I-Beams

! Warning

Any service provided using an I-beam should be considered as experimental and subject to change – without notice – from one release to the next. Any use of I-beams in applications should, therefore, be carefully isolated in cover-functions.

I-beam is a monadic operator that provides a range of system-related services.

Syntax: `R←{X}(A⌶)Y`

where:

- `A` is an integer that specifies the type of operation to be performed
- `X` (optionally) and `Y` are described in the following table
- `R` is the result of the derived function

A	Derived Function	Notes
8	Inverted Table Index-of	X and Y are inverted tables.
13	Log Use of Deprecated Features	Records information in the log file set by `109⌶` about the deprecated feature names or keywords identified in Y (a character vector or vector of character vectors). R lists the feature names for which warnings have been enabled.
43	Monadic Operator Generator *DEPRECATED*	Generates a monadic operator with functionality determined by Y. Possible values of Y are: • `632`: a .NET-specific operator that can create generic classes and execute generic methods. Not supported when using .NET Framework.
85	Execute Expression	X is a scalar specifying whether to retain the shy result obtained by executing expression Y (a character vector). Possible values are: • `0` : retain shy results • `1` : discard shy results (default)

109	Log File for Deprecations	Sets/queries the log file that identifies occurrences of deprecated features/APL code. Used with `13⌶` and `14⌶`. Y can be:
		• `0` : R depends on X:
		  • if X is omitted (monadic use), R is the current filename or `''` if none
		  • if X is a new filename or `''` for none, R is the previous filename, or `''` if there was none
		• `1` : R is the current file status. X is not defined.

120	Generate UUID	Generates *Universally Unique IDentifiers* (UUIDs). Y is the UUID version according to the RFC 9562 specification. Possible values of Y are: • `4` : random values • `7` : combination of time-ordered values (based on Unix Epoch) and random values R is a vector containing a 36-character UUID of the specified version.
127	Overwrite Free Pockets	Overwrites all unused data pockets in the workspace, thereby removing any remnants of potentially secure data. Returns `1` when successful. Y is any empty array, preferably `⍬`.
180	Canonical Representation	Similar to monadic `⎕CR` but enables the canonical representation to be obtained for methods in classes as well as functions and operators. Y is a simple character scalar or vector comprising the name of a defined, system or primitive function or operator or the class.method name.
181	Unsqueezed Type	Similar to monadic `⎕DR` but does not squeeze the data. Returns an integer indicating the array type. Y is any array.
200	Syntax Colouring	Returns syntax colouring information for the APL code specified in Y (a vector of character vectors containing the `⎕NR` representation of a function or operator). The output of `201⌶` can be used to interpret the returned information.
201	Syntax Colour Tokens	Returns a 4-column matrix (token type, value, specific token and TTY colour number) of syntax colouring descriptions. Y is `⍬`.

219	Compress/Decompress Vector of Short Integers	X is a scalar or 1-item (optionally, 2-item) vector specifying the compression library to use. Possible values are: • 1 : use the LZ4 compression library • 2 : use the zlib compression library • 3 : use the gzip compression library • 4 : use the LZ4 compression library with frames (compresses arrays >2GB) If X[1] is positive, then compress. In this situation: • X[2] specifies the compression level in the range 0 to 9 (only if X[1] is not 1) • Y must be a simple integer vector of input data with items in the range -128 to 127 If X[1] is negative, then decompress. In this situation: • X[2] specifies the length of the uncompressed data • Y must be a simple integer vector of compressed data with items in the range -128 to 127 If X is 0, then decompress. In this situation: • Y must be the 2-item vector of vectors produced by a previous 219⌶ compression
220	Serialise/Deserialise Array	X specifies whether Y is to be serialised or deserialised. Possible values are: • 1 : Y can be any array – this array is then serialised • 0 : Y must be a simple integer vector with items in the range -128 to 127 that must have been serialised using this I-Beam – this array is then deserialised

