

Appendix V: How to Read Card Summaries

LST has written the LS-DYNA manual an encyclopedic reference for LS-DYNA. The first goal of the manual is to fully document all features and their corresponding input structures. Secondly, we have structured the manual so that any chunk of keyword input can be quickly interpreted. As of R11 we have introduced a new standard format aimed at making the manual even more useful for *quickly* interpreting keyword input. The definitions below, which are provided for clarity, are followed by a lengthy discussion of this new format.

Definitions: Decks, Cards, and Keywords

1. **Decks.** The text file containing the LS-DYNA input is called *the input deck*. The term *deck* is historical in origin, originally referring to actual punch-cards.
2. **Cards.** Each line of the input file is called a *card*, as in a *punch-card*. Taken together the *cards* comprise the *deck*.
3. **Keywords.** A typical input deck consists of two kinds of cards: (1) keyword cards and (2) data cards. Keyword cards are easily recognized by the "*" -character in the first column followed directly by a (case insensitive) *keyword*. These cards form the skeleton of the deck. LS-DYNA treats each keyword card as a kind of command (or function or subroutine) for which the arguments are passed with data cards.
4. **Keyword Options.** Each keyword card consists of a base keyword combined with option words. Option words are appended to the base keyword using underscores. For example, to invoke the *node* keyword with the *merge* option one would include a card (a line) consisting of the text "*NODE_MERGE" where the "*" -character is in the first column. We have, for clarity, adopted a convention of capitalizing (the case insensitive) keywords.

Introduction to Card Summaries

The keyword manual (the enormous PDF you're looking at) is organized by keyword. As mentioned in item 3 above, keywords are like subroutines (or functions, if you program in C). Like subroutines, keywords take input. Unlike subroutines, keyword input is not divided into a fixed number of strongly-typed fields. For instance, for a given keyword the number of fields and their data types can, and often does, depend on what is contained in the input itself. Keyword input is, then, best described by flow chart.

APPENDIX V

After processing a keyword card all cards until the next keyword card are treated as the keyword's input. Collectively, these cards are called the *data cards*. Most of the LS-DYNA manual is devoted to describing how each keyword interprets the contents of its data cards (the flowchart mentioned in the previous paragraph). *Beginning in 2018 (R11), new and updated entries to the manual will contain a section called "card summary" that expresses the structure of the keyword's data in a concise and standardized format.* We hope that these summaries will be simple enough that you can understand them without reading this document.

Motivation for Introducing the Card Summary Section.

1. **"Optional" Cards/Blank Cards.** Traditionally, some data cards have been called *optional* and others have been called *required*. This can be confusing. Some examples of frequently asked questions are:

Can the second optional card be omitted? The answer is: not if you intend to include the third optional card.

Can I add blank lines to make my deck more human-readable? The answer is "no" because LS-DYNA interprets blank lines as data. For instance, each data card after a *NODE is expected to contain a node definition. A blank card is interpreted as an invalid node definition.

2. **Branching (and Switch Constructs).** Often the format of the next data cards depends on what was in the preceding cards or on the options applied to the keyword. Sometimes the branching can be complicated. We have adopted a compact convention to express branching in the *card summaries*.
3. **Repeated Cards (DO/FOR Loops Constructs).** There has been confusion over the meaning of the *card number* as presented in traditional manual entries. In some cases, a data card's card number indicates its position relative to the keyword; in other cases, the card number serves as a kind of format identifier. Iterated cards, however, make the line-number interpretation impossible without saying something like *Card n*, where *n* is an integer, and we almost never do that. Card summaries clear up this confusion.
4. **Default Values.** When a field is blank or set to 0, then LS-DYNA applies a default value. The meaning of "0" for objects like node sets can be confusing and, in the past, hasn't been explicitly spelled out. As part of the format revamp we are adopting [new conventions](#) for ambiguous cases.

“Optional” Cards (How LS-DYNA Parses Blank Lines).

