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 2: Control.*

ISO/IEC 9636-3 : 1991 *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 3: Output.*

ISO/IEC 9636-4 : 1991 *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 4: Segments.*

ISO/IEC 9636-6 : 1991 *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 6: Raster.*

ISO/IEC 9637-1 : -¹⁾ *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Data stream binding — Part 1: Character encoding.*

ISO/IEC 9637-2 : -¹⁾ *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Data stream binding — Part 2: Binary encoding.*

ISO/IEC TR 9973 : 1988 *Information processing — Procedures for registration of graphical items.*

[ISO/IEC 9636-5:1991](https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991)

<https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991>

¹⁾ To be published.

3 Concepts

3.1 Introduction

This part of ISO/IEC 9636 defines those functions of the Computer Graphics Interface concerned with input and echoing. This functionality is divided into the following areas

- *Input control functions*, which provide control over initialization and deallocation of Logical Input Devices (LIDs), together with the means to tailor their characteristics.
- *Request and sample functions*, which permit LIDs to be used with Request and Sample input methods.
- *Echo request input functions*, which permit LIDs to be used with the Echo Request input method, a special type of Request method in which changes to the LID's measure value can be tracked by the client.
- *Event input functions*, which permit LIDs to be used with the Event input method; this allows the client to control a number of active LIDs simultaneously while concurrently performing graphical output.
- *Echo output functions*, which allow values to be echoed on a given CGI Virtual Device when the source of such values was not the given device.
- *Input and echoing inquiry functions*, which provide access to the description tables and state lists concerned with input and echoing.

3.2 Basic input model

The functions specified by this part of ISO/IEC 9636 are defined in terms of a model of the operation of graphical input. This sub-clause defines the relationship between the top level concepts involved, and the following sub-clauses detail each of the elements of the input model.

An INPUT or OUTIN CGI Virtual Device provides input capabilities through *Logical Input Devices (LIDs)*.

Input may be acquired by means of different *input methods*, and the client of the CGI may also exert control of the operation of *prompt*, *echo*, and *acknowledgement* capabilities of the Virtual Device.

3.3 Logical input devices

A logical input device consists of a *measure*, *state information* and a set of associated *triggers*. (A LID's set of associated triggers may be empty.)

A LID is identified by the *input class* of its measure (see table 1) and by its *LID index*. There is a separate set of LID indices for each class of measure.

Each LID has a description table and a state list; each is divided into a class-independent and a class-specific portion. The Class-Independent Logical Input Device State List holds entries for the input device state, the sample input state, and the controls for prompt, echo and acknowledgement. The Class-Specific Logical Input Device State List defines the control that is applied to the LID's measure. Both the class-specific and the class-independent state lists form the Logical Input Device State List, which exists for every LID. It is not required that the list of available logical input device indices for a specific input class be dense.

Table 1 – Input Classes

Input classes		
Input Class	Data returned	Example
LOCATOR	a single VDC point	digitizer, mouse
STROKE	a sequence of VDC points	digitizer
VALUATOR	a number from a continuous range of alternatives	potentiometer
CHOICE	an integer from a bounded range of alternatives	button box
PICK	a pick status, and the list of pick values, containing a pick identifier and segment identifier	light pen
STRING	a character string	alphanumeric keyboard
RASTER	an input colour value array	scanner
GENERAL	a data record	voice input

3.4 Measures

A measure is the modelled counterpart of the real-world entity that generates values that feed into the CGI input mechanisms. For the purposes of the input model, a measure has three components, an *input tool*, a *current value* and a *measure validity*. Both the current value and the measure validity are interrogated by the LID as part of any input action; the measure validity status determines completeness and reliability of the measure. If the measure validity is VALID, the measure value is complete and reliable and may be returned. If the measure validity is INVALID, the current value of the measure may not be returned and no meaning should be attached to the returned parameter value.

In normal operation, the input tool modifies the current value and the measure validity under the influence of an outside agent (for example, an operator).