400	Compiler Control	Controls the actions of the Compiler. The nature of Y and R depend on the value of X. Possible values for X are:
		• 0 : set automatic compilation options (default)
		  • If Y is 0, disable automatic compilation (initial setting)
		  • If Y is 1, compile functions when they are fixed (with ⎕FX or from the function editor)
		  • If Y is 2, compile operators the first time they are executed
		  • If Y is 3, compile functions when they are fixed (with ⎕FX or from the function editor) and compile operators the first time they are executed
		• 1 : determine whether the function/operator Y has been successfully compiled
		  • Y must be a character vector, matrix or vector of vectors specifying the name of a function or operator or a list of such names
		• 2 : compile the function/operator Y
		  • Y must be a character vector, matrix or vector of vectors specifying the name of a function or operator or a list of such names that should be compiled
		• 3 : uncompile the function/operator Y
		  • Y is a character vector, matrix or vector of vectors specifying the name of a function/operator (or a list of such names) for which to discard any compiled bytecode. If empty, discard all compiled bytecode in the workspace
		• 4 : show bytecode for the compiled function/operator Y
		  • Y must be a character vector, matrix or vector of vectors specifying the name of a function or operator or a list of such names
		• A namespace : compile the function/operator Y using callbacks to provide information about global names
		  • Y must be a character vector, matrix or vector of vectors specifying the name of a function or operator or a list of such names

600	Disable Traps	Controls whether the trapping mechanism is active. Y is an integer whose possible values are: • 0 : all traps are enabled • 1 : all traps are disabled • 2 : when in suspended functions, errors generated by expressions typed in the Session do not fire traps (default)
739	Temporary Directory *DEPRECATED*	Returns the name of a temporary system directory suitable for user files (no trailing separator is included). Y is 0.
900	Called Monadically?	When included within a tradfn/tradop, returns a Boolean value indicating whether the function/operator was called monadically (1) or not (0). Y is any array (ignored).
950	List Loaded Libraries	Lists the dynamic link libraries that have been loaded by ⎕NA and are still loaded. Y is the empty vector ⍬.

1010	Set Shell Script Debug Options	Sets options for debugging APL "shell scripts". Y is an integer whose possible values are: • 1 : lines in the script are echoed to stderr prior to execution • 2 : behaves as if ⎕TRACE is set for every line of every function in the script • 3 : a combination of the other two options If Y is 1, X optionally specifies a character scalar/vector that prefixes each line of output (the default is '+'). If Y is not specified, the previous value of Y is returned.
1111	Number of Threads	Y is an integer specifying one of the following: • the number of threads to be used for parallel execution (the previous value is returned) • ⍬ (the number of virtual processors in the machine is returned)
1112	Parallel Execution Threshold	Y is an integer specifying the array size threshold at which parallel execution takes place (the previous value is returned).
1159	Update Function Time and User Stamp	X is an array of function attributes in same format as the output of ⎕AT Y is an array of function names in same format as the right argument of ⎕AT
1200	Format Date-Time *DEPRECATED*	X is a character scalar or vector specifying the formatting to apply to the elements in Y Y is a numeric array of any shape, where every element contains a Dyalog Date Number that represents a date between 1 January 0001 and 28 February 4000
1302	aplcore Parameters	Sets/queries values for the aplcore-related configuration parameters. Y can be: • a simple character scalar/vector specifying the new value for AplCoreName (or '') • a simple integer singleton specifying the new value for MaxAplCores (or ⍬) • a 2-element vector in which [1] is a character vector (AplCoreName value) and [2] is an integer (MaxAplCores value) If Y is '' or ⍬, no changes are made. Always returns the previous values.

1500	Hash Array	Y is any array. R is dependent on X. Possible values of X are:
		• 1 : R is an integer reporting on the hash status of Y. Possible values of R are:
		  • 0 : Y has not been marked for hashing
		  • 1 : Y has been marked for hashing but does not yet have a hash table
		  • 2 : Y has a hash table
		• 2 : R is the unhashed form of Y
		If X is not specified, R is a copy of array Y that has been marked for hashing (the hash table will be created the first time the array is used as an argument to ⍳ or other set functions).