When LS-DYNA reads an input deck some data cards contain fields having no default value. These data cards are said to be *required*; LS-DYNA will error terminate when required cards are missing. In contrast, data cards consisting entirely of fields that have default values can, generally, be left blank. Such cards are called *optional*.

The details of the algorithm for processing optional cards are subtle, but the application of optional cards is nonetheless mostly intuitive. An *optional card* is only optional in the sense that if *all the cards* that come after it are also optional, then the card can be omitted, but *only if all the subsequent cards are also omitted*. For example, if a keyword has three optional data cards and some of the values on the third optional card need to be changed from their default values, then the two previous optional cards must be included, at the very least, as blank lines. Note that there are a very small number of irregular cases where an omitted card is treated differently from a blank card; such special cases are documented when they appear.

Example:

The example below illustrates some of the subtleties involved in deciding when to omit an optional card. In this example, two rigid wall cylinders are defined with a single instance of the *RIGIDWALL_GEOMETRIC_CYLINDER_ID_DISPLAY_MOTION keyword by including two sets of data cards, one for each cylinder. The DISPLAY option activates an optional data card. Since two cylinders are being defined, the DISPLAY data card must be included, at least as a blank line, for the first cylinder; otherwise, LS-DYNA will read the first line for the second cylinder as new defaults for the DISPLAY option data card causing an error termination due to format mismatch. The last card included in the input below for the second cylinder is *not* the DISPLAY data card. For this case, it does not have to be input since it is optional and no further data cards follow. A blank line may be included for the second DISPLAY data card, but is not necessary, and excluding it makes no difference.

```

:
*RIGIDWALL_GEOMETRIC_CYLINDER_ID_DISPLAY_MOTION
$ first cylinder
$     ID
      100
$     nsid    nsidex    boxid    birth    death
          0          0     0.000     0.000
$     xt        yt        zt        xh        yh        zh        fric
-20.000000     0.000-50.000000-21.000000     0.000-50.000000     0.000
$     radcyl   lencyl
      20.000000  30.000000
$     lcid      opt       vx       vy       vz
          99          0     0.000   1.000000     0.000
$
$ Blank line for card data associated with DISPLAY. This blank line is necessary since
$ 2 card sets are being used even though this card is optional. If the blank line was
$ not included, then LS-DYNA would read 101 as the DISPLAY data card line which would
$ lead to an error termination
$
$ second cylinder

```

APPENDIX V

```
101
          0      0     0.000    0.000
-20.000000  0.000 10.000000-21.000000    0.000 10.000000    0.000
 20.000000 30.000000
          2      0.000 1.000000    0.000
$ No blank line is needed here since this is the last card set and the DISPLAY data
$ card is optional.
*control_contact
:
```

The Structure of Manual Entries.

The following table compares traditional entries to summary table formatted entries:

Old Format (no summary table)	Summary Table Format
1. The entry begins with a “purpose” statement	1. No change.
2. The purpose statement is followed by an explanation of options, if there are any. Sometimes the purpose statement and option sections are reversed.	2. No change.
3.	3. Summary table
4. Several small tables of between 2 and 4 rows are included, roughly, one for each type of data card. Each table contains a short note about how many times its repeated.	4 & 5. Each card table is followed directly by a short variable list. Generally, each branching is represented by a <i>different</i> card table.
5. These tables are followed by a single long “variable list table,” which defines the meaning of all the fields on all of the preceding tables. Branching is mostly explained in the variable list.	
6. Lastly, manual entries often contain a “remarks” section.	6. No change.

As indicated in the above table, the differences between the two formats are confined to items 3, 4, and 5. The traditional format, when applied to keywords with large data sets suffers a few drawbacks which the summary table format addresses:

1. For keywords having many data cards (> 5) the variable lists may be several pages removed from the original data card. In the summary table format a data card’s fields are explained right after the data card.