The CGI client may also set the current value and the measure validity, using the function PUT CURRENT <input class> MEASURE. Some physical implementations of measures may render implementation of this function impractical; this situation is indicated by the Put Current Measure Effective entry in the Class-Independent Logical Input Device Description Table.

3.4.1 Measures of input class LOCATOR

The value of a LOCATOR measure is a point in VDC space. A separate VDC coordinate system, independent of that used for graphical output, may be specified for each LOCATOR LID. (See 3.5.)

The criterion for the validity of a LOCATOR measure is that the point measured should be within the Input Extent.

Measures of input class LOCATOR may be implemented, for example, by a mouse, trackball, tablet puck, joystick, thumb-wheels, and by digitizing tables.

3.4.2 Measures of input class STROKE

The value of a STROKE measure is a list of points in VDC space. A separate VDC coordinate system, independent of that used for graphical output, may be specified for each STROKE LID. (See 3.5.)

There is no criterion for a STROKE measure to be invalid; points outside the Input Extent are rejected.

An implementation of a STROKE measure will typically involve several external actions before a measure value becomes available. Successive use of SAMPLE input from a STROKE measure will characteristically produce values that (at least in part) repeat one another. The “complete” measure value is that returned to a REQUEST or ECHO REQUEST input function, or that entered in the event queue when a trigger fires. After the complete measure value has been made available, use of SAMPLE will return an “empty” measure value until further external action has occurred.

Measures of input class STROKE may be implemented by the same sorts of devices as for class LOCATOR, with a (possibly) separate method of signalling “input complete”.

3.4.3 Measures of input class VALUATOR

The value of a VALUATOR measure is a real number in a range defined by maximum and minimum values specified by entries in the Class-Specific Logical Input Device State List.

The criterion for the validity of a VALUATOR measure is that its value should fall in the specified range.

Measures of input class VALUATOR may be implemented, for example, by a single thumb-wheel or by a potentiometer knob.

3.4.4 Measures of input class CHOICE

The value of a CHOICE measure is an integer in a range whose lower bound is 1 and whose upper bound is defined in the Class-Specific Logical Input Device Description Table.

The criterion for the validity of a CHOICE measure is that its value should fall in the specified range.

Measures of input class CHOICE may be implemented, for example, by a button box, by a tablet, or screen menu.

3.4.5 Measures of input class PICK

The value of a PICK measure is a list of pick values, (see the definition of data type PV in ISO/IEC 9636-1, 5.2.10). The implementation of input class PICK requires an OUTIN Virtual Device to support the capabilities defined in ISO/IEC 9636-4. A separate VDC coordinate system, independent of that used for graphical output, may be specified for each PICK LID. (See 3.5.) This VDC coordinate system is used to define the pick aperture relative to the pick location.

The criteria that define which segments are picked are defined in ISO/IEC 9636-4, 3.8. PICK measures may exist only on Virtual Devices of device class OUTIN, since they derive values from an interaction between an input tool and segment storage of the Virtual Device. The PICK measure is INVALID if the number of pick values exceeds the maximum specified in the Class-Specific Logical Input Device State List.

Measures of input class PICK may be implemented by the same sort of devices as for class LOCATOR with some means to link it to segment storage.

[ISO/IEC 9636-5:1991](https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-f03ce93f/iso-iec-9636-5-1991)

[https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-](https://standards.iteh.ai/catalog/standards/sist/070f1084-5b9f-497e-8be9-f03ce93f/iso-iec-9636-5-1991)

3.4.6 Measures of input class STRING

The value of a STRING measure is a string of characters which is subject to the Input Character Set Index, Alternate Input Character Set Index, and the Input Character Coding Announcer entries in the Class-Specific Logical Input Device State List.

The STRING measure is INVALID if the number of characters exceeds the maximum specified in the Class-Specific Logical Input Device State List.

An implementation of a STRING measure will typically involve several external actions before a measure value becomes available. Successive use of SAMPLE input from a STRING measure will characteristically produce values that (at least in part) repeat one another. The “complete” measure value is that returned to a REQUEST or ECHO REQUEST input function, or that entered in the event queue when a trigger fires. After the complete measure value has been made available, use of SAMPLE will return an “empty” measure value until further external action has occurred.