2000	Memory Manager Statistics	Y is an integer or vector of integers specifying the statistics to be displayed (if X is not specified) or set (if X is specified). Possible values are: • 0 : workspace available • 1 : workspace used • 2 : number of compactions since loading workspace • 3 : number of garbage collections that found garbage • 4 : number of garbage pockets currently in workspace • 9 : number of free pockets currently in workspace • 10 : number of used pockets currently in workspace • 12 : sediment size • 13 : amount of memory currently being used in workspace • 14 : maximum amount of memory used since workspace was loaded • 15 : limit on minimum workspace allocation • 16 : limit on maximum workspace allocation • 19: number of calls to ⎕WA or 2002⌶ since the last time 2000⌶ was called (or since the process started if this is the first call to 2000⌶) • 20: requested size of the WS FULL buffer • 21: actual size of the WS FULL buffer • 22: number of WS FULL handlers currently running • 23: number of WS FULL errors that have occurred • 24: number of WS FULL errors that have been trapped Optionally, X is an integer or vector of integers of the same length as Y specifying the value to change the specified Y item to. Possible values are: • for Y is 2, X must be 0 (resets the compaction count) • for Y is 3, X must be 0 (resets the garbage collection count) • for Y is 14, X must be 0 (resets the amount of memory used since ws was loaded) • for Y is 15, X must be between 0 and the current workspace allocation (sets the minimum workspace allocation) • for Y is 16, X must be between the current workspace allocation and MAXWS (sets the maximum workspace allocation) • for Y is 19, X must be 0 (resets the compaction count) • for Y is 20, X must be the required size of the WS FULL buffer
2002	Specify Workspace Available	Similar to ⎕WA but allows the memory allocation to be specified explicitly. Returns an integer indicating the size (in bytes) of the memory committed for the workspace. Y is an integer specifying the size (in bytes) of the extra memory that is added to the compacted workspace before de-committing the remainder.
2007	Disable Global Triggers	Controls whether global triggers are active (useful when databinding) – only active in the APL thread in which it is called. Y is an integer whose possible values are: • 0 : all global triggers are enabled (default) • 1 : all global triggers are disabled
OS WIN 2010	Update DataTable	NOTE: Not supported when using .NET. X is a Boolean vector with same number of items as the instance of System.Data.DataTable matrix has columns (a 1 indicates that the corresponding column contains strings that must be passed to the Parse method of the data type). Y is a 2, 3 or 4-item array comprising (in this order): • a reference to the instance of System.Data.DataTable • a matrix with the same number of columns as the instance of System.Data.DataTable • a vector with the same number of items as the instance of System.Data.DataTable matrix has columns, with each item specifying the value to map to DBNull when this column is written to the instance of System.Data.DataTable • Row indices (zero origin) of the rows to be updated; if omitted, then data will be appended to the instance of System.Data.DataTable