2. The entire set of data card tables, itself, may span several pages. Ideally, it should be on one page so that a user can look at a single page when trying to interpret his input deck. The card summary table is compact (and more likely to fit on one page) and contains enough information to match each field in an input deck to the name of the field in the manual.
3. In cases where the meaning of a data card's field depends on previous data cards, the traditional format puts the branching explanations entirely in the variable list. Interpreting an input deck then requires lots of page flipping. Moreover, since groups of fields often branch together, the same branching verbiage is often repeated in field after field requiring you to read each-and-every repetition in case one field works differently from the others, which sometimes happens.

In the card summary format branching is explained in the summary table and sometimes repeated on the card representations. Since the fields are explained after the card, the branching is *not* generally dealt with in the variable lists.

Summary Table Format

The summary table itself consists of two rows per data card (see [Example 1](#) directly below). The first row consists of pseudo-code containing an identifier for the card type, the logic used to determine when a data card is included, and the logic used to determine how many times it is repeated. The second row represents the data card's fields following the convention from the traditional manual entry.

The summary tables will, for the most part, adhere to the conventions described below.

1. Cards are labeled as "Card [identifier]". In the simplest case, the identifier is an integer. See [Example 1](#).
2. In the case of a group of mutually exclusive cards (or sets of cards), the variants are expressed by adding a letter onto the identifier. Letters increment alphabetically. See [Example 2](#).
3. Cards that are grouped together (in contrast to ones that are mutually exclusive) are indicated by adding a ".*n*" onto the base identifier, where *n* is the card number with the first card in the group being numbered "1". This construct is analogous to increasing indentation in python or nesting with curly braces in C. See [Example 3](#).
4. Cards with italicized names are optional. See [Example 2](#).

APPENDIX V

Example 1.

The data cards for the *RIGIDWALL_PLANAR keyword have neither exclusive branches or conditional sets. The cards are, therefore, numbered linearly. To the extent there is branching, it is only to include extra data cards as options are appended to the keyword.

ID Card. First data card of the card set if the ID keyword option is used in the keyword name. Otherwise Card 1 is the first data card.

ID							
----	--	--	--	--	--	--	--

Card 1. This card is required.

NSID	NSIDEX	BOXID	OFFSET	BIRTH	DEATH	RWKSF	
------	--------	-------	--------	-------	-------	-------	--

Card 2. This card is required.

XT	YT	ZT	XH	YH	ZH	FRIC	WVEL
----	----	----	----	----	----	------	------

Card 3. This card is included only if the ORTHO option is used.

SFRICA	SFRICB	DFRICA	DFRICB	DECAYA	DECAYB		
--------	--------	--------	--------	--------	--------	--	--

Card 4. This card is included only if the ORTHO option is used.

NODE1	NODE2	D1	D2	D3			
-------	-------	----	----	----	--	--	--

Card 5. This card is included only if the FRICTION option is used.

XHEV	YHEV	ZHEV	LENL	LENM			
------	------	------	------	------	--	--	--

Card 6. This card is included only if the MOTION option is used.

MASS	V0						
------	----	--	--	--	--	--	--

Card 7. This card is included only if the FORCES option is used.

SOFT	SSID	N1	N2	N3	N4		
------	------	----	----	----	----	--	--

APPENDIX V

For the *RIGIDWALL_PLANAR_MOTION_FORCE_ID keyword the summary table is interpreted as

ID	ID						
1	NSID	NSIDEX	BOXID	OFFSET	BIRTH	DEATH	RWKSF
2	XT	YT	ZT	XH	YH	ZH	FRIC
6	MASS	V0					
7	SOFT	SSID	N1	N2	N3	N4	

For the *RIGIDWALL_PLANAR_ORTHO_FRICTION keyword the summary table is interpreted as

1	NSID	NSIDEX	BOXID	OFFSET	BIRTH	DEATH	RWKSF	
2	XT	YT	ZT	XH	YH	ZH	FRIC	WVEL
3	SFRICA	SFRICB	DFRICA	DFRICB	DECAYA	DECAYB		
4	NODE1	NODE2	D1	D2	D3			
5	XHEV	YHEV	ZHEV	LENL	LENM			

Example 2.