A measure of input class STRING may be implemented, for example, by means of a computer keyboard.

3.4.7 Measures of input class RASTER

The value of a RASTER measure is an array of input colour values (typically colour or grey-scale intensities of an external image). The representation of each colour value is determined by the entries Colour and Bits Per Colour in the Class-Specific Logical Input Device State List. The Source Window, specified in pixels, in the Client-Specific Logical Input Device State List, determines the region from which the returned input colour values are obtained. Invoking the function RASTER DEVICE DATA can be used to change the setting of this Source Window. The pixel offsets used to specify the source window may result in the return of an invalid RASTER measure.

The measure of input class RASTER may be implemented, for example, by a video camera, a scanner, facsimile machine, or remote-sensing equipment. A conforming measure of input class RASTER shall not be implemented using the contents of a CGI bitmap. (GET PIXEL ARRAY, defined in ISO/IEC 9636-6, can be used for this purpose.)

3.4.8 Measures of input class GENERAL

The value of a GENERAL measure is a data record, whose format is determined by a measure format identifier. The GENERAL input class provides a mechanism for addressing, within the CGI model of input, logical input devices which do not match one of the other standardized input classes. The GENERAL measure is INVALID if the data record size exceeds the maximum specified in the Class-Specific Logical Input Device State List.

Non-negative measure format identifiers for input class GENERAL are reserved for those registered with the ISO Registration Authority. Negative measure format identifiers may be used for implementation-dependent formats.

3.5 Coordinate systems for VDC measures

LOCATOR and STROKE devices deliver VDC measure values. A GENERAL input device may also deliver a VDC measure value. Although PICK devices do not deliver VDC measure values, pick input uses a pick aperture specified in VDC that is used in determining the list of pick values returned.

The characteristics of the physical device that supports such logical input devices appear in the Class-Specific Logical Input Device Description Table. These physical characteristics include:

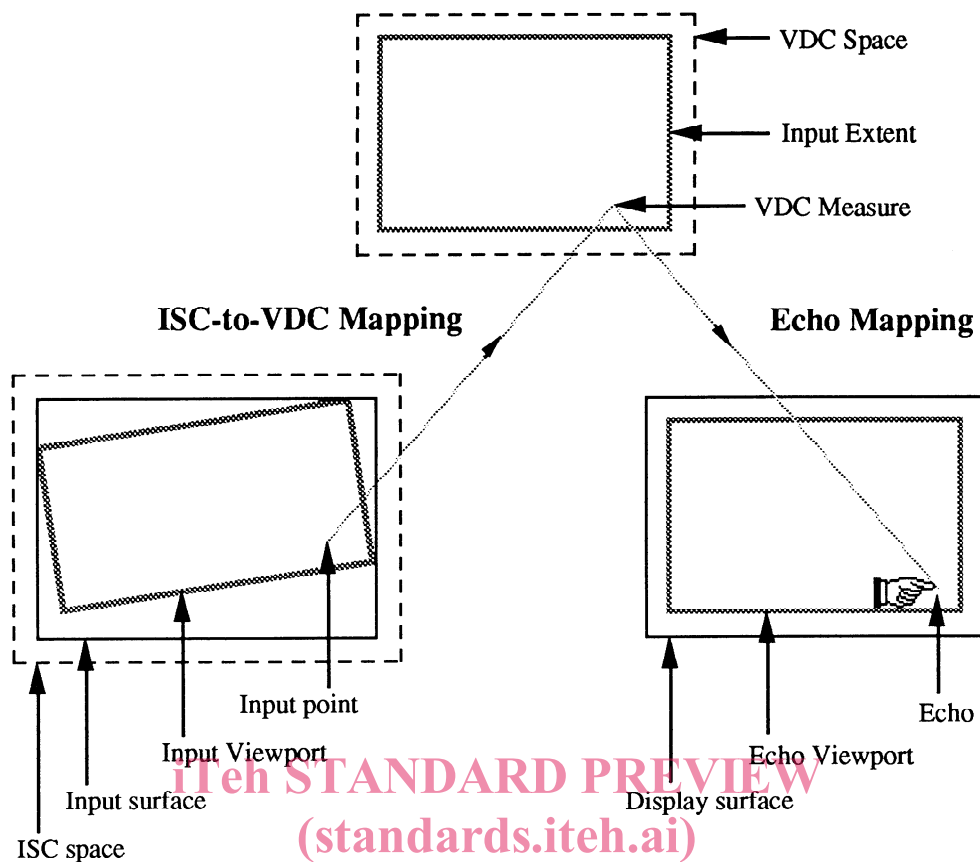
- Input device address space. This is the native coordinate range of the input device, and is expressed as a pair of ISPs, being the extreme lower left and upper right corners of the address range.
- Input device physical size. This is expressed as a pair of real numbers which gives the dimensions of the device in millimetres. For certain types of device (e.g. mouse, trackball) this may refer to a nominal rectangle rather than an actual size.
- Input device resolution. This is expressed as the number of resolvable steps over the complete address range of each axis. This information is of an advisory nature, since the resolution may vary across a device's range and through its operational lifetime.