OS WIN 2011	Read DataTable	NOTE: Not supported when using .NET. Y is a 1 or 2-item array (scalar or vector) comprising (in this order): • a reference to the instance of System.Data.DataTable • a vector with the same number of items as the instance of System.Data.DataTable has columns, with each item specifying the value that a DBNull in the column should be mapped to when this column is read The **Invert** variant option (default = 0) determines R: • 0 : R is a matrix with the same shape as the DataTable referenced by ⊃Y • 1 : R is a vector whose length is the same as the number of columns in the DataTable referenced by ⊃Y X is a numeric vector whose length is the same as the number of columns in the DataTable referenced by ⊃Y (if X has fewer elements than there are columns then the missing values are assumed to be 0 and those columns are not transformed): • 1 : Specifies that the corresponding column of the result should be converted to a string using the `ToString` method of the data type of the column. • 2 : Specifies that numbers of type `System.Int64` in the corresponding column of the result should be converted to DECFs rather than to .NET objects (which is the default) • 4 : Only applies when the **Invert** variant option is 1 and the type of the corresponding column is `System.String`. Specifies that the entire column should be returned as a character matrix rather than as a vector of character vectors (any nulls will be replaced with a row of spaces). • 5 : Combines 1 and 4 (that is, generates strings and then generates a matrix from those strings).
OS WIN 2014	Remove Data Binding	NOTE: Not supported when using .NET. Disassociates a data-bound variable from its data binding source. Returns 1 when successful. Y must be a character vector containing the name of the data-bound variable to be disassociated (otherwise all databinding is removed from the workspace).
OS WIN 2015	Create Data Binding Source	NOTE: Not supported when using .NET. X is optional; if omitted, then default binding types are used. If included, its structure is dependent on the content of Y. Y is a character vector comprising the name of one of the following: • a matrix: X is a two-column matrix specifying the name and binding type for each column in matrix Y • a variable : X is a single .NET type specifying the binding type for variable Y • a namespace containing variables(s) : X is a two-column matrix specifying the name and binding type for each variable in namespace Y • a variable containing vector of refs to namespaces containing variables(s) : X is a two-column matrix specifying the name and binding type for each variable in each namespace
OS WIN 2016	Create .NET Delegate	NOTE: Not supported when using .NET. Returns an instance of the .NET type specified in Y[1] that can be used to invoke the function in Y[2]. Y is a vector comprising: • [1] is a .NET type that derives from `System.Delegate`, for example, `System.EventHandler` • [2] is either the name or the ⎕OR of a function to be used as a callback.
OS WIN 2017	Identify .NET Type	NOTE: Not supported when using .NET. Returns the .NET type of Y for types that are located in any assembly that has been loaded into the current AppDomain by calling ⎕USING or `:using` (the assembly-qualified name is required by `System.Type.GetType`). Y is a character string containing the name of a .NET object (namespace names can be omitted if they are provided in elements of ⎕USING).
OS WIN 2022	Flush Session Caption	Updates the Session caption. Y is any array (ignored).
2023	Close all Windows	Closes all open **Edit**/**Trace** windows without resetting the state indicator. Returns 1 when successful. Y is any array (ignored).
OS WIN 2035	Set Dyalog Pixel Type	Specifies how Coord 'Pixel' is interpreted by all GUI operations. Y is a character vector whose possible values are: • 'ScaledPixel' • 'RealPixel'

OS WIN	2041	Override COM Default Value	By default, if a COM object is in error or is of a type that cannot be represented in APL, then an error is generated in the Session; if the COM object is of type VT_EMPTY then ⎕NULL is returned. Y is an integer whose possible values are: • 1 : X specifies the value that is returned instead of ⎕NULL when the COM object is of type VT_EMPTY • 2 : X specifies the value that is returned when the COM object is in error or is of a type that cannot be represented in APL Omitting X restores the default behaviour.
OS WIN	2100	Export to Memory	Exports the active workspace as an in-memory .NET assembly. Returns 1 when successful. Y is any array (ignored).
OS WIN	2101	Close .NET AppDomain	NOTE: Not supported when using .NET. Close the current .NET AppDomain (started by the current APL process). Returns 0 when successful, otherwise returns a non-zero integer. Y is any array (ignored).
	2250	Verify .NET Interface	Provides information about the Dyalog-.NET interface. Y must be 0 and is ignored. R is a vector of vectors in which ⎕[1] indicates .NET support, ⎕[2] indicates failure (0) or success (1) in loading, and ⎕[3] is a text vector containing error messages generated during load. Possible values of R⎕[1] are: • ¯1 : .NET is not supported • 0 : .NET is supported but not configured • 1 : Configured to use .NET • 2 : Configured to use .NET Framework (Microsoft Windows only)
	2400	Set Workspace Save Options (workspace specific)	Specifies whether *Trace*, *Stop* and *Monitor* settings are cleared when workspace is saved. Y is an integer whose possible values are: • 0 : settings are not cleared on saving (default) • 1 : settings are cleared on saving
	2401	Expose Root Properties	Specifies whether Root Properties, Events and Methods are exposed. Y is an integer whose possible values are: • 0 : no further Root Properties, Events and Methods are exposed (default) • 1 : Root Properties, Events and Methods are exposed
OS WIN	2501	Discard Thread on Exit	Specifies whether threads created to serve incoming .NET requests are destroyed or persist (the default) on completion of their task. Y is an integer singleton; when set to 0 on the current thread, that thread is destroyed on termination rather than persisting.
OS WIN	2502	Discard Parked Threads	Destroys all persistent threads in the workspace. Y is any array (ignored).
	2503	Mark Thread as Uninterruptible	Specifies whether a thread and/or its children respond to a weak interrupt. Y is an integer whose possible values are: • 0 : the thread and its children are interruptible (default) • 1 : mark thread as uninterruptible • 2 : mark children of the thread as uninterruptible • 3 : mark thread and its children as uninterruptible
OS WIN	2520	Use Separate Thread for .NET	NOTE: Not supported when using .NET. Specifies whether .NET code run on APL thread 0 runs on the same operating system thread as the session. Y is an integer whose possible values are: • 0 : .NET code runs on the same thread as the session (default) • 1 : .NET code called from APL thread 0 runs on its own thread
	2704	Continue Autosave	Enables or disables the automatic saving of a CONTINUE workspace when Dyalog exits. Y is an integer whose possible values are: • 0 : disable the automatic saving of a CONTINUE workspace • 1 : enable the automatic saving of a CONTINUE workspace
	3002	Disable Component Checksum Validation (system wide)	Specifies whether checksums are validated by ⎕FREAD. Y is an integer whose possible values are: • 0 : ⎕FREAD will not validate checksums • 1 : ⎕FREAD will validate checksums (default)