Card 1. This card is required.

NEIG	CENTER	LFLAG	LFTEND	RFLAG	RHTEND	EIGMTH	SHFSCL
------	--------	-------	--------	-------	--------	--------	--------

Card 2. Card 2 and beyond are optional. If Card 3a or 3b are included, then Card 2 must be included; Card 2 can be a blank line in this case, so default values are used.

ISOLID	IBEAM	ISHELL	ITSHELL	MSTRES	EVDUMP	MSTRSCL	
--------	-------	--------	---------	--------	--------	---------	--

Card 3a. Card 3a is read only when EIGMTH = 101. It is optional.

IPARM1	IPARM2	IPARM3	IPARM4	RPARM1			
--------	--------	--------	--------	--------	--	--	--

Card 3b. Card 3b is read only when EIGMTH = 102. It is optional.

IPARM1	IPARM2			RPARM1	RPARM2		
--------	--------	--	--	--------	--------	--	--

APPENDIX V

Example 3.

ID Card. First data card of the card set if the ID keyword option is used in the keyword name. Otherwise Card 1 is the first data card.

RWID	HEADING						
------	---------	--	--	--	--	--	--

Card 1. This card is required.

NSID	NSIDEX	BOXID	BIRTH	DEATH			
------	--------	-------	-------	-------	--	--	--

Card 2. This card is required.

XT	YT	ZT	XH	YH	ZH	FRIC	
----	----	----	----	----	----	------	--

Card 3a. This card is included only for the FLAT shape.

XHEV	YHEV	ZHEV	LENL	LENM			
------	------	------	------	------	--	--	--

Card 3b. This card is included only for the PRISM shape.

XHEV	YHEV	ZHEV	LENL	LENM	LENP		
------	------	------	------	------	------	--	--

Card 3c. This card is included only for the CYLINDER shape.

RADCYL	LENCYL	NSEGS					
--------	--------	-------	--	--	--	--	--

Card 3c.1. NSEGS instances of this card are included only for the CYLINDER shape.

VL	HEIGHT						
----	--------	--	--	--	--	--	--

Card 3d. This card is included only for the SPHERE shape.

RADSPH							
--------	--	--	--	--	--	--	--

Card 4. This card is required if the MOTION keyword option is used.

LCID	OPT	VX	VY	VZ			
------	-----	----	----	----	--	--	--

Card 5. This card is read only if the DISPLAY keyword option is used. It is optional and may be omitted unless more than one card set is being read in; if more than one card set is being used, then at least a blank line must be included for all but the last card set. If not input, the defaults will be used.

PID	R0	E	PR	↓			
-----	----	---	----	---	--	--	--

Default Values for Data Cards.

The keyword reader always, with a vanishingly small number of exceptions, fills in blank fields with the value 0. LS-DYNA, then replaces the value of 0 with *the default value*. The default value row of a keyword card table adheres to the following conventions:

1. Often, the row of default values is missing from the keyword card tables. In this case see the variable list following the table (or after the tables for traditional format) for information about default values.
2. A “↓” character indicates that the default value is explained in the variable list. This, likely, indicates that the explanation involves more content than the table can accommodate.
3. If the field is required, then the default value is written as *none*.
4. The symbol {Ø} indicates an empty set.
5. “Rem *n*” where *n* is an integer indicates that the field is explained in a remark.
6. If the default value is listed as being 0, then one of two things is being described: (1) the default value is *actually* 0, or (2) the table is indicating the tautological, namely, that the keyword reader fills blank fields with zero, which always happens for every field. In the second case the meaning of zero is almost always explained within the variable list. Going forward, LST will try to avoid this second case.