For device of class LOCATOR, STROKE, and PICK, the Class-Specific Logical Input Device Description Table specifies a Physical Input Surface Size entry, in millimetres, and an Input Surface Size Interpretation entry with possible values of NOMINAL, ACTUAL, and UNLIMITED. The value ACTUAL is used for input devices for which an exact physical size is meaningful (e.g. tablets). The value NOMINAL is used for input devices for which an effective bounding region of a conceptual approximate size is more appropriate (e.g. a mouse or trackball). The value UNLIMITED indicates that no bound is meaningful and the Physical Input Surface Size entry only relates to the interpretation of the Number of Distinguishable Steps in XY Directions entry in the Class Specific Logical Input Device Description Table.

Logical Input Devices of class LOCATOR, STROKE, and PICK all start from a point in ISC space which determines a point in VDC space. The ISC-to-VDC Mapping for such a LID maps points on its input surface to VDC space. In order to echo such a point for an operator, the resulting VDC point is transformed by the echo mapping associated with the LID.

The ISC-to-VDC Mapping is specified in terms of an Input Viewport in ISC space and an Input Extent for the LID. The Input Viewport is a parallelogram on the input surface. It is nominally a rectangle aligned with the ISC axes; however, provision is made for the possibility that this rectangle is rotated or skewed. The Input Viewport rectangle is parameterized by three ISPs, conceptually corresponding to bottom left, bottom right and top right. The resulting ISC-to-VDC Mapping is a transformation which maps the first point of the Input Viewport onto the first point of the Input Extent, the second point of the Input Viewport onto a VDC point with the same x-coordinate as the second point of the Input Extent, and the same y-coordinate as the first point of the Input Extent. The third point of the Input Viewport is mapped onto the second point of the Input Extent.

For devices of class OUTIN, the echo mapping is determined by the Input Extent for the LID and an Echo Viewport for the LID on the display surface. The Echo Viewport is specified as two VPs. The echo mapping is linear in each of x and y and maps the first point of the Input Extent onto the first point of the Echo Viewport, and similarly for the second points, see figure 1.



ISO/IEC 9636-5:1991
<https://standards.iteh.ai/catalog/standards/sist/07961084-5b9f-497e-8be9-024cf03ce93f/iso-iec-9636-5-1991>
Figure 1 – Input Surface Coordinate mapping

When a LID with a VDC measure is initialized on a device of class OUTIN, the state list entries controlling its ISC-to-VDC Mapping and its echo mapping are initialized in such a way that the Input Viewport is mapped isotropically onto the entire display surface. This Input Viewport is initialized to the largest rectangle within the input surface, aligned at the lower left with respect to the input surface, having the same aspect ratio as the display surface, where both aspect ratios are measured in length units as perceived by an operator. The echo mapping (determined by the Input Extent and Echo Viewport state list entries for the LID) is initialized in such a way that the transformations for the echo mapping and the VDC-to-Device Mapping (for the display surface) are effectively the same, although they need not be parameterized in the same way. Specifically, the Echo Viewport is initialized to the entire display surface and the Input Extent is initialized to the image of this Echo Viewport under the inverse of the VDC-to-Device Mapping for the display surface.