3012	Enable Compression of Large Components	Specifies whether large components (> 2GB) can be compressed. Y is an integer whose possible values are: • 0 : Large components are not compressed. • 1 : Large components are compressed if Z property is 1 (see ⎕FPROPS) Versions of Dyalog prior to v19.0 cannot read compressed large components.
3500	Send Text to Ride-embedded Browser	Optionally, X is a simple character vector, the contents of which are used as the caption for the tab in Ride that contains the embedded browser. If omitted, then the default is "HTML". Y is a simple character vector the contents of which are displayed in the embedded browser tab. R identifies whether the write to Ride was successful. Possible values are: • 0 : the write to Ride was successful • ¯1 : Ride is not enabled • any other value : contact support@dyalog.com
3501	Connected to Ride?	X and Y are any value (ignored). R identifies whether the Session is running through Ride. Possible values are: • 0 : the Session is not running through Ride • 1 : the Session is running through Ride
3502	Manage Ride Connections	Controls connections between Ride and an interpreter. Returns 0 if successful or a positive or negative integer if unsuccessful. Y has the following possible values: • 0 : disable any active Ride connections – only valid when Ride is enabled • 1 : enable Ride using the initialisation string defined in the RIDE_INIT configuration parameter – only valid when Ride is not enabled • a simple character vector : specifies an initialisation string that replaces the RIDE_INIT configuration parameter – only valid when Ride is not enabled On a run-time interpreter, 3502⌶1 is the only way to enable Ride.
3535	Scan For Deprecated Files	Scans the specified directory (and, optionally, sub-directories) for deprecated workspaces, component files, and external variables, and returns a matrix of filenames and explanations.
OS UNIX 4000	Fork New Task	Initiates a new APL process with the same execution stack and runs the task in both processes. Returns 0 in the child process and the child's process ID in the parent process. Y is a simple empty vector (ignored).
OS UNIX OS MAC 4001	Change User (system wide)	Should only be run as *root*. Changes the effective user ID for the process. Runs the user name specified in Y (a character vector specifying a valid UNIX name) if successful.
OS UNIX 4002	Reap Forked Tasks	Returns a matrix of newly-terminated child processes along with information about each of those processes (including whether the process terminated normally or as a result of a signal). The first three of the 18 columns indicate: • R[;1] is the process ID of the terminated child • R[;2] is the exit code of the child process (¯1 if the process terminated as the result of a signal) • R[;3] is the signal number that caused the child process to terminate (¯1 if the process terminated normally) Y is a simple empty vector (ignored).
OS UNIX OS MAC 4007	Signal Counts	Returns an integer vector of signal counts whose length corresponds to the number of signals supported by the operating system. Elements are the counts of SIGHUP, SIGINT, SIGQUIT, SIGTERM and SIGWINCH signals (others are 0). Y is a simple empty vector (ignored).
5171	Discard Source Information	Removes source code and file information for scripted objects, namespaces, classes, functions, and operators that is saved in the workspace. Y is a vector or scalar containing zero or more references to # and ⎕SE, and specifies the namespaces from which the information is removed.
5172	Discard Source Code	Specifies whether source code is discarded for functions and operators when they are created by the editor or by ⎕FIX. Y is an integer whose possible values are: • 0 : source code is retained in the workspace when an object is fixed (default) • 1 : source code is discarded from the workspace when an object is fixed (source code already retained in the workspace is not discarded)
5176	List Loaded Files	Returns a list of all of the files that are associated with objects in the active workspace, together with information about those files. Y is any array (ignored).
5177	List Loaded File Objects	Returns details of all the objects in the active workspace that are associated with a file. Y is an empty array (ignored).
5178	Remove Loaded File Object Info	Removes file/script information about single workspace object Y from the workspace.