The entries for Input Extent, Input Viewport and Echo Viewport are maintained in the Class-Specific Logical Input Device State List. Using the function <input class> DEVICE DATA, the client may redefine the ISC-to-VDC Mapping in a relatively unconstrained manner, as the Input Viewport is not required to be entirely inside the input surface. For example, the client might wish to map the entire input surface onto an Echo Viewport that is smaller than the display surface and accessed by (input) VDC points which bear no direct relation to VDCs being used for graphic output. Conversely, the client might map only a subset of the input surface onto the entire display surface.

For devices of class INPUT, when the LID is initialized the Input Extent is set to an implementation-dependent rectangle and the Input Viewport is set to the full range of the input device.

The Input Extent entry is interpreted in accordance with the VDC type and precision in effect at the time of its definition. Hence, changing the VDC type sets the Input Extent to the default for the chosen type.

The settings of the Device Viewport Specification Mode and Device Viewport Mapping have no relevance to the Input Viewport, which is always specified in native input coordinates (ISPs). Its mapping to the Input Extent is not necessarily isotropic.

For devices of class OUTIN, the position of a geometric echo on the display surface is determined by the current value of the measure. The appearance of an echo should be altered as its image in VDC passes outside of the Input Extent. This part of ISO/IEC 9636 allows latitude from this preferred behaviour and whether the preferred behaviour is supported is indicated by the Echo Change Support entry in the Class-Independent Logical Input Device Description Table. If the LID is sampled in such circumstances, the measure validity status is INVALID.

For devices of class OUTIN, the Echo Area rectangle, which is used to position certain types of non-geometric echo on the display surface, has no relevance to geometric echoing which automatically tracks the current value of a measure over the display surface.

3.6 Triggers

A trigger is typically (though not exclusively) a physical control manipulated by an operator. When a defined set of conditions is satisfied, the trigger is said to *fire*. The trigger firing defines the moment in time at which an input action takes place; if a LID with which the trigger is associated is enabled for request, echo request or event input, the current value of the LID's measure is stored, and thereby made available to the client. If the LID's Acknowledgement Control is ENABLED, then acknowledgement output is also performed when the trigger fires for request or echo request input or when the event report is enqueued for event input.

3.6.1 Trigger association

Triggers may exist independently of any particular LID. For trigger firing to have effect, it is necessary that the trigger be associated with at least one LID.

The CGI offers the client a mechanism for creating and deleting associations between triggers and LIDs. Each LID's Class-Independent Logical Input Device Description Table holds a list of trigger indices indicating those triggers that may be associated with the LID; a (possible empty) subset of this list is designated as triggers that are necessarily associated with the LID, and may not be dissociated. A single trigger may be associated with more than one LID; in this case, when the trigger fires each LID with which it is associated may simultaneously deliver input events to the client.

3.6.2 Timeouts

Some input functions offer a timeout capability. This occurs in situations where the function may not be able to return until the operator takes some action. In case the operator has left or is not paying attention, this provides a mechanism whereby control will eventually be returned to the client in any case.

The duration of a timeout interval is specified in seconds by a real number. A negative value of timeout indicates *wait forever*. That is, there is no possibility of the function returning due to an elapsed timeout interval.

Even if a Virtual Device cannot present a true timeout capability, this wait forever capability is always available. In these cases, a positive value indicates wait forever for the functions REQUEST and INITIALIZE ECHO REQUEST, and indicates a zero timeout for the function AWAIT EVENT. The Timeout Capability entry in the Input Device Description Table specifies the implemented timeout behaviour. The preferred behaviour is FULL timeout capability.

3.6.3 The break action

The operator may abort input by executing a *break action*. The break action cause the LID to abort any input, and to return a pseudo-event to the client, which has no valid measure associated with it. The precise effect of the break action on a LID depends on its input class.

There are two possibilities for the method of triggering a break action. A break action may be initiated by a single method that is common to the entire Virtual Device and applies over a multiplicity of LIDs (in which case, no LID class is associated with it). On the other hand, one may be initiated by a method that is specifically associated with a particular LID (e.g. one of the buttons on a mouse).

If the operator initiates a break action when a LID is in the state REQUEST PENDING or ECHO REQUEST PENDING, then the effect of that break is limited to the (single) LID in that state and the REQUEST or ECHO REQUEST soliciting function returns immediately. At other times, there may be multiple LIDs which are all active for sampling or event input, in which

case the effect of the break (discarding all partially constructed input) is applicable to all active LIDs of class STRING or STROKE.

3.7 Input methods and state model

3.7.1 Logical input device model

There are four methods of performing input with any logical input device of the input classes described above. These four methods are available through groups of functions for request, sample, event, and echo request input.

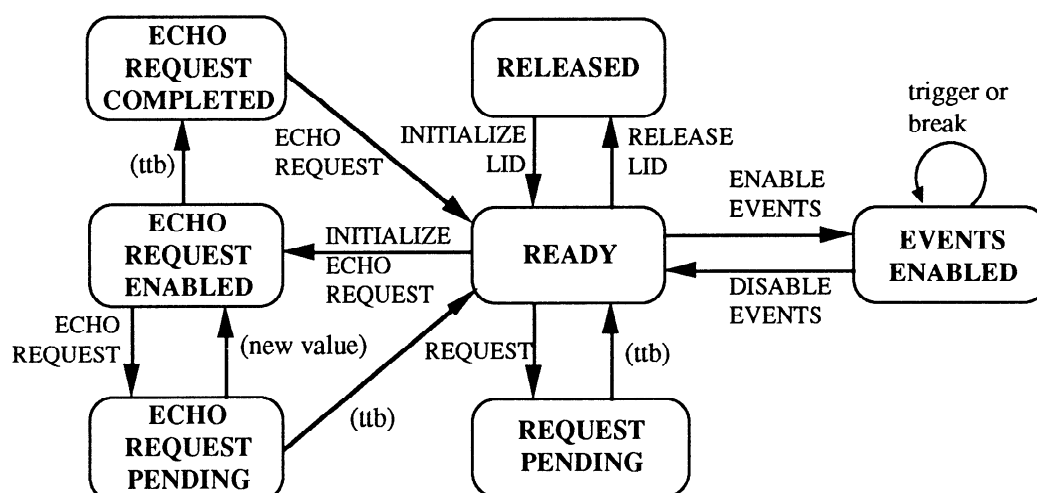
The interface to CGI input functions is a synchronous interface, although asynchronous actions may occur inside the Virtual Device.

The operating state of each logical input device is specified in the appropriate Class-Independent Logical Input Device State List by two input device state variables, the Input Device State and the Sampling State. In addition, the Event Queue State variable in the Event Queue State List defines further input states which are applicable when any input device is using the event input method.

The states in which it is not permissible to invoke the various input functions are defined in tables 13 and 14 and in terms of the logical input device state diagram, figure 2. This state diagram depicts transitions in the value of the Input Device State variable. The transitions take place either as a result of the client invoking a function, as a result of operator action, or as a result of certain other standardized conditions within the LID.

The Input Device State and the Sampling State (see 3.7.3) are defined separately for every instance of a logical input device. Thus, state restrictions apply on an individual basis for each logical input device.

The initial Input Device State of a LID is RELEASED and its measure is unavailable. A LID Input Device State will be set to state READY by the function INITIALIZE LOGICAL INPUT DEVICE. The function RELEASE LOGICAL INPUT DEVICE may be invoked in any state of the LID, the state after invoking RELEASE LOGICAL INPUT DEVICE is always RELEASED. The function INITIALIZE LOGICAL INPUT DEVICE may be invoked in any state of the LID, the state after invoking INITIALIZE LOGICAL INPUT DEVICE is always READY. Note, those transitions are not shown in the state diagrams to avoid unnecessary complexity in the figures.



ttb = trigger, timeout, or break

Figure 2 – Logical Input Device state diagram