5179	Loaded File Object Info	Returns file/script information about single workspace object Y.
5581	Unicode Normalisation	Returns Y with all character data within it normalised to the form specified in X (all four Unicode normalisation forms are supported – NFC, NFD, NFKC, and NFKD).
7162	JSON Translate Name	X (scalar) specifies how name Y (a character vector or scalar) is converted between APL and JSON formats. Possible values are: • 0 : Y is converted from a JSON name into a valid APL name • 1 : Y is converted from an APL name into a valid JSON name
8373	Shell Process Control	Returns the child process ID for APL thread number Y (or all process IDs running after ⎕SHELL has completed if Y is ⍬). If X is an integer scalar representing a signal number that is to be sent to the child process and Y is an integer scalar identifying the child process by APL thread number (positive) or process ID (negative), then R is a Boolean scalar indicating whether the signal was successfully sent (1) or not (0).
8415	Singular Value Decomposition	Computes the singular value decomposition of a matrix Y ; useful when ⌹ cannot compute an inverse due to Y being singular or nearly singular. Returns a nested vector U S V f (where Y ≡ U +.× S +.× ⍉V) in which: • U and V are unitary matrices • S is a diagonal matrix • f is a Boolean indicating whether the algorithm converged (1) or not (0)
8659	List Shared Code Files/Attached Names	64-bit Unicode only. R depends on Y: • If Y is ⍬, R is a 2-column matrix listing the shared code files that are attached to the current workspace • If Y is an integer vector that would be a valid right argument to ⎕NL, and X is the slot (integer in range 1 to 8) in which the file is saved, R lists the nameclasses and subclasses for which the names should be listed
8666	Attach/Assimilate/ Detach Shared Code Files	64-bit Unicode only. The behaviour depends on Y: • If Y is a single character vector or a vector of character vectors of shared code files, they are loaded ("attached"). Optionally, X specifies nameclasses to include • If Y is ⎕NULL, all referenced objects in the shared code files are copied into the active workspace • If Y is 0⍴⊂'', any existing attached shared code files are detached
8667	Save Shared Code Files	64-bit Unicode only. Saves a shared code file. Y is a 2-item vector specifying the slot (integer in range 1 to 8) in which to save the file and the filename. Optionally, X restricts the functions/operators/variables in the active workspace that are saved.
16808	Sample Probability Distribution	Generates an array of random numbers from a named probability distribution. Y is a 2-item vector specifying the name of the probability distribution and the shape of the result. X is a scalar or 1- or 2-item numeric vector that specifies parameters for the named distribution.
50100	Line Count	Restricts the number of calls to ⎕LC, thereby potentially improving performance. Y is any positive integer; R returns at most the first Y elements of ⎕LC.